Unsupervised Image Feature Extraction Based on Scattering Transform and Self-supervised Learning with Highly Training Efficiency

To cite this article: Guoshu Zheng and Qiuyu Zhu 2019 J. Phys.: Conf. Ser. 1237 032044

View the article online for updates and enhancements.
Unsupervised Image Feature Extraction Based on Scattering Transform and Self-supervised Learning with Highly Training Efficiency

Guoshu Zheng\textsuperscript{1*} and Qiuyu Zhu\textsuperscript{1}

\textsuperscript{1}School of Communication and Information Engineering, Shanghai University, Shanghai, 200444, China

*Corresponding author's e-mail: guoshu@shu.edu.cn

Abstract. Convolutional neural networks (CNN) can effectively extract high-level semantic features of images, however, to learn these features, a large number of labelled data is required. In order to extract image features without labelled data, this paper proposes an unsupervised image feature extraction method based on self-supervised learning. To learn image features, we train the neural network to identify the two-dimensional rotation applied to the image. The first few layers of the convolution network are replaced with a scattering network to speed up training process, and get good image features in the last few layers of the convolutional network. We input the extracted features into convolutional network for supervised learning, and take recognition accuracy as a criterion of feature validity. The experimental result show that the recognition accuracy gets 84\% on CIFAR10, reaching the mainstream results of unsupervised method; and 62.17\% on CIFAR100, which is very close to the rate of supervised learning. This method can be applied to the applications that do not have massive labelled training data and have limited computing resources.

1. Introduction
In recent years, the widespread adoption of deep convolutional neural networks has led to tremendous advances in supervised learning in computer vision. Specifically, in an object recognition or a scene classification task, a large amount of manually tagged data are used, and the convolutional network seeks to learn a powerful visual representation suitable for image understanding tasks. Image features learned in this supervised manner have achieved good results when transferred to other visual tasks, such as object detection, semantic segmentation or image subtitles.

However, the main limitation of supervised feature learning is the large amount of manual labelling work required, which is expensive and infeasible for the large amount of visual data available today. Therefore, learning high level convolution features with unsupervised manner is getting more popular because of no need for manually labelled data. Among them, a prominent example is self-supervised learning, which uses only the visual information present on the image or video to define an unlabelled task to provide an alternate supervisory signal for feature learning. For example, to learn features, [1][2] train convolutional network to colorize gray scale images. SIFT[3] has been widely used in image feature learning and has been used as a feature extractor in classifiers[4]. This image representation is typically encoded by unsupervised FV and inputted to a linear SVM classifier. A significant improvement to SIFT can be found in the scatter transform[5][6], which is a deep convolutional network that allows for the preservation of discriminant information discarded by SIFT,
while introducing geometric invariance and stability. Compared to other unsupervised representations, scatter transformations have been shown to produce best representations for complex image datasets[6].

Our work is based on [7], which proposed to learn image features by training convolutional networks to identify geometric transformations applied to the input image. Our improvement to the method of [7] is that we use a scattering network to replace the convolutional layers in the first few layers. The scatter transform proposed by Edouard Oyallon is a convolutional network, but its architecture and filters are predefined wavelets. Later, Edouard Oyallon improved the network[8], using a scattering network as a general and fixed initial layer for supervising mixed convolutional networks. Since these initial layers do not need to be learned, the training time is reduced, while the recognition performance is close to the mainstream convolutional network. The experiment shows that the first few layers of the convolutional network do not necessarily need to be learned, and can be replaced by a scattering network, so the training difficulty and complexity of the subsequent network layer can be significantly reduced.

We use a self-supervised network to train the model. The network is shown in figure 1, where ScatterNet represents the scattering network and ResNet consists of multiple layers of resnet. The network learns four types of rotation directions of the image. The image is rotated in four directions of 0°, 90°, 180° and 270°, and the rotation direction of the image is the self-supervised learning label. The rotated image is input into entire network to learn the direction of rotation. ScatterNet and its previous layers are solidified, only the ResNet and fully connected layers need to be trained.

Figure 1. Structure of self-supervised training network

After training the self-supervised network, we use the above model for feature extraction. The image, not rotated in advance, directly input into the ScatterNet and ResNet of the above model. Fine image features can be obtained in the ResNet layer, and the quality of the features obtained in different layers of ResNet is different. We output image features in the penultimate layer. After obtaining the features, we use a convolutional network for feature evaluation. The structure of the feature extraction and evaluation network is shown in figure 2. The network is a supervised network, and the label is the original image label. The image features are extracted using ScatterNet and ResNet of figure 1, then features are input into the convolutional network (Conv), and the image classification label are output. The quality of the feature is assessed by the original label identification accuracy.

