Maxout Filter Networks Referencing Morphological Filters

Makoto NAKASHIZUKA, Kei-ichiro KOBAYASHI, Toru ISHIKAWA and Kiyoaki ITOI
Faculty of Engineering, Chiba Institute of Technology
2 – 17 – 1, Tsudanuma, Nakashino, Chiba, 275–0016, Japan
Email: nkszk@sky.it-chiba.ac.jp

***Abstract*—This paper presents nonlinear filters that are obtained from extensions of morphological filters. The proposed nonlinear filter consists of a convex and concave filter that are extensions of the dilation and erosion of morphological filter with the maxout activation function. Maxout can approximate arbitrary convex functions as piecewise linear functions, including the max function of the morphological filters. The class of the convex function hence includes the morphological dilation and can be trained for specific image processing tasks. In this paper, the closing filter is extended to a convex–concave filter with maxout. The convex–concave filter is trained for noise and mask removal with a training set. The examples of noise and mask removal show that the convex–concave filter can obtain a recovered image, whose quality is comparable to inpainting by using the total variation minimization with reduced computational cost without mask information of the corrupted pixels.**

***Index Terms*—Mathematical morphology, maxout, noise removal, nonlinear filter, neural network.**

I. Introduction

Nonlinear filters have been widely applied to many image processing tasks. Many classes of nonlinear filters have been proposed. Mathematical morphology[1][2] is a framework of nonlinear image processing and morphological filters are major class of the nonlinear filters. Morphological filters are implemented by connecting basic building blocks: dilation and erosion. Impulsive noise reduction, feature detection and other image enhancement tasks[2] are realized by the connections of the erosion and dilation filters. For gray scale images, the outputs of dilation and erosion are, respectively, the maximum and minimum of the biased intensities of the pixels within a local window.

Recently, deep neural networks[3][4][5] have been successfully applied to object recognition and other image processing tasks. For convolutional neural networks (CNNs)[3][4], the unit input of the first layer is obtained from a sliding window over the image. The units of a layer share the same parameter set and the output of the layer is translation–invariant with respect to the input. When the resolution of the input is equal to the resolution of the output, the CNN layer can be interpreted as a nonlinear filter. For the recently proposed CNN, a rectified linear unit(ReLU)[3][4][5] is employed for the activation function. The ReLU outputs the larger value of zero and the sum of a linear combination of inputs and a bias.

Maxout, which is an extension of the ReLU, is proposed in Ref. [6]. The output of maxout is the maximum value over sums of the weighted inputs and biases. The epigraph of the transfer function between the input and output of maxout is the union of the epigraphs of the hyper planes, each of which is defined by the weighting parameters and bias. Therefore, the maxout transfer function is a convex function and can approximate any convex function as a piecewise–linear convex function[6]. Maxout network has been successfully applied to pattern and speech recognition problems[6][7].

Further, the dilation of morphology can be realized by maxout, if the parameters are restricted to zero and one. Thus, the convolutional layer composed of the maxout functions is interpreted as an extension of dilation. Moreover, erosion can be implemented as the negative of the dilation, of which the signs of the input are inverted. Erosion can also be implemented as the negative maxout, which can approximate any concave function. Since the class of the function that can be realized by maxout is broader than those by dilation and erosion, the maxout network that yields from a morphological filter by replacing dilations and erosions with maxout will outperform the original morphological filter.

In this study, we construct maxout networks based on morphological filters. The dilation filter is extended to a convex filter with maxout, whereas the erosion filter is extended to a concave filter with maxout by inverting the sign of the output. By replacing the dilation and erosion of the opening and closing filters with convex and concave filters, we propose novel nonlinear filters for image noise and mask removal.

In applying deep neural networks to image processing, super resolution[4] and completion[5] have been realized by CNNs with the ReLU. In such applications, the number of hidden layer is greater than one. In this study, we demonstrate that the proposed nonlinear filter networks can perform noise reduction and completion with only one hidden layer, owing to the filter configuration based on the morphological filters.

