Improved Loss Functions for SSD Object Detection Model

Yixuan Lin
Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
yixuanli@andrew.cmu.edu

Yannan Chen
Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
yannanc@andrew.cmu.edu

Chujun Ni
Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
chujunn@andrew.cmu.edu

Abstract

SSD is a state-of-the-art one-stage detection approach, which shows fundamental increase in speed and keeps accurate at the same time. We propose to modify SSD algorithm by enhancing the cluster of intra class features therefore improving discrimination between classes. Center loss has been proved useful in face recognition problem because of its discriminative power of the deeply learned feature. By combining the center loss with the softmax loss, the minimum degree of changes can be achieved within the class while maintaining the distinguish ability between classes. The proposed method applys center loss in object detection under SSD model in order to enhance the classification robustness. In experiment, the proposed modified model is trained and tested on VOC2007 and compared with original SSD model, which yields reasonable results.

1 Introduction

As a single-shot detector for multiple categories, SSD[1] gives objects location and classification directly at one stage, which shows fundamental increase in speed and keeps accurate at the same time. SSD algorithm combines the regression method of YOLO[2] and the Anchor method of Faster RCNN. SSD is faster than YOLO, and as accurate as slower techniques that perform explicit region proposals and pooling (including Faster R-CNN). Similar to the Anchor approach of Faster RCNN, SSD uses a series of default bounding boxes as the proposed regions, which are matched with ground truth boxes during training.

The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections. Here we use the VGG-16 network as a base and truncate it before any classification layers. Then convolutional feature layers are added to the end of the truncated base network. These layers decrease in size progressively and allow predictions of detections at multiple scales by using a set of convolutional filters[1]. These are indicated on top of the SSD network architecture in Figure 1.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

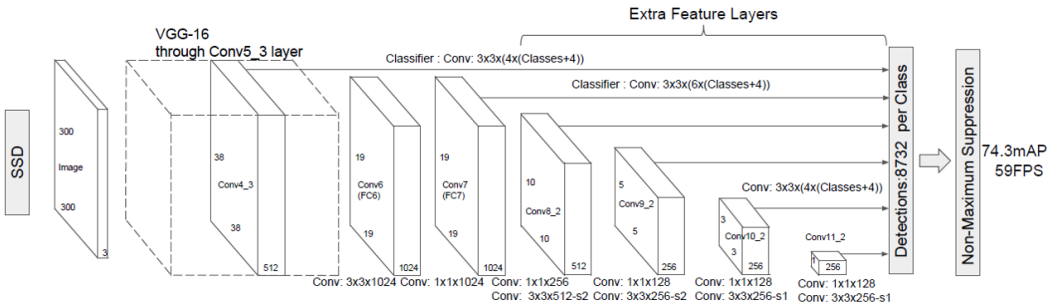


Figure 1: SSD network architecture[1]

The objective loss function of SSD is a weighted summation of the confidence loss and the localization loss, which uses softmax Loss and Smooth L1 Loss of Fast R-CNN respectively. However, features can be distinguished by softmax loss but not sufficiently distinguishable.

In order to obtain robust object detection, the discriminative power of the deeply learned feature should be strong. As a result, we plan to apply center loss function to the classification subnet to improve the performance of the SSD object detection model.

2 Related work

The objective loss function of SSD is a weighted summation of softmax Loss and Smooth L1 Loss. As people continue to explore, many improvements have occurred on these two loss functions to yield better model results.

Yandong Wen etc. propose a new supervision signal, called center loss, with the joint supervision of softmax loss, in order to enhance the discriminative power of the deeply learned features for face recognition task[3]. Jiahui Yu etc. introduce a novel Intersection over Union (IoU) loss function for bounding box prediction, which regresses the four bounds of a predicted box as a whole unit. By taking the advantages of IoU loss and deep fully convolutional networks, the UnitBox is introduced and applied on face detection task, which achieves the best performance among all published methods on the FDDB benchmark[4]. In 2018, Yutong Zheng, Dipan K. Pal and Marios Savvides motivate and present Ring loss, a simple and elegant feature normalization approach for deep networks designed to augment standard loss functions such as Softmax[5].

3 Proposed method

3.1 Center loss

In most of the available convolutional neural networks, the softmax loss function is used as the supervision signal to train the deep model, which is presented as Eq 1.

$$L_s = -\sum_{i=1}^{N} x_i^p log(\hat{c}_i^p) \tag{1}$$

In Eq 1, $x_i^p \in R^d$ denotes the i-th deep feature, belonging to the p-th class. d is the feature dimension. $\hat{c}_i^p$ is the class confidence of the p-th class.

