# ICASSO: SOFTWARE FOR INVESTIGATING THE RELIABILITY OF ICA ESTIMATES BY CLUSTERING AND VISUALIZATION

Johan Himberg[1] and Aapo Hyvärinen[1,2]
1) Neural Networks Research Centre
Helsinki Univ. of Technology, P.O. Box 5400, 02015 HUT, Finland
2) Helsinki Institute for Information Technology/BRU
Dept. of Computer Science, Univ. of Helsinki, Finland
johan.himberg@hut.fi, aapo.hyvarinen@helsinki.fi

**Abstract. A major problem in application of independent component analysis (ICA) is that the reliability of the estimated independent components is not known. Firstly, the finite sample size induces statistical errors in the estimation. Secondly, as real data never exactly follows the ICA model, the contrast function used in the estimation may have many local minima which are all equally good, or the practical algorithm may not always perform properly, for example getting stuck in local minima with strongly suboptimal values of the contrast function. We present an explorative visualization method for investigating the relations between estimates from FastICA. The algorithmic and statistical reliability is investigated by running the algorithm many times with different initial values or with differently bootstrapped data sets, respectively. Resulting estimates are compared by visualizing their clustering according to a suitable similarity measure. Reliable estimates correspond to tight clusters, and unreliable ones to points which do not belong to any such cluster. We have developed a software package called *Icasso* to implement these operations. We also present results of this method when applying *Icasso* on biomedical data.**

## INTRODUCTION

Independent component analysis (ICA) is a general-purpose statistical model that has been used in many applications, see [7] for a review. In data analysis applications, it is hoped that the independent components reveal something interesting of a multi-dimensional data set. Some recent applications include brain imaging applications [10, 12, 20] and bioinformatics [9].

A major problem in application of ICA to data analysis is that the reliability of the estimated independent components is not known. The ICA

algorithm gives a specified number of components, but it is not known which ones are to be taken seriously. In fact, most algorithms give different components when run multiple times. This is in stark contrast to such methods as principal component analysis, whose results are unique. One reason for this is that most algorithms only find a local minimum of the objective or "contrast" function they are trying to optimize.

Algorithmic uncertainty is only part of the problem. Even with algorithms which are deterministic and always find the global optimum of their objective function, the valid interpretation of the results need some analysis of the statistical reliability or significance of the components. There are two different reasons for this. Firstly, as real data never exactly follows the ICA model, the contrast function used in the estimation may have many local minima which are all equally good. The independent components are simply not well-defined in this case. Secondly, even in the extreme where the data is exactly generated according to the ICA model, the finite sample size induces statistical errors in the estimation—this is the case where classical analysis of statistical significance and confidence intervals would be needed [13].

Here, we present a tool for investigating the reliability of the independent components. The method is based on estimating a large number of independent components, and visualizing their clustering in the signal space. Each estimated independent component is one point in the space. If an independent component is reliable, (almost) every run of the algorithm should produce a point that is very close to the ideal component corresponding to the cluster center. Thus, reliable independent components correspond to tight clusters, and unreliable ones correspond to points which do not belong to any cluster.

We investigate both algorithmic and statistical reliability by running the ICA algorithm many times with different initial values, or with different bootstrapped data sets, respectively. Obviously, randomization for ICA has previously been used *ad hoc*[1] (see also [13]). Our focus is on constructing a comprehensive set of methods supported by explorative data analysis and visualization.

Furthermore, we are able to combine information from several runs of the algorithms and obtain a set of components (or cluster centers) that is better than any of component sets provided by a single run. Note also that even if we were able to compare the sets using a global goodness measure (such as the squared reconstruction error in k-means), and choose the set that is the best, we would thus gain no information on the reliability on each cluster center or independent component.

We have developed a software package called *Icasso*[2] to implement these operations and visualize the results.

[1] R. Vigário, personal communication, 2003

[2] The MATLAB package is available at http://www.cis.hut.fi/jhimberg/icasso

## METHODS

In the following sections, we explain the technical details of each phase that is implemented in *Icasso*, a tool that is specialized for studying independent component estimates. The same approach basically holds for comparing randomized estimates originating from some other method when a similarity measure between the estimates is available.

In general, *Icasso* consists of the following steps:

1. Parameters for the estimation algorithm(s) are selected: e.g., for FastICA the estimation approach (symmetrical or deflatory), contrast function, etc.

2. The estimation is run $M$ times using the selected training parameters. Each time the data is bootstrapped and/or the initial conditions of the estimation algorithm are changed.

