

Image recognition with deep neural networks in presence of noise – Dealing with and taking advantage of distortions

Michał Koziarski* and Bogusław Cyganek
AGH University of Science and Technology, Kraków, Poland

Abstract. Data classification in presence of noise can lead to much worse results than expected for pure patterns. In this paper we investigate this problem in the case of deep convolutional neural networks in order to propose solutions that can mitigate influence of noise. The main contributions presented in this paper are experimental examination of influence of different types of noise on the convolutional neural network, proposition of a deep neural network operating as a denoiser, investigation of a deep network training with noise contaminated patterns, and finally an analysis of noise addition during the training process of a deep network as a form of regularization. Our main findings are construction of the deep network based denoising filter which outperforms state-of-the-art solutions, as well as proposition of a practical method of deep neural network training with noisy patterns for improvement against the noisy test patterns. All results are underpinned by experiments which show high efficacy and possibly broad applications of the proposed solutions.

Keywords: Image recognition, deep neural networks, convolutional neural networks, noise, image denoising, regularization

1. Introduction

Research field of computer vision changed significantly over the recent years, mostly due to the advances made in the area of deep learning [28]. In this time, deep neural networks were successfully used in various practical applications [35,37,38]. In particular, convolutional neural networks were able to achieve state-of-the-art results in the task of image recognition [20,45,47], in many cases surpassing the human capabilities. Despite the significant amount of research done in this area, most of the work revolves around benchmark datasets, consisting of fairly high quality images. In real-life applications, however, we are often faced with low quality data, distorted by different types of noise, affected by motion blur, difficult lighting and weather conditions, low resolution, or a combination of

these factors, to name a few. Furthermore, their nature is not always known *a priori*. Thus, in many cases resilience to previously unmet types of distortions is necessary. The impact of image quality on performance of computer vision algorithms is often overlooked, which may in turn lead to unrealistic expectations in practical applications.

In this paper we try to answer the questions on influence of the presence of various types of noise on the image recognition task with deep neural networks. We further investigate how severely can it affect the classification accuracy and what are the possible solutions to eventual drop in performance. Finally, we address the question if presence of noise can be used to our advantage. To answer these questions we performed an extensive experimental study on one of the landmark neural architectures of recent years, VGG model [45] (proposed by and named after Oxford Visual Geometry Group). We measured the impact of various noise conditions, with both known and unknown distributions, on the classification performance. We evaluated different possibilities of dealing

*Corresponding author: Michał Koziarski, AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Kraków, Poland. E-mail: michal.koziarski@agh.edu.pl.

with those distortions, namely augmenting the training data versus applying fully convolutional denoising prior to classification. Finally, we evaluated the possibility of artificially inducing noise as a form of regularization, in hope of observing a boost in performance. It was shown that small doses of synthetic distortions applied during the training procedure are equivalent to certain forms of regularization [5,34,54,55]. However, this further complicates the relation between noise levels present in images and performance in classification task, especially considering the prevalence of regularization techniques already used in combination with the deep learning models.

The rest of the paper is organized as follows. In Section 2 we briefly outline related works on deep learning and noisy pattern classification. In Section 3 we describe mathematical noise models used throughout the experimental evaluation. Section 4 presents a possibility of taking advantage of noise in image recognition tasks with deep neural architectures. In this respect two strategies are considered, namely data augmentation (Section 4.1) as well as prior data denoising (Section 4.2). In Section 4.3 the problem of noise as a form of regularization of the deep neural network is discussed. Experimental results are presented in Section 5. Conclusions and further research directions are outlined in Section 6.

2. Related work

In recent years deep learning [18] received significant amount of attention from research community, leading to numerous breakthroughs in the area of pattern classification. In context of image recognition problem, advances made in crafting neural architectures were of particular interest. Increasing depth of the networks was for a long time one of the main challenges associated with neural models. Deeper networks, while offering a promise of better discriminative properties, were always difficult to train. Various novel approaches to increasing depth of networks were presented over the recent years [20,28,45,47]. This together with larger and larger training datasets, processed by powerful graphical processing units (GPU), led to ever-increasing performance on benchmark data.

Most of the research on neural architectures was conducted in rather sterile conditions, however. Images in commonly used benchmarks usually contain relatively small amounts of distortions. On the other hand, in practical applications noise is often ubiquitous. As-

sessing the impact of distortions on image recognition accuracy was, therefore, an important research endeavor. The problem was relatively a new one, though. Only several papers examined how image quality affects convolutional neural networks [15,24,27,42,51]. In this respect, various forms of degradation were considered, including noise, blur, contrast and occlusion. Research on the impact of noise is also not limited to the image distortions: Massouh et al. [33] measured experimentally how presence of label noise affects the classification accuracy of deep neural networks. Some work has also been done in combination with other types of classification algorithms [13,16,57]. Based on the existing research it is clear that even relatively small levels of distortions can significantly influence image recognition task. Presence of noise in test data negatively affects the classification accuracy, oftentimes making the correct prediction infeasible.

Various techniques of dealing with low-quality data in image classification have been presented in the literature [48,49]. For instance, Tan and Triggs proposed using special feature sets in presence of difficult lighting conditions. Even though described method was not relying on neural classifiers, it could be speculated that learning features with similar characteristics is possible for convolutional networks [48]. On the other hand, Peng et al. [36] examined the case of low-resolution images. They employed transfer learning to reuse knowledge gained from high-resolution data to low-resolution case. This approach is particularly useful when low-quality images are difficult to obtain. The possibility of incorporating image quality measures into the classification procedure was also examined in several papers [3,29].

Another approach to deal with distortions relies on applying restoration techniques prior to classification. Possibility of using neural networks for image restoration has been extensively studied, particularly in combination with noise [4,22,26] and blur [7,10]. Saatci and Tavsanoğlu [41] applied convolutional networks for image enhancement. Eigen et al. [17] considered distortions with characteristic spatial structure, namely rain and dirt. Finally, Chaudhury and Roy [9] evaluated the possibility of using convolutional networks in general restoration problem [25]. Most important contributions of these papers are presented therein models of the neural networks, capable of highly effective restoration of images. Overall, neural networks achieve state-of-the-art results in image restoration task and display high robustness to the type of distortion. An interesting approach to learning to deblur with con-

volutional neural networks is proposed by Schuler et al. [43]. In their approach a learning-based deep structure for blind image deconvolution is employed. Their system is trained end-to-end on a set of artificially generated blurred training examples. The system then automatically learns the deconvolution kernel. Nevertheless, despite very good results for smaller kernels, scalability of the proposed method to larger ones is still an issue. Moreover, the mentioned method by Schuler et al. [43] is only for image deconvolution without considering further classification step.

