POLAR COORDINATE CONVOLUTIONAL NEURAL NETWORK: FROM ROTATION-INVARIANCE TO TRANSLATION-INVARIANCE

*Ruoqiao Jiang, Shaohui Mei**

School of Electronics and Information, Northwestern Polytechnical University, Xi'an, China 710129

ABSTRACT

Convolutional neural network (CNN) has been famous for its translation-invariant ability in feature learning. In order to further encounter rotation-invariant, data augmentation by rotation of training samples should be considered for multiple-branch based structure using maximum operator or average operator. In this paper, a novel Polar Coordinate CNN (PC-CNN) is proposed for rotation-invariant feature learning. Specifically, training samples are first input to a polar coordinate transform layer by which rotation-invariance is converted into translation-invariance. Consequently, rotation-invariance problem in feature learning can be easily encountered by traditional CNNs without the multiple-branch structure. Experimental results over two benchmark data sets demonstrate that the proposed polar transformation is very effective to encounter rotation-invariant into traditional CNNs and outperforms several state-of-the-art rotation-invariant CNNs.

Index Terms— rotation-invariant, convolutional neural network, image classification, polar coordinate

1. INTRODUCTION

After AlexNet [1] was proposed in 2012, deep learning based researches have achieved great success. Compare with traditional feature representation methods, such as scale-invariant feature transform (SIFT) [2], histogram of oriented gradient (HOG) [3, 4], features learned in deep CNNs has more powerful description capability and generalization capability. However, the learned deep features are not the acme of perfection. For example, though max-pooling layer in CNNs enables slight rotation-invariant features, such rotation-invariance is far from meeting the requirement of practical applications. On the converse, rotation-invariant feature is vital in many applications, such as target detection in remote sensing images and galaxy image classification. Therefore, how to guarantee rotation-invariance in feature learning of CNNs is of great importance.

*Corresponding author. This work is supported by National Natural Science Foundation of China (61671383), the Fundamental Research Funds for the Central Universities (3102018AX001), the Natural Science Foundation of Shaanxi Province (2018JM6005), and the National Defense Basic Scientific Research Project (JCKY2016203C067). Email: meish@nwpu.edu.cn.

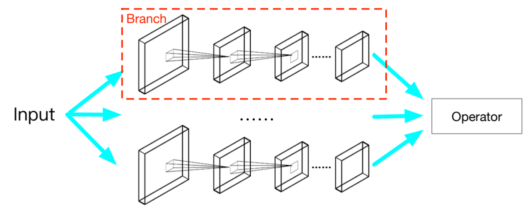

Fig. 1. Architecture of multi-branch CNN for rotation-invariant feature learning.

Recently, rotation-invariant deep feature learning has gained lots of attentions and many rotation-invariant CNNs has been proposed. Actually, most of the state-of-the-art methods adopted a multi-branch structure for rotation-invariant feature learning and achieved impressive performance in many application areas [5, 6, 7]. As shown in Fig. 1, the multi-branch CNN contains multiple identical branches, each of which consists of convolution layers and pooling layers. In order to extract rotation-invariant features, the input samples are rotated to various angles and fed to each branch for feature learning. The features learned in different angles are then fused using different pooling operators to achieve rotation-invariant. For example, in galaxy morphology prediction task, the output of each branch is concatenated and processed by a stack of dense layers to obtain rotation-invariant features [6]. Inspired by max-pooling and multi-instance learning, TI-Pooling used the maximum operator to process output of each branches to obtain rotation-invariant features [5]. The rotation-invariant CNN (RICNN) applied the average operator to the outputs of each breaches [7]. Moreover, a regularization constraint is added to the multinomial logistic regression objective, which explicitly enforces the feature representations of the training samples before and after rotating to be mapped close to each other. In these works, the performance of rotation-invariance is highly influenced by the number of branches in the structure. Generally, these algorithms achieve their best performance when the number of branch is set as from 24 to 36. This is because the max-pooling layer in each branch can only encounter a maximum rotation of 10-15 degree in feature learning of CNNs [5].

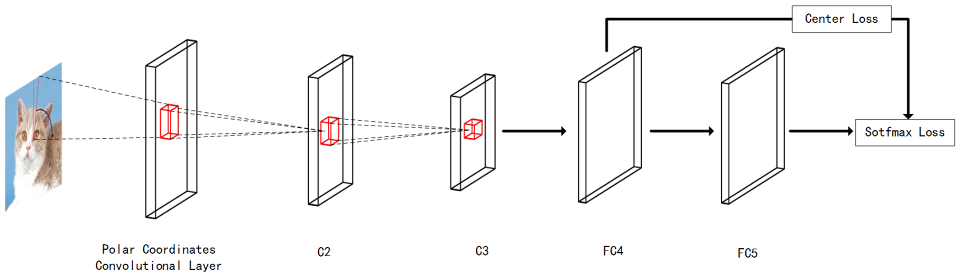

Fig. 2. Architecture of the proposed PC-CNN for rotation-invariant feature learning.

