ConfNet: Predict with Confidence

Sheng Wan [1], Tung-Yu Wu[2], Wing H. Wong [2][3] and Chen-Yi Lee[1]

[1] Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan
[2] Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, USA
[3] Department of Statistics, Stanford University, Stanford, CA, USA

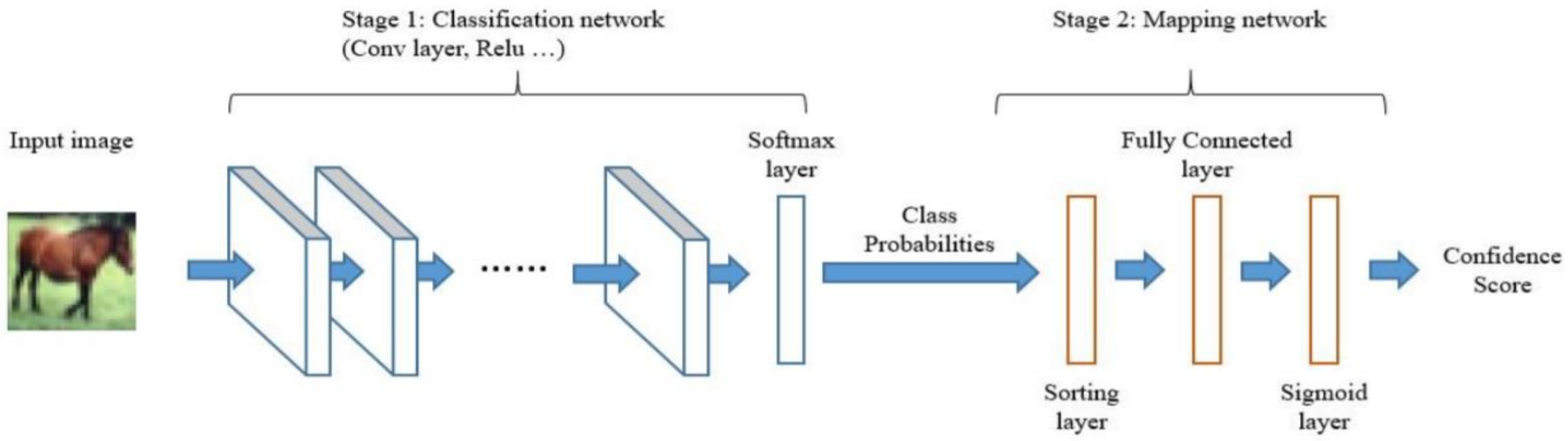


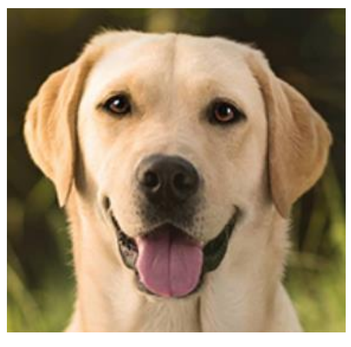
Fig. 1: The structure of Confidence Network.

● *Abstract*

In this paper, we propose Confidence Network (ConfNet) which not only makes predictions on input images but also generates a confidence score that estimates the probability of correctness of each prediction. Furthermore, Confidence Loss is proposed to make ConfNet automatically learn confidence scores in the training phase. The experiments on two public datasets show that the confidence scores generated by ConfNet are highly correlated with the model accuracy and outperforms two related methods. When stacking two ConfNets in a cascade structure, **3.8x** computational cost can be saved compared to the single state-of-the-art model with only **0.1%** increase of error rate.

● *Introduction*

Recently, Convolutional Neural Network (CNN) has been widely applied to real-world tasks, such as speech recognition, face detection, and audio classification. Despite its high accuracy, each CNN model tends to have difficulty in predicting certain classes due to the common training issues including unbalanced training dataset and limited model capacity.



Dog: 60%
Wolf: 25%
others: 15%

√ High confidence

Dog: 60%
Wolf: 40%
others: 0%

Low confidence

Therefore, estimating how much confidence a CNN model has in its prediction is a crucial problem. In this paper, we take one step further to ask, **"Can CNN models explicitly estimate the probability of correctness of each classification prediction?"** In other words, we aim to design a network which gives high confidence scores when it has strong faith in its predictions and provides low confidence scores when facing unrecognized input samples.

● *Proposed Network*

As shown in Fig 1, ConfNet consists of two stages. The first stage is a general classification network which takes images as input and generates the class probabiliies. In general, the class with maximum probability is selected as the model prediction for the input sample. It is worth noting that the framework of the classification network can be replaced with different CNN frameworks that match the resource restrictions (latency, accuracy) such as Alexnet, VGG-16, and Resnet.

The second stage of ConfNet is a mapping network that maps the class probabilities generated in the previous stage to a single confidence score. The mapping network is composed of a Sorting layer, a Fully Connected layer and a Sigmoid layer. The Sorting layer takes class probabilities as input and sorts the probabilities in descending order. The number of output neurons of the Sorting layer is the same as the number of classes. This layer removes the class information from the class probabilities and the following layers are hence able to focus on mapping the distribution of class probabilities to the model accuracy.

ConfNet: Predict with Confidence

Sheng Wan [1], Tung-Yu Wu[2], Wing H. Wong [2][3] and Chen-Yi Lee[1]

[1] Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan
[2] Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, USA
[3] Department of Statistics, Stanford University, Stanford, CA, USA

After the Sorting layer, a Fully Connected layer with one output neuron is employed to derive the mapping. Weights in Fully Connected layer represent the contribution of each probability to the confidence score. Since the confidence score is expected to estimate the probability of prediction correctness, a Sigmoid layer is adopted as the last layer of the mapping network to limit the output between 0 and 1.

