sensors

MDPI

Article

Accurate Hand Detection from Single-Color Images by Reconstructing Hand Appearances

Chi Xu [1,2], Wendi Cai [1,2,*,†], Yongbo Li [1,2,*], Jun Zhou[1,2] and Longsheng Wei [1,2]

[1] School of Automation, China University of Geosciences, Wuhan 430074, China; xuchi@cug.edu.cn (C.X.); jchowcug@gmail.com (J.Z.); weilongsheng@163.com (L.W.)
[2] Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, Wuhan 430074, China
* Correspondence: caiwendi@cug.edu.cn (W.C.); ybli@cug.edu.cn (Y.L.)
† This author contributed equally to this work.

Received: 21 October 2019; Accepted: 27 December 2019; Published: 29 December 2019

Abstract: Hand detection is a crucial pre-processing procedure for many human hand related computer vision tasks, such as hand pose estimation, hand gesture recognition, human activity analysis, and so on. However, reliably detecting multiple hands from cluttering scenes remains to be a challenging task because of complex appearance diversities of dexterous human hands (e.g., different hand shapes, skin colors, illuminations, orientations, and scales, etc.) in color images. To tackle this problem, an accurate hand detection method is proposed to reliably detect multiple hands from a single color image using a hybrid detection/reconstruction convolutional neural networks (CNN) framework, in which regions of hands are detected and appearances of hands are reconstructed in parallel by sharing features extracted from a region proposal layer, and the proposed model is trained in an end-to-end manner. Furthermore, it is observed that the generative adversarial network (GAN) could further boost the detection performance by generating more realistic hand appearances. The experimental results show that the proposed approach outperforms the state-of-the-art on public challenging hand detection benchmarks.

Keywords: hands detection; hand appearance reconstruction; convolutional neural networks; generative adversarial network; human–computer interaction

1. Introduction

The human hand plays a very important role in communication when people interact with each other and with the environment in everyday life. The recognition of hand gestures and human activities is closely related to the locations and directions of human hands. Therefore, detecting human hand reliably [1] from single color images or videos which are captured from common image sensors plays an important role in many computer vision-related applications, such as human–computer interaction [2,3], human hand pose estimation [4–6], human gesture recognition [7,8], human activity analysis [9], and so on.

In the computer vision field, the pipeline of hand-related applications usually contains three steps: (1) hand detection, (2) hand pose estimation and (3) static gesture recognition or dynamic action recognition. The second step is optional because gesture/action recognition can be performed with or without hand pose estimation. About five years ago, hand pose estimation and action recognition were the most challenging steps (or bottlenecks) in the pipeline even in the constrained environment (normally only single hand and simple background in an image) from which the hand can be easily detected or assumed to be already cropped. Therefore, the hand-related community primarily focused on the second and third steps in the past. However, nowadays hand pose estimation

and gesture/action recognition in the constrained environments are reaching the mature level, and hand-related applications in an unconstrained environment (complex background and the number of hands in an image is unknown) will be an important trend in the future. In this condition, hand detection in an unconstrained environment becomes a new bottleneck in the hand-related works. Thus, the high precision hand detection method will be a crucial step in the pipeline for hand-related applications in unconstrained environment. In this paper, we focus on the hand detection algorithm.

Traditional hand detection methods primarily utilize low-level image features such as skin color [10] and shape [11,12] for hand region detection. Nowadays, convolutional neural networks (CNN) based detection approaches [13–19] are proved to be more robust and accurate [20–22] due to the discriminative deep features learned. However, compared to common objects, human hands are highly articulated, appearing in various orientations, scales, shapes, skin colors, and sometimes partial occlusions, therefore reliably detecting multiple human hands from unconstrained cluttering scenes remains to be a challenging problem.

To tackle this problem, we propose an approach to accurately detect human hands from single color images by reconstructing the hand appearances. It can also be applied to video clips, as video clips can be considered as sequences of single images. The spirit of our approach is primarily oriented from multitask learning [23] which improves generalization of the network by learning tasks in parallel while using a shared representation. The quality of the hand detection task is closely related to the diversity of hand appearances in terms of hand shape, skin color, orientation, scale, and partial occlusion, etc. As a result, the shared information contained in the training signal of the hand appearance reconstruction task can be utilized as an inductive bias to improve the performance of the hand detection task.

The general process of the proposed approach is illustrated in Figure 1. Firstly, feature maps of the whole input image are extracted by shared convolutional layers. Then, they are fed into a region proposal network (RPN) to generate possible region proposals, a.k.a. region of interest (RoI). Finally, the feature maps of RoIs are used to classify the corresponding labels (hand/background), to refine the locations of detected hands, and to reconstruct the appearances of hands simultaneously.

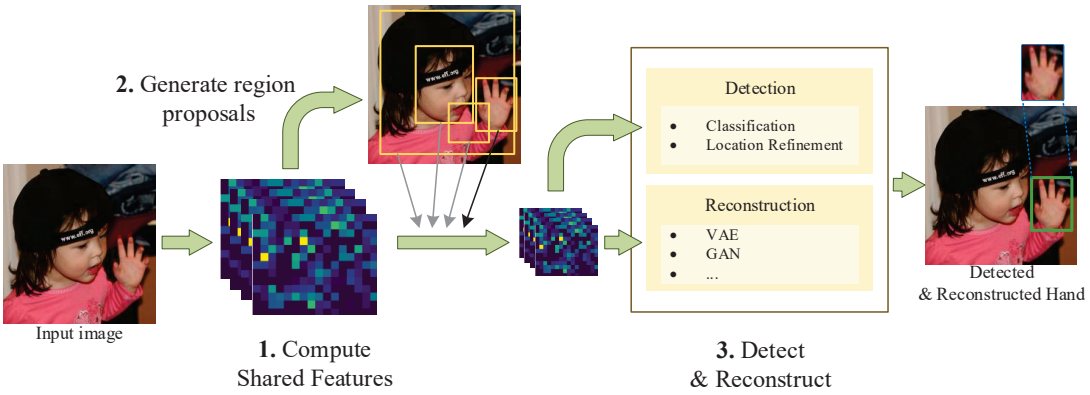

Figure 1. General process of our hybrid detection/reconstruction framework.

In this paper, a new hybrid detection/reconstruction CNN framework is present. The detection branch and reconstruction branch share the same feature presentation, so that the shared information contained in the hand appearance reconstruction branch can be utilized to boost the performance of the hand detection branch. We adopt the idea of variational autoencoder (VAE) [24,25] for hand appearance reconstruction. On the one hand, there is no previous work which adopted VAE in hand or object detection framework as far as we know. On the other hand, our approach is not a simple combination of the faster R-CNN, VAE, and GAN. Different from the traditional VAE model, our VAE model is asymmetric. During the detection, the shared information contained in the shared CNN and RPN layers can be effectively used by the asymmetric detection-VAE structure. Besides, it is

found that GAN [26] could further improve the performance of the model by generating more realistic hand appearances.

To evaluate the proposed approach, we compare our approach with existing state-of-the-art methods on public hand detection benchmarks, i.e., Oxford hand dataset [27] and EgoHand dataset [28]. Experimental results show that our approach achieves the highest detection accuracy among the state-of-the-arts.

It is noted that there exist some related works which utilize generative methods for detection purpose as well, but their contributions are very different from ours. In [29] VAE is used to reconstruct traffic trajectories so that anomalies can be detected from videos. In [30] GAN is used to super-resolve small objects so that traffic signs can be detected reliably, and in [31] hard positive examples are generated by GAN to train a classifier which recognizes anomaly hard examples better. However, we aim at the challenging multiple hands detection problem which is very different from the detection tasks mentioned above. In our approach, the detection and reconstruction branches share the same features, and the detection performance is improved by introducing the shared information contained in the hand appearances reconstruction branch.

The rest of the paper is organized as follows. In Section 2, related works about hand detection and image reconstruction are reviewed. Details of the proposed approach are provided in Section 3. Experimental results are presented in Section 4. The details of our motivation and asymmetric VAE are discussed in Section 5. A brief conclusion is given in Section 6.

2. Related Works

Robust human hand detection in an unconstrained complex environment is one of the most challenging tasks in computer vision. The existing hand detection methods can be classified into two categories as follows.

Traditional hand detection methods use artificially designed weak features, such as skin color and shape features. These hand detection algorithms occupy less computational resources. In order to solve the skin color diversity problem caused by human races and illumination change, Girondel et al. [32] tried several color spaces and found that *Cb* and *Cr* channels in *YCbCr* color space worked well in skin detection task. Sigal et al. [10] proposed the Gaussian mixture model that performed well in different illumination conditions. But, the skin color based method is susceptible to the background that has similar color of skin. There are also some approaches that use shape features to detect hands. Guo et al. [12] trained a SVM classifier based on HOG features for detection. Based on [12], Mittal et al. [27] developed a mixture of deformable parts based method to detect hand precisely. Additionally, Karlinsky et al. [33] proposed an approach to locate hands by detecting the relative positions between the hands and other human body parts. However, because of the multi-scale and various rotations of human hands in a single picture, it is difficult to train a model suitable for unconstrained complex environment.