Under the supervision of softmax loss, the deeply learned features are separable, but the deep features are not discriminative enough, since they still show significant intra-class variations.

In order to enhance the discriminative power of the deeply learned features in neural networks, we need to minimize the intra-class variations while keeping the features of dierent classes separable. So a new supervision signal, called center loss [3], is proposed and formulated in Eq 2. In fact, center loss has been proved useful in face recognition problem because of its discriminative power of the deeply learned feature.

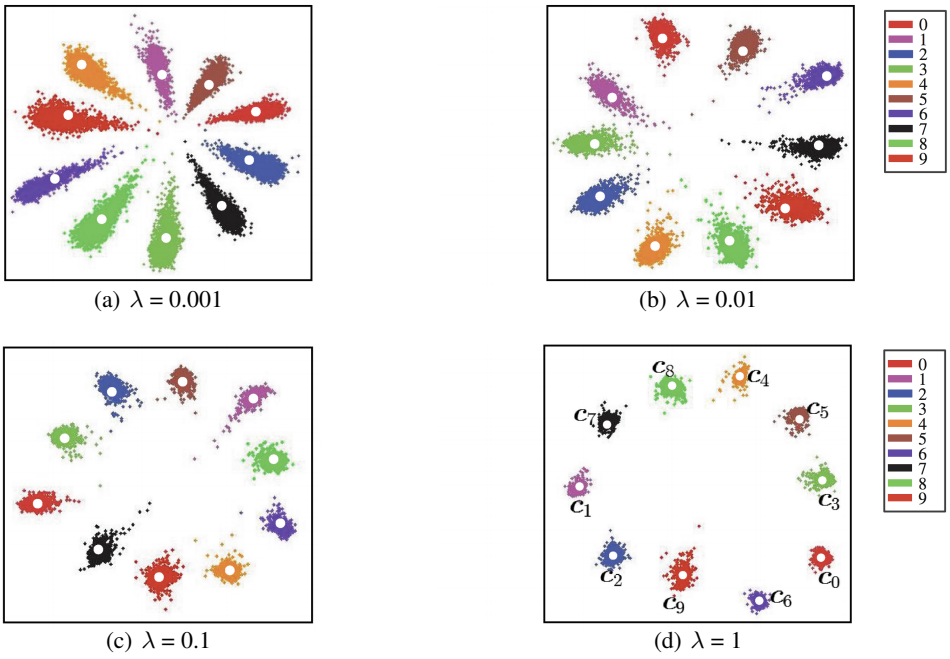
$$L_c = \frac{1}{2}\sum_{i=1}^{N} ||x_i^p - c_i^p||_2^2 \tag{2}$$

In Eq 2, $c_i^p \in R^d$ denotes the p-th class center of deep features. The formulation effectively characterizes the intra-class variations.

With the joint supervision of softmax loss and center loss, we can train the CNNs for discriminative feature learning, with a λ parameter to balance the two supervision signals. The joint supervision is presented in Eq 3.

$$L_{conf}(x, c) = L_s + \lambda L_c = -\sum_{i=1}^{N} x_i^p log(\hat{c}_i^p) + \frac{\lambda}{2}\sum_{i=1}^{N} ||x_i^p - c_i^p||_2^2 \tag{3}$$

Figure 2 shows that different λ leads to different deep feature distributions. With proper λ, the discriminative power of deep features can be significantly enhanced.



(a) λ = 0.001 (b) λ = 0.01

(c) λ = 0.1 (d) λ = 1

Figure 2: Visualization of impact of center loss on deep features[3]

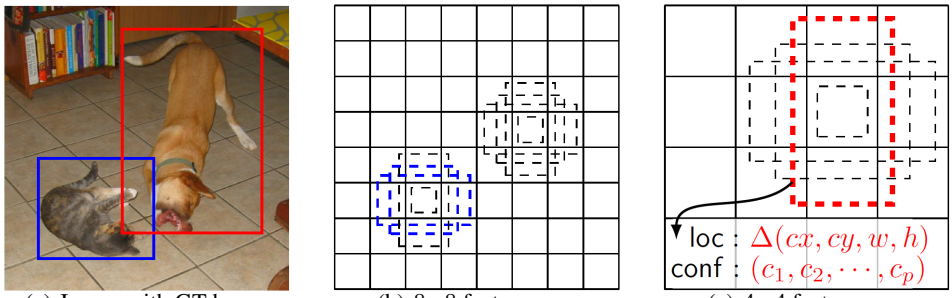
In fact, the principle of the center loss is very similar to that of the Linear Discriminant Analysis(LDA) mentioned in class. In LDA, its goal is to maximize the distance between the projected means (e.g. maximize $|\tilde{\mu}_1 - \tilde{\mu}_2|^2$) while minimize the total scatter of each class in projected space (e.g. minimize $\tilde{s}_1^2 + \tilde{s}_2^2$), where $\tilde{\mu}_i$ means the projected mean of class i onto LDA direction vector w and $\tilde{s}_i^2$ means the variance or scatter of the projected samples from class i. So LDA's objective function for 2-class problem can be formulated in Eq 4.