In this paper, instead of the multi-branch structure, a novel polar coordinate CNN (PC-CNN) is proposed for rotation-invariant feature learning. A polar transformation layer is first added to traditional CNNs, which converts rotation-invariant problem into translation-invariant problem. As a result, the rotation-invariant can be easily encountered by traditional CNNs. Moreover, the proposed PC-CNN adopts the center loss [8] function to learn more discriminative rotation-invariant features. Finally, experiments over two benchmark data sets are carried out to evaluate the effectiveness of the proposed PC-CNN for rotation-invariant feature learning.

2. PROPOSED METHOD

2.1. Overview

The architecture of the proposed PC-CNN is shown in Fig. 2. A very simple structure is adopted in the proposed PC-CNN, which consists of two convolutional layers (C2 and C3) and two full connected layers (FC4 and FC5). Similar to traditional CNNs, after conducting classification tasks in Layer FC5, effective features can be learned in FC4. In order to encounter rotation-invariance, a Polar Coordinate Convolutional Layer (PCCL) is added prior to convolutional layers, This layer converts rotation-invariance in feature learning into translation-invariance, which can easily explored by convolutional layers and full connected layers in a CNN. In order to train the proposed PC-CNN for feature learning, The center loss function in [8] is adopted:

$$J = J_s + \lambda J_c, \tag{1}$$

where λ controls the relative importance of embedded tern in object function, J_s and J_c denote softmax loss function and center loss function, respectively. For training samples $x_i \in \mathcal{X}$, let $\boldsymbol{O}_4(x_i)$ be output of FC4 and $\boldsymbol{O}_5(x_i)$ be output of FC5. J_s and J_c is defined as:

$$J_s = -\frac{1}{N}\sum_{i=1}^{N} \langle y_i, log(\boldsymbol{O}_5(x_i))\rangle \,, \tag{2}$$

$$J_c = \frac{1}{2}\sum_{i=1}^{N} \|\boldsymbol{O}_4(x_i) - c_{y_i}\|^2, \tag{3}$$

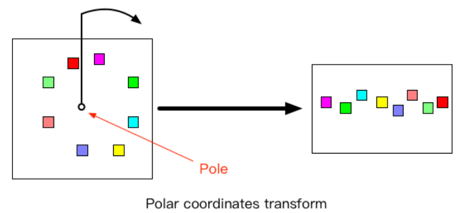

Fig. 3. Illustration of transformation from Cartesian coordinate system into a polar coordinate system.

where c_{y_i} is denoted the y_i class center of deep feature, and $\langle \cdot, \cdot \rangle$ denotes the inner-product of two vectors.

2.2. Polar Coordinate Convolutional Layer

In order to encounter rotation-invariance, the proposed PCCL contains two parts: polar coordinate transformation and convolution. The polar coordinate transformation maps pixels of input image sample from Cartesian coordinate system into polar coordinate, and then fed to a convolution layer to yield polar coordinate feature maps. By applying feature learning over these polar coordinate feature maps, rotation-invariance can be easily realized by translation-invariance.

Generally, in Cartesian coordinate system, the input image can be defined as a 2-dimensional plane $I\ (W, H)$, where W, H denote the width and height of the input, respectively. In order to convert rotation into translation, as shown in Fig. 3, the pole of polar coordinate system is defined in the center of the input image. Let $\bar{I}\ (X, Y)\ (X = W,\ Y = \frac{3}{4}H)$ represents transformed polar coordinate input, so the polar coordinate mapping between X and Y is defined as:

$$x = \|h \cdot (\sin\theta)\| + \frac{H}{2}\,, \tag{4}$$

$$y = \|h \cdot (\cos\theta)\| + \frac{W}{2}, \tag{5}$$

where $h \in \{0, 1, 2, \dots, H - 1\}$.

In order to retain enough information of raw pixels, the

Algorithm 1 Polar Coordinate Transformation

Input: Image $\boldsymbol{I}(W, H)$

Output: Image $\bar{\boldsymbol{I}}(X, Y)$

$\bar{\boldsymbol{I}}(X, Y) = 0$
$h = 0$
$w = 0$
while $w \leq W$ **do**
  $\theta = 2\pi \frac{w}{W}$
  while $h \leq H$ **do**
    $x = \|h \cdot (\sin\theta)\| + \frac{H}{2}$
    $y = \|h \cdot (\cos\theta)\| + \frac{W}{2}$
    if $0 \leq x \leq W$ & $0 \leq y \leq \frac{3}{4}H$ **then**
      $\bar{\boldsymbol{I}}(X,Y) = (x,y)$
    end if
    Update iteration counter: $h \leftarrow h + 1$
  end while
  Update iteration counter: $w \leftarrow w + 1$
end while

