The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Visual Recognition in Very Low-Quality Settings: Delving into the Power of Pre-Training

Bowen Cheng,[*†] **Ding Liu,**[*†] **Zhangyang Wang,**[‡] **Haichao Zhang,**[§] **Thomas S. Huang**[†]
†Beckman Institute, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
‡Department of Computer Science and Engineering, Texas A&M University, College Station, TX 77843, USA
§Baidu Research, Sunnyvale, CA 94089, USA

Abstract

Visual recognition from very low-quality images is an extremely challenging task with great practical values. While deep networks have been extensively applied to low-quality image restoration and high-quality image recognition tasks respectively, few works have been done on the important problem of recognition from very low-quality images. This paper presents a degradation-robust pre-training approach on improving deep learning models towards this direction. Extensive experiments on different datasets validate the effectiveness of our proposed method.

Introduction

While visual recognition has made tremendous progress in recent years, most models are trained and evaluated on high quality image datasets, such as LFW and ImageNet. For most practical applications, the input images cannot be assumed to be of high quality. Therefore, it becomes highly desirable to investigate and improve the robustness of visual recognition systems in very low-quality settings.

Exiting studies demonstrate that most state-of-the-art models appear fragile when applied on low quality data. The literature has confirmed the significant effects of quality factors such as low resolution, contrast, brightness, sharpness, focus, and illumination on commercial face recognition systems. Besides face recognition, the low quality data are also found to negatively affect other recognition applications, such as hand-written digit recognition and style recognition (Wang et al. 2016).

We study this important but less explored problem, and carry out a systematic study on improving deep learning models towards the task. We generalize conventional unsupervised pre-training and data augmentation methods, and propose the Degradation-Robust Pre-Training algorithm, that is generally applicable to handling various low-quality inputs. The proposed algorithms are thoroughly evaluated on various datasets, with highly impressive performance improvements achieved.

Proposed Method

Notations We define the visual recognition model $\mathcal{M}$ that predicts the category labels $\{l_i\}_{i=1}^N$ from the images $\{\mathbf{y}_i\}_{i=1}^N$. In the very low-quality (LQ) settings, $\{\mathbf{y}_i\}_{i=1}^N$ can be viewed as LQ images, degraded from original high-quality (HQ) images $\{\mathbf{x}_i\}_{i=1}^N$. In testing, our model operates with only LQ inputs. We use a CNN based image recognition model $\mathcal{M}$ with d layers. The first d_1 layers are convolutional, while the remaining $d - d_1$ layers are fully connected. The i-th convolutional layer, denoted as $conv_i$ $(i = 1, \cdots, d_1)$, contains n_i filters of size $c_i{\times}c_i$, with default stride size 1 and zero-padding. The j-th fully-connected (fc) layer, denoted as fc_j, $j = 1, \cdots, d - d_1$, has a dimensionality of m_j. We use ReLU activation and apply dropout with a rate of 0.5 to fully-connected layers. Softmax loss is adopted for classification, while mean square error (MSE) for reconstruction tasks.

Algorithm 1 Degradation-robust pre-training

```
Input: Configuration of $\mathcal{M}$; $\{\mathbf{x}_i\}$ and $\{l_i\}$, $i = 1, ..., N$; the
    choice of $k$; two degradation factors $\alpha$ and $\beta$ $(\beta \geq \alpha)$.
1: Generate $\{\mathbf{y}_i\}$, $\{\mathbf{z}_i\}$ from $\{\mathbf{x}_i\}$, based on two degradation processes parameterized by $\alpha$ and $\beta$, respectively.
2: Construct the $(k+1)$-layer sub-model $\mathcal{M}_s$. Its first $k$ layers are configured identically to those of $\mathcal{M}$.
3: Train $\mathcal{M}_s$ to reconstruct $\{\mathbf{x}_i\}$ from $\{\mathbf{z}_i\}$, using MSE.
4: Export the first $k$ layers from $\mathcal{M}_s$ to initialize the first $k$ layers of $\mathcal{M}$.
5: Tune $\mathcal{M}$ over $\{\{\mathbf{y}_i\}, \{l_i\}\}$, using the softmax loss.
Output: $\mathcal{M}$.
```