CNN-based detection methods are becoming a popular research topic in computer vision field recently, because higher-level deep features can be learned from the networks. The multi-scale and various rotations problems mentioned above can be well addressed by using CNN. Recent research has focused on three principal directions on developing better object detection systems, and these principals are also suitable for CNN-based hand detection. The three principal directions are introduced as follows.

The first principal direction is to change the base architecture of these networks. The main idea is adopting state-of-the-art convolutional networks to extract robust features, and it not only can benefit the classification but also detect location. Some recent work include ResNet [34], ResNet-Inception [35], and FPN [36] for object detection. Le et al. [21] proposed to fuse the multi-scale feature map directly for detection and classification to avoid missing small hands. Qing et al. [37] proposed a feature-map-fused SSD that used deconvolutional to combine deep layers with shallow layers for hand detection.

However, the best detection precision in this direction is relatively low, i.e., 75.1% [21] on Oxford dataset. It is necessary to explore other directions to further improve the detection performance.

The second principal direction is to exploit the data itself by augmenting the diversity characteristics of the training data. Traditionally, the training data can be efficiently augmented by randomly generated spatial transformations [14,15], such as random translation, rotation, scaling and cropping, and it is a common technology widely used by the computer vision community. Besides, generative methods can also be utilized for data augmentation in hand-related applications. For example, in [38] GAN is adopted to augment training data for hand gesture recognition, and in [39,40] VAE and GAN are utilized to augment dataset for hand pose estimation. The generative methods can be employed to augment cropped image patches with single hand, but no previous work augments data using generative methods for hand detection task. The reason may be that, an unconstrainted scene usually contains multiple hands, and the contextual information among the hands and background is very hard to be modeled and randomized.

The third principal direction is to utilize contextual reasoning and proxy tasks for reasoning and other top-down mechanisms for improving representations of object detection. He et al. [16] proposed to use the segmentation as a way to contextually prime object detectors and provided feedback to initial layers. In [20,22,41], a hand rotation information was introduced to improve the detection precision. Deng et al. [20] adopted CNN to learn the rotation angles of hands, so that the directions of human hands were regulated by a spatial transformation network. Based on the idea of [20], Li et al. [22] proposed an embedded implementation of SSD (Single Shot Multi-Boxes) based hand detection and orientation estimation algorithm that can detect hand more efficiently. Narasimhaswamy et al. [41] introduced a contextual attention module for hand location and orientation estimation in unconstrained images. Most of the related works in this direction improve the representation for hand detection by utilizing the hand rotation information, and the best detection precision in this direction is 83.2% [22] on Oxford Dataset. However, we argue that the hand detection representation can be further improved by exploring more comprehensive information.

Our work follows this third principal direction. Hand appearances reconstruction is utilized to introduce shared information into our detection framework. Different from previous hand detection tasks, which only introduced rotation or orientation related information, reconstruction can deal with much more complex information of hand, such as scales, shapes, skin colors, and sometimes partial occlusions of hand.

To the best of our knowledge, there is no VAE based method for hand detection. VAE [24,42–44] can be used to reconstruct images effectively, but the VAE methods only minimized mean square error (MSE) between the reconstructed and original images, and the reconstructed image might look blurry. Therefore, we utilize GAN to improve the reconstructed image.

There are some related works that applying generative methods in detecting tasks, but their ideas are very different from ours. Kelathodi et al. [29] proposed a video object trajectory classification and anomaly detection method, in which VAE was used to reconstruct gradient conversion of trajectories and anomalies were detected by t-SNE. Li et al. [30] proposed perceptual generative adversarial networks by which small objects (e.g., traffic signs) were super-resolved to narrow the representation differences between objects of various scales. Wang et al. [31] utilized GAN to generate hard positive examples for training of a classifier which could recognize anomaly hard examples more accurately.

GAN based methods are also utilized in hand-related tasks such as abnormal dynamic gesture recognition [38] and gesture translation [45], but these works are very different from the image-based hand detection task which this work focuses on. For example, in [38] is addressed by dynamic hand gesture classification and abnormal hand gesture detection problemes using GAN, but their method was based on Electromyography (EMG) signals acquired from the forearm muscles (i.e., UC2018 DualMyo and UC2017 SG dataset). In [45] GAN was adopted to translate one hand gesture to another where only single hand appeared in per image. To the best of our knowledge, the GAN based method has not been used in hand detection applications previously.

3. Approach

In this work, we aim to train a model which can predict hand regions precisely and reconstruct the images of hand regions by using the RoI features. To achieve this, we present an end-to-end optimized detection and reconstruction framework, as shown in Figure 2. Our framework consists of two independent branches. One is the detection branch which is based on Faster R-CNN, and the other is the reconstruction branch which can reconstruct hand appearances in RoIs based on VAE. Both of these two branches share the same RoI features. Additionally, a GAN module is introduced to make the generated images look more realistic.

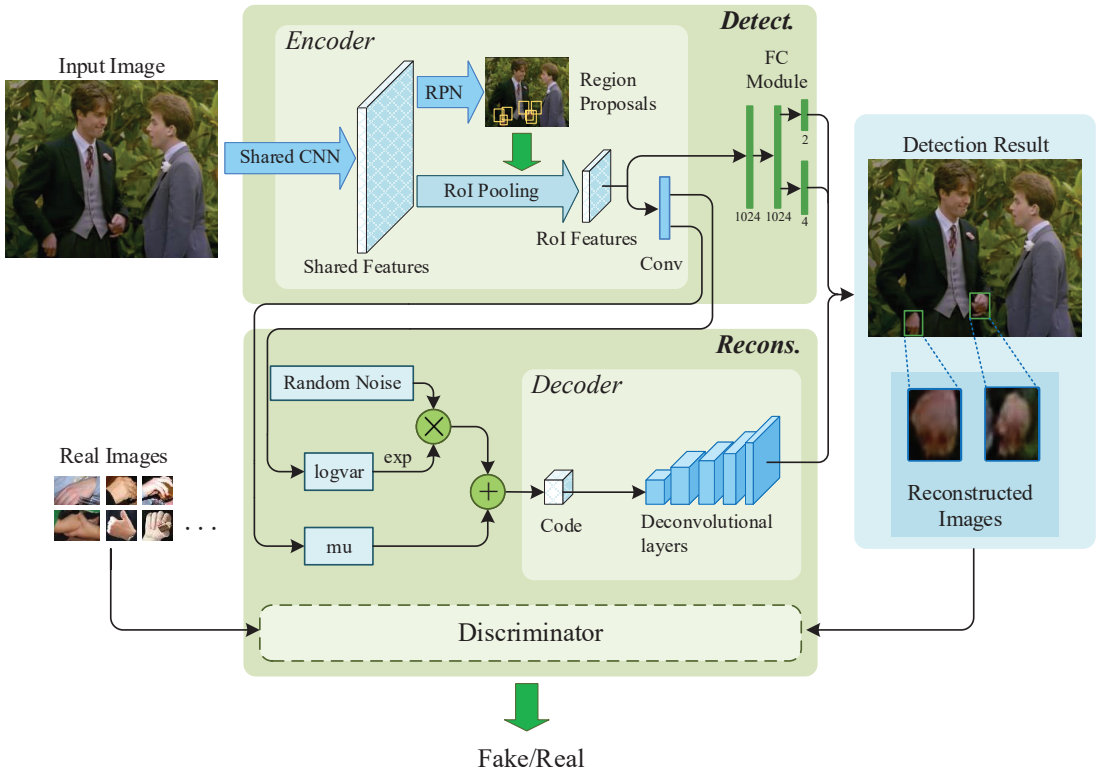

Figure 2. The framework of our model. The model consists of two branches, the detection branch and the reconstruction branch. Firstly, the input images are fed into shared convolutional neural networks (CNN) layers to compute shared features. Then, a region proposal network (RPN) is applied to generate region proposals, and pooling the feature of region proposal into the same shape. Finally, the region of interest (RoI) feature is used to detect hand and reconstruct hand appearance. Additionally, a discriminator is introduced to improve the reconstructed images.

3.1. Detection Branch

In this part we adopt the backbone of the Faster R-CNN for hand detection. The modules of this branch can be divided into three parts, as follows.

The shared convolutional module computes shared features for region proposals generation, detection refinement and reconstruction. In general object detection network, VGG [46], ResNet [34,35] etc. are usually used as its shared convolutional layers. As can be seen in Figure 2, shared features are calculated by the shared convolution module, and then the features are used for generating region proposals, refining the bounding box location, and reconstructing hand appearances tasks. We augment the shared features of [16] by using feature pyramid network (FPN) [36]. The structure of FPN involves a bottom-up pathway, a top-down pathway, and lateral connections. The bottom-up pathway is the feedforward computation of the ResNet-101. The top-down pathway hallucinates

higher resolution features by upsampling spatially coarser, but semantically stronger, feature maps from higher pyramid levels. Each lateral connection merges feature maps of the same spatial size from the bottom-up pathway and the top-down pathway.

