Real-Time Imaging **6**, 69–76 (2000)
doi:10.1006/rtim.1998.0160, available online at http://www.idealibrary.com on IDEAL®

Real-time Visualization of Brain Electrical Activity

New visualization methods can improve understanding of complex scientific phenomena, such as the brain electrical activity. Real-time execution facilitates assessment of genuine dynamics of brain operation. This paper presents methods for visualization of the brain electrical activity, and real-time implications of the standard computer implementation. The authors present essential trade-off and lessons learned during the realization of two different EEG visualization environments. Real-time visualization provides assessment of genuine brain dynamics, and perception of inherent spatio-temporal patterns of brain electrical activity.

Aleksandar B. Samardzic[1], **Emil S. Jovanov**[2] **and Dusan B. Starcevic**[3]

[1]*School of Electrical Engineering, University of Belgrade, Belgrade, Yugoslavia*
[2]*Electrical and Computer Engineering, The University of Alabama in Huntsville, AL 35899, USA*
[3]*Faculty of Organizational Sciences, University of Belgrade, Belgrade, Yugoslavia*

Introduction

Scientific visualization, strengthened by the tremendous increase in the computer performance, makes possible efficient presentation of vast amount of data. Development of graphic subsystems, as well as system support for multimedia applications expanded its use to biomedical applications. Innate multidimensionality of biomedical data naturally requires the most sophisticated modes of presentation.

Although real-time visualization of brain electrical activity has existed for nearly half a century [1], early visualization systems required extremely expensive equipment and timely application supports. Fortunately, the present microcomputer revolution provides supercomputer performance on commercially available platforms. Moreover, multimedia systems contributed further hardware acceleration, operation system support, and quality of presentation.

Brain operation is one of the most important challenges of the contemporary science. Brain electrical activity represents an available, non-invasive method of assessing brain operation. Using visualization of the brain's electromagnetic activities could facilitate possible insight into brain functions. The most important means of tracing the brain's electro-magnetic activity is to record scalp electric potentials known as electroencephalography (EEG), or induced magnetic fields in the vicinity of the head, called magnetoencephalography (MEG). Both methods provide a very good temporal resolution in the order of milliseconds, and therefore dynamic characterization of brain operations. Here we survey real-time issues in the visualization of brain electrical activity, and present our experience gained from the realization of two different 3D visualization environments.

Corresponding author: E.S. Jovanov. E-mail: jovanov@ece.uah.edu

1077-2014/00/020069 + 08 $35.00/0

Visualization of Brain Electrical Activity

EEG has been routinely used as a diagnostic tool since it was introduced by Hans Berger in 1929 [1–3]. Since head inhomogeneities (caused by liquor, skull, and skin) considerably influence EEG recordings, magnetoecephalography (MEG) in recent years has been used in addition to complete the picture of underlying processes [4]. The main reason is that the head is almost transparent to magnetic fields.

A traditional EEG plot relieves only temporal changes. Therefore, a topographic mapping had been introduced to represent the spatial distribution of activity [1,5]. A single topographic map represents the global view of spatial distribution of brain activity. Contrary to CT and PET images, where all data points represent real data, EEG topographic maps contain real data values only on electrode positions. Consequently, all other points must be spatially interpolated using known values calculated on electrode positions. Therefore, higher numbers of electrodes provides for more reliable topographic mapping.

The most important problem of planar (two-dimensional) topographic maps is the projection from the scalp onto a plane. Various projections had been used, but every projections leads to deformations of certain parts of the image. This can be resolved using topographic maps created on a 3D human head model. Then, the choice of observation viewpoint provides realistic perception of electrode positions, inter-electrode distances and the spatial distribution of brain electrical activity.

In addition to the most frequently used standard spectral scores, the system can visualize different custom scores. We have found promising implementation of a band power envelope and a strange attractor dimension mapping [6].

Brain topography is gradually becoming a clinical tool. It is mostly used in conjunction with the standard polygraph EEG. The same visualization environment provides support for visualization of both EEG and MEG scores.

Real-time issues

A visualization environment must be able to support dynamics of observed phenomena in real-time. It should be limited only by the characteristics of user perception. The most important parameters affecting the real-time visualization performance are:

- Animation rate
- Image size
- Calculation of parameters (EEG scores).

Commercially available computer platforms can create an animation rate in order of 10 frames/s, according to the image size and score calculation complexity [7]. Although the animation rate can go up to 50 frames/s, the actual rate must be matched to the information processing capabilities of a human observer. Otherwise, problems such as temporal summation and visual masking may arise [8]. Both effects occur if the frame rate is too high, when details on adjacent maps interfere creating false percepts. However, the animation rate must be kept constant to preserve the original dynamics of brain electrical activities.

