CIRL: A Category-Instance Representation Learning Framework for Tropical Cyclone Intensity Estimation*

Dengke Wang[1], Yajing Xu[1], Yicheng Luo[1], Qifeng Qian[2], and Lv Yuan[1]

[1] Beijing University of Posts and Telecommunications
{wangdk,xyj,luoyicheng,lvyuan}@bupt.edu.cn
[2] National Meteorological Centery
qianqf@qq.com

Abstract. Tropical Cyclone (TC) intensity estimation is a continuous label classification problem, which aims to build a mapping relationship from TC images to intensities. Due to the similar visual appearance of TCs in adjacent intensities, the discriminative image representation plays an important role in TC intensity estimation. Existing works mainly revolve around the continuity of intensity which may result in a crowded feature distribution and perform poorly at distinguishing the boundaries of categories. In this paper, we focus on jointly learning category-level and instance-level representations from tropical cyclone images. Specially, we propose a general framework containing a CI-extractor and a classifier, inside which the CI-extractor is used to extract an instance-separable and category-discriminative representation between images. Meanwhile, an inter-class distance consistency (IDC) loss is applied on top of the framework which can lead to a more uniform feature distribution. In addition, a non-parameter smoothing algorithm is proposed to aggregate temporal information from the image sequence. Extensive experiments demonstrate that our method, with the result of 7.35 knots at RMSE, outperforms the state-of-the-art TC intensity estimation method on the TCIR dataset.

Keywords: Tropical Cyclone · Intensity estimation · Representation learning.

1 Introduction

Tropical Cyclone (TC) is one of the natural disasters that bring out severe threats to human society. The intensity of TC, which is defined as the largest continuous surface wind near the center of the TC, is an important indicator of its destructiveness. Estimating the intensity of TC can help mankind effectively reduce the damage caused by TC.

* Supported by the National Natural Science Foundation of China (NSFC No.62076031).

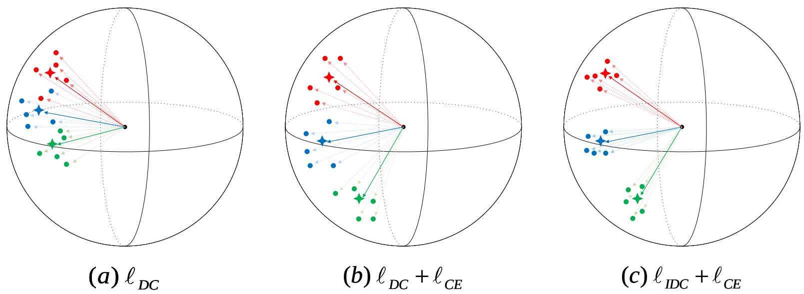


Fig. 1. Illustration of DC loss (a), DC loss + CE loss (b) and IDC + CE loss (c) based representation learning for TC intensity estimation. Stars represent the center of classes and circles represent samples. DC loss (a) learns the crowded feature distribution and has poor performance at boundaries. DC + CE loss (b) learns sparse embedding distribution, but it is scattered within the class. And IDC + CE loss (c) can lead to a uniform and discriminative feature distribution.

The essence of TC intensity estimation is establishing a mapping relationship from TC images to intensities. Since the change of TC is continuous during its life cycle, images of TC at adjacent moments may have similar visual appearances but different intensities. The appearance of TC with the same intensity may also vary greatly. These bring great challenges to TC intensity estimation.

The most classical method for TC intensity estimation is the Dvorak technique[7] which relies on TC cloud characteristics established on statistical experience. Over recent years, with the rapid development of deep learning, convolutional neural networks (CNNs) have achieved great success in TC intensity estimation. Among them, most methods [1, 2, 28, 3] treat the intensity as continuous values and construct regression networks to estimate the exact intensity. [14] consider intensity estimation as a classification task. Different with most classification tasks, TC intensity estimation focuses on not only the right or wrong classifications but also the influence of different degrees of errors when classifying. Therefore, a distance consistency (DC) loss is proposed to keep the distance between representations in proportion to the distance between labels to reduce the errors. DC loss focuses on the relationship between instances to learn instance-level representations and achieved good performance. However, it neglects that the feature distribution of different classes should be separate, and the intra-class and inter-class samples should be treated differently when optimizing the embedding space.