The region proposal module generates region proposals. There are several methods, such as selective search [47], objectness [48] etc. However, these methods need to be trained independently, and are relatively inefficient. Therefore, we adopt RPN [15] which consists of 3×3 convolutional kernels to generate region proposals and can be integrated into an end-to-end training framework. Intersection-over-union (IoU) is usually employed to measure the overlap rate of two boxes, and the IoU is defined as

$$\mathrm{IoU} = \frac{\mathrm{Box}_1 \cap \mathrm{Box}_2}{\mathrm{Box}_1 \cup \mathrm{Box}_2}. \tag{1}$$

For each generated region proposal, we consider it as positive if the IoU between region proposal and ground truth is larger than 0.7, and as negative if the IoU is smaller than 0.3. Then the non-maximal suppression (NMS) [49] method is used to remove region proposals with higher IoU overlap rate. In order to pool the region proposals with different scales and aspect ratios into same size (in this work, the size of RoI feature is 7×7), we generate a regular grid of size 14×14 evenly cover the region proposal for sampling purpose. As the region proposals are with different scales and aspect ratios, the x, y coordinates on the grid are float but not integer. A bilinear interpolation algorithm is used to compute the exact value of the shared feature on the grid, and then a max-pooling with stride of 2 is used to aggregate the feature to 7×7.

Classification and refinement modules are used to classify the regions and to refine the detected boxes. By utilizing fully connection layers (details refer to FC module of Figure 2), the RoI features are mapped to two vectors. One vector is softmax probabilities of background and hand (two-dimensional), the other vector is bounding-box regression offsets (four-dimensional).

3.2. Reconstruction Branch

In this part, we introduce a reconstruction branch (refer to Figures 3 and 4) to reconstruct the hand appearances by using RoI features. Different from the traditional VAE model, in our model the structures of encoder and decoder are very different from each other (refer to Figure 3). Due to the input of our detection/reconstruction hybrid network is the whole image with arbitrary sizes, and the output of the reconstruction branch is image patches of uniform size (i.e., 32×32) that contain hands. To address the asymmetric of input and output, we design an asymmetric structure.

The encoder module is used to extract features of hand, in other words, encoding the image. It consists of two parts, shared convolutional module and Region proposal module which are described in the detection branch.

The decoder module is used to reconstruct hand appearances. Firstly, the RoI features are taken as the input of the convolutional layers with kernel size of 1×1 to calculate the mean vector $\boldsymbol{\mu}$ and the logarithmic standard deviation vector $\boldsymbol{\sigma}$ of the RoI features. Then a Gaussian distributed noise Φ which has the same dimension of $\boldsymbol{\mu}$ or $\boldsymbol{\sigma}$ is generated, and the latent vector $\boldsymbol{c}$ is calculated as follows

$$\boldsymbol{c} = \boldsymbol{\mu} + \frac{e^{\boldsymbol{\sigma}}}{2} \times \Phi. \tag{2}$$

This operation makes $\boldsymbol{c}$ follow a standard normal distribution roughly. Finally, $\boldsymbol{c}$ is fed into five deconvolutional layers and a sigmoid layer (refer to Figure 3) to generate hand appearance image.

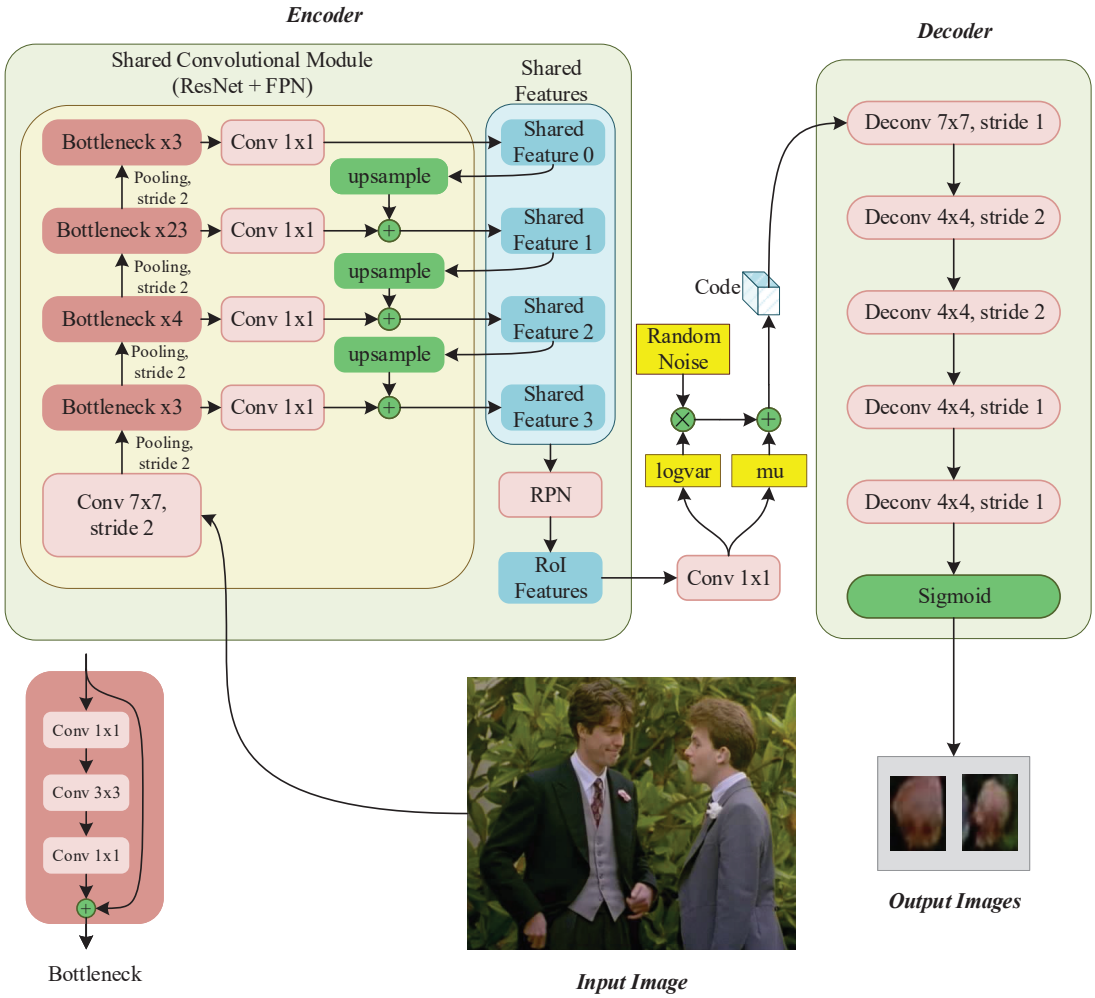

Figure 3. Asymmetric network structure of the encoder/decoder module.

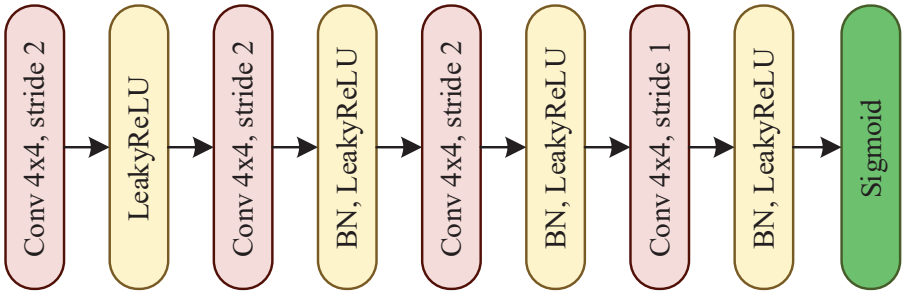

Figure 4. Network structure of the discriminator module.

The discriminator module discriminates whether the input image is fake or real. VAE based reconstruction model is trained by minimizing the element-wise error by default. The element-wise metric is simple but it does not model the properties of human visual perception, and the reconstruction of VAE tends to be blurry. An appealing property of GAN is that its discriminator network implicitly has to learn a rich similarity metric for images, so as to discriminate them from "unrealistic". Thus, we introduce the discriminator into the reconstruction branch, which can make it learns more similarity features of hand appearance. By utilizing GAN, local details of the reconstructed hand tend to be finer and sharper. Because better local details contribute positively to the localization of hand, our approach

with GAN demonstrates more accurate location prediction ability. In our scenario, the inputs of the discriminator module are the ground-truth hand patches cropped or reconstructed images outputted by the deconvolutional module. The discriminator module consists of four convolutional layers (refer to Figure 4). At the end of the module, a sigmoid function is applied to generate a probability which discriminates whether the input image is fake or real.

3.3. Loss Function

The loss of our proposed hybrid detection/reconstruction model is defined as

$$L = L_{detection} + \lambda L_{recons}, \tag{3}$$

where $L_{detection}$ is the loss of the detection branch, and L_{recons} is the loss of the reconstruction branch.

For the detection branch, we mainly refer to a faster R-CNN and its loss is defined as

$$L_{detection} = L_{rpn} + \gamma_1 L_{detreg} + \gamma_2 L_{cls}, \tag{4}$$

where L_{rpn} is the loss of RPN, L_{detreg} is the loss of detection regression, and L_{cls} is the loss of classification.

