Deep Learning networks with p-norm loss layers for spatial resolution enhancement of 3D medical images

Karl Thurnhofer-Hemsi[1][0000−0001−6519−1213], Ezequiel López-Rubio[1][0000−0001−8231−5687], Núria Roé-Vellvé[2], and Miguel A. Molina-Cabello[1][0000−0002−8929−6017]

[1] Department of Computer Languages and Computer Science
University of Málaga, Bulevar Louis Pasteur, 35, 29071 Málaga, Spain
{karlkhader,ezeqlr,miguelangel}@lcc.uma.es
Institute of Biomedical Research of Málaga (IBIMA)
[2] Biomedical Research Networking Center in Bioengineering
Biomaterials and Nanomedicine (CIBER-BBN), Barcelona, Spain
nuriaroe@gmail.com

Abstract. Nowadays, obtaining high-quality magnetic resonance (MR) images is a complex problem due to several acquisition factors, but is crucial in order to perform good diagnostics. The enhancement of the resolution is a typical procedure applied after the image generation. State-of-the-art works gather a large variety of methods for super-resolution (SR), among which deep learning has become very popular during the last years. Most of the SR deep-learning methods are based on the minimization of the residuals by the use of Euclidean loss layers. In this paper, we propose an SR model based on the use of a p-norm loss layer to improve the learning process and obtain a better high-resolution (HR) image. This method was implemented using a three-dimensional convolutional neural network (CNN), and tested for several norms in order to determine the most robust fit. The proposed methodology was trained and tested with sets of MR structural T1-weighted images and showed better outcomes quantitatively, in terms of Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM), and the restored and the calculated residual images showed better CNN outputs.

Keywords: Super resolution · magnetic resonance images · convolutional neural networks.

1 Introduction

Obtaining high quality, high resolution images is critical for medical diagnosis, due to the impact of the subsequent clinical decisions. Even though acquisition techniques for MR imaging are continuously being improved, resolution is always limited by a variety of factors. This calls for the application of post-processing algorithms to optimize image quality. Resolution enhancement is of particular

interest, and the improvement of super-resolution (SR) algorithms is also a subject of constant interest. Many different approaches have been applied to obtain resolution-enhanced images in the recent years, and, among them, deep learning has been gaining increasing popularity. Most of these methods are based on the minimization of the residuals using Euclidean loss layers.

In recent times, the use of p-norm methods in optimization algorithms has drawn attention for various applications. They have been used for machine learning binary classifiers [5, 4], and in the framework of optimal control [2]. Due to the properties of the p-norm when dealing with with sparse vectors and matrices [3, 8, 14, 1], it has also been applied for feature selection [13].

One of the advantages of using a p-norm, with $p < 2$, is that it can allow reducing the effect of outliers in a minimization problem. Noise and artifacts in the images of the training set for MR superresolution are outliers that have to be avoided so that a SR algorithm provides realistic high quality solutions. In this work, the p-norm with $p < 2$ is proposed as the loss function for neural layers of super resolution convolutional neural networks, and the most suitable values of p are studied.

The rest of paper is organized as follows: Section 2 contains the theoretical background of our model. Then, in Section 3 the experiments carried out and the outcomes are described. Finally, the conclusions and future works are presented in Section 4.

2   The model

In this section we present the learning rule for neural layers with p-norm loss function. The rationale behind our proposal is that the classic quadratic loss function, which corresponds to $p = 2$, may be outperformed by loss functions based on the p-th power of the absolute value of the error. For $p < 2$, this increases the robustness of the learning rule against outliers, i.e. training samples with extremely large values of the error.