On the other hand, considering the change of TC is a continuous process, the TC intensity at current time is related to those in the past. And generally, there is no violent shaking. Therefore, it is necessary to combine historical information to smooth the estimated intensity. [1] use fixed weight smooth methods to combine historical information. [14] adopts the transformer model to learn the change of intensity from a series of typhoon images. The application of transformer effectively utilizes the temporal information between typhoon samples but also introduces a large number of extra parameters to the estimation model.

In this paper, We focus on jointly learning category-level and instance-level representations from TC images. While instance-level learning aims to learn a uniform distribution between instances, the category-level representations attend to reduce the intra-class variance and distinguish boundaries between categories. Specifically, we describe our ideas in Fig. 1.

As shown in Fig. 1a, the feature distribution learned from DC loss can be crowded, since the the ratio of distance between feature vectors to label distance is not supervised, which increases the difficulty of classification. In Fig. 1b, the distance between the feature of different classes still maintains a proportional relationship, but the inter-class feature distribution becomes separable with a CE loss. In Fig. 1c, an IDC loss with CE loss learns more intra-class compact and inter-class separable features, which further reduces the probability of the feature being classified incorrectly.

Motivated by above, we propose CIRL: a Category-Instance joint Representation Learning framework for TC intensity estimation with a CI-extractor and a classifier. As instance-level representation learning aims to obtain uniform distribution, category-level representation learning aims to distinguish the boundaries of categories and make the intra-class samples converge. Further, an IDC loss is proposed to optimize the backbone together with the CE loss. In IDC loss, the distance consistency is only maintained between categories. And the intra-class distance is optimized by the CE loss. Finally, we proposed a new smoothing algorithm, which can use historical information of any length to smooth the intensity estimate at the current moment. Without bells and whistles, our method achieved better performance than existing methods.

The contributions of our work can be summarized as follows:

- We propose a framework with a CI-extractor and a classifier aiming to learn a discriminative feature distribution which can take into account both instance-level and class-level representation learning. It is also general for continuous label classification problems.
- We propose a new inter-class distance consistency loss, which can learn a uniform feature distribution better.
- We explored a simple and fast smoothing post-processing algorithm without any parameters and find it is more accurate in real-time intensity estimation.
- Extensive experiments on the TCIR dataset demonstrate the effectiveness of our proposed approach, which outperforms the existing methods.

2 Related Works

Our work is closely related to both TC intensity estimation and metric learning.

2.1 Tropical Cyclone Intensity Estimation

Some results have been achieved by using CNN to estimate the intensity of the TC. [18] first propose to estimate the TC intensity by using a CNN-based classification network. However, only an approximate intensity range is obtained and the training data and test data are related in [18]. In [1, 2], a regression network is further designed to to estimate the intensity accurately and more information beyond the image is taken into account, such as latitude, longitude, and date. [28] choose to divide the TC sample into three different categories and for each category, different regression networks are constructed to estimate the intensity. [27] proposed a context-aware cycleGAN to solve the problem of an unbalanced distribution of sample categories. [3] designed a Tensor Network to solve the asynchronous problem in remote sensing dataset to utilize more channels of data. In [14], a combined model of CNN and transformer is used to capture TC temporal information.

However, existing methods only notice the continuity of intensity and ignore that the sample of different intensities (labels) should be separable. [14] considers TC intensity estimation as a classification problem but still learns representations from instances. In contrast, our proposed framework jointly learns category-level and instance-level representations and aims to obtain a separable and discriminative feature distribution.

2.2 Metric Learning

Metric learning uses the distance to measure the similarity of samples and constraints to make similar samples close, and different samples far away. To our best knowledge, [4, 8] introduced deep neural networks into metric learning for the first time. On this basis, the triple loss is proposed by [26, 19] in which the relationship between inter-class samples and intra-class samples is further considered. [20] expands the number of positive and negative samples in a tuple and proposed an N-tuple loss. [21] integrated the above loss to cope with the situation of multiple positives. [13] combined the softmax and cross-entropy loss to softmax loss and proposed to increase the margin. [24, 23, 25, 6] step further to optimize the margin in different opinions. The common idea of these methods is to minimize the intra-class distance and maximize the inter-class distance.