Closely related to the idea of using restoration as a form of preprocessing is the notion of autoencoder [30, 32, 52, 53]. Autoencoders can be used to extract useful image representations in an unsupervised manner. Denoising autoencoders achieve that as a result of reconstructing artificially distorted images. Because of that, produced feature representations should, in principle, be robust to the used type of signal corruption.

Finally, the addition of noise to training images can, in some cases, lead to increased generalization performance. Bishop [5] proved that adding noise is equivalent to another established regularization technique, Tikhonov regularization. Further theoretical analysis of noise injection was later conducted by Grandvalet et al. [19], Rifai et al. [39], as well as by Simard et al. [44]. Neelakantan et al. [34] evaluated the possibility of noise injection at the gradient level, which was shown to not only reduce overfitting, but also to lower the training loss and reduce the impact of poor initialization in very deep networks.

3. Noise models

Noise is unwanted signal that affects the original one. It comes as an effect of some physical phenomena encountered in the process of signal acquisition and transmission [6, 12]. Noise is usually modeled as a random multiplicative or additive component added to the pure signal. Assuming that $s(x)$ denotes a pure signal, the two aforementioned noise models are given respectively as follows

$$\hat{s}(x) = \eta + s(x), \quad (1)$$

$$\hat{s}(x) = \eta s(x), \quad (2)$$

where \hat{s} denotes an observable signal contaminated with noise, whereas η is a random variable characterized by its probability density function specific to the type of noise, as will be discussed. Noise introduces

distortions which make detection and analysis of a pure signal component difficult. Noise has also an influence on pattern classification, since deteriorating the training and test patterns with noise also affects their statistical properties. Moreover, in practice it is usually difficult to tell a type of noise which affect the input signals. However, as we show in this paper, design of classifiers that account for this phenomena can mitigate its deteriorating influence to a high degree.

3.1. Gaussian noise

Thermal effects in electronic devices, as well as photon counting and film grain phenomena lead to a type of noise which is represented with the additive noise model Eq. (1) with random variable characteristic of the Gaussian density function. Therefore, this type is called the Gaussian noise. The associated Gaussian density function is given as follows

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (3)$$

where μ and σ , in this case, are noise parameters. Usually $\mu = 0$ and noise is controlled only by the parameter σ . Such setting with one control parameter σ is also assumed in our paper, as will be further discussed.

3.2. Quantization noise

Quantization noise arises as a result of discretization of a continuous signal into its discrete counterpart. Each signal sample has assigned a finite number of bits, which inevitably superimposes a limit on the lowest value that can be represented with no errors. For a sufficient number of quantization levels, this type of noise is modelled in accordance with Eq. (1) and η being a random variable that fulfills the following inequality

$$-\frac{1}{2}q \leq \eta \leq +\frac{1}{2}q, \quad (4)$$

where q denotes a quantization level, and characterized by the uniform probability distribution function p , as follows

$$p(x) = \begin{cases} \frac{1}{x_{\max} - x_{\min}}, & \text{for } x_{\max} \leq x \leq x_{\min} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where x_{\max} and x_{\min} are the maximum and minimum values of the argument x . Thus, the random variable η takes the values $\pm \frac{1}{2}q$ with a uniform distribution Eq. (5), where q is a quantization parameter set in the experiments, as will be discussed.

3.3. Salt and pepper noise

Salt-and-pepper noise arises as a result of transmission bit errors or an analog-to-digital converter errors. Its name comes from a characteristic white and black spots in an image caused by a flip of usually the most significant bit of a pixel representation. This type of noise can be modelled by a combination of the multiplicative Eq. (2) and the additive Eq. (1) noise models, respectively, as follows [8]

$$\hat{s}(x) = (1 - \eta)s(x) + \eta\beta, \quad (6)$$

where $s(x)$ and $\hat{s}(x)$ denote pure and noise contaminated signals, respectively, η is a random variable with the probability $p = Pr(\eta = 1)$ and β is a random variable such that $Pr(\beta = s_{\max}) = Pr(\beta = s_{\min}) = 0.5$. The process of generating salt and pepper noise can be interpreted as double drawing process. At first a random variable η is generated with a probability p of the event $\eta = 1$. Then, if $\eta = 1$ occurred, β is drawn and checked if it is s_{\max} or s_{\min} . Thus, p is the only control parameter for this type of noise, as will be discussed when presenting experimental results.

3.4. Randomly selected noise models

In a practical setting it is often difficult to determine the type of the noise affecting the signal. Images can be influenced by a multiple sources of noise simultaneously, each one with its own characteristic. Furthermore, severity of noise usually also varies between the images. To capture this additional level of uncertainty we introduce two extended noise models, as follows.

- In the *noise of random intensity* the parameter associated with the noise model, specifically: σ for the Gaussian noise, q for the quantization noise and p for the salt & pepper noise, will be sampled randomly from a uniform distribution.
- In the *noise of random type* the underlying noise model will be selected randomly, with equal probabilities being assigned to picking each model.

Finally, let us notice that in practice adding noise to the images should be careful to avoid numerical overruns which would result in additional noise by themselves. This happens for instance when using the additive noise model Eq. (1) causes pixel value overflow, i.e. the new value exceeds a number of bits assigned to a single pixel. In such a case, instead of adding the Gaussian noise Eq. (1) we would generate the salt and pepper one. Further practical ways of noise addition to images for experimentation can be looked up in [12].

4. Dealing with and taking advantage of noise in image recognition task

Considering the possible influence of distortions on image recognition task with deep neural networks, we focus on two data-centric methods of dealing with noise: training with noise-augmented patterns and using denoising as a form of preprocessing. In the remainder of this section we discuss them both in more detail, describing possible advantages and limitations of using them. Finally, we discuss the possibility of using noise as a regularizer, especially in context of training with the data augmentation strategy.

4.1. Augmenting training data

Conceptually simplest approach to deal with noise in pattern recognition task is augmenting the training data with some form of an expected noise. It is particularly convenient if large quantities of images with real distortions are available at our disposal. In practical applications it might not be the case, however. Obtaining labeled data might be also expensive, especially if it has to be distorted in a particular way. Furthermore, characteristics of noise present in images might change over time, additionally increasing difficulty of capturing distorted data.