In the RPN module. At each pixel location on the shared features, we simultaneously predict K region proposals which are parameterized relative to K reference boxes, called anchors. Each anchor is centered at each pixel on the shared features, and is associated with scales and aspect ratios. Specifically, following [15], we use three scales and three aspect ratios, yielding $K = 9$ anchors at each pixel. For regression of region proposal boxes, we adopt parameterizations of four coordinates as follows:

$$t_x = \frac{x - x_a}{w_a},\ t_y = \frac{y - y_a}{h_a},\ t_w = \log\frac{w}{w_a},\ t_h = \log\frac{h}{h_a}, \tag{5}$$

$$t_x^* = \frac{x^* - x_a}{w_a},\ t_y^* = \frac{y^* - y_a}{h_a},\ t_w^* = \log\frac{w^*}{w_a},\ t_h^* = \log\frac{h^*}{h_a}, \tag{6}$$

where x, y, w and h denote the two coordinates of the box center, width, and height. Variables x^*, x_a and x are for the predict box, anchor box, and ground truth box respectively (likewise for y, w, h).

The anchor is matched with ground-truth firstly. It is counted as positive when IoU between the anchor and the ground-truth is larger than 0.7, and negative when the IoU is smaller than 0.3. For the positive anchor we set its label l as 1, and 0 for negative anchor. L_{rpn} contains two parts, proposal regression loss $L_{propreg}$ and object score loss L_{obj}.

The proposal regression loss $L_{propreg}$ represents the relative distance between the region proposal and ground-truth, and it is calculated as follow

$$L_{propreg}(\boldsymbol{t}, \boldsymbol{t}^*) = \frac{1}{N_D}\sum \begin{cases} 0.5|\boldsymbol{t} - \boldsymbol{t}^*|^2, & \text{if } |\boldsymbol{t} - \boldsymbol{t}^*| < 1 \\ |\boldsymbol{t} - \boldsymbol{t}^*| - 0.5, & \text{otherwise} \end{cases}, \tag{7}$$

where $\boldsymbol{t}$ is the parameter of the ground truth relative to the anchor offset which is calculated by using Equation (5). $\boldsymbol{t}^*$ is the hypotheses for region proposals relative to the anchor offset corresponded to $\boldsymbol{t}$. N_D is the number of positive region proposals.

The object score loss L_{obj} is log loss over two classes (background or object), it is defined as follow

$$L_{obj}(\boldsymbol{l}, \boldsymbol{l}^*) = -\frac{1}{M_D}\sum[\boldsymbol{l}^* \log(\boldsymbol{l}) + (1 - \boldsymbol{l}^*) \cdot \log(1 - \boldsymbol{l})], \tag{8}$$

where $\boldsymbol{l}$ is the label vector of anchors. $\boldsymbol{l}^*$ are the hypotheses for region proposals label vector corresponded to $\boldsymbol{l}$, and M_D is the sum of negative and positive region proposals.

Due to the numbers of the human hand in different images are different, so for each image, N_D and M_D may be different. The regression loss of bounding-box refinement and classification loss are calculated similar to $L_{propreg}$ and L_{obj} respectively.

For the reconstruction branch, we calculate the MSE loss between the real image and the reconstructed image as

$$L_{recons}(\boldsymbol{P}, \boldsymbol{P}^*) = \frac{1}{N_R} \sum |\boldsymbol{P} - \boldsymbol{P}^*|^2, \tag{9}$$

where $\boldsymbol{P}$ is the reconstructed image patch of cropped hand region, and $\boldsymbol{P}^*$ is real image patch of ground-truth. N_R is the number of reconstructed images.

Additionally, we also introduce the discriminator of GAN into our framework to further improve the performance. In this part, we need to distinguish whether the images which are fed into the discriminator are real or fake, so we adopt the cross-entropy loss function as

$$L(\boldsymbol{p}, \boldsymbol{p}^*) = -\frac{1}{M_R} \sum [\boldsymbol{p}^* \log(\boldsymbol{p}) + (1 - \boldsymbol{p}^*) \cdot \log(1 - \boldsymbol{p})], \tag{10}$$

where $\boldsymbol{p}^*$ is the label vector of input images (the real image is labeled as 1, and the fake as 0), and $\boldsymbol{p}$ is the probability vector output by the discriminator. M_R is the number of input images. During the training, the loss of GAN is calculated as follows

$$L_D = -\frac{1}{M_{R_1}} \sum (\log(D(\boldsymbol{real})) + \log(1 - D(\boldsymbol{fake}))), \tag{11}$$

$$L_G = -\frac{1}{M_{R_2}} \sum \log(D(G(\boldsymbol{r}))), \tag{12}$$

where, D is the discriminator, G is the re-constructor, ***real*** is the real image of ground truth, ***fake*** is the reconstructed image, $\boldsymbol{r}$ is the feature output by region proposal network, M_{R_1} is the number of all input images which include the real and reconstructed images, and M_{R_2} is the number of reconstructed images. The training of GAN is split up into two main steps. In the first step, we train the discriminator to maximize the probability of correctly classifying a given input as real or fake. We assign the label of real image cropped on original image as 1 and the label of fake image generated by the reconstruction branch as 0. Then the real and fake images are fed into the discriminator to minimize the loss of discriminator L_D, and only the weights of discriminator are updated. In the second step, we minimize the loss L_G by updating the weights of the whole network except the discriminator.

4. Experiments

In the following subsections, experiments will be conducted on the Oxford Hand Dataset and EgoHand Dataset to evaluate the effectiveness of our proposed approach. Our approach has two variations: (1) ours without GAN denotes to detect hand and reconstruct hand appearances without GAN; (2) ours with GAN denotes to detect hand and reconstruct hand appearances with GAN.

For each experiment, training samples are augmented by image processing operations such as resizing, shifting, flipping images, etc. which can further increase the number of training data, and prevent the over-fitting. In terms of training model, we adopt stochastic gradient descent (SGD) algorithm with a weight decay of 0.0005 and a momentum of 0.9 to update the weight of the model. The initial learning rate is 0.005, and then it will be multiplied by 0.1 per 3 epoch. The total training epoch is 10. All experiments are performed on a workstation with an Intel iCore 7 CPU, 32G RAM and a GTX 1080Ti GPU with 11G Memory capacity, and the program of our proposed approach is developed under the PyTorch deep learning framework.

4.1. Oxford Hand Dataset

This dataset [27] is a comprehensive images dataset of hands which are collected from various different public image dataset sources. In each image, all the hand instances that can be perceived clearly by humans are annotated. For the whole dataset, there are 13,050 hand instances. In the training set, there are 11,019 hand instances, and for the testing set, there are 2031 hand instances.

On the Oxford Hand Dataset, the proposed approach is compared with [13,15,16,20–22,27,41]. We adopt the evaluation metric proposed in [20], which uses the typical average precision when the threshold of IoU is 0.5 (AP_{50}).

The AP_{50} of the state-of-the-art methods on the Oxford Hand Dataset are illustrated in Table 1, and their corresponding precision-recall curves are shown in Figure 5. We compared the results of our approach on the Oxford Hand Dataset with that of the state-of-the-art methods. When the threshold of IoU is 0.5, the average precision of our proposed approach without GAN (AP_{50} = 87.0%) is higher than that of other methods. Our proposed approach with GAN (AP_{50} = 87.6%) improves the average precision of the state-of-the-art method proposed in [22] by 4.4 points.

Table 1. Average precision on the Oxford Hand Dataset when the threshold of IoU is 0.5.

Model	**AP_{50}**
R-CNN [13]	42.3
Multi Proposals with NMS [27]	48.2
Faster R-CNN [15]	55.7
Rotation Estimation [20]	58.1
Mask R-CNN [16]	70.5
MS-RFCN [21]	75.1
Hand-CNN [41]	78.8
SSD-Hand [22]	83.2
Ours without GAN	87.0
Ours with GAN	**87.6**

To further investigate the performance of our proposed approach, we conducted several experiments by using following variations: (1) baseline 1 [16] is the backbone of our approach, which to detect hand only without reconstruction, (2) baseline 2 augments *Baseline 1* by utilizing multi-scale feature maps [36], (3) ours without GAN and (4) ours with GAN mentioned above. We employ recently widely used evaluation metric [50], which includes the mean average precision at the threshold of IoU in the range from 0.5 to 0.95 (mAP), average precisions when the thresholds of IoU are 0.5 and 0.75 respectively (AP_{50} and AP_{75}). The evaluation results are shown in Table 2. The mean average precision of our model without GAN (mAP = 44.0%) improves 8.8% than the baseline 2 (mAP = 35.2%). It indicates that to reconstruct the hand appearances is helpful to improve the detection performance. As for the GAN, AP_{75} of our method with GAN is 42.0% which is 4.1 points higher than ours without GAN (AP_{75} = 37.9%). It indicates that the GAN could improve the IoU between ground truth and predicted bounding-box. In other words, location prediction ability is improved by the GAN. Furthermore, the mAP of our method with GAN is 46.2% which is 2.2 point higher than ours without GAN (mAP = 44.0%). It shows that the comprehensive detection capability of ours with GAN is better than ours without GAN.