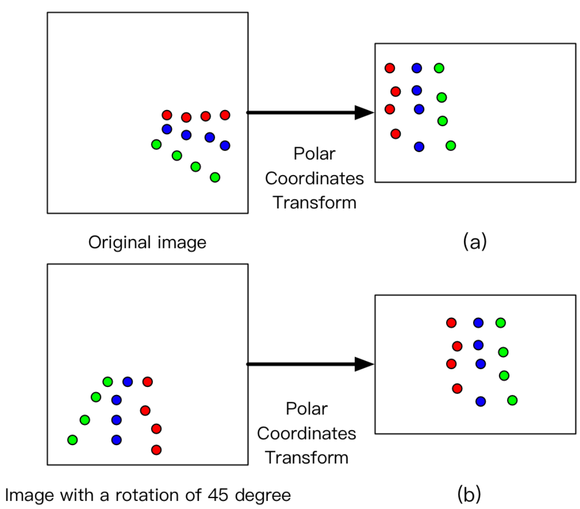

Fig. 4. Illustration of transforming Rotation-invariance into Translation-Invariance using the proposed PCCL.

number of θ is limited, which is defined by:

$$\theta = 2\pi \frac{w}{W}, \tag{6}$$

where $w \in \{0, 1, 2, ... , W - 1\}$. In addition, the mapped coordinates (X, Y) must stay in the range of image $\bar{I}$. Therefore, $0 \leq x \leq W$, $0 \leq y \leq \frac{3}{4}H$.

In summary, the proposed polar coordinate transformation is summarized in Algorithm 1.

2.3. Discussion

Generally, traditional CNNs are difficult to process rotation of input which contains position shift of pixels. However, in polar coordinate system, only the angular changed when the input is rotated if the pole is centered in the input image. Therefore, the rotation in Cartesian coordinate system is mapped into angular shift in polar coordinate system. As shown in Fig. 4, even if original image is rotated 45 degree, the distribution of pixels only translate in polar coordinate. As a result, by using the proposed PCCL to pre-process training samples, rotation-Invariance can be easily encountered in feature learning using traditional CNNs.

Fig. 5. Sample images from Minst-rot-12k data set.

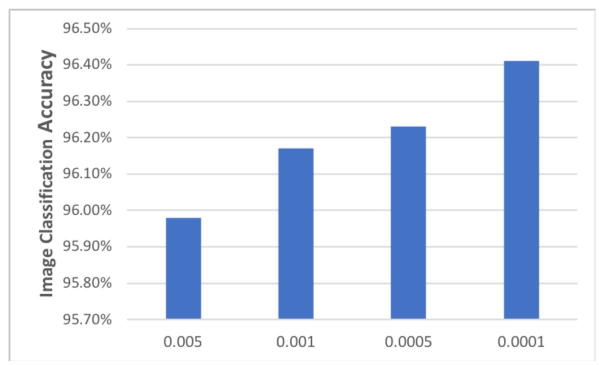

Fig. 6. Image classification results with different λ in the proposed PC-CNN on Minst-rot-12k data set.

3. EXPERIMENT

3.1. Experiment Over Minst-rot-12K Data Set

The minst-rot-12K data set [9, 10], which contains images of handwriting numbers rotated between 0 to 360 degree angles, is selected in this experiment. This data set owns 12000 training images and 50000 testing samples. Fig. 5 lists several sample images from Minst-rot-12K data set. Obviously, this data set is mainly used to evaluate the performance of rotation-invariant feature extraction. In this experiment, all of these method are implemented using Tensorflow platform and evaluated on PC with four GPUs.

3.1.1. Implementation Details

First, the balance parameter λ in the loss function defined in (1) is selected from $\{0.005, 0.001, 0.0005, 0.0001\}$. The ex-

Table 1. Comparative results of the proposed PC-CNN over traditional CNNs on Minst-rot-12k data set.

Method	Testing Accuracy
PC-CNN	**96.41**%
PC-CNN without PCCL	95.30%
AlexNet	93.434%

Table 2. Comparative results of the proposed PC-CNN over two multi-branch CNNs on Minst-rot-12k data set.

Method	Testing Accuracy
PC-CNN	**97.80**%
Ti-Pooling	97.30%
RICNN	97.53%

perimental results of the proposed PC-CNN with various λ is shown in Fig. 6. It is observed that the performance of the proposed PC-CNN varies slightly for different λ, and its best accuracy is obtained when $\lambda = 0.0001$.