If the properties of expected distortions can be described mathematically, a suitable alternative might be augmenting data with synthetic noise. This, however, requires knowing the type of distortion that will affect the images prior to training. Augmenting data with too high amounts of noise or wrong type of distortion might negatively influence further classification performance. Impact of such augmentation is also affected by the quality of used noise model.

4.2. Prior data denoising

An alternative approach to classify noisy patterns is to train a classification model on undistorted data, and afterwards to apply denoising as a form of preprocessing. Recent advances in neural denoising might indicate that such preprocessing will be sufficient to obtain images of necessary quality. We consider solely the denoising with neural networks for several reasons. First of all, it often outperforms other state-of-the-art methods [43]. While to do so, large quantities of data are often necessary, this issue is less severe since the data is required for classification either way. Even more so, images used for training denoising model do not have

to be labeled. In that context, training denoising model can be viewed as an unsupervised pretraining of the final classifier, possibly done using larger distribution of unlabeled data.

Using neural architecture to denoise images has some additional benefits. First of all, it enables the possibility of finetuning of the final model. Secondly, since denoising in principle produces images of sufficient quality, transfer of further layers from preexisting models, trained on undistorted data, is possible. This is particularly beneficial due to the long training times of classification architectures.

4.3. Using noise as a form of regularization

Regularization is a well studied problem in context of neural networks. Applying small doses of noise during training procedure is, in particular, known to improve generalization properties of networks [5]. However, given the abundance of other regularization techniques such as weight decay, dropout [54] and adaptive regularization [55], it is questionable whether applying yet another form of regularization is beneficial.

Furthermore, the possible negative influence of applying noise as a form of regularization is also relevant. It is of particular importance when considering the training data augmentation strategy. If the noise conditions during model evaluation are not certain, we might augment data with too severe distortions. In this context, by measuring negative impact of too severe regularization, we examine how model would behave when augmentation noise is not chosen correctly.

5. Experimental study

To evaluate the impact of noise on image recognition task we performed an extensive experimental study. We tested VGG [45] architecture under various noise conditions, both known and unknown. As already mentioned, we evaluated two approaches of dealing with noise, augmenting training data and applying denoising. Finally, we measured the impact of noise in training data, and its usefulness as an additional form of regularization. In the remainder of this section we give a detailed description of experimental set-up, present achieved results and discuss their implications.

5.1. Experimental set-up

All of the conducted experiments were implemented in Python programming language and were using Ten-

sorFlow [2] library for numeric computation. Produced code, sufficient to easily repeat all the experiments, was made publicly available.¹

Several types of synthetic noise models, already discussed in Section 3, were used with various parameters. Gaussian, quantization and salt & pepper noise models with respective parameters taking values from the set {0.05, 0.1, 0.2, 0.5} were used to evaluate the case with known noise conditions. The choice of parameters for noise models was dictated by the need to cover both less severe, as well as very severe noise conditions. At the same time the number of considered parameters had to be limited due to the computational constraints. On the other hand, to evaluate the situation in which noise conditions are unknown, we used the same noise models with parameters sampled uniformly from range from 0.0 to 0.5 at every iteration. Finally, we considered the case in which the noise model itself, as well as distortion intensity, are chosen randomly.

5.2. Dataset

ImageNet [40] dataset, consisting of color images of varied resolution, was used throughout all the stages of the conducted experimental study. Specifically, subset of ImageNet images provided during the Large Scale Visual Recognition Challenge 2011 (ILSVRC2011) was used [1]. It consisted of 1.2 million train images, as well as 50 thousand labeled validation images, used to evaluate the final performance of the models. Each of the images was assigned a single label, depicting one of the 1000 possible object categories. ImageNet dataset, in particular its subset provided during the ILSVRC2011, is publicly available at [1] and can be used to reproduce the results achieved in this paper.

5.3. Network architectures

During the classification task, the VGG network architecture presented by Simonyan et al. was used [45]. Since achieving the highest possible accuracy was not the main goal of the presented experimental study, to accelerate the computation speed the simplest of the proposed models was employed. It consisted of 8 convolutional layers, with pooling applied after first, second, fourth, sixth and eighth layer. The convolutional layers were comprised of 3×3 filters, with the number of filters increasing from 64 in the first layer, to 128 in

¹<https://github.com/michalkoziarski/dnoise>.

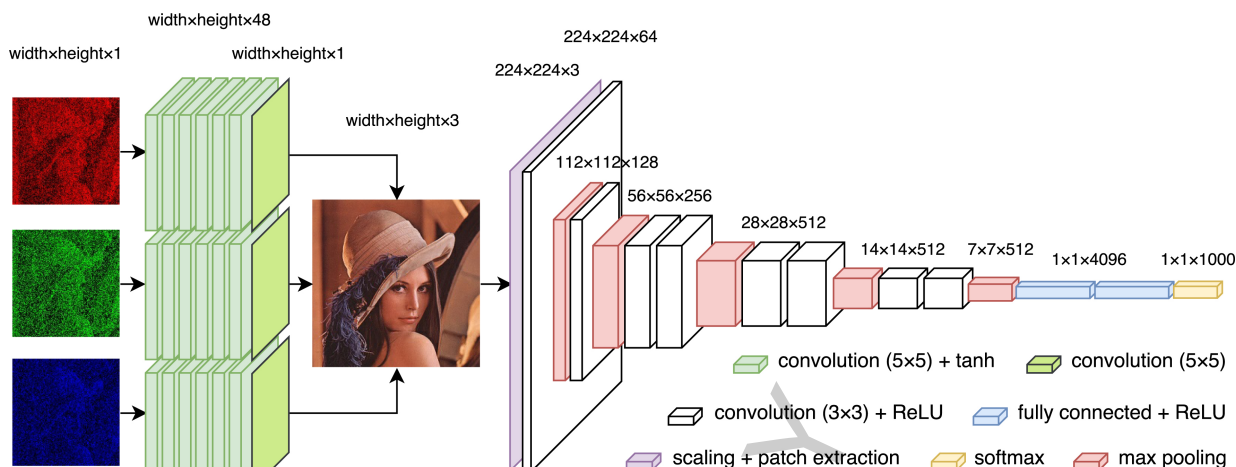


Fig. 1. Graphical representation of combined denoising and classification architectures. Dimension of data after passing through specific layers indicated at the top.

the second, 256 in the third and fourth, up to 512 in the remaining layers. After the convolutional layers, network consisted of 3 fully connected layers, with 4096 neurons per first two layers and 1000 neurons in the last layer. Rectified linear unit (ReLU) activation function was used after all the layers except the last, after which softmax activation function was used.