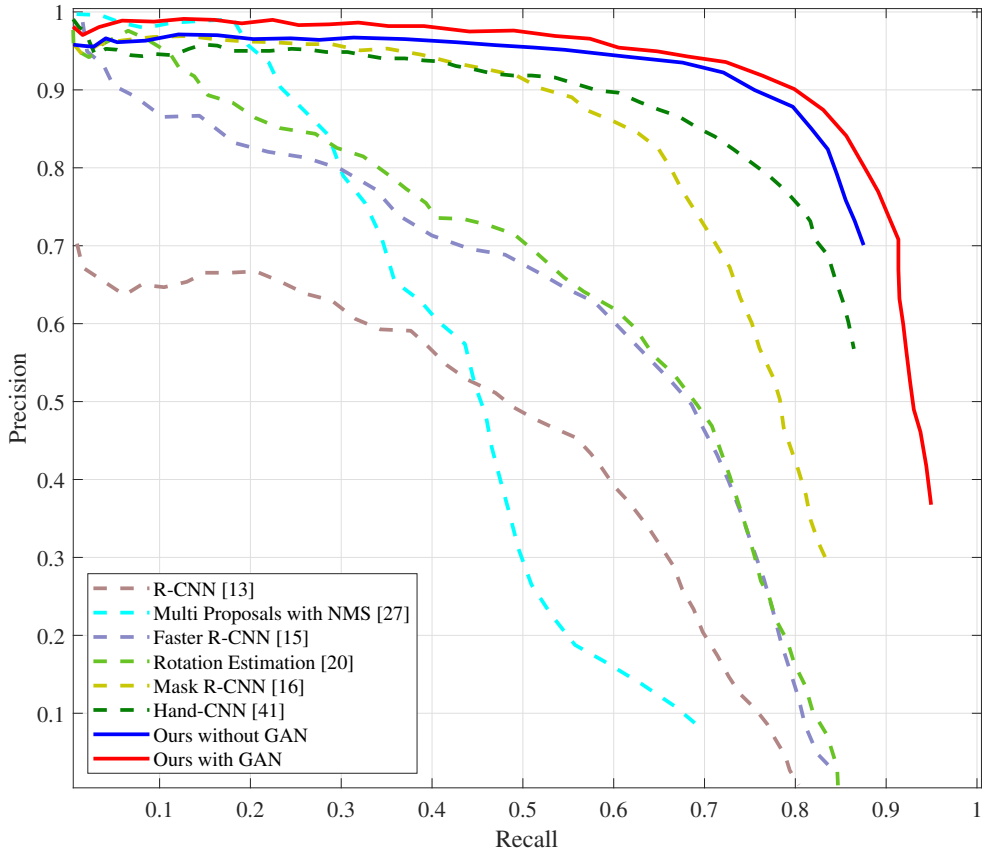

Figure 5. Precision-recall curves on the Oxford Hand Dataset. The curves of MS-FRCNN (Multiple Scale Region-based Fully Convolutional Networks) and SSD-hand are not shown because they are not provided in [21,22] (This figure is best viewed in color).

Table 2. Further comparison between our proposed approach and baseline.

Model	mAP	AP_{50}	AP_{75}
Baseline 1	31.5	70.5	22.1
Baseline 2	35.2	78.1	25.3
Ours without GAN	44.0	87.0	37.9
Ours with GAN	**46.2**	**87.6**	**42.0**

In [20], the authors also measured the localization precision by using mean average best overlap (MABO) proposed in [47] to evaluate the localization ability of RPN, and determine the number of region proposals generated by RPN. Because the number region proposals is closely related to the efficiency of the network, in other words, the more region proposals generated, the more computing resources taken. In our work, we adopt the same RPN structure which is proposed in [14]. As shown in Figure 6, the MABOs of RPN in "ours without GAN" and "ours with GAN" converge to 79.4% and 79.7% respectively when the number of windows (region proposals) reaches 300. The RPN in [20], and its MABO converges to 79.2% when the number of windows reaches 300. Note that, [20] proposed a new RPN named "deep feature proposal" which achieves 100% recall when the number of windows is 17,489 (please refer to Table 3), but its MABO converges to 76.1% which is lower than that of the standard RPN. Comparing with the standard RPN, the recall of RPN in our approach is also improved.

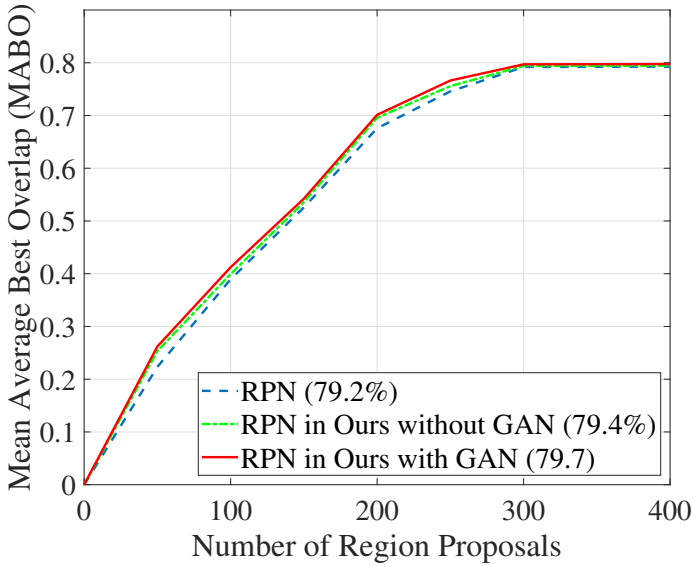

Figure 6. Trade-off between mean average best overlap and the number of region proposals.

Table 3. Region proposals network performance on the Oxford Hand test data.

Model	Recall	MABO	#Proposals
Selective Search [47]	46.1%	41.9%	13,771
Objectness [48]	90.2%	61.6%	13,000
Deep Feature Proposal [20]	100%	76.1%	17,489
RPN [20]	95.9%	79.2%	300
RPN in Ours without GAN	96.8%	79.4%	300
RPN in Ours with GAN	98.7%	79.7%	300

Figure 7 shows some detection examples on Oxford Hand Dataset. Meanwhile, we also present the reconstructed images of hand appearances which are demonstrated in the blue boxes. Besides, from Figure 8, we can see that reconstructed images generated by VAE without GAN are blurry. By utilizing GAN, local details of the reconstructed hand tend to be finer and sharper. Although our model can precisely detect hands, there are still some false detections or missing detections. For example, in Figure 9a, the appearance of the baby's foot is very similar to that of human hand in terms of color and shape, so there exists a certain probability that a human foot would be falsely detected as a human hand. As can be seen in the bottom right corner of Figure 9b, when two hands are spatially close to each other, they may be falsely detected as a single hand. In the top left corner of Figure 9c, when the human hand is heavily occluded, the scale is too small and the appearance is too dark, it may cause missing detection. Additionally, "false detection" may be caused by false labels in dataset. For example, in the center of Figure 9d, correctly detected hand maybe considered as a "false" detection due to the mislabel of this hand in the dataset.