3.1.2. *Comparison with Traditional CNNs*

In order to verify that the proposed PCCL offers rotation-invariance in feature learning, we compare the proposed PC-CNN with PC-CNN without PCCL. In addition, the well-known AlexNet [1] is also selected as a benchmark. The experimental results of these three algorithms over Minst-rot-12K Data Set are shown in Table. 1. Obviously, the proposed PC-CNN outperforms AlexNet CNN and PC-CNN without PCCL, demonstrating that the proposed PCCL is very effective to encounter the rotation-invariant character in feature learning of CNNs.

3.1.3. *Comparison with Multi-branch CNNs*

In this experiment, two state-of-the-art CNNs for rotation-invariant feature learning is selected for comparison, including Ti-Pooling [5] and RICNN [7]. In both of these two CNNs, the number of branch is set as 4. In order to show fair comparison, samples that fed to train the proposed PC-CNN is also rotated by 4 angles. The experimental results of these algorithms are shown in Table. 2. It is observed that our proposed PC-CNN clearly outperform the other two state-of-the-art rotation-invariant feature learning CNNs.

3.2. Experiment Over NWPU-VHR-10 Data Set

NWPU VHR-10 is a challenging ten-class VHR optical remote sensing image data set [11], including: airplane, storage tank, tennis court, basketball court, harbor, bridge, ship, ground track field, and vehicle. This data set was obtained from Google Earth with a spatial resolution ranging from 0.5 to 2 m. NWPU VHR-10 data set is divided into two image set, including a positive data set and a negative data set, in which the positive data set contain 757 airplanes, 302 ships, 390 baseball diamonds, 655 storage tanks, 124 bridges, 244 harbors, 163 ground track fields, and 477 vehicles. Though it has been widely used for validating the performance of object detection methods, in this experiment, the ground truth of this data set is used to validate performance of rotation-invariant feature learning for image classification. Some sample images of NWPU VHR-10 data set are shown in Fig. 7.

Table 3. Experimental results over NWPU-VHR-10 data set.

Method	Testing Accuracy
PC-CNN	**88.01**%
Ti-Pooling	81.950%
RICNN	84.341%

Fig. 7. Sample images from the NWPU-VHR-10 data set.

In this experiment, 10 samples per class are selected to train the proposed PC-CNN, TI-Pooling [5], and RICNN [7], while other 2385 image samples are used for testing. The number of branch in Ti-Pooling and RICNN is set as 4 and samples that fed to PC-CNN is also rotated 4 angles. In the proposed PC-CNN, λ is set as 0.0001. The experimental results of these algorithms are listed in Table. 3. It is also confirmed that the proposed PC-CNN is very effective for rotation-invariant feature learning of remote sensing images, which obviously outperforms other two stat-of-the-art methods by more than 3%.

4. CONCLUSION

In this paper, a novel PC-CNN is proposed to learn rotation-invariant feature. Instead of adopting multi-branch structure, the proposed PC-CNN uses polar coordinate transformation to convert rotation into translation, which can be easily encounter by traditional CNNs. Experimental results on two benchmark data sets demonstrate the proposed PC-CNN is very effective to encounter rotation-invariance using PCCL.

5. REFERENCES

[1] Alex Krizhevsky, IIya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Conference on Neural Information Processing Systems*, 2012.

[2] David G.Lowe, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999.

[3] Wanceng Zhang, Xian Sun, Kun Fu, Chenyuan Wang, and Wang Hongqi, "Object detection in high-resolution remote sensing images using rotation invariant parts based model," *IEEE Geoscience and Remote Sensing Letter*, vol. 11, pp. 74–78, 2014.

[4] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[5] Dmitry Laptev, Nikolay Savinov, Joachim M. Buhmann, and Marc Pollefeys, "Ti-pooling: Transformation-invariant pooling for feature learning in convolutional neural networks," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, pp. 289–297.

[6] Sander Dieleman, Kyle W. Willett, and Joni Dambre, "Rotation-invariant convolutional neural networks for galaxy morphology prediction," *Monthly Notice of the Royal Astronomical Society*, vol. 450(2), pp. 1441–1459, 2015.

[7] Gong Cheng, Peicheng Zhou, and Junwei Han, "Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images," *TGRS*, vol. 54(12), pp. 7405–7415, 2016.

[8] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, "A discriminative feature learning approach for deep face recognition," in *ICCV*, 2016.

[9] Cheng-Lin Liu, Kazuki Nakashima, Hiroshi Sako, and Hiromichi Fujisawa, "Handwritten digit recognition: benchmarking of state-of-the-art techniques," *Pattern Recognition*, vol. 36, pp. 2271–2285, October 2016.

[10] Shaohui Mei, Ruoqiao Jiang, Jingyu Ji, Jun Sun, Yang Peng, and Yifan Zhang, "Invariant feature extraction for image classification via multi-channel convolutional neural network," in *International Symposium on Intelligent Signal Processing and Communication Systems*, 2017.

[11] Gong Cheng, Junwei Han, peicheng Zhou, and Lei Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 98, pp. 119–132, December 2014.