Image sizes introduce complexity related to the complexity of the human head model and the speed of rendering. Although it is possible to interpolate a score for every pixel of the topographic map, usually the surface consists of color "patches" with a uniform or linearly interpolated color. The computer model frequently uses triangles, and calculates a single color for the triangle surface, or distinct colors for vertices with the color interpolation. Large numbers of triangles induce a significant calculation time, while small number of surface elements provides a "rough" surface with poor spatial resolution. Naturally, the system has to provide sufficient temporal and spatial resolution of animated sequences, and therefore optimal granularity has to be found for applications under consideration.

Brain electrical activities could be characterized in a parameter space using different signal processing procedures [2,3,9,10]. The score calculation used to be the greatest obstacle on previous computer generations. Fortunately, a great effort has been made to improve scientific computing on standard computers and workstations in the last two decades. Therefore, the calculation of standard scores, such as spectral scores may be performed in a fraction of second, even using the existing CPU.

The visualization application is designed for a multiprocessor Windows NT environment. In order to achieve the best possible real-time performance in Windows NT environment, we organized our application as a single process with multiple execution threads.

The threads are synchronized using critical sections, as the fastest synchronization mechanism in Windows NT environment. In single processor systems the animation cycle is given by:

$$T_{ac} = T_{acq} + T_{sc} + T_{vis} + T_{do}$$

where T_{acq} is the data acquisition time, T_{sc} is the score calculation time, T_{vis} is the visualization time (consisting of model processing and data display times), and T_{do} is the data overhead. The data overhead consists of context switching and data transfer time between threads.

Calculations of scores could be further accelerated using

- Parallel processing on multiple processors,
- Specialized digital signal processing processors (DSPs)

Most signal processing procedures could be effectively parallelized, and executed on multiple processors within the same computer, which is often the case with visualization workstations. Dedicated DSP processors, with architecture optimized for fast signal processing could further significantly accelerate score calculations. Still, some scores could be hardly calculated in real-time [11,12].

Head model

An initial attempt was made to develop our own model of the human head by digitizing a realistic plaster bust [13]. Unfortunately, the acquired precision and problems with 3D digitization of small (but visually important) details were more serious than we expected, and therefore we decided to use existing head model from the *3DStudio* program. Custom routines are developed for conversion from **.dxf** file format exported by *3DStudio* to an internal model with **.mdl** file extension.

The head model consists of a set of triangles. It is split into two objects (surfaces): *head* and *scalp* area. Proposed organization improves program efficiency, because of the fact that head area is changed only when viewing angle is changed, while scalp area is changed in every animation cycle. We devised a dedicated program to perform interactive selection of triangles belonging to the scalp area.

Extended model parameters (for example normal vectors in each model vertex) are precalculated and stored as a part of model description into **.mdl** file. This eliminates the necessity for re-calculation of these parameters on start-up.

Everyday clinical use necessitates simple changes of electrode arrangement. Different arrangements are often used to improve spatial resolution of EEG recordings. Therefore, we supported the generation of custom electrode arrangements as shown in Figure 1. Electrode positions are stored in a separate file (with **.src** extension) and physical channels are mapped on a head model. It is possible to position either a single electrode or a symmetrical pair of electrodes. The default electrode arrangement is set according to the International 10-20 Standard.

System Organization

The first version of the program TEMPO is developed in Visual C++ for Windows 95 in a single processor operating environment. The second version is developed for multi-processor Windows NT real-time systems. The system consists of one process with three parallel execution threads: *data acquisition*, *score calculation* and *score visualization*. They are synchronized by means

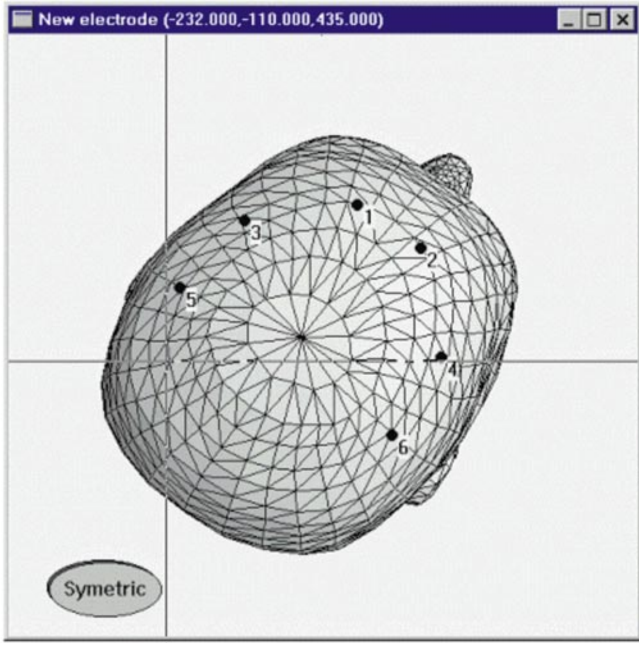

Figure 1. Custom model electrode placement.