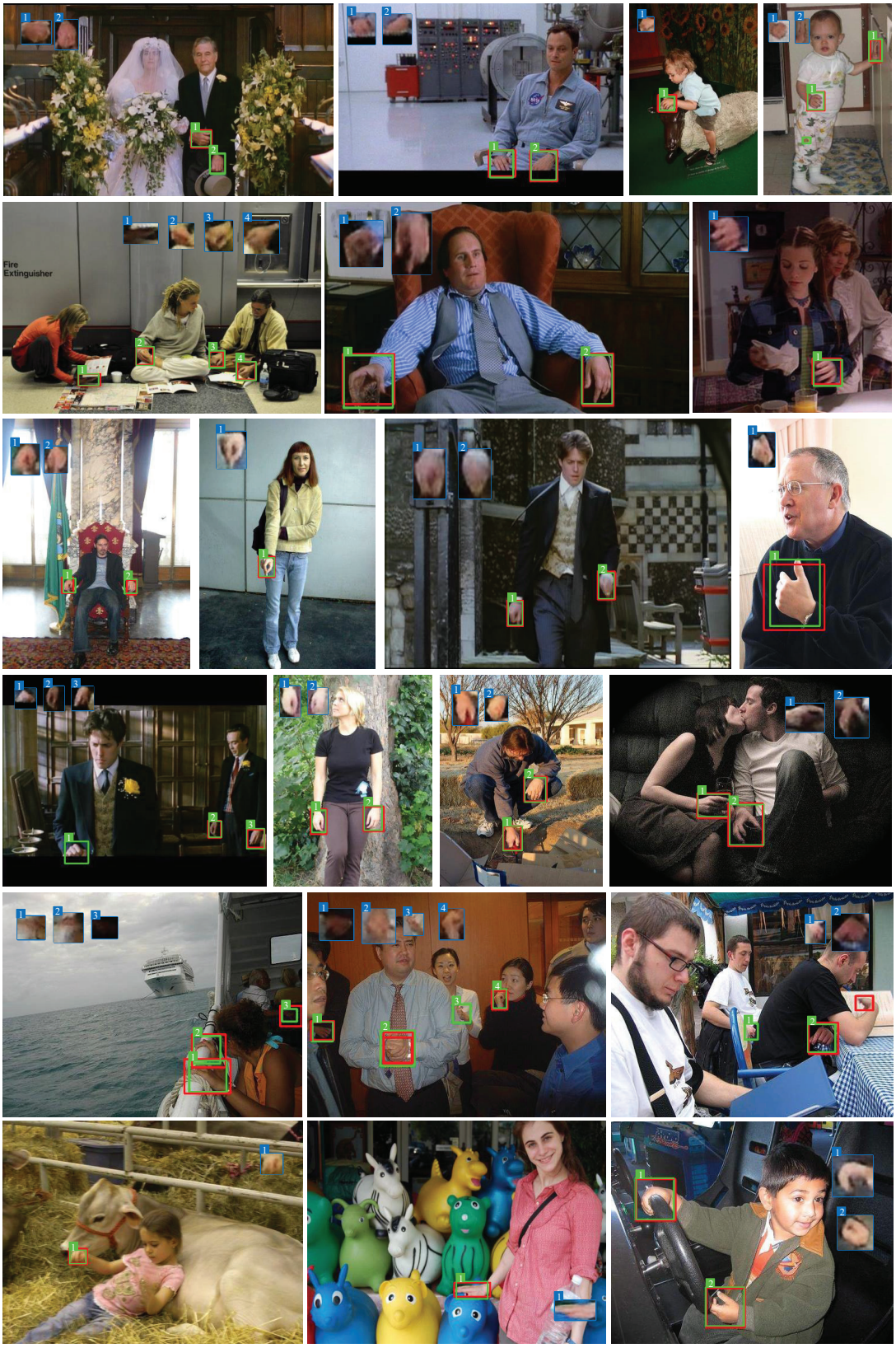

Figure 7. The detection results of our approach (ours with GAN) on the Oxford Hand Dataset. The red boxes denote the ground-truth of hands; the green boxes denote the predicted boxes output by our model; the blue boxes denote the reconstructed images of corresponding hand regions. Multiple hands are annotated with serial numbers on top left corners of boxes. (This figure is best viewed in color).

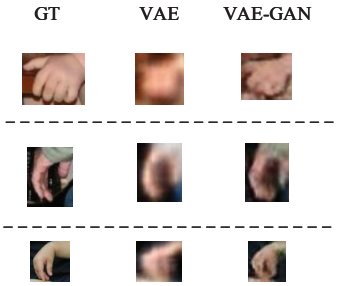

Figure 8. Comparison of reconstructed hand appearances between ours without GAN and ours with GAN. Each row corresponds to a sample. The first column denotes the ground truth, the second column denotes reconstructed images by ours without GAN, and the third column denotes reconstructed images by Ours with GAN.

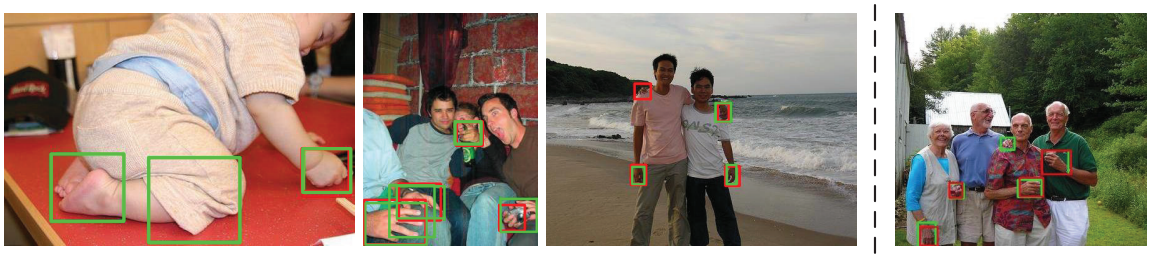

Figure 9. The failure case of our detection approach on the Oxford Hand Dataset. The red boxes denote the ground-truth of hands; the green boxes denote the predicted boxes output by our model. (This figure is best viewed in color).

The running time of our approach is compared with that of the previous state-of-the-art methods [15,20–22,27] in Table 4. Our approach is very efficient, because it takes about 0.1 s per image. Ours spends less time than most of the state-of-the-arts except [15,22], because we introduce the additional reconstruction branch which requires slightly more computational time but improves the detection precision significantly. In these works, almost all computing is done by GPU except [27] and our approach is primarily compared with the GPU based method. In our works, our GPU is GTX 1080 Ti with 3584 CUDA core, and 11 GB memory capacity, and the other works used TITAN X with 3072 CUDA core and 12GB memory capacity. Since we do not have a TITAN X GPU, we cannot test our model on it. Assuming that the computing speed is proportional to the number of CUDA cores, the speed of ours could be about 0.1297 s/frame on TITAN X GPU. Thus, ours is faster than [20,21], but slower than [15] and [22]. In [21], it uses a more complex feature extraction network which slows down its detection speed. In [20], due to the rotation estimation and the derotation modules, it is less efficient than ours. What is more, our approach is almost real-time.

Table 4. Average time to detect hands in the testing stage.

Model	Time	Test Environment	Framework
Multi proposals with NMS [27]	120 s	2.50 GHz CPU	Caffe
Faster R-CNN [15]	0.08 s	2.9 GHz CPU and Titan X	Caffe
RPN [20]	0.1 s	2.9 GHz CPU and Titan X	Caffe
Rotation estimation [20]	1.0 s	2.9 GHz CPU and Titan X	Caffe
MS-RFCN [21]	0.2150 s	3.5 GHz CPU and Titan X	Caffe
SSD-Hand [22]	0.0072 s	3.0 GHz CPU and Titan X	PyTorch
Ours without GAN	0.1112 s	3.5 GHz CPU and GTX1080Ti	PyTorch
Ours with GAN	0.1121 s	3.5 GHz CPU and GTX1080Ti	PyTorch

4.2. EgoHand Dataset

This dataset [28] contains 48 Google Glass videos of complex and first-person interactions, such as, playing cards, chess, puzzle and Jenga, between two people in different locations. There are 130,000 frames of videos in total, of which 4800 frames have been labeled, and 15,053 hand instances have been annotated.

In order to prove the hand detection ability of our approaches proposed in this work, we also evaluate our model on the EgoHand Dataset by calculating the average precision when the threshold of IoU is 0.5. The precision-recall curves are shown in the Figure 10, and the corresponding AP_{50} is illustrated in Table 5. The average precision of our proposed approach with GAN improves the highest score in literature by 10 points when the threshold of IoU is 0.5. Figure 11 shows some detection examples on Egohand Dataset. Besides, we also present the reconstructed images of hand regions which are demonstrated in the blue boxes.

In additionally, we also refer to the training method that is used in [37]. First, the model was pre-trained using the Oxford dataset. Then the EgoHand dataset is used to fine-tune the model. The model is evaluated by using evaluation metric proposed in [50]. The experimental results are shown in Table 6. It can be clearly seen that the pre-training and fine-tuning training method improves the performance of the model on the EgoHand dataset. The AP_{75} of ours with GAN improved 2.3% points.

Furthermore, to evaluate the performance of our approach in real-life environment, we test the model on color images captured by an off-the-shelf digital camera. As can be seen in Figure 12, our approach can reliably and precisely detect multiple human hands in unconstrained real-life environment.

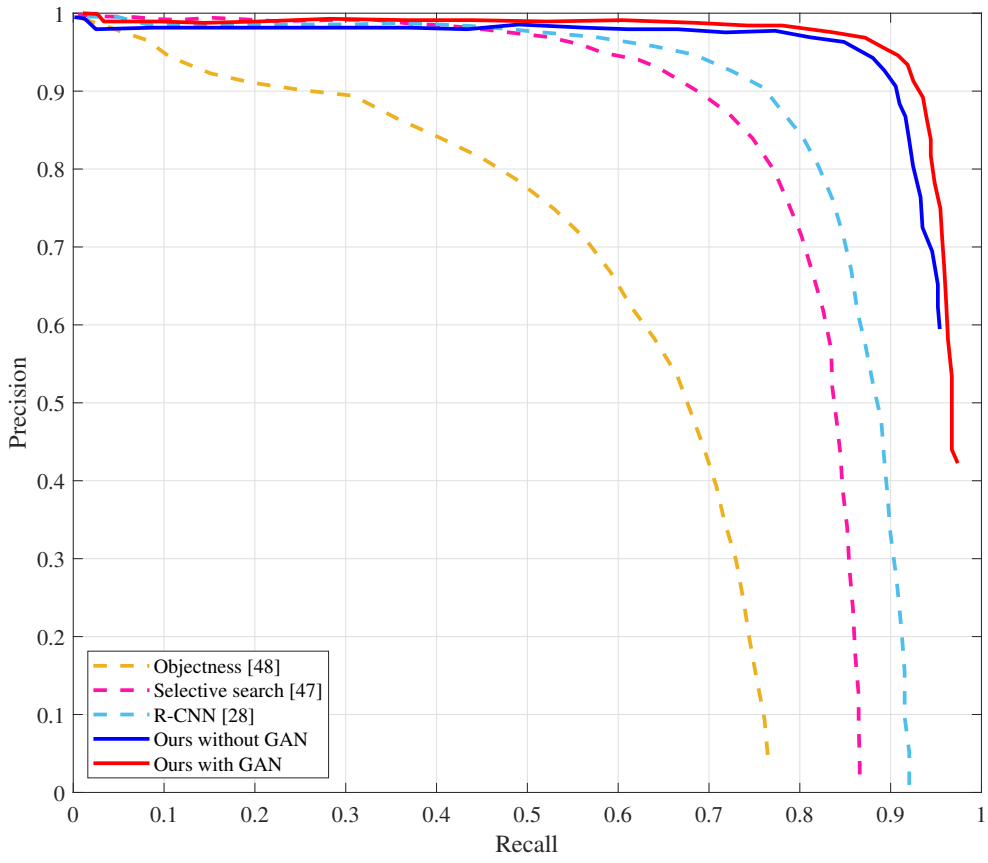

Figure 10. Precision-recall curves on the EgoHand Dataset. (This figure is best viewed in color).