The p-norm of a D-dimensional vector $\mathbf{v} \in \mathbb{R}^D$ is defined as:

$$\|\mathbf{v}\|_p = \sum_{j=1}^{D} |v_j|^p \tag{1}$$

For deep learning neural networks, the standard loss function is the square of the Euclidean (2-norm) of the difference between the desired output vector and the output vector obtained from the network, averaged for all available training samples. While this choice yields excellent results for many applications, there is room for improvement, since the exponent p in (1) determines the importance that the loss function gives to those training samples which have components with higher absolute values $|v_j|$. The higher p, the more importance that is given to extreme values of $|v_j|$. It must be considered that maybe those extreme values of $|v_j|$ correspond to badly measured training samples, or irrelevant observations. Therefore, a loss function which does not get too influenced by those extreme

errors might obtain better values than the 2-norm. The most promising values of p are those for which the p-norm fulfils the mathematical definition of norm, i.e. $1 \leq p \leq 2$.

Given the above considerations, we propose to employ neural layers with a p-norm loss function. The loss function is defined as follows, for N samples of dimension D:

$$E = \sum_{i=1}^{N} \sum_{j=1}^{D} |y_{ij} - z_{ij}|^p \tag{2}$$

where y_{ij} is the j-th component of the i-th obtained output vector from the network and z_{ij} is the desired output.

The derivative of the loss function with respect to a synaptic weight w is:

$$\frac{\partial E}{\partial w} = \sum_{i=1}^{N} \sum_{j=1}^{D} p \, |y_{ij} - z_{ij}|^{p-1} \operatorname{sign}(y_{ij} - z_{ij}) \frac{\partial y_{ij}}{\partial w} \tag{3}$$

where the sign function is defined as follows:

$$\operatorname{sign}(x) = \begin{cases} -1 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \tag{4}$$

Stochastic gradient methods can be applied to the gradient of the loss function (3), in order to train a p-norm neural layer within a deep neural network.

3   Experimental results

This section describes the experiments we carried on. First, Subsection 3.2 explains the low-resolution image generation, as well as the software and hardware employed, and the selected performance metrics for comparison between p-norms. The input datasets are described in Subsection 3.1. Finally, we report the findings of the experiments in Subsection 3.3.

3.1   Datasets

Six different T1-weighted MR images are considered for the evaluation of the p-norms:

- 2 images of the Kirby 21 (images 10 and 11) [7]. These data were acquired using a 3-T MR scanner with a $1.0 \times 1.0 \times 1.2 \, mm^3$ voxel resolution over an field-of-view (FOV) of $240 \times 204 \times 256 \, mm$ acquired in the sagittal plane.
- 2 images of the OASIS dataset (images 1 and 2 of the cross-sectional data) [9] . Data were acquired on a 1.5-T Vision scanner with a $1.0 \times 1.0 \times 1.25 \, mm^3$ voxel resolution over an FOV of $256 \times 256 \, mm$.
- 1 image of the IBSR public dataset [12]. It is named IBSR_07, it has image size $256 \times 256 \times 128$, with $1.5 \times 1.0 \times 1.0 \, mm^3$ voxel resolution.

– 1 T1-weighted image from CIMES[3] using a 3-T MR scanner with a $0.93 \times 0.93 \times 1.0\,mm^3$ voxel resolution over an FOV of $256 \times 256\,mm$.

3.2   Methods

The deep network used to evaluate our proposal was the SRCNN3D method [10], which is a super-resolution convolutional neural network for three-dimensional MR images. The convolutional network have been developed using Caffe package [6] on a Python framework. One of the motivations of selecting this network is its simplicity to understand and modify the source code to create our customized p-norm loss layer.

Given an image $\mathbf{Y}$ and its respective LR one $\mathbf{X}$, this CNN is based on the application of three blocks of convolutional Rectified Linear Unit (ReLU) layers successively, to a pre-interpolated image $\mathbf{Z} = I(\mathbf{X})$. This step is internal to the method. Thus, the net computes a super-resoluted HR image by the minimization of the Euclidean loss between the output of the CNN, $s(\mathbf{Z})$, and the original HR image $\mathbf{Y}$.

$$f = arg\,min_s \sum ||\mathbf{Y} - s(\mathbf{Z})||^2 \qquad (5)$$

This network is trained using overlapping patches that are extracted from a set of HR reference images. A down-sampling and up-sampling is applied to each patch and a set of pairs input-target is created in order to learn an end-to-end function between low and high resolution images. Specific details of the implementation of this network can be found in the literature.