For denoising we used extension of the previously tested architecture, presented by Koziarski and Cyganek [26] and based on the work by Jain and Seung [22]. It consisted of 7 convolutional layers per color channel. First 6 layers were comprised of $48 \ 5 \times 5$ filters each, followed by the hyperbolic tangent activation function. The final layer consisted of a single 5×5 filter, introduced to preserve the original shape of images after passing through the network.

Graphical representation of our complete neural architecture, consisting of both denoising and classification networks, is presented in Fig. 1.

In the both cases of classification and denoising networks, the choice of the model was dictated by the availability of the previous experimental studies ([45] in the case of classification, and [26] for denoising), confirming the validity of choice of the architecture. It allowed us to limit the tests of various model variations. Specifically, no further tests of different variants of classification network were conducted. In the case of the denoising model, training separate network for each color channel turned out to be crucial to achieve a good performance.

In the case of classification model, as far as the resilience to the noise is considered, it is not clear whether the trends observed for VGG architecture will

hold for different models. Having said that, results presented in this paper show similar trends to our previous work [26,27], in which impact of noise on a different neural architectures is measured on STL-10 [11] and GTSRB [21] datasets. While it may suggest that observed trends are more general, both across datasets and neural models, further evaluation would be required to confirm this hypothesis. In this paper we limited the evaluation to a single architecture due to the high computational overhead associated with training of the networks.

5.4. Training procedure

In classification task, the stochastic gradient descent method was used to minimize the cross-entropy objective function. Constant learning rate of 0.001, with momentum of 0.9, were used throughout the training. The weight decay was set to 0.0005 and a dropout of 0.5 after all hidden, fully-connected layers were used as a form of regularization. Choice of the hyperparameters was motivated by their values reported in previous research [45]. Random patches were extracted from the original images. First of all, images were rescaled so that their shorter dimension was equal to 224 pixels. Secondly, they were cropped to the size of 224×224 pixels. Cropping was performed randomly every time the image was fetched. Training was conducted in batch mode, with batches consisting of 50 images each. The whole training procedure lasted 100 epochs. After the training we observed that the classification accuracy saturated sooner in most cases, oftentimes as

Table 1

Average values of PSNR for different types of noise and denoising methods. Distortion level shown in respect to an original image. The best filtering result shown in bold. The method proposed in this paper (CNN, the convolutional neural network) was compared with reference algorithms. Chosen values of parameters shown in subscript, namely: window size for median filtering, σ_s and σ_r for bilateral filtering, and σ for the BM3D algorithm

Type of noise	Distortion level	Median	Bilateral	BM3D	CNN (this paper)
Gaussian (0.05)	26.40	26.87 ₍₃₎	31.70 _(0.05, 3)	31.10 _(0.05)	28.04
Gaussian (0.1)	20.60	24.44 ₍₃₎	27.50 _(0.1, 3)	27.22 _(0.1)	25.99
Gaussian (0.2)	15.09	21.98 ₍₇₎	23.08 _(0.2, 3)	22.97 _(0.2)	23.91
Gaussian (0.5)	9.18	18.55 ₍₉₎	17.07 _(0.4, 7)	16.86 _(0.3)	21.32
Quantization (0.05)	31.19	26.53 ₍₃₎	29.74 _(0.05, 3)	29.44 _(0.05)	29.89
Quantization (0.1)	25.24	23.63 ₍₃₎	25.53 _(0.05, 3)	25.36 _(0.05)	28.71
Quantization (0.2)	19.37	19.29 ₍₃₎	20.20 _(0.05, 3)	20.06 _(0.05)	26.74
Quantization (0.5)	12.01	12.40 ₍₃₎	12.87 _(0.2, 3)	12.76 _(0.1)	23.61
Salt & pepper (0.05)	17.74	28.21 ₍₃₎	25.36 _(0.2, 5)	25.06 _(0.1)	24.79
Salt & pepper (0.1)	14.73	27.32 ₍₃₎	23.38 _(0.2, 7)	23.05 _(0.2)	23.84
Salt & pepper (0.2)	11.72	24.66 ₍₃₎	20.81 _(0.3, 7)	19.93 _(0.3)	22.59
Salt & pepper (0.5)	7.74	20.24 ₍₇₎	15.36 _(0.5, 7)	14.89 _(0.4)	20.42
Random (Gaussian)	15.90	21.31 ₍₇₎	21.49 _(0.3, 3)	20.75 _(0.3)	21.75
Random (quantization)	20.30	18.73 ₍₃₎	19.91 _(0.05, 3)	19.60 _(0.05)	22.32
Random (salt & pepper)	12.27	23.47 ₍₅₎	20.00 _(0.3, 7)	19.14 _(0.3)	20.65
Random (mixture)	16.13	20.94 ₍₅₎	20.06 _(0.3, 3)	19.26 _(0.3)	20.45

soon as after 50 epochs. However, further training did not lead to overfitting.

As an optimization criterion in the denoising task, the mean squared error between the original and the artificially distorted images was chosen. The same learning rate, momentum and batch size as in the classification task were used. No weight decay was employed, however. Different choices of learning rate, momentum and weight decay were evaluated, but conducted tests indicate that neither of these parameters has a significant impact on the denoising performance. Prior to denoising images were normalized to the range 0 to 1. During learning, randomly selected 64×64 patches were used to speed up the training procedure. In this case training lasted 10 epochs. Post-training observation of denoising accuracy, measured as a Peak Signal-to-Noise Ratio (PSNR) [6], showed significant decrease in improvement speed. However, it did not fully saturate, which may indicate the possibility of achieving a slight improvement over the reported results. Using PSNR measure in this type of comparison is frequent among researchers since it is defined in the same way regardless of the image content [6,31]. We also consider it suitable to the presented tests. However, other measures, such as the psychovisual ones, can be also considered in future research [56].

