Visualization of Neural Networks Using Saliency Maps

Niels J. S. Mørch[+‡] Ulrik Kjems[+] Lars Kai Hansen[+]
Claus Svarer[‡] Ian Law[‡] Benny Lautrup[†] Steve Strother[♮] Kelly Rehm[♮]

[+] Electronics Institute
Technical University of Denmark
DK-2800 Lyngby, Denmark

[‡] Department of Neurology
National University Hospital, Rigshospitalet
DK-2100 Copenhagen Ø, Denmark

[†] Niels Bohr Institute
University of Copenhagen
DK-2100 Copenhagen Ø, Denmark

[♮] PET Imaging Service, Va Medical Center
Radiology and Health Informatics Depts.
University of Minnesota, Minneapolis
Minnesota, 55417, USA

E-Mail : `nmorch@ei.dtu.dk`

ABSTRACT

The saliency map is proposed as a new method for understanding and visualizing the non-linearities embedded in feed-forward neural networks, with emphasis on the ill-posed case, where the dimensionality of the input-field by far exceeds the number of examples. Several levels of approximations are discussed. The saliency maps are applied to medical imaging (PET-scans) for identification of paradigm-relevant regions in the human brain.
Keywords: saliency map, model interpretation, ill-posed learning, PCA, SVD, PET.

1. Introduction

Mathematical modeling is of increasing importance in medical informatics. In bio-medical context the aim of neural network modeling is often twofold. Besides using empirical relations established within a given model, there is typically a wish to interpret the model in order to achieve an *understanding* of the processes underlying and generating the data. This paper presents a new tool for such opening of the neural network "black box".

Our method is aimed at neural network applications where the network is trained to provide a relation between huge, highly correlated, measurements and simple "labels". The measurement could e.g. be a spectrum, an image, or as in our particular case a brain scan volume. The label could be a concentration, a diagnosis etc.

In neural network applications, an important aspect of the training process is the architecture synthesis. An architecturally optimized network supplies structural information about the input field as used by the model, thus giving a qualitative measure of importance.

The output of our new procedure is a "map" *quantifying* the importance (*saliency* c.f. [7]) of each individual component of the measurement (i.e. pin, pixel, or voxel) with respect to the obtained empirical relation. Hopefully, this so-called *saliency map* will assist the modeler in interpreting the model and in communicating the interpretation to the end-user.

In bio-medical context it is often hard (not to say expensive) to gather large samples of data. Hence, if modeling from high dimensional data based on small samples, one faces an extremely ill-posed learning problem and standard practice has been to apply hand crafted tools ("a priori knowledge") for preprocessing and data reduction in order to bring down the dimensionality of the neural network. However, we have recently shown that one may cure this extremely ill-posed problem using straightforward linear algebra *without loss of information* [2], [5]. The scheme achieves *massive weight sharing* [7] by projecting the high dimensional data onto a low dimensional basis spanning the so-called signal space of the training set input vectors. The saliency map is an attempt to visualize this induced geometry and the specific manner in which this geometry is used by the trained network.

As a specific case, we consider modeling of images obtained from Positron-Emission-Tomography

(PET)-scans which is a technique offering 3-dimensional volume measurements of human brain activity. A neural network may be trained using supervised learning on a given training set of PET-scans [2], [5]. We investigate two cases, based on two sets of 64 scans each (8 subjects scanned 8 times): one where the subjects perform an eye movement task according to a graduated (parameterized) paradigm [6], and one where they perform a finger opposition task [12]. In the first case the network is trained to predict the paradigm graduation parameter—the frequency of the saccadic eye movements—using the measured activation patterns in the brain volume as input. In the latter the network is trained to classify the measured activation patterns as rest or activated (i.e. doing the finger opposition task). Since the models are nonlinear, the interpretations are not straightforward. In this particular case the saliency map can be viewed as a tool for visualizing the regions in the brain, which are related most strongly to the specific tasks.