Our contributions in this paper are:
1. We use self-supervised learning to train the hybrid convolutional network of solidified scattering network and resnet to improve speed.
2. We evaluate our feature extraction methods in various data sets (CIFAR10, CIFAR100, ImageNet), and experiments have shown that our method can efficiently extract the effective features of images.

Figure 2. Structure of feature extraction and evaluation network

2. Rotating Network
A rotating network (RotNet)[7] is a convolutional network that identifies two-dimensional rotations on an input image. Suppose $g(\cdot | y)$ stands for geometry operation, we can define $G = \{g(\cdot | y)\}_{y=1}^{K}$ as a set of $K$ discrete geometric transformations, $X^y = g(X|y)$ is the $y$-geometric operation applied to the input image $X$. The convolutional network model $F(\cdot)$ receives the geometric transform $X^y$ of the input image and outputs the probability distribution of the image in all geometric transformations:
\[ F(X^{y'}|\theta) = \{F^y(X^{y'}|\theta)\}_{y=1}^K \]  \hspace{1cm} (1)

where \( F^y(X^{y'}|\theta) \) is the predicted probability of the geometric transformation label \( y \) under the convolutional network parameter \( \theta \).

Therefore, given \( N \) training images \( D = \{X_i\}_{i=0}^N \), the goal of self-supervised learning is that the convolutional network model needs to learn to find:

\[ \min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \text{loss}(X_i, \theta) \]  \hspace{1cm} (2)

where the loss function \( \text{loss}() \) is defined as:

\[ \text{loss}(X_i, \theta) = -\frac{1}{N} \log(F^y(g(X_i|y)|\theta)) \]  \hspace{1cm} (3)

In the above formula, the geometric transformation \( G \) must define a classification task that force the convolutional network model to learn the semantic features that are useful for visually aware tasks. In this paper, the input image is rotated by a plurality of 90 degrees as a geometric transformation \( G \). The description of the network is shown in figure 3.

3. Scattering Hybrid Network

3.1 Scattering Network

The scattering network is a new method based on wavelet transform. Stable signal can be obtained by performing wavelet processing on the high-frequency information of the image, and then performing modulo and low-pass filtering.

The wavelet transform uses the expansion and rotation of the wavelet to calculate the convolution. Wavelets are local waveforms and are therefore stable to deformation. Scattering transform establishes nonlinear invariant based on wavelet coefficients. Let \( G \) denote a set of rotations \( r \) with an angle of \( 2k\pi/K \), where \( 0 \leq k \leq K \). Two-dimensional wavelets can be expressed as:

\[ \psi_2(u) = 2^{-2j}\psi(2^{-j}r^{-1}u) = 2^{-j}r \]  \hspace{1cm} (4)

where \( \psi \) is a bandpass filter.

The wavelet transform of \( x \) is \( \{x * \psi_2(u)\}_\lambda \), which is a redundant transform without orthogonality. In the discrete image, to avoid aliasing, we only capture frequencies of \( |\omega| \leq \pi \), and most of the energy in the camera image outside this frequency range can be ignored.

Let \( U[\lambda]x = \{x * \psi_2\} \), the sequence \( p = (\lambda_1, \lambda_2, ..., \lambda_m) \) defines a path along which the ordered
product of nonlinear and non-commutative operators is calculated:
\[ U[p]x = U[\lambda_m] \ldots U[\lambda_2]U[\lambda_1]x = |||x \ast \psi_{\lambda_1} \ast \psi_{\lambda_2} \ldots \ast \psi_{\lambda_m}| (5) \]

particularly, \( U[\emptyset]x = x \).

The scatter transformation for path \( p \) can be expressed as:
\[ S[p|x(u) == |||x \ast \psi_{\lambda_1} \ast \psi_{\lambda_2} \ldots \ast \psi_{\lambda_m} \ast \phi_{2J}(u) \] (6)

where \( \phi_{2J}(u) \) is a low pass filter. In particular, \( S[\emptyset|x = x \ast \phi_{2J} \).

Unlike a standard convolutional network, the output scattering coefficient is produced by each layer, not the last layer. Filters are not learned from the data, but are predefined wavelets.

If \( p = (\lambda_1, \lambda_2, \ldots, \lambda_m) \) is the path of length \( m \), then \( S[p|x(u) \) is called an \( m \)-th order window scattering coefficient. It is calculated at \( m \) layers of the specified convolutional network. For large-scale invariants, multiple layers are needed to avoid losing critical information. The structure of the scattering network is shown in figure 4.

3.2 Deep Hybrid Networks
Taking the results obtained by the scattering network as input, a supervised learning by a classifier such as SVM can obtain good recognition accuracy. Subsequently, Oyallon et al. proposed a network that cascades the scattering network with the ordinary CNN network[7], called the deep hybrid network. The paper combines the scattering network and resnet to obtain better performance.