5.5. Denoising

Experimental evaluation began with assessing the performance of the proposed denoising strategy. It was compared with three other denoising algorithms,

which do not rely on neural learning. These are the median filter [31], the bilateral filter [50] and the BM3D filtering [14]. Further information on these and other state-of-the-art image filtering methods can be accessed e.g. in [6,31,46]. Parameters of the baseline algorithms were finetuned for specific noise conditions. For median filtering, windows of size 3×3 , 5×5 , 7×7 , 9×9 , 11×11 and 13×13 were considered. On the other hand, for the bilateral filtering, values of its control parameters $\sigma_s \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$ and $\sigma_r \in \{3, 5, 7\}$ were employed. Lastly, for the BM3D method, the values $\sigma \in \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$ were used.

Comparison of denoising strategies was conducted on 2000 randomly selected images from the ImageNet dataset. Chosen number of images was limited due to the computational constraints. Having said that, evaluation was first conducted on 20, later on 50, and finally on 2000 images. In all the cases trends were identical, whereas the differences between average values of PSNR were negligible. Results are presented in Table 1. For all the baseline methods, only the best choice of parameters for particular noise condition was reported. It is well visible that the deep network denoising method, proposed and evaluated in this paper, outperformed each of the aforementioned reference methods. However, interestingly enough, the deep architecture is especially efficient for larger levels of distortions. On the other hand, an exception is the case of the mild salt & pepper noise. The last effect might be caused by poor operation of the convolutional layers with this way distorted signals.



Fig. 2. Sample images before (lower left half) and after denoising with our proposed method (upper right half). Gaussian, quantization and salt & pepper noise of varying intensity was considered. Peak signal-to-noise ratio after denoising was also specified.

To assess the statistical significance of the observed results we conducted the Wilcoxon signed-rank test [23]. The proposed denoising strategy based on the convolutional neural network achieved significantly better performance than the bilateral filtering and the BM3D algorithm, at the significance level of $\alpha = 0.05$. On the other hand, we were unable to reject the null hypothesis when compared with the median filter. However, analyzing the results in Table 1 we notice that their competence regions are complementary and the median filter performs the best when processing salt & pepper noise, which is not a surprise. Nevertheless, the results of the statistical analysis further validate the choice of the convolutional network over conventional denoising algorithms, especially when connected with the classification process, as will be presented in the next section.

Our proposed method achieved high robustness to both type and intensity of noise, and relatively high quality of denoising. Particularly good performance was observed when dealing with quantization noise, as well as the most severe distortions of other types. Especially the latter property is very encouraging.

Sample images which were denoised using our proposed approach are presented in Fig. 2.

5.6. Classification in presence of noise

Performance of the discussed neural network was evaluated under varying noise conditions, present in either the training data, the test data, or both. The metric used to measure the performance of the model was the classification accuracy: proportion of test images for which the ground truth label and the prediction of the model matched. Evaluation began by measuring the accuracy of the network without any distortions applied (baseline case, referred to as C2C). Afterwards, an impact of the noise in the test data, not accounted for during the training procedure was measured (C2N). Relationship of the final classification accuracy in respect to different types and levels of noise is presented in Fig. 3. Even relatively small amounts of distortions significantly influenced performance of the network. Presence of noise with higher intensity, when not accounted for, made the network unable to properly recognize presented objects.

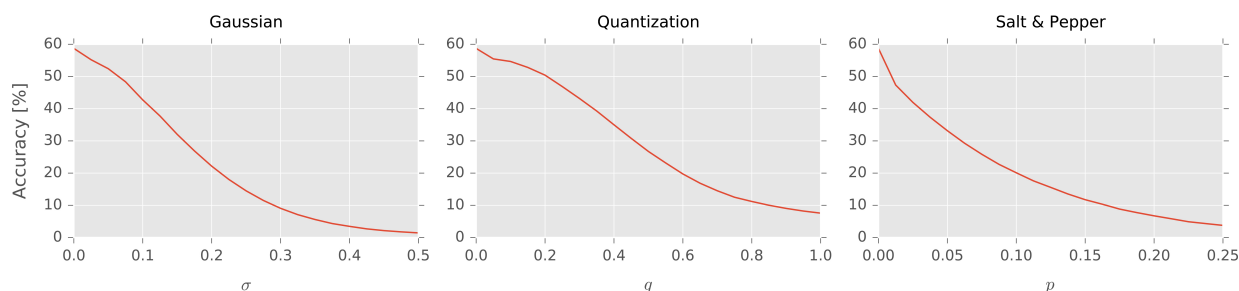


Fig. 3. Relation between intensity of various types of noise and classification accuracy. Standard deviation for Gaussian noise, probability of flipping pixel for salt & pepper noise and the range of distortion for quantization noise were adjusted. Noise was applied before resizing images to fit the network.

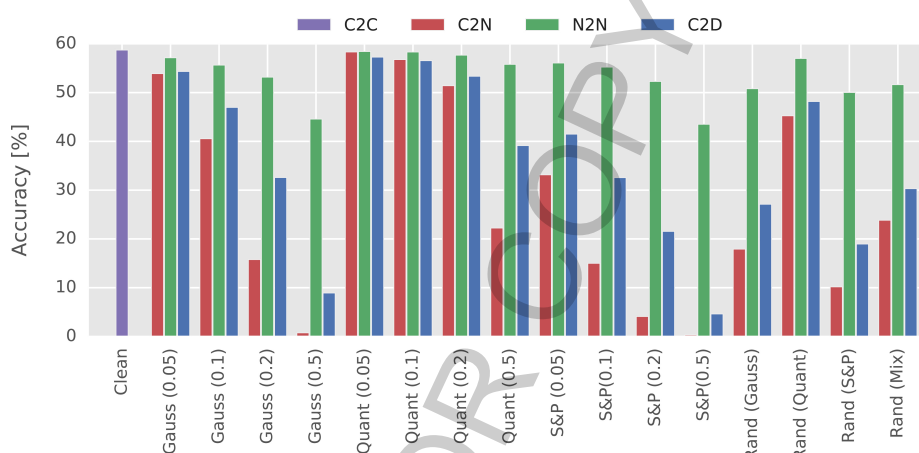


Fig. 4. Classification accuracy depending on type of artificial distortion applied, in four different cases: with no artificial noise (C2C), with only test set distorted (C2N), with both training and test sets distorted (N2N), and with test set distorted and denoised (C2D).

To reduce the severe impact of distortions on classification accuracy, the two already mentioned strategies of dealing with noise were evaluated. In the first one, i.e. training data augmentation, images were distorted during the training procedure with the same type of noise that was later present during evaluation. This case was denoted as N2N. In the second strategy, i.e. image denoising, classification network was trained on undistorted data. Second, smaller network was however trained explicitly to denoise images prior to classification. This case was denoted by C2D.