2. The Saliency Map

It is well-known that affine preprocessing [8, 10] can assist training and generalization significantly. Affine preprocessing of an input vector $\boldsymbol{x}_j$ (i.e. an element of the training set of inputs $\mathbf{X} = [\boldsymbol{x}_1 \dots \boldsymbol{x}_J]$) can be expressed as $\boldsymbol{v}_j = \mathbf{B}^T(\boldsymbol{x}_j - \boldsymbol{c})$. In fact, translating by the training set averaged input vector $\boldsymbol{c} = \bar{\boldsymbol{x}}$ and computing the projection matrix $\mathbf{B}$ from a diagonalization of the input covariance matrix we may obtain $\boldsymbol{v}_j$ as the principal components[1] of $\mathbf{X}$. For simplicity we replace $\boldsymbol{x}_j - \boldsymbol{c}$ with $\boldsymbol{x}_j$ in the following, without loss of generality.

In image or volume processing, where the number of input channels I is often much greater than the number of examples J, a transformation like above can be used to reduce the dimensionality of the data-representation. However, it should be noted that within our scheme for handling extremely ill-posed problems the preprocessing doesn't necessarily reduce the data[2], in contrast to what is often the purpose when employing PCA, but may merely transform the data to a convenient (orthogonal) basis—thus we may have $\text{rank}(\mathbf{X}) = \text{rank}([\boldsymbol{v}_1 \dots \boldsymbol{v}_J])$. In this way we map the high dimensional input data vector onto a much smaller data vector of *projections*—hence, enforcing relations between elements of the weights connecting input to hidden units of the feed forward neural network, in other words we achieve a massive weight sharing. For a more detailed description see [2], [5]. Spelled out in terms of the neural network this can be written,

$$\begin{aligned} F(\mathcal{W}, \mathbf{B}, \boldsymbol{x}) &= F(\mathcal{W}, \mathbf{B}^T \boldsymbol{x}) \\ &= \sum_a W_a \tanh\left(\boldsymbol{w}_a^T \mathbf{B}^T \boldsymbol{x}\right) \end{aligned} \tag{1}$$

which is now a function of the input $\boldsymbol{x}$ projected on the set of $K \leq \text{rank}(\mathbf{X})$ basis vectors[3] $\boldsymbol{b}_k$ forming the basis $\mathbf{B} = \left[\boldsymbol{b}_1 \dots \boldsymbol{b}_K\right]$ and a set of weight parameters $\mathcal{W} = \{W_a, \boldsymbol{w}_a\}$. The constrained weights are in turn optimized using a training set[4] $\mathcal{T} = \left\{(\boldsymbol{x}_j, y_j) \,|\, j = 1, \dots, J\right\}$ by minimizing the cost function with respect to $\mathcal{W}$

$$E(\mathcal{W}, \mathbf{B}, \mathcal{T}) = \frac{1}{J} \sum_{j=1}^{J} \left(y_j - F(\mathcal{W}, \mathbf{B}^T \boldsymbol{x}_j)\right)^2, \tag{2}$$

and we define:

> The saliency of input channel i (or pixel i if $\boldsymbol{x}$ is an image vector) is the change in the cost-function when the i'th input channel is removed.

This removal can be thought of as changing the basis vectors in $\mathbf{B}$, resulting in the new basis $\check{\mathbf{B}}^i$

$$\check{\boldsymbol{b}}^i_{k,i'} = \begin{cases} \boldsymbol{b}_{k,i'} & i' \neq i \\ 0 & i' = i \end{cases} \tag{3}$$

i.e. setting the i'th component[5] of all basis vectors to 0. Introducing this new basis, the model should be retrained to yield a new set of weight parameters $\check{\mathcal{W}}^i$. The saliency of input channel i is therefore

$$\delta E_i = E(\check{\mathcal{W}}^i, \check{\mathbf{B}}^i, \mathcal{T}) - E(\mathcal{W}, \mathbf{B}, \mathcal{T}). \tag{4}$$

If pruning is used to eliminate the effect of noise it should be applied to the full network prior to the calculation of the saliency map, so the retraining after removing the individual inputs conserves the network architecture.