4. Experiment

4.1 CIFAR Dataset Structure

4.1.1 Network Structure. Our convolutional network uses hybrid network combined the scatter network and resnet[9], where scattering network is a three-layer network. The parameter \( J \) of the scattering network is 2, so the convolutional network that will be cascaded later has a pixel resolution of 8*8, and a dimension of 243. For three color channels, each of which is 81-dimensional. After the scattering network, we cascade two layers of resnet, the network is represented by table 1.

| Stage     | Output size | Detail   |
|-----------|-------------|----------|
| scattering| 8*8         | J=2      |
| conv      | 8*8         | 256 channels |
| res1      | 8*8         | 256 channels |
| res2      | 4*4         | 256 channels |
We iterate the network 90 epochs to minimize the loss, using the gradient descent algorithm SGD, the momentum is 0.9, and the batch size is 128. We use 5e-3 weight decay, the initial learning rate is 0.1, and after 30, 60, and 80 iterations, the learning rate is reduced by a factor of 5.

After training the rotating network, we evaluate the features obtained. The evaluation method is feeding the output features to a supervised pipeline, and the accuracy is used as the basis for the feature quality. We output the extracted features in the penultimate layer of the convolutional network. For the CIFAR dataset network, we output features at the res1 layer with the output feature size of 8*8 and 256 channels. For the last layer of the convolutional network, the feature pixels are too small to reflect the location characteristics of the image. Table 2 is a comparison of different features’ accuracy getting from the last layer and the penultimate layer of the convolutional network, which shows that the quality of the features output from the penultimate layer is better than the last layer.

### Table 2. The accuracy of the last layer and the penultimate layer

| Method          | CIFAR10 Accuracy | CIFAR100 Accuracy | ImageNet-10 Accuracy |
|-----------------|------------------|-------------------|----------------------|
| The last layer  | 77.18            | 58.20             | 65.61                |
| The penultimate layer | 83.57           | 62.17             | 70.40                |

Our feature evaluation network uses the output features from the penultimate layer of the above network and feeds features into a convolutional layer and two fully connected layers for supervised learning. Therefore, the evaluation network can be divided into three blocks. The first two blocks are the solidified scattering network and resnet, which are used to extract features. The last block is the convolution network, which is used to learn the output features and classify. We judge the feature quality based on the accuracy of the classification. We iterate the evaluation network 100 epochs, and the learning rate, the weight decay, the learning rate drop and other parameters are the same as the above rotating network.

### 4.1.2 Results on CIFAR10 Dataset.

On the CIFAR10 dataset, the tested results are shown in Table 3.

### Table 3. CIFAR10 result comparing with other supervised and unsupervised method

| Method          | Accuracy |
|-----------------|----------|
| Supervised      |          |
| WRN 28-10[10]   | 96.0     |
| WRN 16-8[10]    | 95.7     |
| ALL-CNN[11]     | 92.8     |
| Scat+Resnet[8]  | 93.1     |
| Unsupervised    |          |
| Roto-Scat +SVM[6] | 82.3    |
| RotNet+non-linear[7] | 89.06   |
| RotScatRes+conv(ours) | 83.57  |

We compare the experimental results with the most advanced research on CIFAR10, which was implemented using an end-to-end convolutional network. Scat+Resnet is the method proposed in [8], and its accuracy rate is 93.1%, which is close to the most advanced supervised algorithm. This shows that the scattering network plus resnet can extract features well. We also use the scattering network plus resnet. But unlike [8], we use this network for unsupervised learning, which learns the rotation of images. In CIFAR10's most advanced unsupervised learning method, [7] uses a convolutional network to identify rotation, but the convolutional network used is NIN (Network In Network)[11], and the training time and parameters are larger than our network. Table 4 shows the difference between our network and RotNet[7] on CIFAR10. The experimental machine and settings are the same as well.