3. The estimates are clustered according to their mutual similarities. In principle, the clustering method can be freely selected. We apply agglomerative clustering with average-linkage criterion.

4. The clustering is visualized as a 2-D plot. The user investigates how dense the clusters are. The clustering of the estimates is expected to yield information on the reliability (robustness) of estimation. A compact cluster emerges when a similar estimate repeatedly comes up despite of the randomization.

5. The user can retrieve the estimates belonging to certain cluster(s) for further analysis and visualization.

To complete steps 1–3 the user simply sets the FastICA parameters and launches a resampling and clustering application. In step 4, the user explores the clustering by launching an interactive visualization application. The user may examine the quality of the clusters and rank them accordingly. Subsequently, *Icasso* visualizes the similarity matrix between all the estimates and their partition into clusters in a single graph. Thus, the user can examine relationships between estimates and clusters in detail. In step 5, the user can retrieve any set of estimates that belong to certain cluster(s). After this, it is up to the user's needs what to do with the results. For example, one can form the average or the centrotype of the estimates belonging to a single compact cluster. This can be considered to be a more reliable estimate of a component than an estimate from a single run of ICA algorithm.

Our criteria for selecting the specific methods for *Icasso* were that i) methods for completing each subtask are well-known, ii) they support visualization and explorative data-analysis, and iii) in order to avoid redundant work, existing, publicly available building blocks should be used.[3]

[3]We use FastICA Toolbox 2.1 and SOM Toolbox 2.0 [18] for MATLAB, both freely available from http://www.cis.hut.fi/research/software.shtml

### Generating the estimates and comparing them

We consider the standard linear, noise-free ICA model $\mathbf{x} = \mathbf{A}\mathbf{s}$ of independent sources $\mathbf{s}$ and a mixing matrix $\mathbf{A}$. However, what is often estimated in practice, is the demixing matrix $\mathbf{W}$ for $\mathbf{s} = \mathbf{W}\mathbf{x}$, where $\mathbf{W}$ is a (pseudo)inverse of $\mathbf{A}$ [7].

The algorithm is run $M$ times on data $\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2 \dots \mathbf{x}_N]$ consisting of $N$ samples of $k$-dimensional vectors. The estimates of demixing matrices $\hat{\mathbf{W}}_i$ from each run $i = 1, 2, \dots, M$ are collected into a single matrix $\hat{\mathbf{W}} = [\hat{\mathbf{W}}_1^T\ \hat{\mathbf{W}}_2^T \cdots \hat{\mathbf{W}}_M^T]^T$. If $n_i$ independent components are estimated on each round, we get $K = \sum_i n_i$ estimates, and the size of $\hat{\mathbf{W}}$ will be $K \times k$.

We can resample independent component estimates by a) *Randomizing the initial condition*: FastICA is run $M$ times for the same data $\mathbf{X}$, so that for each run the algorithm starts from a new random initial condition; b) *Bootstrapping*: FastICA is run $M$ times. The initial condition is kept the same in every run, but the data is bootstrapped every time; and c) *Bootstrapping with randomized initial condition* as a combination of a) and b).

A natural measure of similarity between the estimated independent components is the absolute value of their mutual correlation coefficients $r_{ij}$, $i, j = 1, \dots, K$. Straightforward calculations show that they can be obtained as elements of $\mathbf{R} = \hat{\mathbf{W}}\mathbf{\Sigma}\hat{\mathbf{W}}^T$ where $\mathbf{\Sigma}$ is the covariance matrix for $\mathbf{X}$. The final similarity matrix has then elements

$$\sigma_{ij} = |r_{ij}|. \tag{1}$$

Later, we use clustering methods and validity indices that expect dissimilarities (distances) in their standard form. We transform the similarity matrix into a dissimilarity matrix with elements $d_{ij}$. A simple way to make this transformation is obviously [4]:

$$d_{ij} = 1 - \sigma_{ij}. \tag{2}$$

### Clustering the estimates

We can partition the set of all estimates $C$ into $L$ disjoint clusters $C = \bigcup_{m=1}^{L} C_m$ using some basic clustering algorithm and the dissimilarity measure in Eq. 2. Agglomerative hierarchical clustering is a well-known method for a modest number of objects [4, 5]). The tree-like hierarchy (dendrogram) produced by agglomeration is intuitively appealing in the sense that all clusters implied by lower levels of the tree are always subsets of clusters at the higher levels. As a result, the user is able to explore and compare the different level(s) of clustering that are readily computed. The simplest way to obtain a partition of $L$ clusters from a dendrogram is to cut it at level where $L$ clusters are present. There are numerous reviews and studies on the multitude of agglomeration strategies and cluster validity indices, see, e.g., [1, 2, 4, 5, 11]. Unfortunately, there is no easy way of selecting the optimal agglomeration strategy for a specific data, and the selection must be

based on problem specific considerations. The same applies also to selecting a clustering validity index for determining a "natural" number of clusters [2]. Three basic agglomeration strategies that operate directly on the similarity matrix are single-link (SL), complete-link (CL), and group average-link (AL).