Degradation-Robust Pre-Training Algorithm Since the performance of $\mathcal{M}$ trained over $\{\{\mathbf{x}_i\}, \{l_i\}\}$ will be drastically degraded when applied to LQ subjects, our main intuition is to jointly utilize $\{\{\mathbf{x}_i\}, \{\mathbf{y}_i\}, \{l_i\}\}$, such that the feature extraction from $\{\mathbf{y}_i\}$ could be enhanced and regularized by the ground-truth $\{\mathbf{x}_i\}$, whereas the entire $\mathcal{M}$ is also well adapted for the mapping relationship from $\{\mathbf{y}_i\}$ to $\{l_i\}$.

We propose to pre-train the first k layers of $\mathcal{M}$ in model $\mathcal{M}_s$, to reconstruct $\{\mathbf{x}_i\}$ from $\{\mathbf{y}_i\}$. Since the set $\{\mathbf{x}_i\}$ introduces auxiliary information to these feature extraction layers, it guides them to discriminate the true signal information from the degradation errors. After that, the pre-trained k feature extraction layers are jointly tuned with the remaining layers of $\mathcal{M}$, for the classification task.

Note that different from the testing case where only LQ

[*] denotes equal contribution.

	HQ	LQ-2	LQ-2-non-joint	LQ-2	LQ-2-4	LQ-2-8	LQ-2-12	LQ-2-16
Top-1	67.43%	60.79%	46.89%	62.12%	62.80%	**63.31%**	62.91%	62.56%
Top-5	96.61%	95.32%	90.77%	95.10%	95.52%	**95.80%**	95.34%	95.10%

Table 1: The top-1 and top-5 classification accuracies on the CIFAR-10 dataset, where LQ images are generated by downsampling the original images with a factor of α=2.

	HQ	LQ-50%	LQ-50%-no-joint	LQ-50%
Top-1	67.43%	33.46%	38.64%	**50.32%**
Top-5	96.61%	83.22%	86.86%	**92.03%**

Table 2: The top-1 and top-5 classification accuracies on the CIFAR-10 dataset, where LQ images are generated by adding α=50% salt & pepper noise.

	HQ	LQ-2	LQ-2-non-joint	LQ-2	LQ-2-5	LQ-2-8	LQ-2-9
Top-1	67.43%	52.62%	39.80%	54.73%	54.77%	**55.67%**	54.35%
Top-5	96.61%	92.70%	87.34%	93.24%	93.50%	**93.52%**	93.15%

Table 3: The top-1 and top-5 classification accuracies on the CIFAR-10 dataset, where LQ images are generated by blurring original images (HQ), with Gaussian kernel of std α=2.

	HQ	LQ-8	LQ-8-non-joint	LQ-8-joint	LQ-8-16-joint
Top-1	89.23%	19.60%	45.98%	51.00%	**51.17%**
Top-5	98.57%	65.44%	87.08%	**89.15%**	89.06%

Table 4: The top-1 and top-5 digit recognition accuracies on the SVHN dataset, where LQ images are downsampling the original images (HQ) by factor of α=8.

	HQ	LQ-2	LQ-2	LQ-2-	LQ-2-5-	LQ-2-8-
			non-joint	-joint	joint	joint
Top-1	89.23%	85.40%	83.84%	82.47%	**89.40%**	88.29%
Top-5	98.57%	97.55%	96.92%	96.82%	**98.32%**	98.09%

Table 5: The top-1 and top-5 digit recognition accuracies on the SVHN dataset, where LQ images are generated by blurring the original images (HQ), with the Gaussian kernel of standard deviation α=2.

images are available, we have the flexibility to generate LQ images for training at our will. Furthermore, we explore the possibility to train $\mathcal{M}$ and $\mathcal{M}_s$ on different LQ settings, expecting that $\mathcal{M}_s$ could learn more robust feature mapping. This algorithm, termed as Degradation-Robust Pre-Training, is outlined in Algorithm 1.

