Image Representation Learning by Transformation Regression

Xifeng Guo, Jiyuan Liu*, Sihang Zhou, En Zhu*, Shihao Dong

National University of Defense Technology, Changsha, China
guoxifeng1990@163.com,
enzhu@nudt.edu.cn

Introduction

Motivation

Existing self-supervised learning methods usually define a label free surrogate task to provide a pretext supervision signal for feature learning. For example, AutoEncoder (AE) targets at reconstructing original data by minimizing the error between network output and corresponding data. Variational Auto-Encoder (VAE) and Generative Adversarial Network (GAN) are also taking the data content as supervision, though they have very different objectives. These methods focus too much on pixel details and thus are hard to learn high level semantic representation. Some methods apply many groups of transformations to the original data and teach the CNN model to recognize which group the input data comes from. Fundamentally, they usually construct a classification task with finite discrete labels, resulting in insufficient supervisory signals, which in turn harms the performance of representation learning.

Our Contributions

- 1) This is the first work to create sufficient auxiliary supervisory signals by regressing continuous transformation parameters.
- 2) We achieve the state-of-the-art performance on four popular image benchmark datasets by performing classification on the representation learned through our proposed transformation regression method.

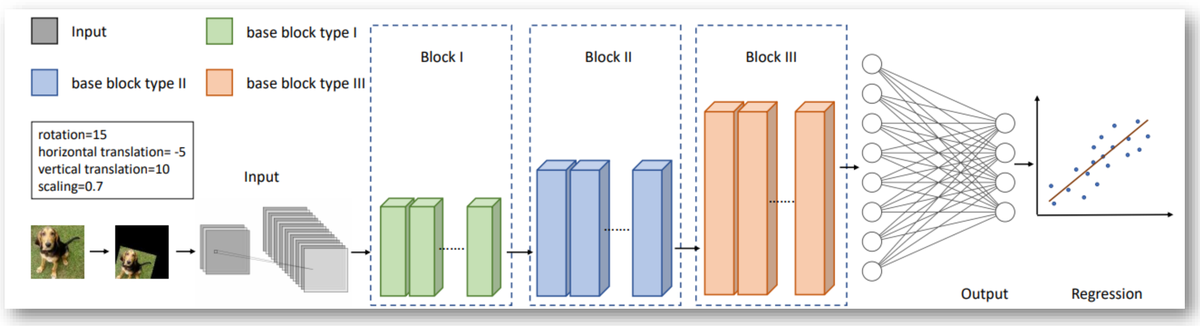
Our Approach

Objective

It is modeled as a regression task, where x_i is the ith sample, y_i is the transformation parameters and regression labels, T represents the transformation mapping (image rotation, translation, scaling or composition of them) parameterized by y_i , F is the neural network model parameterized by $\mathbf{W}$.

$$\min_{\mathbf{W}} \frac{1}{n} \sum_{i=1}^{n} \left\| \mathcal{F}\left(\mathcal{T}\left(\mathbf{x}_i; y_i\right); \mathbf{W}\right) - y_i \right\|_2^2$$

The whole framework is shown in the following figure.



Algorithm

Algorithm 1 Image Representation Learning Algorithm by Transformation Regression

Input: Image Dataset X; Transformation function $\mathcal{T}(\cdot, y)$ Neural Network $\mathcal{F}(\cdot, \mathbf{W})$.
Output: The parameters of the neural network $\mathbf{W}$.

```
Initialize the neural network parameter W;
for t in 1 to T do
    for i in 1 to n do
        Random sample a value y_i;
        Apply transformation: x̂_i = T(x_i; y_i).
        Forward pass to get the output F(x̂_i; W).
    end for
    Compute the loss L = (1/n) Σ_{i=1}^{n} ||F(x̂_i; W) − y_i||_2^2.
    Update the W by gradient descent.
end for
return W.
```

Empirical Results

We conduct an extensive evaluation of our method on the most commonly used image datasets, i.e., CIFAR10, CIFAR100, STL10, and SVHN. The classification task is chosen to evaluate the discriminability of the representation learned by our transformation regression learning method.

Data sets and Protocol

	#examples	#training examples	#testing examples	#classes	image size
CIFAR10	60,000	50,000	10,000	10	$32 \times 32 \times 3$
CIFAR100	60,000	50,000	10,000	100	$32 \times 32 \times 3$
STL10	13,000	10,000	3,000	10	$32 \times 32 \times 3$
SVHN	99,289	73,257	26,032	10	$32 \times 32 \times 3$

Protocol: 1) Train a network in a self-supervised way to learn representations. 2) Extract representations from different layers. 3) Use a classification model to validate the quality of representations (higher accuracy corresponds to better quality)

Result

Observations:
1) The method can learn good representations by comparing the Input with Block_i;

2) The intermediate representation (Block_1 or 2) is the best.

3) More samples lead to better representation by comparing STL10-10k and STL10-100k

Table: Classification accuracy of a MLP on representations

	Input	Block_1	Block_2	Block_3
CIFAR10	66.84	81.37	84.88	76.16
CIFAR100	39.60	53.19	57.06	44.07
STL10-10k	54.20	68.53	72.67	64.33
STL10-100k	54.20	71.00	77.03	67.80
SVHN	82.25	92.29	94.28	91.52

Table: Classification accuracy of a CNN on representations

	Input	Block_1	Block_2	Block_3
CIFAR10	93.37	93.65	90.75	81.44
CIFAR100	70.84	71.42	65.69	49.01
STL10-10k	77.93	79.93	77.20	65.50
STL10-100k	77.93	81.03	79.60	67.17
SVHN	96.12	96.54	95.67	92.63

State of the art:
Our method outperforms the other self-supervised methods and approaches the supervised method.

Table: Compare with the state of the art

	CIFAR10	CIFAR100	STL10	SVHN
DCGAN* [12]	82.80	–	–	–
Split-Brain† [27]	67.10	39.00	–	77.30
Counting† [28]	50.90	18.20	–	63.40
AND† [21]	77.60	47.90	–	93.70
RotNet* [15]	91.16	–	–	–
TR (Ours)	93.65	71.42	79.93	96.54
Supervised	94.92	75.76	80.10	96.45

Ablation Study

1) The composition of any two types of transformations leads to good performance
2) All three types of transformations achieve the best performance.

Rotation	Translation	Scaling	ACC (%)
0	0	[0.5, 1.5]	90.97
0	[-10, 10]	1.0	87.97
0	[-10, 10]	[0.5, 1.5]	91.48
[-180, 180]	0	1.0	87.99
[-180, 180]	0	[0.5, 1.5]	92.21
[-180, 180]	[-10, 10]	1.0	93.24
[-180, 180]	[-10, 10]	[0.5, 1.5]	**93.65**

Conclusion

We propose a new image representation learning method by constructing a regression task whose target is to predict the continuous parameters of some transformations applied to the input image. Extensive experiments on various image datasets validate the effectiveness and discriminability of representation learned by our proposed transformation regression method

Future work includes: 1) exploring other types of transformations like image flipping, cropping, and color jitter; and 2) eliminating the edge effect when applying some transformations like image rotation.