$$J(w) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{s}_1^2 + \tilde{s}_2^2} \tag{4}$$

With Eq 4, LDA can maximize the between-class scatter and minimize the within-class scatter at the same time. Similarly, for the joint supervision of softmax loss and center loss, the softmax loss can help train the model with inter-class dispersion while the center loss can help train the model with intra-class compactness.

3.2 Modified SSD Model

During SSD training, for each ground truth box we assign a default box with a best match among different locations and scales, as shown in Figure 3. The training objective loss function of SSD (Eq 5) is a weighted summation of the localization loss and the confidence loss. Usually we set the weight term α to 1.



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

Figure 3: SSD feature map corresponding to default boxes[1]

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \tag{5}$$

Let N be the number of matched default boxes. Let $x_{ij}^p = \{1, 0\}$ be an indicator for matching the i-th default box to the j-th ground truth box of category p.

In SSD framework, smooth L1 loss is used as localization loss between the predicted box (l) and the ground truth box (g), as shown in Eq 6.

$$L_{loc}(x, l, g) = \sum_{i \in Pos}^{N} \sum_{m \in cx, cy, w, h} x_{ij}^p smooth_{L1}(l_i^m - \hat{g}_j^m) \tag{6}$$

The softmax loss is used as confidence loss over multiple classes confidences(c), as shown in Eq 7, where $\hat{c}_i^p = \frac{exp(c_i^p)}{\sum_p exp(c_i^p)}$.

$$L_{conf}(x, c) = -\sum_{i \in Pos}^{N} x_{ij}^p log(\hat{c}_i^p) - \sum_{i \in Neg} log(\hat{c}_i^0) \tag{7}$$

As mentioned above, under the supervision of the softmax loss, the deeply learned features are separable, but they still show significant intra-class variations. In order to enhance the discriminative power of the deeply learned features in neural networks, we need to minimize the intra-class variations while keeping the features of different classes separable.

How to do that? We decide to modify the confidence loss of SSD by adding the center loss (Eq 8) to the confidence loss with a parameter λ for each layer in the network, which is formulated in Eq 9.

$$L_c = \frac{1}{2} \sum_{k=1}^{K} \sum_{i \in Pos}^{N} ||f_i^p - C_i^p||_2^2 \tag{8}$$

$$L_{conf}(x, c) = -\sum_{i \in Pos}^{N} x_{ij}^p log(\hat{c}_i^p) - \sum_{i \in Neg} log(\hat{c}_i^0) + \frac{\lambda}{2} \sum_{k=1}^{K} \sum_{i \in Pos}^{N} ||f_i^p - C_i^p||_2^2 \tag{9}$$

In Eq 8, f_i^p denotes the feature of the p-th class in the k-th layer and C_i^p denotes the center of the p-th class in the k-th layer corresponding to the feature. After several attempts, we set λ to 0.01 to balance the softmax loss and the center loss.

Ideally, the center C_i^p should be updated as the deep features changed. In other words, we need to take the entire training set into account and average the features of every class in each iteration, which is very inefficient.

To address this problem, we set a scalar α to 0.5 to control the learning rate of the centers and perform the update based on mini-batch instead of updating the centers with respect to the entire training set[3]. To be specific, at the end of each iteration, the centers of each layer are updated by computing the average features of the corresponding classes.

Intuitively, the softmax loss forces the deep features of different classes staying apart. The center loss efficiently pulls the deep features of the same class to their centers by simultaneously learning a center for deep features of each class and penalizing the distances between features and their class centers. With the joint supervision of both losses, we can train a robust SSD model to obtain the deep features with the two key learning objectives, inter-class dispersion and intra-class compactness as much as possible.