of the critical sections, and could be executed in parallel on different processors. Therefore, asynchronous data flow is supported which is of great significance for a parallel distributed environment [14]. Even in a single system, data acquisition is usually supervised by an intelligent controller on an A/D board, score calculation could be performed by add-on DSP processor, while the graphic coprocessor could handle visualization tasks.

EEG data could be retrieved either on-line (from an A/D converter board) or off-line (from file). Class encapsulating data acquisition tasks pass chunks of EEG samples to score calculation independently of the data source. We implemented reading of standard **.eeg** files generated by *RHYTHM 8.0* (Stellate Systems), and now we implement the ASTM EEG file format standard [14].

It was necessary to develop a general-purpose research visualization environment. We developed TEMPO (**T**opographic **E**EG **M**apping **P**rogram) as the general purpose EEG visualization environment [7,13]. The program was developed to test the user interface and the most important perceptual features of visualization:

- animation control and speed
- evaluation of scores
- color mapping (look up tables)
- background properties
- scene lighting
- model evaluation.

Accordingly, we were using standard multimedia controls for animation, while most visualization parameters are available for easy setting. A typical user interface of TEMPO program is given in Figure 2. TEMPO features an improved *scan-line* hidden surface algorithm to accelerate visualization. It relies on the fact that a single convex object is always rendered, while hidden surfaces were removed using a scan-line algorithm [13].

Practical applications greatly depend on interpolation methods. Having in mind that only electrode positions contain exact score values, while all the others are calculated, we provided the means for change of the interpolation method [15]. The most frequently used interpolation method is to calculate scores according to the distance to the nearest electrodes (nearest neighbor algorithm).

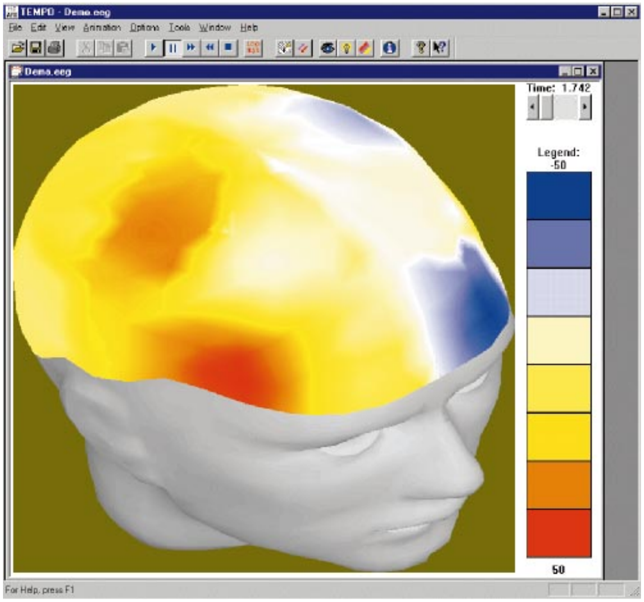

Figure 2. TEMPO visualization environment.

An improved visualization engine for EEG visualization was developed for *Cadwell Laboratories Inc.*, Kennewick, WA for a program BRAINMAP. The system implements different user-interface (Figure 3), and a number of execution optimizations. Further speed improvements are accomplished by remembering a visible triangle for each pixel to be color-coded during image generation, so hidden surfaces removal need not to be repeated until a viewpoint position changes. It that

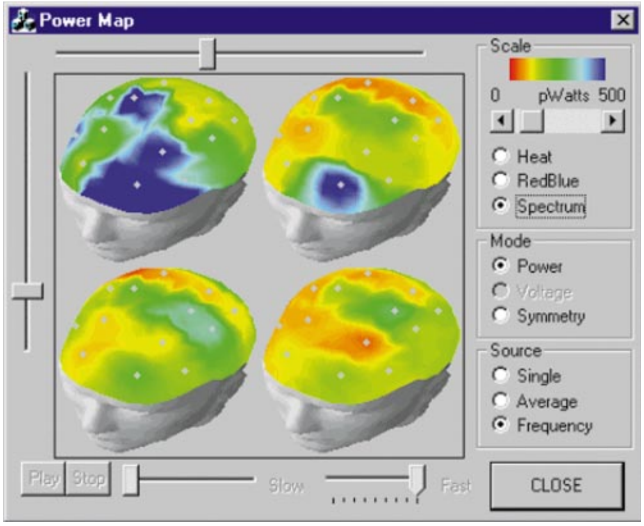

Figure 3. BRAINMAP visualization environment (with permission of Cadwell Laboratories Inc.).

case only values of pixels to be color-coded are recalculated.