*Icasso* uses AL as default choice of agglomeration strategy. This is because, firstly, SL is in general reported to be more sensitive for noise than AL and CL. Secondly, in our experiments with the benchmark data revealed that when $L < k$, CL starts to join clusters inconsistently.

We introduce a conservative cluster quality index $I_q$ in Eq. 3 that reflects the compactness and isolation of a cluster. It is computed as the difference between the average intra-cluster similarities and average extra-cluster similarities:

$$I_q(C_m) = \frac{1}{|C_m|^2} \sum_{i,j \in c_m} \sigma_{ij} - \frac{1}{|C_m||C_{-m}|} \sum_{i \in C_m} \sum_{j \in C_{-m}} \sigma_{ij}, \qquad (3)$$

where $C_{-m} = C - C_m$. Eventually $I_q(C_m)$ is one for an ideal cluster when Eq. 1 is used to compute $\sigma_{ij}$, and decreases when $C_m$becomes less compact and isolated.

We prefer leaving the final selection of the number of clusters $L$ to the user who can interactively explore the results produced by different levels of dendrogram. It is reasonable to start studying the clustering from the number of clusters $L$ equal to the data dimension $k$ and investigate the values of cluster quality index in rank order, see Fig. 1(a).

There are also quantitative indices for suggesting the clustering that best fits to the "natural" structure of the data. We considered five such validity indices that can be computed knowing only the dissimilarity matrix: R-index ($I_R$) referred in [8] and four of the Dunn-like indices in [2]. Empirical studies, e.g., [1, 2, 11] yield often different results depending on the character of the benchmarking data used without no clear indication of general superiority. Our own experiments on these indices did not suggest any definitive winner either. Currently, *Icasso* computes and shows R-index in its user interface, but we note that such an index should be used only by side of the explorative investigation.

### Visualization beyond the dendrogram

*Icasso* provides also a tool for getting a detailed look into the clustering results and relations between the clusters and individual estimates. The result of the hierarchical clustering is typically presented as a dendrogram, but also other types of visualization exist. Here, each estimate $i$ is plotted as a point on the display, and a convex hull bounds the estimates belonging to the same cluster [5]. This presentation allows visualizing similarities like $\sigma_{ij} = |r_{ij}|$ rather explicitly: the points are connected with lines whose thickness/color represent the similarities between them. See Fig. 1(b).

We apply projection methods related to multidimensional scaling (MDS) as suggested in [5] to approximate the original dissimilarities between esti-

mates by Euclidean distances in two dimensions. This should result in a projection, where the smaller a convex hull is, the more compact the corresponding cluster is. An ideal cluster should contract into a single point.

For this purpose, we compared the linear metric MDS (MMDS) [16], and two non-linear methods: Sammon's projection [15], and Curvilinear Component Analysis (CCA) [3]. In addition to visual comparison, we used a trustworthiness index proposed in [17]. Spatial proximity is one of the strongest visual indicators of grouping [21]. In order to be trustworthy, a projection should be such that one can trust the visual proximity as an indicator of similarity. The trustworthiness index in [17] is a function of the visual neighborhood size, and it must be evaluated for the neighborhood sizes of interest: according to [17] it is especially important that the trustworthiness is retained for small neighborhoods.

According to our experiments, CCA produces more trustworthy projections than MMDS and Sammon's method on the benchmarking data for dissimilarity measure in Eq. 2. We considered also Self-Organizing Map (SOM) based visualization since it is reported to be more trustworthy than many MDS-like methods [14, 17]. However, we abandoned SOM since its regular grid visualization forces the lines of the similarity graph to shadow each other more than they do on a non-uniform projection.