We carried out a training over 50000 iterations for each p-norm, using momentum of 0.9, learning rate of 0.0001 and batch size of 256. Stochastic Gradient Descent (SGD) was used for model optimization (all are default parameters). Images 33-42 from Kirby dataset were used for training. Furthermore, zoom factors 2 and 3 were employed in our analysis. Fig. 1 shows the training loss curves for each value of p. We used a fixed number of iterations for every training to make a fair comparison, and the selected number of iterations assess the convergence of all the trainings. We have also tested $1 \leq p < 1.5$ but no convergence was achieved. The comparison experiments have been carried out on a 64-bit Personal Computer with an six-core Intel i7 3.50GHz CPU, 64 GB RAM, with a GPU Nvidia GTX Titan.

On the other hand, LR images were created applying the following algorithm to the HR images:

1. Crop HR image dimensions to make divisible by the scale factor.
2. Apply a 3D Gaussian filter with standard deviation equal to 1.
3. Use an bi-cubic interpolation method to generate LR image

In this work, Matlab fuctions were used with default parameters.

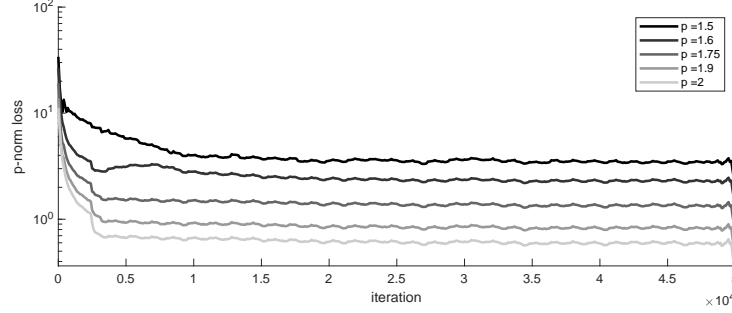Three quality measures were used to evaluate the proposed method:

---

[3] https://fguma.es/unidad-imagen-molecular/

Fig. 1. Training loss for the p-norms ($p \in \{1.5, 1.6, 1.75, 1.9, 2\}$) during 50000 iterations, smoothed using a sliding window of 50.

– *Peak Signal-to-Noise Ratio (PSNR)*, measured in (decibels) dB, and the more high value, the better is the likeness.

$$PSNR = 10 \log_{10} \left(\frac{peakval^2}{MSE} \right) \tag{6}$$

where *peakval* is maximum possible value of the image and MSE refers to the Mean Squared Error.

– *Structural Similarity index (SSIM)* [11], which focuses on structural similarities between images, returning a value between 0 and 1 (higher is better):

$$SSIM(x, y) = \frac{(2\mu_x\mu_y)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{7}$$

where μ_x and μ_y are the mean value of images x and y, σ_x and σ_y are the standard deviation of images x and y, σ_{xy} is the covariance of x and y, $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ (default values were used: $L = 1$ is the dynamic range, $k_1 = 0.01$ and $k_2 = 0.03$).

Besides, we used residual images to analyze the results from a qualitative point of view:

$$residual = h - s \tag{8}$$

where h represents the original HR image and s the output of the CNN. The best performance is such that the residual image is the zero matrix. As it is difficult to distinguish dark values, we subtracted the constant 0.5 to the residual images in order to see the performance differences. Thus, the residual images appear in gray.