Ideally one could estimate the change in generalization ability [11]. Such an estimate would—given a limited amount of data—be quite inaccurate, and since we only want to use the saliency map for comparing the relative input importance, it seems reasonable to consider only the change in the training error as indicated in equation (4).

Further approximations depend on the specific problem: In image processing the number of input channels (pixels) is often much greater than the number of examples, so that the computational burden of the direct computation of the saliency may be impractical. For such applications we develop

[1] The principal components as obtained from SVD (Singular Value Decomposition), or PCA (Principal Component Analysis). In either case the basis vectors correspond to the eigenvectors of the input data covariance matrix, see [4].

[2] In the sense of loosing information.

[3] See also section 2.1 for a more detailed explanation of the notation.

[4] The outputs are assumed scalar for simplicity.

[5] By the notation $\boldsymbol{b}_{k,i}$ we mean the i'th element of $\boldsymbol{b}_k$.

approximations of the saliency map using an expansion of the cost function. This is further described in section 2.1.

Finally, let us note that the saliency map as such is not confined to the ill-posed learning problem. In more conventional neural network applications, where the number of network inputs I is much smaller than the number of examples J, the saliency is similar to the *sensitivity measure* proposed in [14], [13] and [9], and to the Optimal Cell Damage Scheme suggested in [1]. In this case the removal of a single input may cause a notable change in the optimal weights thus making the I network retrainings essential (in contrast to the ill-posed case, as we shall see).

2.1. The Saliency Map in the Ill-Posed Case

As discussed a significant computational reduction can be obtained by projecting on the set of basis vectors $\mathbf{B}$ spanning the signal space[6] $\mathcal{S}$, if $I \gg J$.

It is easily seen [2], [5] that training in this case preserves signal space, i.e., if the initial weights of a hidden unit are confined to signal space they will stay there during training. This is a consequence of the fact that the cost function is independent of any component of the weight parameters outside signal space, $\mathcal{S}$, regardless of the basis $\mathbf{B}$ used for representing the data, as long as $\mathbf{B}$ spans $\mathcal{S}$.

After preprocessing the neural network is not fed the actual pixel data, but the projection of the images on the basis $\mathbf{B}$. This justifies the notation $F(\mathcal{W}, \mathbf{B}^T \boldsymbol{x}_j)$ for the model, in that the model can be said to be working on the projected data $\boldsymbol{v}_j = \mathbf{B}^T \boldsymbol{x}_j$.

2.1.1. Approximating the Saliency Map

If the number of input channels I is large, the task of retraining I networks—i.e. to compute $\check{\mathcal{W}}^i$ as implied by equation (4)—is immense. In this section some approximations are presented to speed up the computation.

The second order expansion of the cost function with respect to the basis vectors and the weight vector $\boldsymbol{u} = \left[\boldsymbol{w}_1^T \ldots \boldsymbol{w}_A^T \; W_1 \ldots W_A\right]^T$ consisting of all the parameters in $\mathcal{W}$ is given by

$$\begin{aligned} \delta E \;\; &\simeq \;\; \sum_{k=1}^{K} \frac{\partial E}{\partial \boldsymbol{b}_k^T} \delta \boldsymbol{b}_k + \frac{\partial E}{\partial \boldsymbol{u}^T} \delta \boldsymbol{u} \\ &+ \;\; \frac{1}{2} \sum_{k=1}^{K} \delta \boldsymbol{b}_k^T \frac{\partial^2 E}{\partial \boldsymbol{b}_k \partial \boldsymbol{b}_k^T} \delta \boldsymbol{b}_k + \frac{1}{2} \delta \boldsymbol{u}^T \frac{\partial^2 E}{\partial \boldsymbol{u} \partial \boldsymbol{u}^T} \delta \boldsymbol{u} \\ &+ \;\; \sum_{k=1}^{K} \delta \boldsymbol{b}_k^T \frac{\partial^2 E}{\partial \boldsymbol{b}_k \partial \boldsymbol{u}^T} \delta \boldsymbol{u}, \end{aligned} \tag{5}$$

[6] We denote the space spanned by the input vectors $\boldsymbol{x}_j$ in the training set $\mathcal{T}$ by signal space $\mathcal{S} = \operatorname{span}\{\boldsymbol{x}_j\}$.

where $\delta \boldsymbol{b}_k$ is the change in the k'th basis vector, and $\delta \boldsymbol{u}$ is the change in the optimal weight parameters, due to the changed basis. If the network is fully trained $\frac{\partial E}{\partial \boldsymbol{u}} = \mathbf{0}$ so the second term vanishes[7].