Recently, [12] noticed that the relationship between samples is not a simple positive and negative, and the distance between the features and the distance of their labels are connected to construct a triple log-ratio loss. [14] extend log-ratio loss to the case of N-tuples and proposed a DC loss. However, their method treats intra-class and inter-class samples equally, which will be harmful to the optimization.

3 The Proposed Approach

In this section, we firstly present our main idea about the Category-Instance fusion framework for TC intensity estimation. Then, we show the idea of IDC loss. Finally, a smoothing algorithm for eliminating fluctuations in intensity estimation is demonstrated.

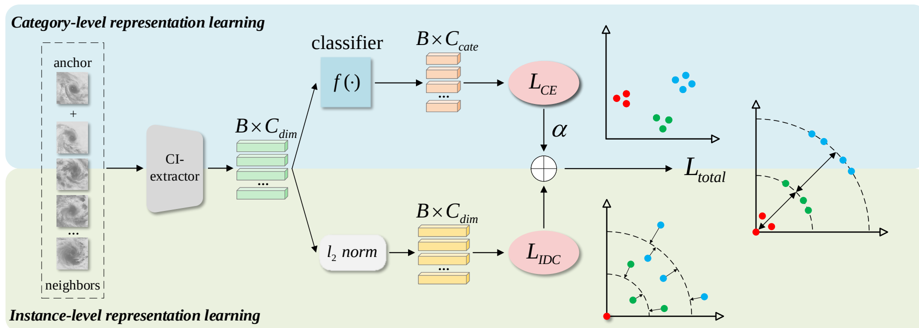


Fig. 2. Overview of the proposed Category-Instance representation learning framework. The framework consists of two loss functions: 1) CE loss for Category-level feature learning and 2) IDC loss for Instance-level feature learning. A CI-extractor is used to extract image representations, after which a multi-layer classifier $f(\cdot)$ is applied on top of the image representations to predict classification logits, and an l_2 normalization is adapted to translate the image representation for IDC loss. The total loss is obtained by the weighted summation of the two loss.

3.1 Category-Instance Representation Learning Framework

Fig. 2 shows the overview of the proposed framework. There are two branches in our framework: one category-level representation learning in the upper part which aims to distinguish the boundaries of categories by CE loss and one instance-level representation learning in the lower part which aims to learn a uniform inter-class embedding distribution by IDC loss. The detail of IDC loss will be introduced in section 3.2. The framework will learn a representation of the two levels at the same time to obtain the best feature distribution.

Formally, we adopt the method in [20] to constuct a mini-batch include M samples as anchors and N samples as neighbors. Each anchor is combined with the N neighbors to construct a set of N+1 tuple $\{x, y\} = \{(x_a, y_a), (x_1, y_1), \ldots, (x_N, y_N)\}$ with an anchor a and N neighbors randomly sampled from the remaining ones in which $x \in R^{2\times H\times W}$ is the images and y is the corresponding intensities. A CI-extractor is used to extract image repersentation $r = \{r_a, r_1, \ldots, r_k\} \in R^{D_E}$ from x. On the one hand, an ℓ_2 normalization is applied to r to get the normalized representaion $z = \{z_a, z_1, \ldots, z_k\} \in R^{D_E}$ to keep the vectors on the same sacle. After that, the IDC loss is applied on top of the normalized representations for instance-level representation learning. On the other hand, a classifier head $f_c(\cdot)$ is adopted to the image representation r to predict the class-wise logits $s = \{s_a, s_1, \ldots, s_k\} \in R^{D_C}$, which are used to compute the CE loss. Motivated by [13], the classifier, which is actually a full connection layers, is constructed without bias. Then, the logits can be writen as:

$$s = \|w\| \, \|r\| \, cos\theta. \tag{1}$$

w is the weight of the fully connected layer which can be regarded as the center of the class. Finally, a CE loss is applied to learn a separable feature which is