This paper is organized as follows. In the next section, dilation and erosion are extended to convex and concave

ISBN 978-0-9928626-7-1

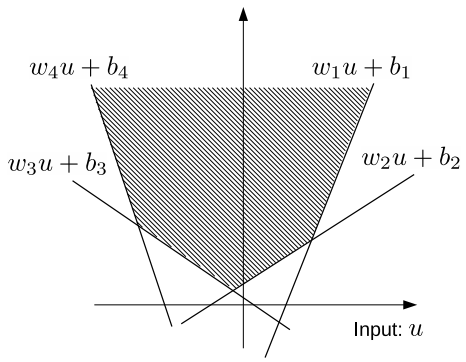

Fig. 1. A maxout example with $K = 4$, N=1.

filters, respectively. In Section 3, the maxout networks are developed as extensions of the opening and closing filters of the mathematical morphology. Parameter training using the stochastic gradient descent method is also explained. Finally, we show several examples of image completion to demonstrate the advantages of our approach.

II. Convex and concave filter with maxout

The ReLU is one of the units of a CNN, of which the transfer function between the input $\mathbf{u} = (u_1, u_2 \cdots u_n)$ and the output $r(\mathbf{u})$ is defined as $r(\mathbf{u}) = \bigvee\{z(\mathbf{u}) + b, 0\}$, where $z(\mathbf{u})$ is the linear combination of the input as $z(\mathbf{u}) = w_1u_1 + w_2u_2 + \cdots + w_Nu_N$. $\bigvee$ obtains the maximum of the set of numbers.

Maxout[6] can be interpreted as an extension of the ReLU. The number of sets of weighting parameters and biases is greater than one for maxout. The maxout output is defined as

$$m(\mathbf{u}) = \bigvee_{k=1\cdots K} (z_k + b_k) \tag{1}$$

where $z_k = w_{1,k}u_1 + w_{2,k}u_2 + \cdots w_{N,k}u_N$. The number of maxout parameters is K times larger than the ReLU. In Fig. 1, schematic outline of maxout is shown. In this figure, the maxout unit has only one input. The transfer function is defined by the union of the epigraphs of the lines. Maxout can approximate an arbitrary convex function as a piecewise–linear convex function. By using maxout for a neural network, the activation function of each unit is learned through training. In Ref. [6], a maxout network is defined as the difference between two maxout units. The negative maxout can approximate concave functions as piecewise–linear concave functions. The network in Ref. [6] can approximate the sum of the convex and concave functions as piecewise–linear functions by using maxout.

Similarly, morphological dilation and erosion are also defined by the max function with respect to the input pixels. Let us suppose that $\{f_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{I}}$ is the set of pixel intensities, of which integer coordinates $\mathbf{x}$ are included in the set $\mathcal{I}$. The intensity of the dilation and erosion of f at coordinate $\mathbf{x}$ are respectively defined as

$$d_s \circ f_{\mathbf{x}} = \bigvee_{\mathbf{y}\in\mathcal{S}} f_{\mathbf{x}+\mathbf{y}} + s_{\mathbf{y}} \quad \text{and} \quad e_s \circ f_{\mathbf{x}} = \bigwedge_{\mathbf{y}\in\mathcal{S}} f_{\mathbf{x}-\mathbf{y}} - s_{\mathbf{y}}, \tag{2}$$

where $\mathcal{S}$ is the small subset of the coordinates. The biases that are allocated to the coordinates in $\mathcal{S}$ are denoted as $\{s_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{S}}$. The pair of the set of coordinate $\mathcal{S}$ and the biases is referred to as the structuring element (SE). By using the definition of dilation, the erosion can be represented as

$$e_s \circ f_{\mathbf{x}} = -d_{s^*} \circ (-f)_{\mathbf{x}}, \tag{3}$$

where s^* denotes the symmetrical SE of s. The SE of s^* is related to s as $s^*_{(m,n)} = -s_{(-m,-n)}$.

Comparing the dilation in (3) and maxout in (1), dilation can be represented as maxout in which input is specified the set of local image intensities. A filtering operation is defined by replacing the maxout input with the image intensities as

$$\hat{d}_w \circ f_{\mathbf{x}} = \bigvee_{k=1\cdots K} \left\{ \left(\sum_{\mathbf{y}\in\mathcal{S}} w_{\mathbf{y},k} f_{\mathbf{x}+\mathbf{y}} \right) + b_k \right\}. \tag{4}$$

This filter is referred to as the convex filter, since maxout can approximate arbitrary convex functions. If each input z_k of the maxout in (1) corresponds to a intensity of the pixel $f_{\mathbf{x}+\mathbf{y}}$, the output of the convex filter is equal to the the dilation output. The concave filter, which is an extension of the erosion filter in (4), is defined as

$$\hat{e}_v \circ f_{\mathbf{x}} = -\bigvee_{k=1\cdots K} \left\{ \left(\sum_{\mathbf{y}\in\mathcal{S}} v_{\mathbf{y},k} f_{\mathbf{x}+\mathbf{y}} \right) + a_k \right\}, \tag{5}$$

where $v_{\mathbf{y},k}$ and a_k are the set of the weighting parameters and biases for the concave filter, respectively. The class of nonlinear filters that are realized by the connection of the convex and concave filters includes the class of morphological filters. In next section, the basic morphological filters are extended to a maxout filter network by replacing dilation and erosion with the convex and concave filters.