The projection can be further controlled by modifying Eq. 2 suitably, e.g.,

$$d^*_{ij} = \sqrt{1 - \sigma_{ij}}. \tag{4}$$

This spreads the distribution of the distances so that differences in size among the most compact clusters can be seen better. For this reason, *Icasso* uses transformation in Eq. 4 instead that of Eq. 2 for making the visualization, though the resulting projection is slightly less trustworthy on the benchmarking data. Using Eq. 4 can be additionally motivated by the fact that for two normalized zero mean vectors $\mathbf{u}_i$ and $\mathbf{u}_j$ their correlation coefficient $r_{ij} = \mathbf{u}_i^T \mathbf{u}_j$ and their Euclidean distance $\| \mathbf{u}_i - \mathbf{u}_j \| \propto \sqrt{1 - r_{ij}}$ [14]. However, for $|r_{ij}|$ the same transformation is not guaranteed to give a distance matrix that would exactly originate from Euclidean distances between points in $\mathbb{R}^k$.

## EXPERIMENTAL RESULTS

We experimented with a biomedical benchmarking data set described in more detail in [19]. The data consist of preprocessed signals originating from 122-channel whole-scalp magnetoencephalographic (MEG) measurements from brain. The original signals are band-pass filtered between $0.5 \ldots 45$ Hz, and the data dimension ($k$) is reduced from 122 to 20 using principal component analysis in order to reduce noise and overlearning [6]. The recording lasts about 2 minutes and contains 17730 samples. The measurements from the brain are disturbed by signals originating from various sources: heart beat, eye blinks and saccade, and other muscular activity—and a digital watch.

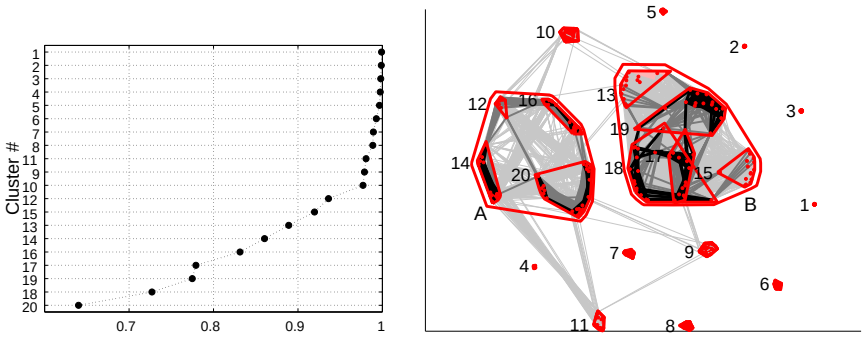
We run *Icasso* five times using three different settings. Setting I: random initial conditions, kurtosis as contrast function; II: as I, but hyperbolic tangent as the contrast function; and III: as I, but using both bootstrapping and random initial conditions. Each time number of randomizations ($M$) was 15, and the symmetrical approach was used in FastICA.

In the following, we present results for a particular test run from setting I. First, we select the number of clusters $L = k = 20$. Fig. 1(a) shows the quality index $I_q$ of ranking for the clusters whose relations are visualized in Fig. 1(b). Note how the diameter of the convex hulls representing the clusters grows when the value of quality index $I_q$ decreases.

We notice a knee in the graph presenting the ranked $I_q$ when moving from cluster #10 to #12 in Fig. 1(a). Also, the clustering validity index $I_R$ has a local minimum for $L = 13$, see Fig. 3(b). Convex hulls marked A and B show how clusters are merged if $L = 13$ is selected instead. The estimated source signals for centrotypes associated to the most robust clusters #1–11 (being outside of convex hulls A and B), are presented in quality rank order in Fig. 2. From the previous studies, we know that source estimates #1 and #2 correspond to eye movements, #3 to heart and #7 to the digital watch. Sources #5 and #6 are related to muscular activities due to biting. As a result, known, strong artifacts are all ranked to the top which is quite reasonable. In repeated experiments, all top 4 estimates were always ranked 1–4 with the first and the second only occasionally changing places. The next seven estimates remained usually in top 11, except that estimates #5–6 related to biting became less reliably estimated, especially in setting II. Consequently, source #4 is extremely interesting since it is clearly well estimated—even in repeated experiments and in other settings—but the physiological explanation, if any, is not yet known. Explanations for source estimates #8–11 are not known to the authors, either.

Fig. 3(a) shows trustworthiness of CCA, MMDS, and Sammon's projection averaged for setting I. The closer the value of the index is to one, the more trustworthy the projection is for that neighborhood size. It can be seen that CCA outperforms the other methods for our benchmarking test on small to modest neighborhoods that are of special interest according to [17]. In settings II and III, the ranking of the projection methods remains the same.