In the ill-posed case, modeling will only be meaningful if the stochastic part of the signal is highly correlated, i.e., the individual pixels are spatially correlated. Thus it can be assumed that the term $\delta \boldsymbol{u}$ roughly scales inversely with the number of inputs, i.e. as $1/I$. We therefore neglect all terms scaling with $\delta \boldsymbol{u}$ yielding

$$\delta E \simeq \sum_{k=1}^{K} \frac{\partial E}{\partial \boldsymbol{b}_k^T} \delta \boldsymbol{b}_k + \frac{1}{2} \sum_{k=1}^{K} \delta \boldsymbol{b}_k^T \frac{\partial^2 E}{\partial \boldsymbol{b}_k \partial \boldsymbol{b}_k^T} \delta \boldsymbol{b}_k, \tag{6}$$

thus eliminating the effect of retraining, effectively estimating $\delta E_i = E(\mathcal{W}, \check{\mathbf{B}}^i, \mathcal{T}) - E(\mathcal{W}, \mathbf{B}, \mathcal{T})$ c.f. equation (4). This is in line with the Optimal Brain Damage scheme [7] for estimating weight saliency and the approximation is indeed supported by the numerical example. Since we compute the saliency for one input channel at a time, the off-diagonal elements of $\frac{\partial^2 E}{\partial \boldsymbol{b}_k \partial \boldsymbol{b}_k^T}$ vanish, so we finally get

$$\delta E_i \simeq \sum_{k=1}^{K} \frac{\partial E}{\partial \boldsymbol{b}_{k,i}} \delta \boldsymbol{b}_{k,i} + \frac{1}{2} \sum_{k=1}^{K} \frac{\partial^2 E}{\partial \boldsymbol{b}_{k,i}^2} \delta \boldsymbol{b}_{k,i}^2. \tag{7}$$

For the two-layer network specified in equation (1), with $h_{aj} = \tanh\left(\boldsymbol{w}_a^T \mathbf{B}^T \boldsymbol{x}_j\right)$ we find[8]

$$\begin{aligned} \frac{\partial E}{\partial \boldsymbol{b}_{k,i}} \;\; &= \;\; -\frac{2}{J} \sum_{j=1}^{J} \Big[\left(y_j - F(\mathcal{W}, \mathbf{B}^T \boldsymbol{x}_j)\right) \\ & \qquad \cdot \sum_a W_a (1 - h_{aj}^2) \boldsymbol{w}_{a,k} \boldsymbol{x}_{j,i} \Big] \\ &= \;\; -\frac{2}{J} \sum_{j=1}^{J} e_j s_{jk} \boldsymbol{x}_{j,i} \end{aligned} \tag{8}$$

where we have introduced the quantities $e_j = y_j - F(\mathbf{B}^T \boldsymbol{x}_j, \mathcal{W})$ and $s_{jk} = \sum_a W_a (1 - h_{aj}^2) \boldsymbol{w}_{a,k}$. By further invoking the Gauss-Newton approximation $\left(\frac{\partial^2 E}{\partial \boldsymbol{b}_k \partial \boldsymbol{b}_k^T} \simeq \sum_{j=1}^{J} \frac{\partial F}{\partial \boldsymbol{b}_k} \frac{\partial F}{\partial \boldsymbol{b}_k^T}\right)$ for least squares problems, see e.g. [7], yielding

$$\frac{\partial^2 E}{\partial \boldsymbol{b}_{k,i}^2} \simeq \frac{2}{J} \sum_{j=1}^{J} s_{jk}^2 \boldsymbol{x}_{j,i}^2, \tag{9}$$