III. Convex filter networks based on the morphological filters

In mathematical morphology, the closing and opening operations with SE s are respectively defined as

$$c_s \circ f_{\mathbf{x}} = e_s \circ d_s \circ f_{\mathbf{x}} \quad \text{and} \quad o_s \circ f_{\mathbf{x}} = d_s \circ e_s \circ f_{\mathbf{x}}. \tag{6}$$

Closing is applied to negative impulsive noise removal, since the complement of the negative impulse cannot include the complement of the SE whose size is greater than one[2]. Opening is the complementary process of the closing. The image is approximated by the union of the translated SEs.

The convex–concave and concave–convex filters that are obtained by replacing the erosion and dilation with convex

ISBN 978-0-9928626-7-1 © EURASIP 2017

and concave filters are respectively defined as

$$\hat{c}_{v,w} \circ f_{\mathbf{x}} = \hat{e}_v \circ \hat{d}_w \circ f_{\mathbf{x}} \text{ and } \hat{o}_{v,w} \circ f_{\mathbf{x}} = \hat{d}_w \circ \hat{e}_v \circ f_{\mathbf{x}}. \quad (7)$$

Compared with the morphological filter, the number of parameters increases with the number of convolutions K. To specify these parameters, we apply training by using stochastic gradient decent[8].

In this paper, we examine the convex–concave filter in (7), which is an extension of the closing filter. Herein, convex–concave filters are only applied to applications that have been realized by closing filters. We therefore apply the convex–concave filter to degraded images that are corrupted by the negative impulsive noises and image masking. Due to the complementary properties of the opening and closing, a concave–convex filter can also be applied to applications based on positive impulse noise reduction. To perform noise and mask removal, the filter parameters must be trained with a set of training images. We present the training procedure for convex-concave filters prior to the demonstration.

A. Parameter training

To train the filter parameters, we employ a training set as the Berkeley segmentation dataset[10], extracting 400 images, whose size is 256×256 pixels. The number of images is increased by inverting the extracted images, – horizontally and vertically–. The total number of images in the training set is, thus, $1,600$. Let us suppose that the i-th image of the data set is $f^{(i)}$. The set of the degraded version of $f^{(i)}$ is $\{g^{(i,j)}\}_{j \in \mathcal{J}}$. The objective function that is reduced through training is the squared error, which is defined by the outputs of the convex–concave filter and the original image,

$$E = \sum_{i,j} \sum_{\mathbf{x}} \left(\hat{c}_{v,w} \circ g^{(i,j)}_{\mathbf{x}} - f^{(i)}_{\mathbf{x}} \right)^2. \quad (8)$$

Parameter sets $\{v, w\}$ are iteratively updated to minimize this squared error. To update the parameter set, we employ a stochastic gradient descent method that is widely employed for training neural networks. For one parameter update iteration, only one image f is chosen from the database. The degraded image g is generated from f for each iteration. The subgradient ∇Q of the error,

$$Q = \sum_{\mathbf{x}} \left(\hat{c}_{v,w} \circ f_{\mathbf{x}} - g_{\mathbf{x}} \right)^2. \quad (9)$$

is computed using the chain rule of differentiation to update the parameter sets. The number of training images is limited; however, the number of degraded images is not, since the chosen image is corrupted randomly in each iteration. The parameter update is performed by adding gradient ∇Q, which is multiplied by the step size obtained by using ADAGRAD[8],which can adaptively specify the gradient size from the number of the iterations.

In noise and mask removal, the window size that covers the input pixels for a maxout unit is empirically specified as 11×11. Thus, the set of coordinates $\mathcal{S}$ is defined as $\mathcal{S} = \{(m, n) \,|\, -M \leq m \leq M, -M \leq n \leq M\}$, where $M = 5$. The initial training parameters are specified to realize the closing filter, whose SE is specified as a 7×7 flat square SE that can remove intensity pits whose size is within 7×7 pixels. Before training, our convex–concave filter hence performs image closing. Noise reduction performance is improved by iterative updates of the stochastic gradient until convergence. The number of convolutions K is specified as 49, which is the same as the number of elements in the 7×7 SE for closing. The initial parameters for biases a_k and b_k are specified as zero.