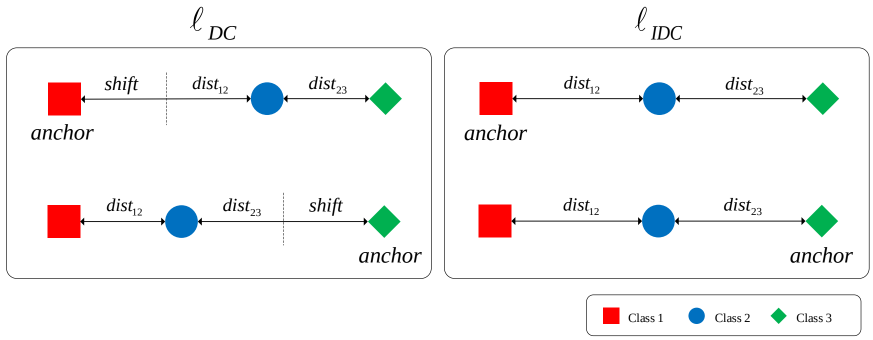


Fig. 3. Comparison of DC loss and IDC loss

important in discriminating class boundaries. As the CE loss aims to maximize the value of logits for the corresponding class, the cosine distance between r and w can also be increased and the intra-class variance is reduced. The final loss function for the framework is:

$$\ell_{total} = \ell_{IDC} + \alpha \ell_{CE}, \tag{2}$$

where ℓ_{IDC} is the IDC loss, ℓ_{CE} is the CE loss and α is a weighting parameter to balance the contribution of different losses.

3.2 Inter-class Distance Consistency Loss

The distance consistency loss[14] takes an anchor a and N neighbors as input. It is designed to penalize the sample for violating the rank constraint, namely, that the feature distance between samples in the embedding space should be consistency to the label distance. And the ratio, which is calculated by dividing the feature distance by the label distance, should be consistent across samples. However, when it comes to the case of intra-class samples, it is difficult to determine the label distance. Ideally, the feature distance between intra-class samples should be minimized and the label distance should be 0. But this would result in an infinite ratio of samples within the class. In [14], a constant is added to all label distances to avoid this which would cause a shift in label distances and be harmful to distance consistency.

Motivated by the above, we proposed an inter-class distance consistency loss to optimize the samples within and between classes respectively. Specially, we only maintain distance consistency across classes. The optimization of intra-class samples is left to CE loss. By doing so, the optimization of inter-class samples wil no-longer be affected by the intra-class samples. For an anchor a and N neighbor, the IDC loss is formulated as:

$$\ell_{IDC} = - \sum_{\substack{i=1 \\ y_i \neq y_a}}^{N} log \frac{r_{ai}}{\sum_{j=1, y_j \neq y_a}^{N} r_{aj}}, \tag{3}$$

$$r_{ij} = \frac{D(f_i, f_j)}{D(y_i, y_j)}, \tag{4}$$

where f is the representation for the sample and $D(\cdot)$ denotes the Euclidean distance.

Compared with DC loss, IDC loss can obtain a more uniform distribution. As shown in Fig. 3, with a shift in label distance, in the same tuple, selecting different samples as anchors will generate different feature distributions in DC loss. This can lead to instability in the learning process and even oscillations in the feature distribution. In contrast, IDC loss can obtain a more uniform and stable distribution without being affected by anchor selection.

3.3 Inference Stage

After training the extractor, the inference stage is started. Considering the temporality of TCs, we sample the images in time order during the inference stage. The classifier is thrown away and only the CI-extractor is used. Following [1], each input image is rotated by four angles and fed into the extractor to obtain the feature embedding. An average operation is used to get the final embedding. We first utilize the backbone for extract features from the whole training dataset and averaged the features by class to construct a global class representation set $C = [c_1, c_2, \ldots]$. The class vectors were used for NN classification. Then, for a sequence of image $X = [x_{t-T}, x_{t-T+1}, \ldots, x_t]$, the process can be represented as

$$R = f(X, \theta) = [r_{t-T}, r_{t-T+1}, \ldots, r_t]. \tag{5}$$

Then, an ℓ_2 normalization was applied to R and get $Z = [z_{t-T}, z_{t-T+1}, \ldots, z_t]$. A nearest neighbor (NN) classifier is adopted to decide on the final intensity of Z, which is given by:

$$\hat{y}_t = \arg\min_i D(c_i, z_t), \tag{6}$$

where $D(\cdot)$ represents Euclidean distance, i represents the label and $\hat{y}_t$ is the estimated intensity at time t.