Fig. 3(b) shows an example of disagreement between the clustering indices in the test run. Index $I_R$ suggests 9 clusters to be the "best" level, yet another local minimum is seen in 13 clusters, supported by the vote of Dunn-like index $\nu_{31}$ for 12 clusters. On the other hand, index $\nu_{32}$ suggest $L = 21$ which is close to the original data dimension. In repeated experiments, including other settings, $\nu_{32}$ typically favors selecting close to $L = k$ while $I_R$ and $\nu_{31}$ favor fewer clusters—but there is no general tendency of $I_R$ favoring the smallest number of clusters, as in this case. Two other Dunn-like indices $\nu_{11}$ (the original Dunn's index) and $\nu_{12}$ were not applicable at all: they suggested always 2 clusters in every experiment.




(a) (b)

Figure 1: The cluster ranking and the similarity graph for $L = 20$. Panel (a) shows the quality index $I_q$ in rank order. Panel (b) shows similarity graph of the estimates. Clusters are indicated by convex hulls. Labels 1–20 correspond to panel (b). Convex hulls A and B show how clusters agglomerate further if $L = 13$ is set instead of $L = 20$. For practical reasons, the tool uses certain heuristics for suppressing lines. For example, if $\sigma_{ij}$ is smaller than a specified threshold (here 0.1) the line is not drawn.

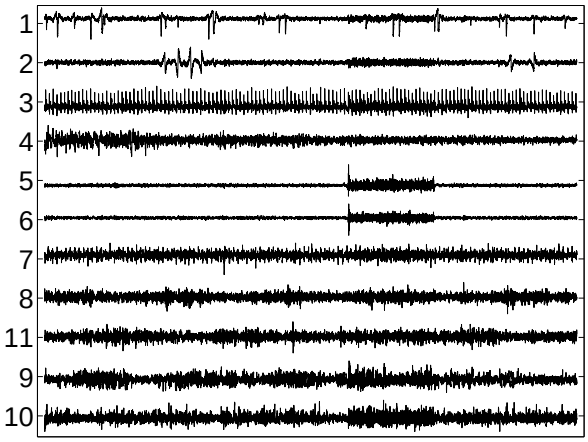
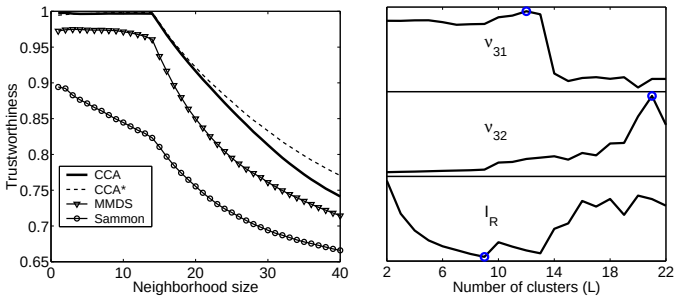


Figure 2: Estimated sources corresponding to centrotypes of clusters #1–11 in Fig. 1.

## CONCLUSIONS

We have developed an interactive visualization method and software package for analyzing the reliability (robustness or significance) of estimated independent components. The basis of the method is running an ICA algorithm many times, and looking at the clustering of the estimated components in the signal space. Basically, each tight cluster corresponds to a component that can be considered reliable. Reliability has two aspects, algorithmic and statistical, which can be probed by running the algorithm with different initial




(a) (b)

Figure 3: Projection and clustering validity indices. Panel (a) presents the trustworthiness of CCA, Sammon's projection and MMDS presented as a function of neighborhood sizes 1–40. CCA* is obtained when Eq. 2 was used, the rest of the graphs result from using Eq. 4. Panel (b) shows three different clustering validity indices ($\nu_{31}$, $\nu_{32}$ and $I_R$) computed for $2 \ldots 22$ clusters induced by the hierarchical clustering. Circle shows the "best" partition according to the index. Note that the best clustering is obtained from the minimum $I_R$ and maximum $\nu_{31}$ and $\nu_{32}$, respectively.

values or bootstrap samples, respectively.

Finding clusters in the high-dimensional signal space involves fixing the number of clusters to be modeled, as well as the values of other internal parameters. Automatic determination of optimal values for these parameters is a most difficult theoretical problem; ultimately, the optimal values also depend on application-specific and subjective considerations. Therefore, we propose an interactive method based on visualization of the clustering structure.

The methods developed here could be applied to investigate the reliability of many other algorithms as well, such as algorithms for estimating cluster centerpoints (in the original data).