Results of this part of experimental analysis were presented in Fig. 4. Importantly, both strategies of dealing with distortions can be successfully applied in the case of unknown noise conditions. Compared to the case in which distortions are not accounted for, both strategies led to an improvement in accuracy. Statistical significance of that improvement was evaluated with the Wilcoxon signed-rank test. At the significance level of $\alpha = 0.05$ both strategies proved to be significantly better than in the case when distortions are not

accounted for. Furthermore, data augmentation strategy proved to be significantly better than applying denoising prior to the classification.

Based on the achieved results, data augmentation strategy allows us to achieve higher performance. However, it should be stated that better classification accuracy comes at price of potentially longer training time. When using the denoising strategy, transfer of weights from previously trained model is easier, since availability of models trained to recognize undistorted data is higher. Additionally, training data augmentation requires either being able to artificially distort the images, or obtaining high amounts of noisy labeled data. This issue is less severe when using the denoising strategy, since images used for training in that case do not need to be labeled.

5.7. Noise as a form of regularization and applying improper strategy of dealing with noise

In the final stage of conducted experimental study we considered the case, in which strategy of dealing

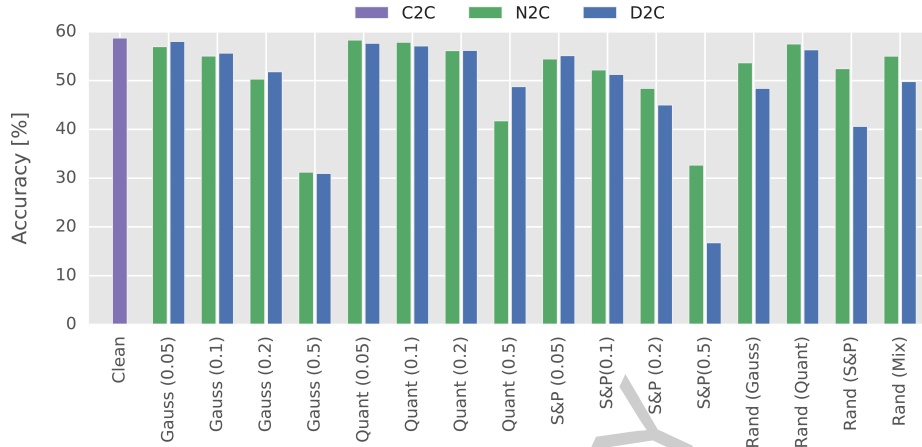


Fig. 5. Classification accuracy after training on data either augmented with noise (N2C) or denoised (D2C), when no distortions were present in test images. Results indicate performance of model trained to recognize noisy images when no distortions are actually present. Alternatively, data augmentation can be viewed as applying additional form of regularization.

with noise is employed even though no distortions are present during evaluation. First of all, this served the purpose of testing the possibility of using noise as another form of regularization. Secondly, it allowed us to assess the negative impact of choosing improper noise model. Both the training data augmentation (N2C) and denoising (D2C) were considered, with the former corresponding also to using noise as a regularizer.

Results of this part of experimental study are presented in Fig. 5. In no case augmenting training data led to improved performance compared to the baseline case. Other applied forms of regularization, namely dropout and weight decay, were sufficient to assure good generalization capabilities of the model. It is possible that augmenting training data could be used instead of them. However, applying it on top of them led to a decreased performance, likely due to the overregularization.

Both strategies led to significant drop in performance in case of most severe distortions, when noise was not present during evaluation. This issue can be partially mitigated by training the model on noise with random intensity, in which case performance decrease was less severe. Despite lower accuracy gain in C2D case, accuracy drop in D2C case was comparable to N2C, oftentimes being even less severe. We speculate that improving the quality of denoising algorithm could escalate this trend even further, leading to denoising approach being the safer of the options.

Finally, we focused on the case in which only some portions of the images were distorted, which is likely to happen in the practical setting. We tried to estimate the probability of distortion sufficient to achieve higher

expected accuracy than in the standard case, in which presence of noise is not accounted for.

Let A_N be the classification accuracy when using training data augmentation, A_C the accuracy when noise is not accounted for, A_{N2N} the accuracy on distorted images when using training data augmentation, A_{N2C} the accuracy on undistorted images in the same case, A_{C2N} the accuracy on distorted images without applying strategy of dealing with noise, and A_{C2C} the accuracy on clean images in the same case. Given the probability of image being distorted p , expected values of A_N and A_C can be defined as $E[A_N]$ and $E[A_C]$, respectively.

$$E[A_N] = p \times A_{N2N} + (1 - p) \times A_{N2C} \quad (7)$$

$$E[A_C] = p \times A_{C2N} + (1 - p) \times A_{C2C} \quad (8)$$

Of particular interest is the case, in which employing data augmentation strategy leads to improved performance. In such a case the following holds

$$E[A_N] \geq E[A_C] \quad (9)$$

This can be reformulated to emphasize the probability of distortion in the input dataset which makes that the proposed data augmentation method leads to better accuracy of the trained network. Inserting Eqs (7) and (8) into Eq. (9) and solving for p yields

$$p \geq \frac{A_{C2C} - A_{N2C}}{A_{C2C} - A_{N2C} + A_{N2N} - A_{C2N}} \quad (10)$$

Analogous calculation can be performed for the case in which denoising is used.

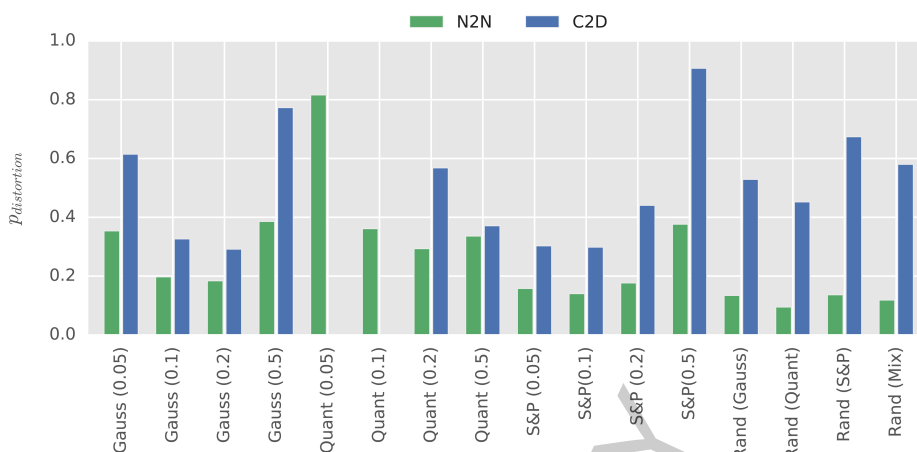


Fig. 6. Probability of observing distorted image sufficient to justify either training data augmentation (N2N) or applying denoising (C2D). Higher probabilities further increase the gain in accuracy compared to traditional training approach. No value was specified for quantization noise with small intensity, since applying denoising decreased performance in those cases.