3.4 A smooth algorithm for intensity estimating

To reduce the shake of TC intensity estimation, we further adopted a weighted average method, named stage smooth, which uses the weighted average of the estimated TC intensity at the current time and the previous N moments as the final intensity estimation at the current time. Considering the TC intensity changes monotonously, we have adopted a basic algorithm, that is, the weight decays as the time increases.

Specifically, given a series of estimated intensites $\hat{Y} = [\hat{y}_{t-T}, \hat{y}_{t-T+1}, \ldots, \hat{y}_t]$, we use a sliding window to smooth the estimated intensity. The intensity estimation at time t was smoothed by the following algorithm 1.

In the early stages of inference, the length of the sequence is less than the length of the window, we just repeat the first value of the sequence at the beginning.

Algorithm 1 Stage-smooth algorithm

Input: $\hat{y}_{t-T}, \hat{y}_{t-T+1}, \ldots, \hat{y}_t$
Output: $\tilde{y}_t$
1: Let $i = T$ and $\tilde{y}_{t-T} = \hat{y}_{t-T}$.
2: **while** $i > 0$ **do**
3: $\quad \tilde{y}_{t-i+1} = 0.5 * (\hat{y}_{t-i+1} + \tilde{y}_{t-i})$
4: $\quad i = i - 1$
5: **end while**
6: **return** $\tilde{y}_t$

4 Experiments

In this section, we firstly introduce the setting for our experiments which includes the dataset and implementation details. After that, we compare our proposed method with the state-of-the-art TC intensity estimating methods. Finally, some ablation studies are given to highlight some important properties of our framework.

4.1 Experimental Settings

Dataset We conducted our experiment on the benchmark TCIR dataset[1] which contains 70501 TC images of 201×201 from 2003 to 2017. And There is 82 different intensity values. We used TC images from 2003 to 2016 for training and TC images from 2017 for testing to make sure the images in the training set and test set are from different typhoons. Followed [2], IR1 and PMW channels of the image in the dataset were used to train our model.

Implementation details Our method is implemented using Pytorch on an Nvidia RTX2080. A ResNet18[10] is adopted as backbone to extract image representations, which has been pre-trained on the ImageNet ILSVRC 2012 dataset[5]. The classifier is consisted by one layer. And all images are resized to 224×224 before feeding into the network. Further, random rotation and random crops were adopted for data augmentation during training, and single-center crops were used for testing. We use SGD with a momentum of 0.9 and an initial learning rate of 5×10^{-4} as the optimizer to train the network. The network is trained for 50 epochs with the learning rate being decayed by a factor of 0.96 after each epoch. Following [2], We adopted a random sampling strategy to construct a mini-batch of a size of 12 from backbone training and the number of anchors is set to 4. The hyperparameter α is set to 1.3 and the T in Stage-smooth algorithm is set to 5.

Evaluation metrics We obtained the final results by searching for the nearest neighbors in the class vectors set. We adopted the root mean squared error (RMSE) and mean absolute error (MAE) as the evaluation metrics.

$$RMSE = \sqrt{\frac{1}{m}\sum^{m}(y_i - \tilde{y}_i)^2}, \tag{7}$$

Table 1. Tropical cyclone estimation results compared with other methods. Temporal means the way of handling temporal data. Bold numbers denote the best results.

	Approach	Temporal	RMSE (kt)
1	Cross-entropy	-	10.36
2	Npair[20]	-	10.75
3	log-ratio[12]	-	10.21
4	CNN-TC[1]	-	10.18
5	DR-extractor[14]	-	8.81
6	**Ours(CI-extractor)**	-	**8.49**
7	ADT[17]	linear	11.79
8	AMSU[11]	linear	14.10
9	SATCON[22]	linear	9.21
10	CNN-TC(S)[2]	five-point smooth	8.39
11	DR-transformer[14]	transformer	7.76
12	**Ours(CI-extractor)**	stage smooth	**7.35**

$$MAE = \frac{1}{m}\sum_{i=1}^{m} |y_i - \tilde{y_i}| \, . \tag{8}$$

4.2 Comparison to state-of-the-art methods

In this section, we compare the proposed CI-extractor to existing TC intensity estimation methods, such as traditional intensity estimation method[17, 11, 22], regreession-based method[1, 2] and our main baseline[14] on TCIR dataset. The results are shown in table 1. Note that kt is a unit commonly used in meteorology. 1kt $\approx$ 0.51m/s.