4 Experiments

4.1 Implementation

Center loss is calculated based on the distance of features and center of the class. Since SSD uses the anchors from multi-layers and features of each layer have different dimensions, we have to compute the loss for each class from each layer separately instead of just add the loss before the last fully connected layer in traditional CNN. Because default boxes are pre-designed in each layer of the network, there is a certain relation between default boxes and corresponding feature vectors, thus making it possible to extract the feature vectors from multiple layers for each detected box respectively. We apply six center loss for block4, block7, block8, block9, block10, and block11 in SSD network. Six center losses as confidence loss and original softmax loss and localization loss are computed separately and together optimize the network.

In order to balance the influence of loss optimization in every layer, we have to carefully set the weight of the center loss of each layer. At first, we treated all center loss equally, without any normalization. It turned out that the first several layers' center loss are quite large and other losses could not converge, which finally made our model failed to make any correct prediction. To solve this problem, we made several attempts. We normalize the loss by number of detected boxes and size of feature vectors, and also adjust the parameter λ of center loss to balance the influence of confidence loss and localization loss. Then the result of our model is much more reasonable.

4.2 Experiment reuslt on VOC 2007

We use the PASCAL VOC2007 trainval dataset (5011 images) to train our model, evaluate it based on VOC2007 test dataset (4952 images) and compare the results against original SSD on this dataset. Our model is constructed with TensorFlow framework and is run on GPU with CUDA 8.0 for 30000 steps training about 10 hours. Figure 4 shows the total loss over training, and it can be seen that the model converges well.

We mainly compare the results by mean accuracy precision (mAP). As is shown in Table 1, after 30000 steps training, the original SSD model can achieve 70.74% mAP while the modified model can achieve 69.04% mAP, which is 1.7% lower than original SSD model.

Table 1: Evaluate on VOC 2007

Model	Training data	Testing data	mAP
SSD-300 VGG-based	VOC07 trainval	VOC07 test	70.74
SSD-300 VGG-based + Center Loss	VOC07 trainval	VOC07 test	69.04

Here we show one example of visualized detection results. From two pictures of Figure 5, we can observe that the detection performance of two models are close, while scores for classification of two models are slightly vary from each other in different objects. The proposed modified algorithm shows advantage in some small objects, but have lower scores in some other objects.

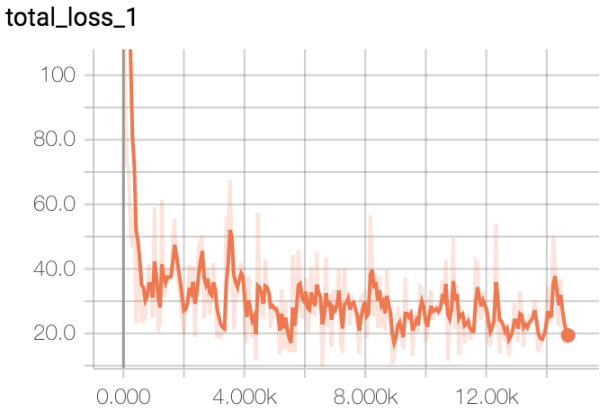


Figure 4: Sample figure caption.