Results of the conducted estimation were presented in Fig. 6. In the case of denoising, no probability was specified for quantization noise with intensity of 0.05 and 0.1. This was due to denoising leading to slightly decreased performance compared to the baseline case. For the remaining cases, probability of distortion sufficient to justify applying strategy of dealing with noise was always smaller for data augmentation. That was especially the case for the random noise types, likely to be most prevalent in practical settings. Probability of distortion of random type sufficient to justify training data augmentation was close to 10%. That is, if proportion of images being distorted is greater than that value, expected accuracy will be higher when using data augmentation than when not accounting for noise. Accuracy gain will further increase as the proportion of distorted images goes up.

6. Conclusions

In this paper we performed a thorough experimental analysis of impact of noise on classification with deep neural networks. We examined the classification performance under various noise settings, with both known and unknown noise models. We evaluated two possible strategies of dealing with noise, that is, training data augmentation and denoising prior to classification. We examined the possibility of using a convolutional neural network as a separate denoising algorithm. Finally, we measured the impact of employing these strategies when no noise is present, which can also correspond to using noise as a form of regularization.

Main findings of this paper are the following:

- The proposed denoising neural network outperforms all tested reference methods (median filtering, bilateral filtering and BM3D) in combination with quantization noise and severe noise conditions of other types, as well as Gaussian noise of random severity. At the same type it offers good performance in the remaining cases, depending on the type and severity of noise outperforming some of the reference methods;
- We experimentally confirmed findings from previous papers, according to which there is a relation between noise severity and deterioration of classification accuracy, up to the point at which correct classification becomes completely infeasible;
- We confirmed that using noise as a form of regularization on top of other regularization techniques, namely weight decay and dropout, does not improve the classification accuracy;
- Finally, we evaluated two methods of dealing with noise in images: training data augmentation and denoising prior to classification. Results of our experimental evaluation indicate that both techniques, depending on a type and severity of noise, can lead to significant improvement over the case in which noise is not accounted for. The training data augmentation proved to be preferable in regard to the classification accuracy. However, it requires large quantities of labeled, noisy data and requires long training of classification network. Using denoising is, therefore, a less expensive choice, albeit leading to worse performance.

Two main directions of further research include employing better denoising strategies and testing the possibility of finetuning the final architecture in the case of denoising. Based on the results presented in this paper, we speculate that it will be necessary to significantly improve the quality of denoising to achieve higher accuracy than in the case of data augmentation. However, training data augmentation is associated with higher cost of training and necessity of having large quantities of labeled, noisy data. Because of that, in practical applications denoising might be more feasible, even despite the lower performance.

Acknowledgments

This work was supported by the Polish National Science Center under the grant no. 2014/15/B/ST6/00609. The support of the PLGrid infrastructure is also greatly appreciated.

References

- [1] Large Scale Visual Recognition Challenge, 2011. <http://image-net.org/challenges/LSVRC/2011/>.
- [2] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:160304467. 2016.
- [3] Abaza A, Harrison MA, Bourlai T, Ross A. Design and evaluation of photometric image quality measures for effective face recognition. *IET Biometrics*. 2014; 3(4): 314-324.
- [4] Agostinelli F, Anderson MR, Lee H. Adaptive multi-column deep neural networks with application to robust image denoising. In: *Advances in Neural Information Processing Systems*. 2013; 1493-1501.
- [5] Bishop CM. Training with noise is equivalent to Tikhonov regularization. *Neural Computation*. 1995; 7(1): 108-116.
- [6] Bovik AC. *The Essential Guide to Video Processing*. 2nd ed. Academic Press; 2009.
- [7] Chakrabarti A. A neural approach to blind motion deblurring. arXiv preprint arXiv: 160304771. 2016.
- [8] Chan T, Shen J. *Image Processing and Analysis*. Society for Industrial and Applied Mathematics. 2005; Available from: <http://epubs.siam.org/doi/abs/10.1137/1.9780898717877>.
- [9] Chaudhury S, Roy H. Can fully convolutional networks perform well for general image restoration problems? arXiv preprint arXiv: 161104481. 2016.
- [10] Cheema TA, Qureshi IM, Jalil A, Naveed A. Artificial neural networks for blur identification and restoration of nonlinearly degraded images. *International Journal of Neural Systems*. 2001; 11(5): 455-461.
- [11] Coates A, Ng A, Lee H. An analysis of single-layer networks in unsupervised feature learning. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 2011; 215-223.
- [12] Cyganek B, Siebert JP. *An introduction to 3D computer vision techniques and algorithms*. John Wiley & Sons; 2009.
- [13] da Costa GBP, Contato WA, Nazare TS, Neto JE, Ponti M. An empirical study on the effects of different types of noise in image classification tasks. arXiv preprint arXiv: 160902781. 2016.
- [14] Dabov K, Foi A, Katkovnik V, Egiazarian K. Image denoising with block-matching and 3D filtering. In: *Electronic Imaging 2006, International Society for Optics and Photonics*. 2006; 606414-606414.
- [15] Dodge S, Karam L. Understanding how image quality affects deep neural networks. arXiv preprint arXiv: 160404004. 2016.
- [16] Dutta A, Veldhuis R, Spreuwers L. In: *The impact of image quality on the performance of face recognition*. Centre for Telematics and Information Technology, University of Twente, 2012; 141-148.
- [17] Eigen D, Krishnan D, Fergus R. Restoring an image taken through a window covered with dirt or rain. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013; 633-640.
- [18] Goodfellow I, Bengio Y, Courville A. *Deep Learning*; 2016. Book in preparation for MIT Press. Available from: <http://www.deeplearningbook.org>.
- [19] Grandvalet Y, Canu S, Boucheron S. Noise injection: Theoretical prospects. *Neural Computation*. 1997; 9(5): 1093-1108.
- [20] 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*. 2016; 770-778.
- [21] Houben S, Stallkamp J, Salmen J, Schlipsing M, Igel C. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In: *Neural Networks (IJCNN), The 2013 International Joint Conference on IEEE*. 2013; 1-8.
- [22] Jain V, Seung S. Natural image denoising with convolutional networks. In: *Advances in Neural Information Processing Systems*. 2009; 769-776.
- [23] Japkowicz N, Shah M. *Evaluating learning algorithms: A classification perspective*. Cambridge University Press; 2011.
- [24] Karahan S, Yildirim MK, Kirtac K, Rende FS, Butun G, Ekenel HK. How image degradations affect deep CNN-based face recognition? In: *Biometrics Special Interest Group (BIOSIG), 2016 International Conference of the IEEE*. 2016; 1-5.
- [25] Khan A, Yin H. Efficient blind image deconvolution using spectral non-Gaussianity. *Integrated Computer-Aided Engineering*. 2012; 19(4): 331-340.
- [26] Koziarski M, Cyganek B. Deep neural image denoising. In: *International Conference on Computer Vision and Graphics*. Springer International Publishing. 2016; 163-173.
- [27] Koziarski M, Cyganek B. Examination of the deep neural networks in classification of distorted signals. In: *International Conference on Artificial Intelligence and Soft Computing*. Springer. 2016; 680-688.
- [28] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*. 2012; 1097-1105.
- [29] Kryszczuk K, Drygajlo A. Improving classification with class-independent quality measures: Q-stack in face verification. In: *International Conference on Biometrics*. Springer. 2007; 1124-1133.
- [30] Längkvist M, Loutfi A. Learning feature representations with a cost-relevant sparse autoencoder. *International Journal of Neural Systems*. 2015; 25(1): 1450034.
- [31] Lukac R, Smolka B, Martin K, Plataniotis KN, Venetsanopoulos AN. Vector filtering for color imaging. *IEEE Signal Pro-*