The first three rows are some traditional methods that are reproduced on the TCIR dataset. The fourth and five rows are the typical regression-based method CNN-TC and our main baseline DR-extractor. Note that the DR-extractor here only includes the backbone and does not use any temporal strategy. The last six rows 7-12 compare experimental results using temporal information. Rows 7-9 are manual intensity estimation methods and linear interpolation is used to math the times of the dataset. CNN-TC(S) is the upgraded version of CNN-TC and a five-point smooth algorithm is applied. DR-transformer is the method that uses the DR-extractor as the backbone and a transformer model is further used to aggregate temporal information.

For rows 1-6, our CI-extractor outperforms other methods by a margin (8.49 vs 8.81) as none of the temporal information was used. Since our CI-extractor pays more attention to the reduction of intra-class variance in the feature space, fewer samples will be misclassified, which leads to a decrease in the RMSE metric. The last six rows (7-12) are results that make use of temporal information. From the table we can see, that our method is superior to the other approach by a

Table 2. Tropical cyclone estimation results with our main baseline. Intensites are devided into eight categories by SSHWS and the number of each is counted. The two methods on the left do not use temporal information which is used in the two on the right. Bold numbers denote the best results and the MAE and RMSE are reported as the final result.

Category	Intensity Range	Numbers	Approach							
			DR-extractor		CI-extractor		DR-transformer		CI-extractor(S)	
			MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
H5	$\geq 137kt$	41	6.04	7.53	7.03	8.48	10.39	11.93	**6.12**	**7.65**
H4	$113-136kt$	93	6.65	8.95	7.24	9.16	6.45	8.17	**5.56**	**6.71**
H3	$96-112kt$	130	9.97	12.44	9.61	11.95	7.68	9.31	**6.89**	**8.88**
H2	$83-95kt$	243	10.10	12.68	9.84	12.25	8.13	10.33	**7.54**	**9.59**
H1	$64-82kt$	468	10.26	12.47	9.18	11.47	8.77	10.96	**8.12**	**10.18**
TS	$34-63kt$	1735	6.29	8.09	6.44	8.20	**5.47**	**7.08**	5.72	7.25
TD	$20-33kt$	1501	5.33	6.92	4.70	6.28	5.04	6.38	**4.28**	**5.71**
NC	$< 20kt$	89	6.74	8.32	7.14	8.58	8.01	9.55	**5.93**	**7.19**
Total	-	4300	6.73	8.81	6.46	8.49	6.02	7.80	**5.62**	**7.35**

Table 3. Evaluation of the effects of different components in our framework. For row (a), the loss is used on top of the backbone. For row (b) and row (c), the category-level representation learning branch is further applied. And for row (d) and row (e), a stage smooth algorithm is adopted at inference stage.

	methods	**MAE**	**RMSE**
(a)	DC	6.73	8.81
(b)	DC+CE	6.59	8.66
(c)	IDC+CE	6.46	8.49
(d)	DC+CE+stage smooth	5.80	7.54
(e)	IDC+CE+stage smooth	5.62	7.35

large margin, with the second-best model (DR-transformer) having circa 0.41 knots higher in the RMSE metric.

Performance in SSHWS We further compare our method with our main baseline DR-transformer to show the performance in different intensity categories. As shown in table 2, the left four columns are the results without temporal strategies which are used in the right four. The TC intensity is split by Saffir-Simpson Hurricane Wind Scale (SSHWS) along with intensity categorization for tropical storms and tropical depressions. The RMSE and MAE were reported. As we can see, in most categories, such as H5, H4, H3, H2, H1, TD, and NC, our method is superior to our baseline. In particular, the RMSE of our method for estimating the intensity of high-intensity (H5) typhoons and low-intensity(NC) typhoons is much lower which is greatly valuable in practical applications.

Table 4. Comparison of smooth methods. CI-extractor is used as backbone in all experiments. The RMSE and MAE is reported.

	methods	**MAE**	**RMSE**
(a)	transformer	5.89	7.69
(b)	five-point	6.85	9.13
(c)	stage-smooth	**5.62**	**7.35**

Table 5. Comparison of classifiers. For all experiments, CI-extractor is used as backbone.

	classifier	**MAE**	**RMSE**
(a)	kNN(k=1)	6.58	8.62
(b)	kNN(k=3)	6.51	8.51
(c)	kNN(k=5)	**6.51**	**8.50**
(d)	softmax	7.11	9.50

4.3 Ablation studies and discussions

In this section, we conduct some ablation studies to characterize our framework. Concretely, we study the effects of the IDC loss, CE loss, and the stage smooth algorithm to the result. We also discuss whether the KNN classifier is better than softmax and the advantage of our smooth algorithm to the transformer.