3.3   Results

First, we evaluate each convolutional neural network from a quantitative point of view. Performance results in terms of PSNR and SSIM for each image applying
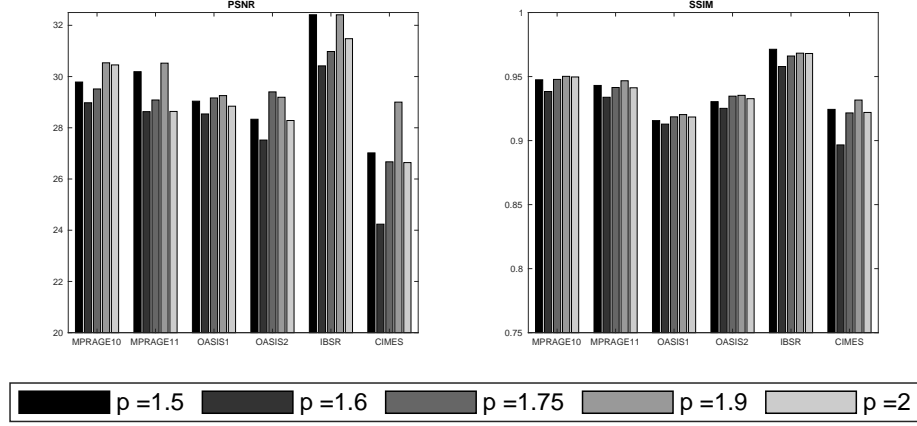
Fig. 2. Comparison of the PSNR and SSIM for the p-norms ($p \in \{1.5, 1.6, 1.75, 1.9, 2\}$) and the six tested images, using a scale factor equal to 2.

a zoom factor of 2, are depicted in Fig. 2. Each color bar represents one value of p. The higher is the bar, the better is the network. In both metrics, for five of the six tested images the "winner" is the 1.9-norm. If we focus on the difference with respect to the euclidean norm, there are great improvements for MPRAGE11 and CIMES images. $p = 1.5$ also shows good performance in PSNR compared to $p = 2$, so it is shown that it is not always the best and we can get better results with a different minimization. However, there are irregularities for $p = 1.6$ and $p = 1.75$, as for some images performs better but for others worse. This analysis can be extrapolated to the SSIM metric too, where the differences between nets are smaller but similar.

Fig. 3 shows the results for zoom factor 3. In general, the performance is worse because we are trying to infer more information from the LR image, and and difference between images is greater. Here again the pattern is the same. The networks based on the 1.5-norm and the 1.9-norm carried out a better prediction of most of the images, being $p = 1.9$ the best one in both PSNR and SSIM.

The general behaviour of each neural network across all images is summarized in Fig. 4. We computed a ranking sorting each p-norm according to its performance with respect each image and we assigned points from 1 to 5. The less punctuation the better is the network. For scale 2, shown in Fig. 4a, the best method is that one based on the 1.9-norm loss layer. The performance of $p = 1.6$ in terms of both PSNR and SSIM is clearly the worst, followed by the 2-norm, specially in PSNR. For the similarity metric there are not great differences between 2, 1.5 and 1.75. Rank of zoom 3 is depicted in Fig. 4b. Here the differences with respect to zoom 2 go in favour of $p = 2$, which is situated in the second place of our SSIM rank. Nevertheless, $p = 1.9$ is still the best method, followed closely by $p = 1.5$ in PSNR.
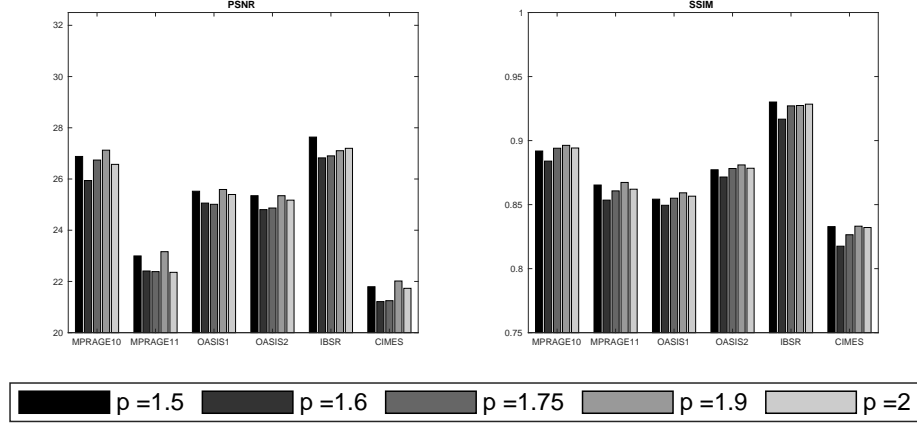
Fig. 3. Comparison of the PSNR and SSIM for the p-norms ($p \in \{1.5, 1.6, 1.75, 1.9, 2\}$) and the six tested images, using a scale factor equal to 3.