Table 5. Average precision on the EgoHand Hand Dataset when the threshold of IoU is 0.5.

Model	AP_{50}
Objectness [48]	56.8
Selective search [47]	72.9
R-CNN [28]	84.2
Ours without GAN	93.3
Ours with GAN	**94.2**

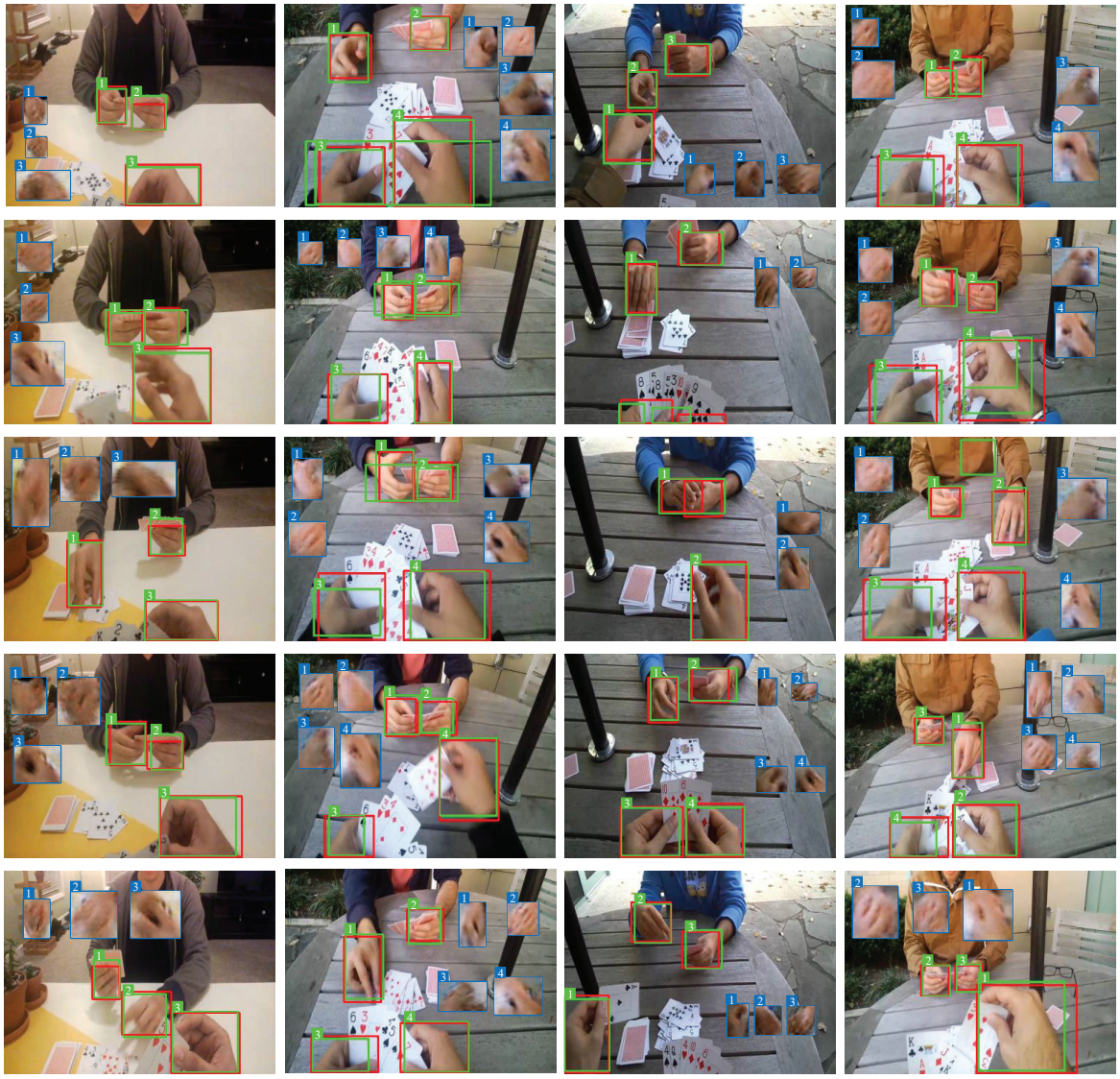

Figure 11. The detection results of our approach (ours with GAN) on the Egohand Dataset. The red boxes denote the ground-truth of hands; the green boxes denote the predicted boxes output by our model; the blue boxes denote the reconstructed images of corresponding hand regions. Multiple hands are annotated with serial numbers on top left corners of boxes. However, there are still some samples in which hands cannot be detected properly. For example, when two hands are very close with each other, the method would detect them as one hand. (This figure is best viewed in color).

Table 6. Comparison between different datasets.

Dataset	Model	mAP	AP_{50}	AP_{75}
EgoHand	Ours without GAN	58.4	93.3	68.3
EgoHand	Ours with GAN	60.2	94.2	70.4
Oxford + EgoHand	Ours without GAN	59.1	93.6	69.0
Oxford + EgoHand	Ours with GAN	61.9	94.4	72.7

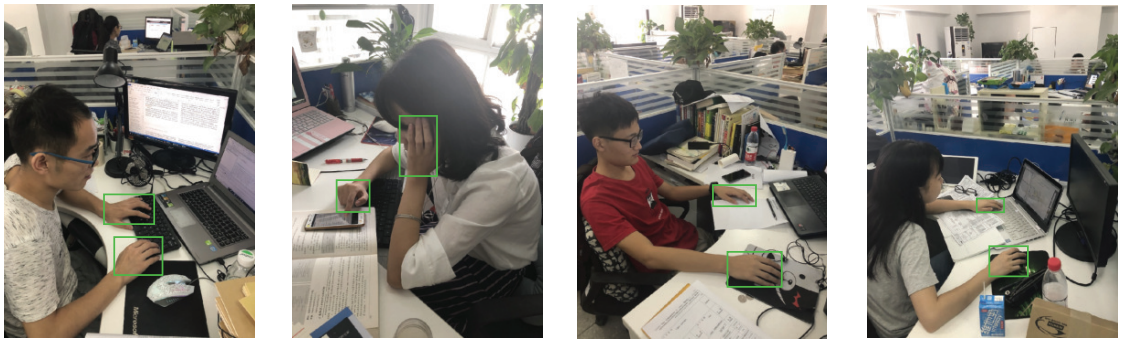

Figure 12. The detection results of our approach in the unconstrained real-life environment. The green boxes denote the predicted boxes output by our model. (This figure is best viewed in color).

5. Discussion

5.1. Motivation of Our Approach

Reliably detecting multiple hands from cluttering scenes is a challenging task because of complex appearance diversities of dexterous human hands in color images, for example, different hand shapes, highly articulated hand poses, skin colors, illuminations, orientations and scales etc. It is impossible to collect training samples evenly cover all these types of hand appearance diversities, and false labels are inevitable. To better understand the motivation of our approach, the details are discussed in the following aspects.

Detection without reconstruction: in this case, the optimizer pursuits merely minimizing the detection loss, and the network learns a model which best fits a specific training dataset. As human hand appearances are very complex in unconstrainted scenarios, the unbalance distribution of training samples, the bias, and false labels of specific training datasets would lead to overfitting and performance degeneration. Detection–only scheme may ignore important hand appearance features which could help generalization.

Detection and reconstruction with VAE: in this case, the optimizer pursuits not only maximizing the detection score, but also reconstructing the hand appearances as much as possible. The learned RoI features contain specific information for hand detection task, as well as comprehensive information for hand appearance reconstruction task. The detection/reconstruction scheme encourages the detector to classify labels and to regress locations based on learned RoI features with comprehensive hand appearance information, but not merely based on features for detection task only. Furthermore, the VAE enforces the learned RoI features (or to say "code") to a normalized gaussian distribution, and this characteristic helps to alleviate the bias and overfitting problem.

Detection and reconstruction with VAE and GAN: an appealing property of GAN is that its discriminator network implicitly learns a rich similarity metric for appearance construction. By utilizing GAN, local details of the reconstructed hand tend to be finer and sharper. As better local details contribute positively to the localization of hand, our approach with GAN demonstrates more accurate location prediction ability.

5.2. Asymmetric VAE

The asymmetry of our VAE contains two aspects: (1) The input and output are asymmetric. (2) The structure of encoder/decoder is asymmetric.

The input and output are asymmetric: In traditional VAE models, one input image corresponds to one output image with the same size, and the contents of the images are the same. Whereas, in our VAE model, one input image corresponds to multiple output images. The input is the whole image with arbitrary size, and the outputs are image patches of uniform size (i.e., 32×32). The content of the input is unconstrained scenario (multiple hands and complex background), and the content of each output is the hand appearance corresponding to each detected region. Our VAE network is different from applying traditional symmetric VAE on the input image with a sliding-window fashion. In sliding-window-based methods, the regions are scanned evenly with constant step. But in our model, the region proposals are generated unevenly, and ratios of the region proposals vary from each other. The proposed method is different from applying traditional symmetric VAE on image patches cropped from detected regions. We do not extract features from cropped image patches, but based on the shared features, a bilinear interpolation algorithm is used to aggregate the features in the region proposal.