[7] If we eliminate overfitting by pruning the network, i.e. forcing some parameters $\boldsymbol{u}'$ to $\mathbf{0}$, only the remaining parameters $\boldsymbol{u}^\star = \boldsymbol{u} \setminus \boldsymbol{u}'$ are optimized so that $\frac{\partial E}{\partial \boldsymbol{u}^\star} = \mathbf{0}$. On the other hand, we will generally have $\frac{\partial E}{\partial \boldsymbol{u}'} \neq \mathbf{0}$, which may cause negative estimates of the saliency. This can be explained as follows: If the network models from a subspace of $\mathcal{S}$, called model-space $\mathcal{M}$, one might say that the basis change in (3) perturbs signal space, so that some of the noise eliminated by pruning re-enters $\mathcal{M}$. Sometimes this will allow the model to perform better on the training set, thus yielding negative saliencies. We therefore choose to interpret these as zero.

[8] Again $\boldsymbol{w}_{a,k}$ means the k'th element of $\boldsymbol{w}_a$, and $\boldsymbol{x}_{j,i}$ the i'th element of $\boldsymbol{x}_j$.

and since we remove only one input channel in the basis, i.e. $\delta \boldsymbol{b}_{k,i} = -\boldsymbol{b}_{k,i}$, we get

$$\delta E_i = \frac{2}{J}\sum_{k=1}^{K}\sum_{j=1}^{J} e_j s_{jk} \boldsymbol{x}_{j,i} \boldsymbol{b}_{k,i} + \frac{1}{J}\sum_{k=1}^{K}\sum_{j=1}^{J} s_{jk}^2 \boldsymbol{x}_{j,i}^2 \boldsymbol{b}_{k,i}^2. \tag{10}$$

as the estimate of the saliency map.

3. Ill-posed Example: Modeling from PET images

We now proceed to demonstrate the practical use of the saliency map. Positron-Emission-Tomography (PET) is a way of indirectly measuring the neural activity of different regions of the human brain, resulting in 3-dimensional images. As the dimension of the images is very large, affine preprocessing (projection of the data on the corresponding PCA-basis) is applied, thus reducing the computational requirement of the modeling.

More specifically, we first examined 64 PET-scans of 8 subjects, each scanned 8 times, exposed to 8 different levels of saccadic eye movement activation [6]. We thus analyze $J = 64$ image vectors of $I = 128 \times 128 \times 48 = 768432$ voxels[9].

A two-layer feed-forward neural network was trained to predict the paradigm activation level (the frequency of the saccadic eye movements) from the 64 3-dimensional brain volumes.

An estimated saliency map was computed employing the approximation in equation (10). In figure 1 iso surfaces (surfaces of equal saliency) capturing the most salient voxels are depicted as bright bodies floating in a box. To help localize the salient areas, slices of a corresponding anatomical brain image (an MR scan) are shown on the walls of the box, with the shadows of the salient bodies projected in black. The slices correspond to the middle of the brain, one in each of the three dimensions.

The result is in correspondence with what has been found using other analysis methods—e.g. Statistical Parametric Mapping (SPM), and the Scaled Subprofile Model (SSM)—on the same data [6], [12]. The larger cluster of salient pixels, as seen in the back of the brain, is identified as the *visual cortex.*

To demonstrate the accuracy of the 1st and 2nd order approximations of the saliency, c.f. equation (10), we computed the images shown in figure 2. The first column shows the true change in the cost function[10] for horizontal slices through the volume corresponding the AC-PC[11] level -17mm, AC-PC, and AC-PC + 17mm. This corresponds to expanding E to infinitely high order with respect to $\boldsymbol{b}$. The second and third columns are the 1st and 2nd order