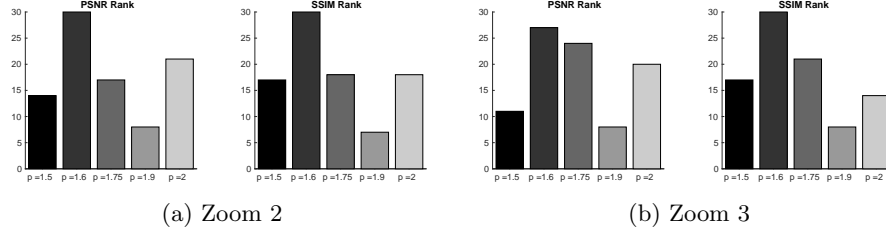(a) Zoom 2   (b) Zoom 3

Fig. 4. Rank of the PSNR and SSIM for the p-norms for scale factors 2 and 3, taking into account all the images.

On the other hand, we compare the output of the networks from a qualitative point of view in Fig. 5 and 6. Residual images are also displayed to have a better discrimination between methods.

Firstly, in Fig. 5 is shown a three-dimensional perspective of the CNN's outcome of image 11 of the Kirby dataset. The differences can be seen on the third row, where the intensity of gray varies from one p-norm to other. The darkest images are the ones corresponding to $p = 1.6$ and $p = 2$, which indicates that the difference between the output of the net and the ground truth is greater. On the other hand, the image where less structures are removed, that is, the most gray uniform one, corresponds to $p = 1.9$. This results matches with the previous quantitative analysis, where its values of PSNR and SSIM are the best.

An axial section of the image 1 of the OASIS dataset is compared in Fig. 6. In this case, a zoom factor of 3 is used. The main differences can be seen in the borders of the lateral ventricle (the uniform centered region). The output with less dark border is the one produced using $p = 1.9$, followed by $p = 2$
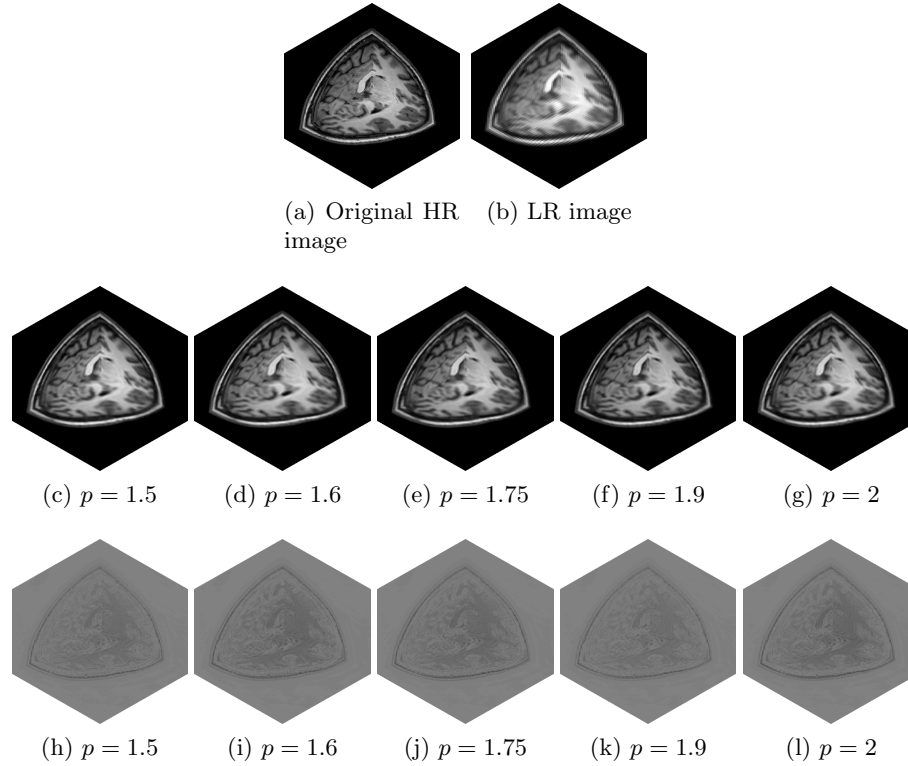
(a) Original HR   (b) LR image
image