The structure of the encoder/decoder is asymmetric: the structure of the encoder and that of the decoder are very different from each other, which is asymmetric too. The encoder contains a shared convolutional module and a region proposal module. Whereas, the decoder only contains five deconvolutional layers and a sigmoid layer (refer to Figure 3 decoder module). For details, refer to Sections 3.1 and 3.2.

6. Conclusions

In this paper we study the hand detection in unconstrained cluttering environment which is a challenging task because of complex appearance diversities of human hands. We propose a new hybrid detection/reconstruction convolutional neural network to reliably detect multiple hands as well as to reconstruct hand appearances. The main contribution of this paper is to improve the precision of hand detection by introducing the hand appearance reconstruction branch. What's more, the detection precision can be further improved by introducing GAN to generate more realistic hand patches. The proposed approach is evaluated and analyzed on two widely-used hand detection benchmarks, i.e., Oxford Hand and EgoHand Datasets. Extensive experiments demonstrate the superiority of our hybrid detection/reconstruction framework for hand detection.

Although our proposed hybrid detection/reconstruction convolutional neural network can effectively detect human hand from unconstrained environments with high average precision, there are also some defects as illustrated in Figure 9. Besides it cannot distinguish right hand from left hand. So, it is important to deal with these issues in the future.

Author Contributions: C.X. conceived this study and designed the method; W.C. developed the program, performed the experiments, and wrote first draft of this manuscript; Y.L. helped to develop the program used in those experiments; J.Z. helped to perform the experiments; L.W. reviewed and edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grants NO.61876170, the National Natural Science Fund Youth Science Fund of China under Grant NO.51805168, the National Natural Science Fund Youth Science Fund of China under Grant NO.61603357, the Fundamental Research Funds for Central Universities, China University of Geosciences NO.CUG170692, and the R&D project of CRRC Zhuzhou Locomotive Co., LTD. No. 2018GY121.

Acknowledgments: The authors would like to express their gratitude to the T.Z., Z.Z. and J.W. from Hubei Key Laboratory of Advanced Control and School of Automation, China University of Geosciences.

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

References

1. Li, C.; Kitani, K. M. Pixel-level hand detection in ego-centric videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, Oregon, OR, USA, 23–28 June 2013; pp. 3570–3577.
2. Paravati, G.; Gatteschi, V. Human-Computer Interaction in Smart Environments. *Sensors* **2015**, *15*, 19487–19494, doi:10.3390/s150819487.
3. Meena, Y. K.; Cecotti, H.; Wong-Lin, K.; Dutta A.; Prasad, G. Toward optimization of gaze-controlled human–computer interaction: Application to hindi virtual keyboard for stroke patients. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2018**, *26*, 911–922, doi:10.1109/TNSRE.2018.2814826.
4. Xu, C.; Govindarajan, L. N.; Zhang, Y.; Cheng, L. Lie-X: Depth Image Based Articulated Object Pose Estimation, Tracking, and Action Recognition on Lie Groups. *Int. J. Comput. Vis.* **2017**, *123*, 454–478, doi:10.1007/s11263-017-0998-6.
5. Xu, C.; Cheng, L. Efficient Hand Pose Estimation from a Single Depth Image. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 3456–3462.
6. Ge, L.; Ren, Z.; Li, Y.; Xue, Z.; Wang, Y.; Cai, J.; Yuan, J. 3D Hand shape and pose estimation from a single RGB image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 10833–10842.
7. Lin, H.; Hsu, M.; Chen, W. Human hand gesture recognition using a convolution neural network. In Proceedings of the IEEE International Conference on Automation Science and Engineering, Taipei, Taiwan, 18–22 August 2014; pp. 1038–1043.
8. Kirishima, T.; Sato, K.; Chihara, K. Real-time gesture recognition by learning and selective control of visual interest points. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 351–364, doi:10.1109/TPAMI.2005.61.
9. Shahroudy, A.; Liu, J.; Ng, T. T.; Wang, G. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019.
10. 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, doi:10.1109/TPAMI.2004.35.
11. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–26 June 2005; pp. 886–893.
12. Guo, J.; Cheng, J.; Pang, J.; Guo, Y. Real-time hand detection based on multi-stage HOG-SVM classifier. In Proceedings of the IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013; pp. 4108–4111.
13. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June; pp. 580–587.
14. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 13–16 December 2015; pp. 1440–1448.
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montréal, QC, Canada, 7–12 December 2015; pp. 91–99.
16. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
17. 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.
18. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C. Y.; Berg, A. C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
19. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
20. 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.* **2017**, *27*, 1888–1900, doi:10.1109/TIP.2017.2779600.

21. 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; pp. 1203–1210.
22. 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**, 1–12, doi:10.1007/s00138-019-01038-4.
23. Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28*, 41–75, doi:10.1023/A:1007379606734.
24. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2014.
25. Larsen, A. B. L.; Sønderby, S. K.; Larochelle, H.; Winther, O. Autoencoding beyond pixels using a learned similarity metric. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; pp. 1558–1566
26. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances in Neural Information Processing Systems, Montréal, QC, Canada, 8–13 December 2014; pp. 2672–2680.
27. Mittal, A.; Zisserman, A.; Torr, P. H. Hand detection using multiple proposals. In Proceedings of the British Machine Vision Conference, University of Dundee, Dundee, UK, 29 August–2 September 2011; pp. 1–11.
28. Bambach, S.; Lee, S.; Crandall, D. J.; Yu, C. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 13–16 December 2015; pp. 1949–1957.
29. Kumaran, S.K.; Dogra, D.P.; Roy, P.P.; Mitra, A. Video Trajectory Classification and Anomaly Detection using Hybrid CNN-VAE. Available online: https://arxiv.org/pdf/1812.07203.pdf (accessed on 18 December 2018).
30. Li, J.; Liang, X.; Wei, Y.; Xu, T.; Feng, J.; Yan, S. Perceptual generative adversarial networks for small object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 1222–1230.
31. Wang, X.; Shrivastava, A.; Gupta, A. A-Fast-RCNN: Hard Positive Generation via Adversary for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 3039–3048.
32. Girondel, V.; Bonnaud, L.; Caplier, A. A human body analysis system. *EURASIP J. Adv. Signal Proc.* **2006**, *2006*, 061927, doi:10.1155/ASP/2006/61927.
33. Karlinsky, L.; Dinerstein, M.; Harari, D.; Ullman, S. The chains model for detecting parts by their context. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 25–32.
34. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
35. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284.
36. Lin, T. Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 2117–2125.
37. Qing G.; Jinguo L.; Zhaojie J. Robust real-time hand detection and localization for space human-robot interaction based on deep learning. *Neurocomputing* **2019**, doi:10.1016/j.neucom.2019.02.066.
38. Miguel S.; Pedro N.; Olivier G. Improving novelty detection with generative adversarial networks on hand gesture data. *Neurocomputing* **2019**, *358*, 437–445, doi:10.1016/j.neucom.2019.05.064.
39. He, W.; Xie, Z.; Li, Y.; Wang, X.; Cai, W. Synthesizing Depth Hand Images with GANs and Style Transfer for Hand Pose Estimation. *Sensors* **2019**, *19*, 2919, doi:10.3390/s19132919.
40. Wan, C.; Probst, T.; Van Gool, L.; Yao, A. Crossing nets: Dual generative models with a shared latent space for hand pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 22–25 July 2017; pp. 3642–3649.
41. Narasimhaswamy, S.; Wei, Z.; Wang, Y.; Zhang, J.; Hoai, M. Contextual Attention for Hand Detection in the Wild. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019.

42. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. beta-VAE: Learning basic visual concepts with a constrained variational framework. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017; p. 6.
43. Van Den Aäron, O.; Nal, K.; Koray, K. Pixel recurrent neural networks. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 1747–1756.
44. Van Den Aäron, O.; Nal, K.; Oriol V.; Lasse, E.; Alex, G.; Koray, K. Conditional image generation with PixelCNN decoders. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 4790–4798.
45. Tang, H.; Wang, W.; Xu, D.; Yan, Y.; Sebe, N. Gesturegan for Hand Gesture-to-Gesture Translation in the Wild. In Proceedings of the 26th ACM International Conference on Multimedia, Seoul, Korea, 22–26 October 2018; pp. 774–782.
46. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations, Banff, AB, Canada, 14–16 April 2015.
47. Uijlings, J. R.; Van De Sande, K. E.; Gevers, T.; Smeulders, A. W. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171, doi:10.1007/s11263-013-0620-5.
48. Alexe, B.; Deselaers, T.; Ferrari, V. Measuring the objectness of image windows. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2189–2202, doi:10.1109/TPAMI.2012.28.
49. Neubeck, A.; Van Gool, L.; Efficient non-maximum suppression. In Proceedings of the International Conference on Pattern Recognition, Hong Kong, China, 20–24 August 2006; pp. 850–855.
50. Lin, T.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C. L. Microsoft COCO: Common Objects in Context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.