Experiments

Object Recognition We experiment with the CIFAR-10 dataset (Krizhevsky 2009) for this task, which consists of 60,000 32×32 color images from 10 classes. Each class has 5,000 training images and 1,000 test images. We choose $\mathcal{M}$ with d=4, with d_1=3 convolutional layers, followed by $d-d_1$=1 fully-connected layers with m_1 always being the number of classes. We set $\mathcal{M}_s$ with k=2, which works well in all experiments. The default configuration of convolutional layers are: n_1=64, c_1=9; n_2=32, c_2=5; n_3=20, c_3=5.

Low Resolution We generate LQ (low-resolution) images $\{\mathbf{y}_i\}$ and compare the following cases: **HQ:** $\mathcal{M}$ is trained and tested on $\{\{\mathbf{x}_i\}, \{l_i\}\}$. **LQ-$\alpha$:** $\mathcal{M}$ is trained and tested on $\{\{\mathbf{y}_i\}, \{l_i\}\}$. **LQ-$\alpha$-non-joint:** Generate both $\{\mathbf{y}_i\}$ and $\{\mathbf{z}_i\}$ with downsampling factor α. $\mathcal{M}_s$ is pre-trained as in Algorithm **??** ($\alpha = \beta$), on $\{\{\mathbf{y}_i\}, \{\mathbf{x}_i\}\}$. The remaining d–k layers of $\mathcal{M}$ are then trained on $\{\{\mathbf{y}_i\}, \{l_i\}\}$, with the first k pre-trained layers fixed. **LQ-α:** $\mathcal{M}$ is trained using Algorithm **??** ($\alpha = \beta$). **LQ-α-β:** $\mathcal{M}$ is trained using Algorithm **??** ($\alpha < \beta$). The evaluation of $\mathcal{M}$ is all performed on the test set of LQ images (except for the HQ baseline), downsampled by the factor α. Table 1 displays the results at α=2.

Noise Since adding moderate Gaussian noise has been standard for data augmentation, we focus on the more destructive salt & pepper noise. The LQ images $\{\mathbf{y}_i\}$ are generated by randomly choosing α=50% pixels in each HQ image $\mathbf{x}_i$ to be replaced with either 0 or 255. We compare HQ, LQ-α, LQ-α-non-joint, and LQ-α, similarly defined as the low-resolution case. The results are displayed in Table 2.

Blur Images commonly suffer from various types of blurs. Here we only focus on the Gaussian blur, while similar strategies can be naturally extended to other types. The LQ images $\{\mathbf{y}_i\}$ are generated by convolving the HR images $\{\mathbf{x}_i\}$ with a Gaussian kernel with std $\alpha = 2$, and the fixed kernel size 9×9. We compare HQ, LQ-α, LQ-α-non-joint, LQ-α, and LQ-α-β. The results are displayed in Table 3.

Overall, our proposed algorithm achieves the best result.

Digit Recognition We use the Street View House Number (SVHN) dataset (Netzer et al. 2011) for this task, which contains 73, 257 digit images of 32×32 for training, and 26, 032 for testing. Our model has a default configuration of d=4, d_1=2; n_1=20, c_1=5; n_2=50, c_2=5; m_1=500; m_2=10 (class number used). $conv_1$ is followed by 2×2 max pooling. Table 4 compares HQ, LQ-α, LQ-α-non-joint, LQ-α-joint and LQ-α-β-joint, in the low resolution case with α=8. Table 5 compares those methods in the Gaussian blur case with standard deviation α=2. Our proposed method performs the best.

References

Krizhevsky, A. 2009. Learning multiple layers of features from tiny images.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; and Ng, A. Y. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop*, 5.

Wang, Z.; Chang, S.; Yang, Y.; Liu, D.; and Huang, T. S. 2016. Studying very low resolution recognition using deep networks. In *CVPR*. IEEE.