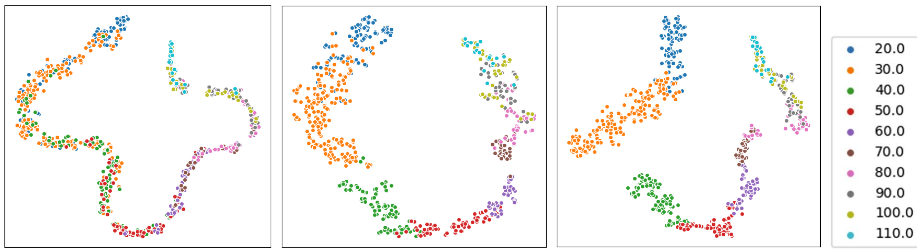
Effects of components For all experiments, the resnet-18[10] is used as backbone to get the extractor. The estimation results are shown in table 3. Results show that both category-level representation learning and inter-class calculation are necessary for CIRL. Especially, IDC loss reduces the RMSE (8.49 vs 8.66) by removing the intra-class instance distance which can cause the distance shifts and prevent the convergence of intra-class samples. The category-level representation learning also has a significant impact on the final result (8.66 vs 8.81) since it performs well in distinguishing class boundaries. And the smooth algorithm greatly improves the performance finally.

We further show the distribution of features which can be seen in Fig. 4. The features which are extracted by backbone are visualized by TSNE[15]. We sample ten categories in order to show them more clearly. In Fig. 4a, the feature distribution is crowded and it is difficult to distinguish different categories. In Fig. 4b, features of the same categories tend to cluster together with category-level representation learning. And in Fig. 4c, the intra-class features get closer which achieved the best performance.

KNN or softmax We further compare the kNN-based classifier and softmax-based classifier on the top of CI-extractor. For the second one, the classifier in our framework is reserved and the output of the classifier is regarded as the final result. The result is shown in table 5. The kNN-based classifier performs better than softmax-based classifier (8.50 vs 9.50). Since our framework

Table 6. Effort of the sequence length in stage-smooth algorithm

Stage-Smooth	T=3	T=5	T=7	T=9
MAE	5.76	5.66	5.65	5.65
RMSE	7.49	7.39	7.38	7.39



(*a*) *DC* (*b*) *DC + CE* (*c*) *IDC + CE*

Fig. 4. The distribution of learned features under the different compose of loss, which lead to different distributions. The point with different colors denotes features of samples from different categories. The number in the legend represent labels and are also intensity values. Best viewed in color.

aims to learn a uniform feature distribution, the softmax classifier is hard to find a dividing line for the embeddings. And a larger number of k has little effect on the results, which also owing to the uniform inter-class distribution.

Stage-smooth or transformer In this work, we replace the stage-smooth algorithm with the transformer module. Specially, we use CI-extractor to obtain the feature embeddings from a sequence of images. Then, the feature embeddings are fed into the encoder of transformer and the one-hot vectors are fed to the decoder. The IDC loss is used to train the transformer. Following [14], the number of layers of encoder and decoder is set to 2 and the sequence length is set to 7. The result is shown in table 4. Since transformer has been proven to effectively utilize temporal information to reduce errors, our method can also achieve the target with less computations. As can be seen, the transformer results in obviously inferior performance to our stage-smooth algorithm, since it only benefits from the continuity of the representations. Further, the effort of the sequence length in stage-smooth is explored in table 6. A longer sequence has little effect on the results since the benefits of smoothing are limited.

5 Conclusion

In this paper, we proposed the CIRL framework which is general for continuous label classification problems. The framework focus on jointly learning a category-instance representation from images which can take full advantage of the separability and continuity of labels. Experiments on the TCIR dataset have convincingly demonstrated the effectiveness of our method. Additionally, we prove that in the continuous label classification problem, it is necessary to consider the separability between labels. We hope that this work will play a role in meteorological observations and in the future, our feature extraction framework can be extended to more continuous label classification problems.