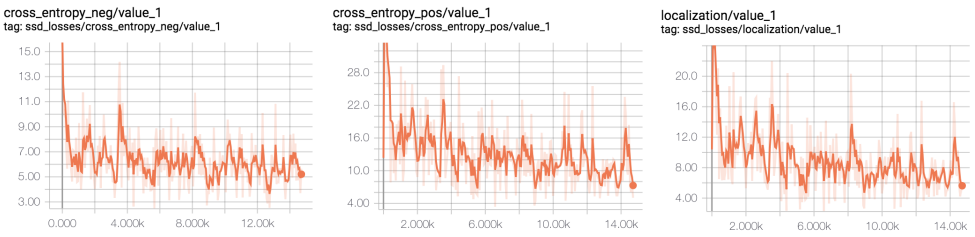
(a) Original model (b) Modified model

Figure 5: Comparison of result

5 Analysis

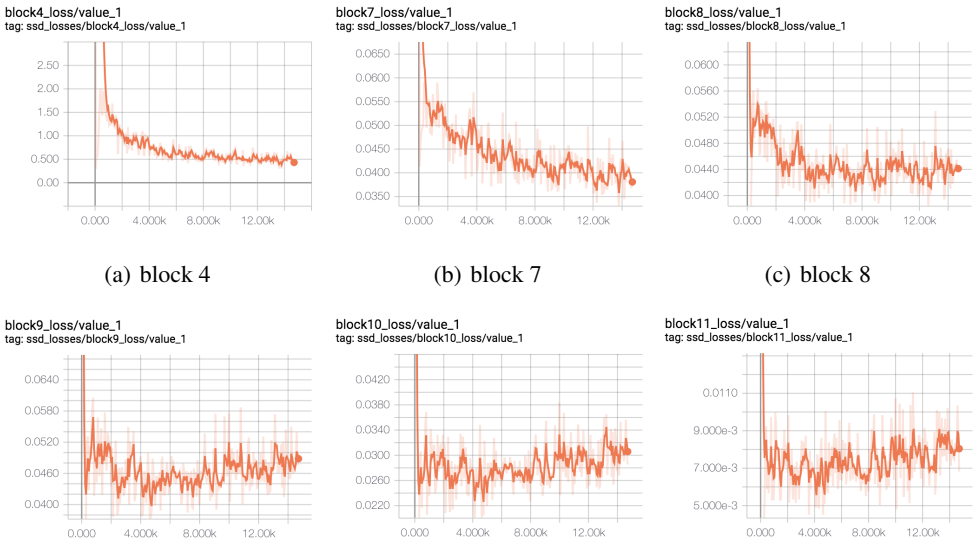
According to the result of our experiments, the application of center loss to original SSD model fails to make the model perform better. Here are some possible reasons for this result.

One explanation is that during training, optimization of center loss for front layers takes predominant role, which can be observed from the loss graph from training process. As shown in Figure 4 and Figure 6, total loss, softmax loss and localization loss of the model all tend to decrease, but not all center losses have such trends. Although the center loss of first two layers and total loss of the model keep decreasing, there is a general uptrend in center losses of latter layers, which shown in Figure 7.



(a) Softmax loss(neg) (b) Softmax loss(pos) (c) Localization loss

Figure 6: Softmax loss and localization loss



(a) block 4 (b) block 7 (c) block 8

(d) block 9 (e) block 10 (f) block 11

Figure 7: Center loss over training

This is due to the unbalanced influence between the center loss of these layers. Since SSD model applies anchor mechanism in multiple layers which has different dimension of feature map, center loss has to be computed separately for these layers. Different layers correspond to bounding boxes of different sizes and ratios, and latter layers' features are more abstract than front layers, so the center loss of different layers do not make equal contribution in training. Therefore there exist a trade-off between several center losses parallelly optimized for these layers in training, which requires a set of reasonable parameters to control the weighing between these center losses. It's a tricky work and need further experiment to find the scheme of setting these parameters.

One observation to backup the explanation which more exploration is needed to search for the best parameters is that the proposed method turns out to have slight advantages over the original model in classifying small objects in some test images. This is because front layers are related to small default boxes and the network tends to fit the feature center of these layers better while fail the latter layers. So the network is likely to have better performance on small objects but fail on others.

Another explanation is that center loss may not play to its strength as easily as in in face recognition task, where center loss was first used and showed good performance. Face recognition is a different scenario from general object detection because the face image of the same person will not vary too much. Data for the same class, i.e. the same person, always look similar only including different light, postures, occlusions and so forth. However, images in general object detection task can significantly vary from each other even for the same class since they are about the whole objects. For example, dogs with completely different appearance share the same class. On the one hand, this can cause it much more difficult to find a center among the features for a certain class, when it is highly comprehensive. On the other hand, attempts to cluster within the class of diversed features may also reduce the network's ability in generalization, thus leading to unsatisfied performance in general.

Future research may focus on making center loss more generalized for diversed features within the class by introducing some other algorithms.

6 Conclusion

This project attempts to modify the SSD object detection algorithm by applying center loss to enhance more accurate object recognition and classification. We add center loss to the confidence loss for each layer separately, train and test our model at VOC2007 and compare the results with the original SSD model.

Experiments have shown that the mAP of our model is a little bit lower than the original SSD model, which means that the application of center loss to SSD model fails to make an improvement as we expected before.

Several reasons may contribute to the decrease of mAP. On one hand, the differences between the length of each layer's feature may result in a weight imbalance between the centers of each layer. On the other hand, the center loss may be suitable only for object recognition problems with smaller differences between classes, such as face recognition.

Future work may focus on adjusting the weight coefficient, balancing the center loss of each convolution layer and further exploring the application of center loss in the field of object detection.

References

[1] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. *CoRR*, abs/1512.02325, 2015.

[2] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. June 2016.

[3] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. pages 499–515, 2016.

[4] Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang. Unitbox: An advanced object detection network. pages 516–520, 2016.

[5] Yutong Zheng, Dipan K Pal, and Marios Savvides. Ring loss: Convex feature normalization for face recognition. pages 5089–5097, 2018.