● *Confidence Loss*

$$J(\alpha) = \frac{1}{N}\sum_{i=1}^{N} CS_i \cdot J_{Entropy}(x_i, y_i) - \alpha log(CS_i)$$

where N denotes the batch size, CSi denotes the confidence score of i-th prediction generated by ConfNet in this batch, and α is a hyper parameter to balance the regularization strength.

The first term of the objective function is the cross entropy loss scaled by the confidence score and the second term is a regularization term. Optimizing this objective function leads to two cases. **(i)** When the input sample can be easily classified and the expected cross entropy loss is low, a higher confidence score is preferred to reduce the regularization term. **(ii)** When a hard sample results in a high cross entropy loss, lower confidence score can effectively reduce the first term of the objective function. In the training phase, the mapping network of ConfNet tries to estimate the cross-entropy loss first and generates the corresponding confidence score to minimize the overall loss. *This objective function allows ConfNet to learn confidence scores automatically without additional labels from the training dataset.*

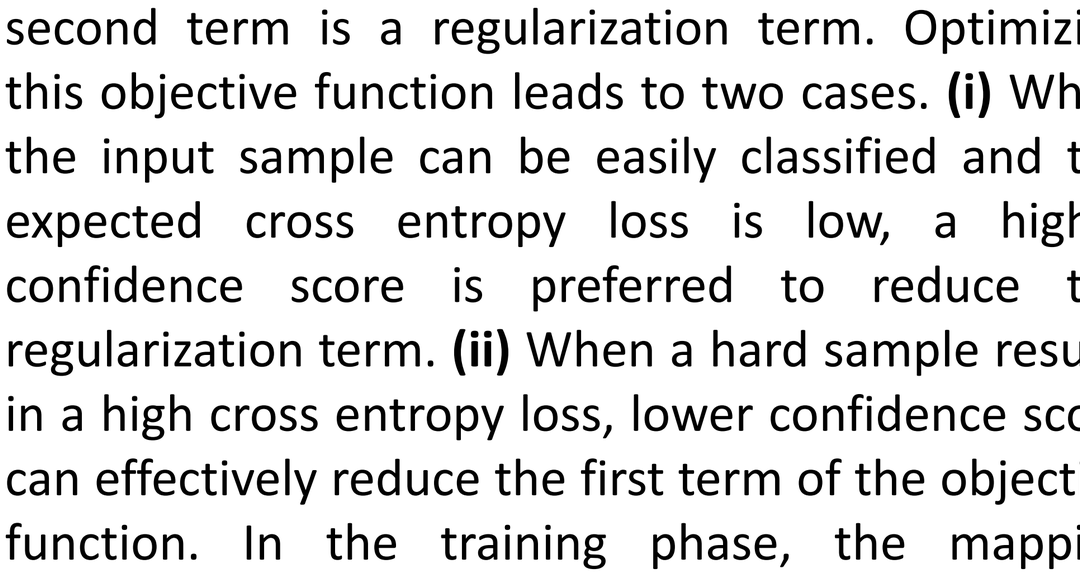


Fig. 3: Precision and Recall under different thresholds of the confidence score. The confidence score highly correlates to model accuracy in both frameworks.

● *Comparison with other methods*

Mean Effective Confidece (MEC):

$$MEC = \frac{1}{n}\sum_{i=1}^{n} C_i * Normalize(CS_i),$$

$$C_i = \begin{cases} 1, & i\text{th prediction is correct.} \\ -1, & i\text{th prediction is wrong.} \end{cases}$$

where n is the number of testing samples, Ci denotes the correctness of i-th prediction.

Table 1: Performance comparison of MEC with two related methods on Cifar10 dataset

Methods	MEC	
	NIN	Resnet-20
Maximum of class probabilities	80.14%	80.74%
Entropy of class probabilities	81.75%	82.50%
ConfNet	**83.48%**	**83.33%**

● *Cascade ConfNet*

Table 2: Performance comparison with state-of-the-art single models on Cifar10 dataset

Single model / Cascade structure	Threshold of Confidence score	Error rate	Average flops per image ($\times 10^7$)
NIN [16]	-	9.34%	22
Resnet-20 [13]	-	8.75%	4.1
Resnet-44 [13]	-	7.17%	9.7
Resnet-56 [13]	-	6.97%	12.5
Resnet-110 [13]	-	6.43%	24.3
DenseNet(L=100,k=12) [18]	-	5.77%	146.2
Stacked ConfNets (Resnet-20 + Resnet-110)	0.960	***7.22%***	***5.3***
Stacked ConfNets (Resnet-20 + Resnet-110)	0.965	***6.93%***	***5.7***
Stacked ConfNets (Resnet-20 + Resnet-110)	0.970	***6.53%***	***6.6***

We evaluate the performance of stacked ConfNets on Cifar10. The error rate and the average flops per image are shown in table 2. Compared with the single Resnet-110 model, our stacked ConfNets which consists of Resnet-20 model and Resnet-110 model can save **3.8x** times computational cost with only **0.1%** increase of error rate.

● *Reference (Partial)*

[1] Xin Wang, Yujia Luo, Daniel Crankshaw, Alexey Tumanov, and Joseph E Gonzalez, “Idk cascades: Fast deep learning by learning not to overthink,” arXiv preprint arXiv:1706.00885v2, 2017.

[2] Thomas P Trappenberg and Andrew D Back, “A classification scheme for applications with ambiguous data,” in Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on. IEEE, 2000, vol. 6, pp. 296–301.