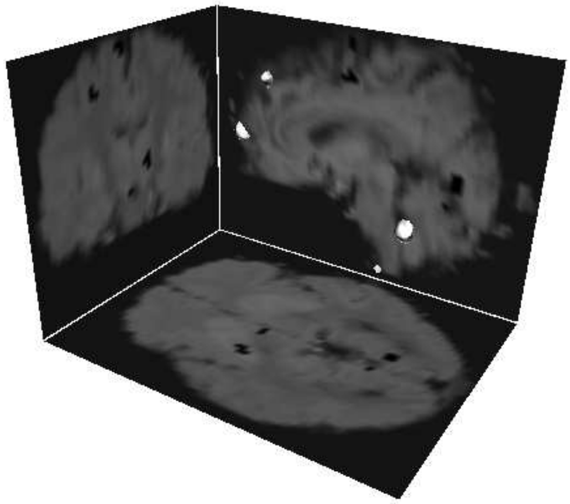

Fig. 1: Using the saliency map to asses paradigm related brain regions in the saccadic eye movement task. The most salient voxels are depicted as iso surfaces (surfaces of equal saliency) here seen as bright bodies floating in a box with slices of a corresponding anatomical brain scan depicted on the walls. Shadows of the iso surfaces are projected in black on the walls. The larger cluster in the back of the brain is the *visual cortex.*

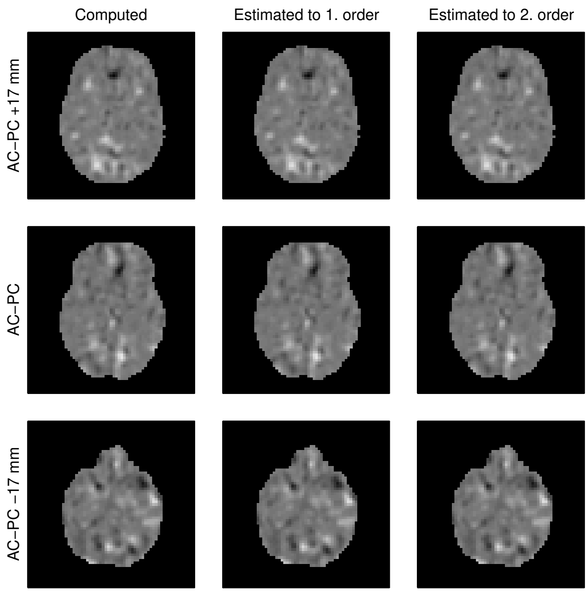

Fig. 2: From left to right: Computed saliency map, 1st order, and 2nd order approximations, all for 3 different slices of the brain. The slices correspond to the AC-PC level -17 mm, the AC-PC level and the AC-PC level + 17mm. Bright areas have high saliencies. In the specific case ($I = 34863$ pixels) all columns are almost identical—thus validating the approximations. In fact, the 2nd order term seems visually negligible.

[9] Of these a large portion is masked out, leaving vectors of "only" active 34863 voxels.

[10] Computed as the change in the cost function *without* retraining $\delta E_i = E(\mathcal{W}, \tilde{\mathbf{B}}^i, \mathcal{T}) - E(\mathcal{W}, \mathbf{B}, \mathcal{T})$, so that only the effects of neglecting the higher order 'pure' $\delta \boldsymbol{b}_k$ terms of (5) and (6) are assessed.

[11] Anterior Comisura - Posterior Comisura, which are easily identified centers in the brain, and thus used for reference.

approximations of (10). It is evident, that even the 1st order term alone is a useful approximation in the case of $I = 34863$ voxels.