During training, the number of updates to the coefficients is limited to 400,000 for a convex-concave filter with $K = 49$. It takes about 9 hours to train the filter using MATLAB parallel processing toolbox with an NVIDIA GTX1070 GPU and Intel Core i7.

IV. Examples of noise and mask removal

In this section, we provide examples of noise and mask removal by the proposed convex–concave filters. In order to evaluate the proposed filters, we employ two noise models. We note that the closing filter can perform noise removal for negative values. The convex–concave filter that extends the closing filter is also appropriate for the removal of negative noises. The first example of noise is pepper noise. The image that is degraded by the pepper noise is generated by replacing the pixel intensities with zero with the noise occurrence probability p. For evaluating median filters, salt–and–pepper noise is often employed, in our case, only the black pixels occur. This model is thought of as pixel defects in imaging devices. We also examine the masking with printed text. In our experiment, the color of the text is black, as in the case of the pepper noise. The probability of occurrence of text–masking noise at a pixel is correlated with that of neighboring pixels. For noise and mask removal, four images (Lena, Goldhill, Man and Boat) which are widely employed in evaluating image processing tasks. The size of the test images is 512×512 pixels.

We first show the result of the removal of pepper noise with various occurrence probabilities p. For comparison, we also examine convex–convex filters that are trained using the same procedure as convex–concave filters to show the advantage of the morphology–based network structure. The closing filter whose SE is trained by stochastic gradient descent is also included in this comparison. Pepper noise removal can be thought of as an image inpainting problem[9] if information about the coordinates of the corrupted pixels are known. We employ the widely applied total variation (TV) minimization method[11][12] for comparison. We note that the TV minimization method requires the mask information that denotes the coordinates at which pixels are replaced with zero. For our filtering methods, mask information is not obtained for the training process. The proposed filter can hence achieve

ISBN 978-0-9928626-7-1 © EURASIP 2017

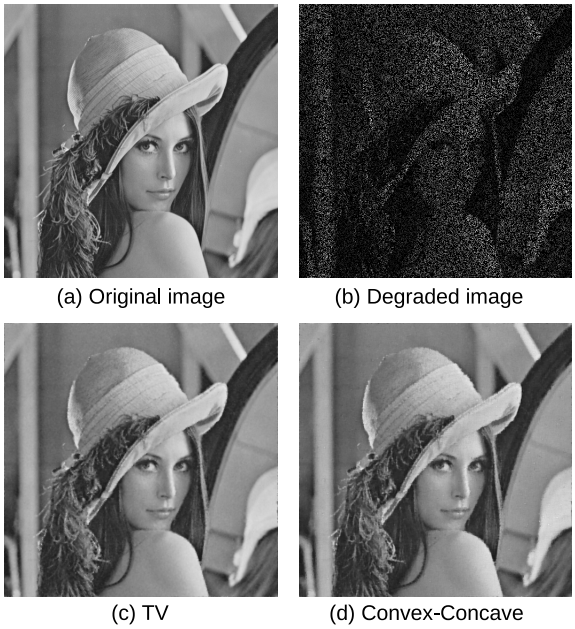

Fig. 2. Examples of pepper noise denoising. (a) Original image, (b) degraded image with $p = 0.75$, (c) result of TV (PSNR:30.77dB) and (d) result of convex –concave filter (PSNR:30.24dB).

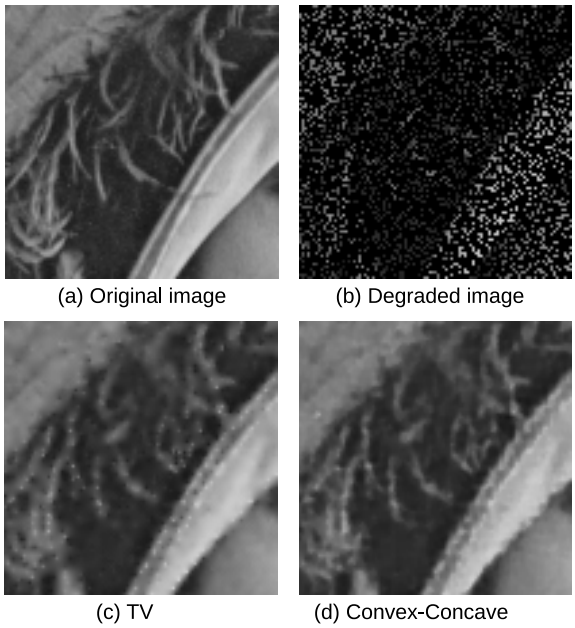

Fig. 3. Parts of Fig. 2

image inpainting, that is, pepper noise removal, without mask information.