References

1. Chen, B., Chen, B.F., Lin, H.T.: Rotation-blended cnns on a new open dataset for tropical cyclone image-to-intensity regression. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 90–99 (2018)
2. Chen, B.F., Chen, B., Lin, H.T., Elsberry, R.L.: Estimating tropical cyclone intensity by satellite imagery utilizing convolutional neural networks. Weather and Forecasting **34**(2), 447–465 (2019)
3. Chen, Z., Yu, X.: A novel tensor network for tropical cyclone intensity estimation. IEEE Transactions on Geoscience and Remote Sensing **59**(4), 3226–3243 (2021). https://doi.org/10.1109/TGRS.2020.3017709
4. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 1, pp. 539–546. IEEE (2005)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
6. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4690–4699 (2019)
7. Dvorak, V.F.: Tropical cyclone intensity analysis and forecasting from satellite imagery. Monthly Weather Review **103**(5), 420–430 (1975)
8. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). vol. 2, pp. 1735–1742. IEEE (2006)
9. Harwood, B., Kumar BG, V., Carneiro, G., Reid, I., Drummond, T.: Smart mining for deep metric learning. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2821–2829 (2017)
10. 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. pp. 770–778 (2016)
11. Kidder, S.Q., Goldberg, M.D., Zehr, R.M., DeMaria, M., Purdom, J.F., Velden, C.S., Grody, N.C., Kusselson, S.J.: Satellite analysis of tropical cyclones using the advanced microwave sounding unit (amsu). Bulletin of the American Meteorological Society **81**(6), 1241–1260 (2000)
12. Kim, S., Seo, M., Laptev, I., Cho, M., Kwak, S.: Deep metric learning beyond binary supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2288–2297 (2019)
13. Liu, W., Wen, Y., Yu, Z., Yang, M.: Large-margin softmax loss for convolutional neural networks. arXiv preprint arXiv:1612.02295 (2016)

14. Luo, Y., Xu, Y., Li, S., Qian, Q., Xiao, B.: Dr-transformer: A multi-features fusion framework for tropical cyclones intensity estimation. Applied Sciences **11**(13), 6208 (2021)
15. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
16. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. arXiv preprint arXiv:1705.10872 (2017)
17. Olander, T.L., Velden, C.S.: The advanced dvorak technique (adt) for estimating tropical cyclone intensity: Update and new capabilities. Weather and Forecasting **34**(4), 905–922 (2019)
18. Pradhan, R., Aygun, R.S., Maskey, M., Ramachandran, R., Cecil, D.J.: Tropical cyclone intensity estimation using a deep convolutional neural network. IEEE Transactions on Image Processing **27**(2), 692–702 (2017)
19. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015)
20. Sohn, K.: Improved deep metric learning with multi-class n-pair loss objective. In: Advances in neural information processing systems. pp. 1857–1865 (2016)
21. Sun, Y., Cheng, C., Zhang, Y., Zhang, C., Zheng, L., Wang, Z., Wei, Y.: Circle loss: A unified perspective of pair similarity optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6398–6407 (2020)
22. Velden, C.S., Herndon, D.: A consensus approach for estimating tropical cyclone intensity from meteorological satellites: Satcon. Weather and Forecasting **35**(4), 1645–1662 (2020)
23. Wang, F., Cheng, J., Liu, W., Liu, H.: Additive margin softmax for face verification. IEEE Signal Processing Letters **25**(7), 926–930 (2018)
24. Wang, F., Xiang, X., Cheng, J., Yuille, A.L.: Normface: L2 hypersphere embedding for face verification. In: Proceedings of the 25th ACM international conference on Multimedia. pp. 1041–1049 (2017)
25. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5265–5274 (2018)
26. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. Journal of machine learning research **10**(2) (2009)
27. Xu, Y., Yang, H., Cheng, M., Li, S.: Cyclone intensity estimate with context-aware cyclegan. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 3417–3421. IEEE (2019)
28. Zhang, C.J., Wang, X.J., Ma, L.M., Lu, X.Q.: Tropical cyclone intensity classification and estimation using infrared satellite images with deep learning. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **14**, 2070–2086 (2021)