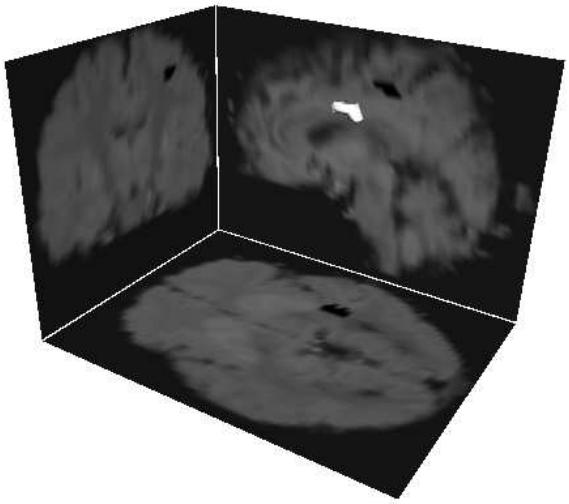

Fig. 3: Saliency map of the finger opposition task. The most salient voxels are depicted as iso surfaces (surfaces of equal saliency) here seen as bright bodies floating in a box with slices of a corresponding anatomical brain scan depicted on the walls. Shadows of the iso surfaces are projected in black on the walls. The salient area identified is the *primary sensory-motor cortex.*

Secondly, the saliency map was computed for a neural network modeling the finger opposition task, which involves areas of the brain controlling motion. The data has previously been analyzed in [12]. Again, 8 subjects were scanned 8 times each, 4 times resting and 4 times doing the finger opposition task. Thus the paradigm is on/off corresponding to a problem of classification[12]. Figure 3 shows the saliency map in a manner similar to figure 1. The method clearly identifies the area known as the *primary sensory-motor cortex.*

Further, we investigated the effect of the dimension of the input-field I, on the approximation (10). For simplicity this is done on a single slice, which is sub-sampled to yield $Q = 9$ datasets with decreasing I. After performing the entire modeling procedure Q times, we measure as a function of I the normalized mean squared error for both the 1st and 2nd order expansions, i.e

$$
\begin{aligned}
f_1(I) &= \frac{\sum_{i=1}^{I}\left(\delta E_{c,i} - \delta E_{1,i}\right)^2}{\sum_{i=1}^{I}\delta E_{c,i}^2} \\
f_2(I) &= \frac{\sum_{i=1}^{I}\left(\delta E_{c,i} - \delta E_{2,i}\right)^2}{\sum_{i=1}^{I}\delta E_{c,i}^2} \\
\delta E_{1,i} &= \sum_{k=1}^{K}\frac{\partial E}{\partial \boldsymbol{b}_{k,i}}\delta \boldsymbol{b}_{k,i} \\
\delta E_{2,i} &= \delta E_{1,i} + \frac{1}{2}\sum_{k=1}^{K}\frac{\partial^2 E}{\partial \boldsymbol{b}_{k,i}^2}\delta \boldsymbol{b}_{k,i}^2 \\
\delta E_{c,i} &= E(\mathcal{T},\mathcal{W},\check{\mathbf{B}}^i) - E(\mathcal{T},\mathcal{W},\mathbf{B})
\end{aligned}
\tag{11}
$$

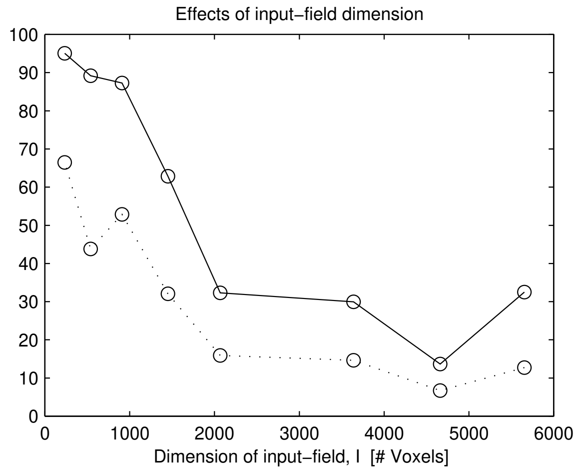

Fig. 4: Normalized mean squared error of the 1st (—) and 2nd (···) order approximations of the saliency. With increasing input-field dimension I, the errors decrease—for large I the 1st order approximation suffices.

These quantities are shown in figure 4. We see that the error introduced by the approximations decreases when I gets large. Further, for very large I, the 2nd order term seems negligible. This is in line with the visual impression of figure 2.

Finally, let us note that the saliency map easily computes for linear models as well.