Table 1 shows the peak signal–to–noise ratios (PSNRs) of the noise removal results for four test images with various noise occurrence probabilities p. In the degradation process in the training phase of the convex–concave filters, we examine two degradation processes. In the first

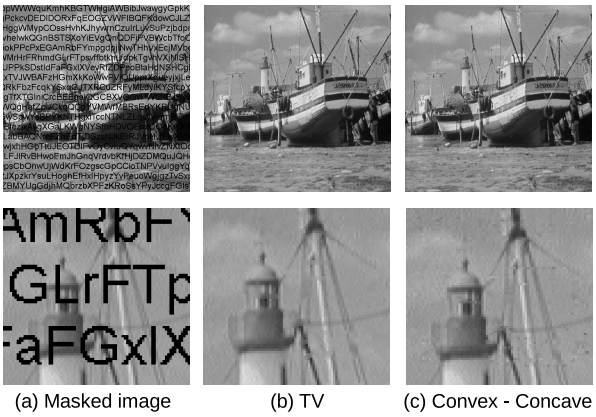

Fig. 4. Example of text mask removal. Entire images are shown in the first row. The parts of the images are shown in the second row.

TABLE I
PSNRs (dB) of the recovered images (pepper noise).

Image	Noise Prob.	Closing	Cnv. - Cnv.	Cnv. - Cnc.*	Cnv. – Cnc.**	TV with mask
Lena	0.500	29.88	31.61	33.70	33.21	35.01
	0.625	28.22	30.06	31.56	31.89	32.78
	0.750	27.48	28.36	30.16	30.24	30.77
	0.875	25.13	27.24	27.82	26.36	28.12
Goldhill	0.500	28.59	30.07	31.54	31.15	32.53
	0.625	27.51	28.97	29.96	30.07	30.81
	0.750	26.27	27.52	28.70	28.51	29.04
	0.875	24.51	26.25	26.54	25.54	26.80
Man	0.500	28.59	29.81	31.34	30.80	32.33
	0.625	27.51	28.74	29.72	29.83	30.57
	0.750	26.27	27.07	28.32	28.24	28.72
	0.875	24.51	25.74	26.08	25.16	26.32
Boat	0.500	28.14	29.08	30.51	30.14	31.33
	0.625	26.83	27.83	28.83	29.03	29.55
	0.750	25.77	26.10	27.21	27.25	27.44
	0.875	23.92	24.81	25.08	24.27	25.14

* Trained with target noise occurrence probability p. ** Trained with $p = [0.5, 0.875]$

degradation process, the noise occurrence probability p is specified as same and the target probability. In the second, p is randomly specified for each iteration of the stochastic gradient descent within the supposed range $[0.5, 0.875]$. For the closing and convex–convex filters, only the results that obtained using filters that trained with the target probability are shown. In this table, we see that the convex–concave filters outperform other filtering method without mask information. Comparing the TV and proposed method, the PSNRs of TV inpainting are always higher than those of the convex–concave filters. Since mask information is utilized in the TV inpainting method, the nondegraded pixel intensities preserved exactly in the

ISBN 978-0-9928626-7-1 © EURASIP 2017

TABLE II
PSNRs (dB) of the recovered images (text mask).

Image	Font Size	Cnv. – Cnc.	TV with mask
Lena	10 pt	36.27	39.07
	24 pt	34.69	36.77
	32 pt	33.06	35.10
Goldhill	10 pt	34.26	36.49
	24 pt	32.43	34.72
	32 pt	32.01	33.71
Man	10 pt	34.38	36.53
	24 pt	32.96	34.92
	32 pt	31.78	33.58
Boat	10 pt	33.05	35.44
	24 pt	31.80	33.61
	32 pt	30.66	32.44

recovered images. However, the improvement in PSNRs due to this advantage decreases as noise probability p increases. When $p = 0.75$, the PSNRs of TV inpainting are higher than those of the convex–concave filter by less than 0.6 dB, which is a small difference for actual recovered images. In Fig. 2, the recovered Lena images with $p = 0.75$ are shown. Large differences are not observed with comparing the TV (c) and convex–concave filters (d). Parts of Fig. 2 are shown in Fig 3. In recovered images using the TV method (c), we see that some thin lines are broken. In the results of the convex–concave filter, some image details that are lost in the TV methods are recovered.