- cessing Magazine. 2005; 22(1): 74-86.
- [32] Masci J, Meier U, Cireşan D, Schmidhuber J. Stacked convolutional auto-encoders for hierarchical feature extraction. In: International Conference on Artificial Neural Networks. Springer. 2011; 52-59.
- [33] Massouh N, Babiloni F, Tommasi T, Young J, Hawes N, Caputo B. Learning deep visual object models from noisy web data: How to make it work. arXiv preprint arXiv: 170208513. 2017.
- [34] Neelakantan A, Vilnis L, Le QV, Sutskever I, Kaiser Ł, Kurach K, et al. Adding gradient noise improves learning for very deep networks. arXiv preprint arXiv: 151106807. 2015.
- [35] Ortiz A, Munilla J, Górriz JM, Ramirez J. Ensembles of deep learning architectures for the early diagnosis of the Alzheimer's disease. International Journal of Neural Systems. 2016; 26(7): 1650025.
- [36] Peng X, Hoffman J, Yu SX, Saenko K. Fine-to-coarse knowledge transfer for low-res image classification. arXiv preprint arXiv: 160506695. 2016.
- [37] Rafiei MH, Adeli H. A novel machine learning model for estimation of sale prices of real estate units. Journal of Construction Engineering and Management. 2015; 142(2): 04015066.
- [38] Rafiei MH, Khushefati WH, Demirboga R, Adeli H. Supervised deep restricted boltzmann machine for estimation of concrete. ACI Materials Journal. 2017; 114(2).
- [39] Rifai S, Glorot X, Bengio Y, Vincent P. Adding noise to the input of a model trained with a regularized objective. arXiv preprint arXiv: 11043250. 2011.
- [40] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet large scale visual recognition challenge. International Journal of Computer Vision. 2015; 115(3): 211-252.
- [41] Saatci E, Tavsanoğlu V. Fingerprint image enhancement using CNN filtering techniques. International Journal of Neural Systems. 2003; 13(6): 453-460.
- [42] Sanchez A, Moreno AB, Vélez D, Vélez JF. Analyzing the influence of contrast in large-scale recognition of natural images. Integrated Computer-Aided Engineering. 2016; 23(3): 221-235.
- [43] Schuler CJ, Hirsch M, Harmeling S, Schölkopf B. Learning to deblur. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2016; 38(7): 1439-1451.
- [44] Simard PY, LeCun YA, Denker JS, Victorri B. Transformation invariance in pattern recognition – tangent distance and tangent propagation. In: Neural Networks: Tricks of the Trade. Springer. 1998; 239-274.
- [45] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 14091556. 2014.
- [46] Smolka B, Cyganek B. Impulsive noise suppression in color images based on the geodesic digital paths. In: SPIE/IS&T Electronic Imaging. International Society for Optics and Photonics. 2015; 94000R-94000R.
- [47] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015; 1-9.
- [48] Tan X, Triggs B. Enhanced local texture feature sets for face recognition under difficult lighting conditions. IEEE Transactions on Image Processing. 2010; 19(6): 1635-1650.
- [49] Tirronen V, Neri F, Kärkkäinen T, Majava K, Rossi T. An enhanced memetic differential evolution in filter design for defect detection in paper production. Evolutionary Computation. 2008; 16(4): 529-555.
- [50] Tomasi C, Manduchi R. Bilateral filtering for gray and color images. In: Computer Vision, 1998 Sixth International Conference on IEEE. 1998; 839-846.
- [51] Vasiljević I, Chakrabarti A, Shakhnarovich G. Examining the impact of blur on recognition by convolutional networks. arXiv preprint arXiv: 161105760. 2016.
- [52] Vincent P, Larochelle H, Bengio Y, Manzagol PA. Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th International Conference on Machine Learning. ACM. 2008; 1096-1103.
- [53] Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. Journal of Machine Learning Research. 2010; 11(Dec): 3371-3408.
- [54] Wager S, Wang S, Liang PS. Dropout training as adaptive regularization. In: Advances in Neural Information Processing Systems. 2013; 351-359.
- [55] Xu B, Guo P, Chen C. An adaptive regularization method for sparse representation. Integrated Computer-Aided Engineering. 2014; 21(1): 91-100.
- [56] Zhai G, Wu X, Yang X, Lin W, Zhang W. A psychovisual quality metric in free-energy principle. IEEE Transactions on Image Processing. 2012; 21(1): 41-52.
- [57] Zou WW, Yuen PC. Very low resolution face recognition problem. IEEE Transactions on Image Processing. 2012; 21(1): 327-340.