4. Discussion

We have proposed the saliency map as a new method for understanding and visualizing feed-forward neural networks. Furthermore, several levels of approximations have been derived providing significant computational savings. The viability of the approach was demonstrated on a series of 3D brain activation volumes.

Though the emphasis has been on the so-called ill-posed case, the proposed technique can easily be

[12]Note that for classification problems better optimization schemes (costfunctions) exist, see e.g [3].

applied to the more standard setting, i.e. the well-posed case.

5. Acknowledgments

This research has been supported by the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center CONNECT, and the US National Institutes of Health's Human Brain Project through grant DA09246.

References

[1] T. Cibas *et al.*, "Variable selection with optimal cell damage," *Proceedings of the International Conference on Artificial Neural Networks, pp. 727–730*, 1994.

[2] L. K. Hansen, B. Lautrup, I. Law, N. Mørch, and J. Thomsen, "Extremely ill-posed learning," CONNECT *Preprint.* Available via anonymous ftp `ei.dtu.dk : dist/hansen.ill-posed.ps.Z.`, Aug. 1994.

[3] M. Hintz-Madsen *et al.*, "Design and evaluation of neural classifiers - application to skin lesion classification," *To appear: 1995 IEEE Workshop on Neural Networks for Signal Processing (NNSP'95).* Available via anonymous ftp `ei.dtu.dk : dist/1995/hintz.nnsp95.ps.Z`, 1995.

[4] J. E. Jackson, *A User's Guide to Principal Components.* Wiley Series on Probability and Statistics, John Wiley and Sons, 1991.

[5] B. Lautrup, L. K. Hansen, I. Law, N. Mørch, C. Svarer, and S. Strother, "Massive weight sharing: A cure for extremely ill-posed problems," in *Proceedings of Workshop on Supercomputing in Brain Research: From Tomography to Neural Networks, HLRZ, KFA Jülich, Germany*, (H. J. Hermann, D. E. Wolf, and E. P. Pöppel, eds.), pp. 137–148, Nov. 1994.

[6] I. Law *et al.*, "A characterization of the frequency related cerebral response during sensory-guided saccades," *In preparation*, 1995.

[7] Y. Le Cun, J. S. Denker, and S. Solla, "Optimal brain damage," *Advances in Neural Information Processing Systems 2, pp. 598–605*, 1990.

[8] Le Cun, Y., I. Kanter, and S. Solla, "Eigenvalues of covariance matrices: Application to neural-network learning," *Physical Review Letters*, vol. 66, Number 18:pp 2396–2399, May 1991.

[9] J. Moody, "Prediction risk and architecture selection for neural networks," in *From Statistics to Neural Networks, Theory and Pattern Recognition Applications*, (V. Cherkassky, J. H. F. H., and H. Wechsler, eds.), pp. 147–165, Springer Verlag, 1992.

[10] J. S. Orfanidis, "Gram-schmidt neural nets," *Neural Computation*, vol. 2, pp 116–126, 1990.

[11] M. W. Pedersen, L. K. Hansen, and J. Larsen, "Pruning with generalization based saliencies: γOBD, γOBS," *Submitted to: Advances in Neural Information Processing Systems (NIPS95).* Available via anonymous ftp `ei.dtu.dk : dist/1995/with.nips95.ps.Z`, 1995.

[12] S. C. Strother, J. R. Anderson, K. A. Schaper, J. J. Sidtis, J. S. Liow, R. P. Woods, and D. A. Rottenberg, "Principal component analysis and the scaled subprofile model compared to intersubject averaging and statistical parametric mapping: I. "Functional connectivity" of the human motor system studied with [15-o]water pet," *Journal of Cerebral Blood Flow and Metabolism*, vol. 15, pp. 738–753, 1995.

[13] J. Utans and J. Moody, "Principled architecture selection for neural networks: Application to corporate bond rating prediction," *Advances in Neural Information Processing Systems 4*, 1991.

[14] J. Utans and J. Moody, "Selecting neural network architectures via the prediction risk: Application to corporate bond rating prediction," *Proc. First International Conference in Artificial Intelligence Applications on Wall Street*, 1991.