TV inpainting is implemented by the augmented Lagrangian method in [12]. It takes about 2.4 seconds to recover the image in Fig. 3(a) using an Intel Core i7 processor and MATLAB. The result of Fig. 3(d) is obtained by the convex–concave filter within 0.8 seconds. If all convolution operations in a layer are performed in parallel, the total computational time is equal to two convolutions and two max operations. By using parallel computations, the computational time of the convex–concave filter can be reduced to 0.15 seconds with an Intel Core i7 CPU and NVIDIA GTX-1070 GPU.

In Table 2, text removal results are shown in terms of PSNR. Comparing the convex–concave filter with TV inpainting that utilizes mask information, the PSNRs of the convex–concave filters are smaller by 2 dB on average. In Fig. 4, the text removal results for the Boat image degraded with 24 pt Arial font are shown. In the recovery result obtained by TV (b), the regions that are masked in the degraded images are smoothed; parts of edge lines behind the mask are hence blurred and seem to be broken in the recovered images. In the image recovered by the convex–concave filter, such over–smoothing is not observed. However, small granular artifacts appear around the masks.

V. Conclusion

In this paper, a novel class of nonlinear filters has been introduced for image processing. The dilation and erosion of morphology are respectively extended to the convex and concave filter with the maxout activation function. By using extended filters, we propose a convex–concave filter that is an extension of closing filter. The convex–concave filter can remove pepper noise with comparable quality as that of TV inpainting, with lower computational costs and without mask information that denotes the occurrence of noise. Moreover, we show that the convex–concave filter can be trained to remove text masks.

This paper presented only the extension of the closing filter to show the advantage of extending morphological filters with maxout. Obviously, existing morphological filters can be extended to maxout filter networks using our approach. In morphological image processing, deeper filter networks have been proposed. For example, the alternating sequential filter[2] cascades connections of closing and opening filters with various SEs. Deeper networks of convex and concave filters will cover many applications of image processing. The applications and training of deeper convex filter networks obtained from the morphological filters are also topics for future research.

References

[1] P. Maragos and R. W. Scafer, "Morphological filters–part I: their set-theoretical analysis and relations to linear shift-invariant filters, " IEEE Trans. Acoustic, Speech and Signal Process., vol. 35, no. 8, pp. 1153–1169, Aug. 1987.

[2] P. Maragos, " Chapter 3. 3: Morphological filtering for image enhancement and feature detection, " The Image and Video Processing Handbook, A. C. Bovik Ed., Elsevier Academic Press, pp. 135-156, 2005.

[3] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks, " Conference on Neural Information Process. Syst., 2012.

[4] C. Dong, C. C. Loy, K. He and X. Tang, "Learning a deep convolutional network for image super resolution, " Proc. Euro. Conf. of Computer Vision, 2014.

[5] R. Köhler, C. Schuler, B. Sch olkopf and S. Harmeling, "Mask–specific inpainting with deep neural networks, " Lecture Notes in Computer Science, Pattern Recognition (GCPR 2014), pp. 523–534, 2014.

[6] I. J. Goodfellow, D. Warde-Farley, A. Courville and Y. Bengio, "Maxout networks, " Proc. 30th Int. Conf. on Machine Learning, 2013.

[7] P. Swietojanski, J. Li and J. T. Huan, "Investigation of maxout networks for speech recognition, " Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process., Florence, May 2014.

[8] J. Duchi, E. Hazan, Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization, " J. Machine Learning Res., no. 12, pp. 2121-2159, 2011.

[9] C. Guillemot and O. L. Meur, "Image inpainting," IEEE Signal Process. Mag., vol. 31, no. 1, pp. 127–144, Jan. 2014.

[10] The Berkeley Segmentation Dataset and Benchmark, https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

[11] A. Chambolle, "An algorithm for total variation minimization and applications, " J. Math. Imaging and Vision, vol. 20, pp. 89–97, 2004.

[12] M. V. Afonso, J. M. Bioucas-Dias and M. A. T. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of image inverse problem, " IEEE Trans. Image Process., vol. 20, no. 3, pp. 681–694, Mar. 2011.

ISBN 978-0-9928626-7-1 © EURASIP 2017