## REFERENCES


[1] S. Bandyopadhay and U. Maulik, "Nonparametric Genetic Clustering: Comparison of Validity Indices," **IEEE Transactions on Systems, Man and Cybernetics: Part C: Applications and Reviews**, vol. 31, no. 1, pp. 120–125, Feb 2001.

[2] J. Bezdek and N. Pal, "Some New Indexes of Cluster Validity," **IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics**, vol. 28, pp. 301–315, Jun 1998.

[3] P. Demartines and J. Hérault, "Curvilinear Component Analysis: A Self-Organizing Neural Network for Nonlinear Mapping of Data Sets," **IEEE Trans. on Neural Networks**, vol. 8, no. 1, pp. 148–154, January 1997.

[4] B. Everitt, **Cluster Analysis**, Arnold, 3rd edn., 1993.

[5] A. Gordon, "A Review of Hierarchical Classification," **Journal of the Royal Statistical Society. Series A (General)**, vol. 150, no. 2, pp. 119–137, 1987.

[6] A. Hyvärinen, P. O. Hoyer and M. Inki, "Topographic Independent Component Analysis," **Neural Computation**, vol. 13, no. 7, pp. 1527–1558, 2001.
[7] A. Hyvärinen, J. Karhunen and E. Oja, **Independent Component Analysis**, Wiley Interscience, 2001.
[8] E. Levine and E. Domany, "Resampling Method For Unsupervised Estimation of Cluster Validity," **Neural Computation**, vol. 13, no. 11, pp. 2573–2593, 2001.
[9] W. Liebermeister, "Linear modes of gene expression determined by independent component analysis," **Bioinformatics**, vol. 18, pp. 51–60, 2002.
[10] S. Makeig, T.-P. Jung, A. J. Bell, D. Ghahramani and T. Sejnowski, "Blind separation of auditory event-related brain responses into independent components," **Proc. National Academy of Sciences (USA)**, vol. 94, pp. 10979–10984, 1997.
[11] U. Maulik and S. Bandyopadhay, "Performance Evaluation of Some Clustering Algorithms and Validity Indices," **IEEE Transactions on Pattern Analysis and Machine Intelligence**, vol. 24, no. 12, pp. 1650–1654, Dec 2002.
[12] M. McKeown, S. Makeig, S. Brown, T.-P. Jung, S. Kindermann, A. Bell, V. Iragui and T. Sejnowski, "Blind separation of functional magnetic resonance imaging (fMRI) data," **Human Brain Mapping**, vol. 6, no. 5-6, pp. 368–372, 1998.
[13] F. Meinecke, A. Ziehe, M. Kawanabe and K.-R. Müller, "Estimating the Reliability of ICA Projections," in **Advances in Neural Information Processing Systems 14**, MIT Press, 2002.
[14] J. Nikkilä, P. Törönen, S. Kaski, J. Venna, E. Castrén and G. Wong, "Analysis and visualization of gene expression data using Self-Organizing Maps," **Neural Networks**, vol. 15, pp. 953–966, 2002.
[15] J. W. Sammon, Jr., "A Nonlinear Mapping for Data Structure Analysis," **IEEE Trans. Comp. C**, vol. 18, no. 5, pp. 401–409, 1969.
[16] W. Torgerson, "Multidimensional Scaling I—Theory and Methods," **Psychometrica**, vol. 17, pp. 401–419, 1952.
[17] J. Venna and S. Kaski, "Neighborhood preservation in nonlinear projection methods: An experimental study," in **Artificial Neural Networks (ICANN 2001)**, Springer, 2001, pp. 485–491.
[18] J. Vesanto, J. Himberg, E. Alhoniemi and J. Parhankangas, "SOM Toolbox for Matlab 5," Report A57, **Helsinki University of Technology, Neural Networks Research Centre**, Espoo, Finland, April 2000.
[19] R. Vigário, V. Jousmäki, M. Hämäläinen, R. Hari and E. Oja, "Independent Component Analysis for Identification of Artifacts in Magnetoencephalographic Recordings," in **Advances in Neural Information Processing Systems**, MIT Press, 1998, vol. 10, pp. 229–235.
[20] R. Vigário, J. Särelä, V. Jousmäki, M. Hämäläinen and E. Oja, "Independent Component Approach to the Analysis of EEG and MEG Recordings," **IEEE Trans. Biomedical Engineering**, vol. 47, no. 5, pp. 589–593, 2000.
[21] C. Ware, **Information Visualization: Perception for Design**, Morgan Kaufmann Publishers, 2000.