### Table 4. Comparison between RotNet and our network on the CIFAR10 dataset

| Method    | Accuracy | Num of MAC   |
|-----------|----------|--------------|
| RotNet[7] | 89.06    | 224,722,944  |
4.1.3 Results on CIFAR100 Dataset. We also test on the CIFAR100 dataset, and the network structure is the same as CIFAR10 dataset. The results are shown in table 5. The results show that the accuracy of our unsupervised algorithm is close to the most advanced supervised algorithm, which shows that our network can effectively extract image features.

| Method                      | Accuracy  |
|-----------------------------|-----------|
| FitNet[12]                  | 64.96     |
| ALL-CNN[11]                 | 66.29     |
| L-Softmax[13]               | 70.47     |
| GenPool[14]                 | 67.63     |
| RotScatRes+non-linear(ours) | 62.17     |

4.2 ImageNet Dataset

For ImageNet, we also use a scattering network cascaded resnet's convolutional network to identify rotation. We use a 3-layer scattering network with a parameter J of 3. During the training process, each image is randomly rescaled, cropped and flipped, and the final image size is 224*224. Therefore, the convolutional network cascaded later will have a pixel space of 28*28 input. Behind the scatter network we cascade four layers of resnet. The network structure is shown in table 6.

| Stage | Output size | Detail       |
|-------|-------------|--------------|
| scattering | 28*28       | J=3          |
| conv   | 28*28       | 256 channels |
| res1   | 28*28       | 256 channels |
| res2   | 14*14       | 256 channels |
| res3   | 7*7         | 512 channels |
| res4   | 3*3         | 512 channels |
| avgpool| 512         |              |
| fc     | 4           |              |

We iterate the network 60 epochs to minimize the loss, using the gradient descent algorithm SGD, the momentum is set to 0.9, and the batch size is 128. We use a 5e-4 weight decay with an initial learning rate of 0.1. After 15, 30 and 50 iterations, we reduce the learning rate by a factor of 10.

Our feature evaluation network still outputs features from the penultimate layer of the convolutional network, namely the res3 layer, followed by a non-linear network and three fully connected layers for supervised learning. We iterate the evaluate network 40 epochs, and the learning rate, weight decay and other parameters are the same as the above rotating network. In the end, our accuracy rating for ImageNet's 10 categories reached 70.4%.

5. Conclusion

This paper proposes a new unsupervised feature extraction method. We train a hybrid network of scattering network and resnet to identify the rotation of the input images. The first few layers of our hybrid network use a scattering network, so there is no need to learn parameters, only numerical operations are required, which can greatly speed up the training. Compared to supervised learning, our method uses only the image's own information and does not require a label. The experimental results show that the accuracy of unsupervised learning is close to supervised learning on different datasets.

In the future, we will improve the experiment from the following aspects: 1. Improve the structure of feature extraction network and improve the ability of feature expression. 2. Study the incremental learning method of feature extraction network to further reduce the training time of incremental learning. 3. Improve the first convolutional layer of the evaluation network, using incremental PCA or
other algorithms to replace the convolutional layer, to enhance the convenience of incremental learning training.

References
[1] Zhang, R., Isola, P., Efros, A.A. (2016) Colorful image colorization. In: European Conference on Computer Vision. Amsterdam. pp. 649–666
[2] Larsson, G., Maire, M., Shakhnarovich, G. (2016) Learning representations for automatic colorization. In European Conference on Computer Vision. Amsterdam. pp. 577–593.
[3] Lowe, D.G. (1999) Object recognition from local scale-invariant features. In: Computer vision. pp. 1150–1157.
[4] S´anchez, J., Perronnin, F., Mensink, T., Verbeek, J. (2013) Image classification with the fisher vector: Theory and practice. International journal of computer vision, 105(3): .222–245.
[5] Bruna, J., Mallat, S. (2013) Invariant scattering convolution networks. IEEE transactions on pattern analysis and machine intelligence, 35(8): 1872–1886.
[6] Oyallon, E., Mallat, S. (2015) Deep roto-translation scattering for object classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston. pp. 2865–2873.
[7] Gidaris, S., Singh, P., Komodakis, N. (2018) Unsupervised representation learning by predicting image rotations. In: International Conference on Learning Representations. Vancouver.
[8] Oyallon, E., Belilovsky, E., Zagoruyko, E. (2018) Scaling the Scattering Transform: Deep Hybrid Networks. In: IEEE International Conference on Computer Vision (ICCV). Venice. pp. 5619-5628.
[9] He, K., Zhang, X., Ren, S., Sun, J. (2015) Deep residual learning for image recognition. In: IEEE International Conference on Computer Vision. Santiago.
[10] Zagoruyko, S., Komodakis, S. (2016) Wide residual networks. In BRITISH MACHINE VISION CONFERENCE. York.
[11] Lin, M., Chen, Q., Yan, S. (2013) Network in network. In: International Conference on Learning Representations. Banff.
[12] Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y. (2014) Fitnets: hints for thin deep nets. Computer Science.
[13] Liu, W., Wen, Y., Yu, Z., Yang, M. (2016) Large-margin softmax loss for convolutional neural networks. In: International Conference on International Conference on Machine Learning.
[14] Lee, C.Y., Gallagher, P.W., Tu, Z. (2015) Generalizing pooling functions in convolutional neural networks: mixed, gated, and tree. Computer Science.