(c) $p = 1.5$ (d) $p = 1.6$ (e) $p = 1.75$ (f) $p = 1.9$ (g) $p = 2$

(h) $p = 1.5$ (i) $p = 1.6$ (j) $p = 1.75$ (k) $p = 1.9$ (l) $p = 2$

Fig. 5. Qualitative results for $MPRAGE11$ image for each p-norm, applied with zoom factor 2. Second row shows the reconstructed image by each algorithm and third row shows residual images between the reconstructed and the original HR image.

and $p = 1.5$. This can indicates that the structural surcus of the cerebrum are restored in a proper way. Moreover, some dark spots are conserved better since they appear clearer in the residual image of our proposed method, which is fundamental to not carry on a bad diagnostic.

4 Conclusion

This work presents a robust three-dimensional super-resolution method for magnetic resonance images. It is based on the use of a p-norm loss layer instead of the usual Euclidean formulation. High-resolution images are obtained by learning a map between an interpolated low-resolution image and the ground truth, where the optimization function is defined as the p-norm to reduce the overall error in the training. T1 structural images from different origins were used to evaluate the efficiency of each variant of the network, achieving better performance. Results show that the Euclidean loss layer is not always the best norm, since PSNR and SSIM have increased for most images and zoom factors. Quali-
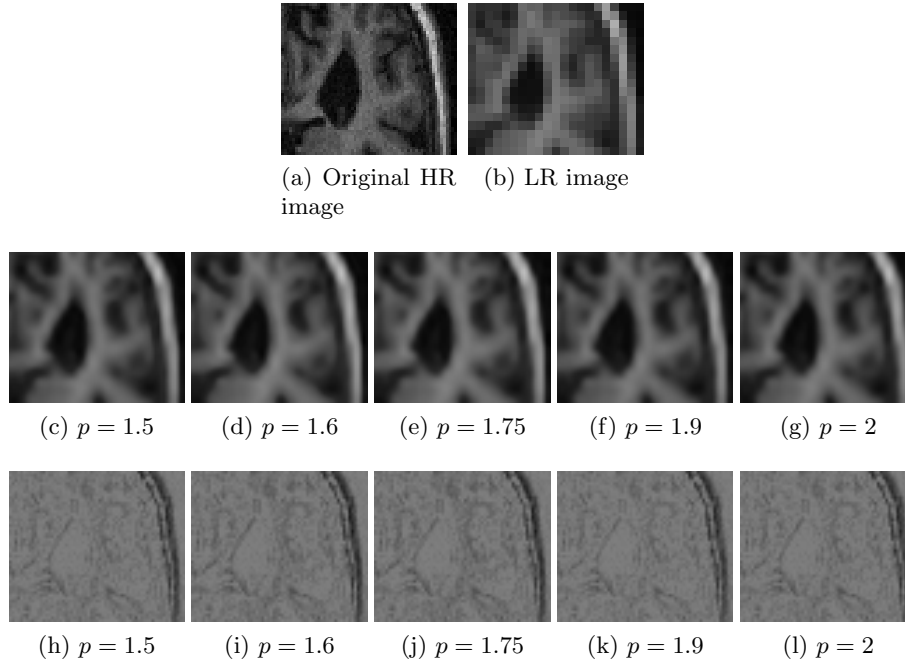
(a) Original HR   (b) LR image
image

(c) $p = 1.5$ (d) $p = 1.6$ (e) $p = 1.75$ (f) $p = 1.9$ (g) $p = 2$

(h) $p = 1.5$ (i) $p = 1.6$ (j) $p = 1.75$ (k) $p = 1.9$ (l) $p = 2$

Fig. 6. Qualitative results for a section of the axial view of the $OASIS1$ image for each p-norm, applied with zoom factor 3. Second row shows the reconstructed image by each algorithm and third row shows residual images between the reconstructed and the original HR image.

tatively, restored images look more refined and with less structural degradation. The proposed approach could be extended to another neural networks in order to improve the quality of the outputs.

Acknowledgements

References

1. Abramovich, F., Benjamini, Y., Donoho, D.L., Johnstone, I.M.: Adapting to unknown sparsity by controlling the false discovery rate. The Annals of Statistics **34**(2), 584–653 (2006)
2. Blueschke, D., Savin, I.: No such thing as a perfect hammer: comparing different objective function specifications for optimal control. Central European Journal of Operations Research **25**(2), 377–392 (2017)
3. Chen, X., Xu, F., Ye, Y.: Lower bound theory of nonzero entries in solutions of l2-lp minimization. SIAM Journal on Scientific Computing **32**(5), 2832–2852 (2010)
4. Gentile, C.: The robustness of the p-norm algorithms. Machine Learning **53**(3), 265–299 (2003)
5. Grove, A.J., Littlestone, N., Schuurmans, D.: General convergence results for linear discriminant updates. Machine Learning **43**(3), 173–210 (Jun 2001)
6. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)
7. Landman, B.A., Huang, A.J., Gifford, A., Vikram, D.S., Lim, I.A.L., Farrell, J.A., Bogovic, J.A., Hua, J., Chen, M., Jarso, S., et al.: Multi-parametric neuroimaging reproducibility: a 3-T resource study. Neuroimage **54**(4), 2854–2866 (2011)
8. Li, Z., Tang, J., He, X.: Robust structured nonnegative matrix factorization for image representation. IEEE Transactions on Neural Networks and Learning Systems **29**(5), 1947–1960 (2018)
9. Marcus, D.S., Wang, T.H., Parker, J., Csernansky, J.G., Morris, J.C., Buckner, R.L.: Open access series of imaging studies (oasis): Cross-sectional MRI data in young, middle aged, nondemented, and demented older adults. J. Cognitive Neuroscience **19**(9), 1498–1507 (Sep 2007). https://doi.org/10.1162/jocn.2007.19.9.1498, http://dx.doi.org/10.1162/jocn.2007.19.9.1498
10. Pham, C.H., Ducournau, A., Fablet, R., Rousseau, F.: Brain mri superresolution using deep 3d convolutional networks. In: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017). pp. 197–200 (April 2017). https://doi.org/10.1109/ISBI.2017.7950500
11. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
12. Worth, A.J.: MGH CMA internet brain segmentation repository (IBSR). http://www.cma.mgh.harvard.edu/ibsr/ (2010)
13. Ye, Y.F., Shao, Y.H., Deng, N.Y., Li, C.N., Hua, X.Y.: Robust Lp-norm least squares support vector regression with feature selection. Applied Mathematics and Computation **305**, 32 – 52 (2017)
14. Zhang, C., Li, D., Tan, J.: The support vector regression with adaptive norms. Procedia Computer Science **18**, 1730 – 1736 (2013)