Rendering algorithm

The most time-consuming part of every visualization application is rendering. Therefore, the selection of a rendering algorithm is crucial for real-time visualization performance, although the choice of optimal rendering algorithm is application-dependent [16]. Both visualization environments use a common graphics engine: a set of custom routines implementing scan-line hidden surface removal algorithm developed by the authors and fine-tuned to maximize the performance of brain topographic mapping. Although the z-buffer hidden surface algorithm is today commonly used for interactive 3D graphics because of its inner loop simplicity and growing hardware support, it is not appropriate for applications where a significant amount of processing is required to obtain the value of each pixel. Namely, the z-buffer algorithm performs scan-converting of each triangle in a scene. Thus, a lot of calculations pertaining to each pixel (spectral analysis, interpolation, color look-up) is lost, since those pixels are later overwritten by pixels belonging to triangles closer to the viewer. On the other side, a scan-line algorithm calculates pixels that will only appear on the final image, without overhead. The cost for such arrangement is the complexity of the scan-line algorithm inner logic, and therefore the main effort of our work targeted optimization of this logic.

Since the most important feature of EEG topographic map visualization is spatial localization, we decided to have the whole head visible in each frame, while zoom-in and zoom-out could be accomplished through resizing of the animation window. The transformation matrix always adapts itself so that the head occupies the available window space. Thus, we are able to drop clipping from the rendering pipeline, which beneficially influenced rendering speed.

Further speed improvements are obtained through the optimization of the scan-line algorithm for the case when there exist only one closed object in the scene — the head in our case. Our implementation of the scan-line algorithm runs usually until line scan-conversion: in a preprocessing step, object edges are sorted in an edge table (ET), and then for each scan-line, those edges are moved between the edge-table and an active edge table (AET). When a scan line starts, all edges intersected by this scan line are enumerated through the active edge table, where they reside, sorted along an increasing x-coordinate. Edges from the active table are processed in sorted order during line scan conversion. A table called triangle table (PT) is created, where references to triangles currently “below” scanline are maintained. Since the head model is closed, we have to consider exactly two triangles when the new edge in the active edge table is encountered. Three different cases are possible:

- neither triangle is enumerated in the triangle table;
- one of the triangles is enumerated in the triangle table, while the other is not;
- both triangles are enumerated in the triangle table.

In the first case, both triangles are inserted into the triangle table. This case always appears when the scan conversion of the line starts. The active triangle the triangle closest to the viewpoint, should be determined then and the procedure accomplished through comparing plane coefficients of the two triangles. The third case usually appears when scan conversion of the line finishes and both triangles are deleted from the triangle table. The active triangle should be determined between the rest of the triangles from the triangle table, but if this appears when the scan conversion of the line finishes, there are no more triangles in the triangle table and no additional calculation is necessary.

In the second case, the first triangle should be deleted from the triangle table and the second triangle added there; this is accomplished through simple replacement of the two triangles involved, without any additional overhead. If the deleted triangle was currently an active triangle, an inserted triangle becomes a new active triangle. This case is dominating during line scan conversion and the simple arrangement described above makes maximum utilization possible of the model coherence and significantly improves rendering time due to the fact that triangle penetration does not exist. After determining the active triangle, a span of pixels from the current edge to the next edge in the active edge table is calculated. Consequently, only visible pixels are calculated and there is no overhead compared to the z-buffer algorithm.

Other system issues

One of the most important perceptual features is the transformation of score values to colors, which is performed with the use of color look-up tables. The choice of three transformations with adjustable sensitivity is provided:

- *Red-blue*, where EEG score is represented by blue hues for low, black for mid-range and red for high values;
- *Heat*, where EEG score is represented by red hues for low, through red and yellow for mid-range to white for high values;
- *Spectrum*, where EEG score is represented by rainbow colors, from red for low through yellow and green for mid-range through blue for high values.

Look-up tables define colors of triangle vertices, while the color of each pixel is determined by linear interpolation between vertices of the triangle pixel which it belongs to. It is interesting to note that the color calculation is so fast that screen refreshing becomes the limitation on fast machines. Namely, the transferal of a map into the video-memory is performed by *StretchDIBits()* API procedure. This procedure is known by its fairly slow *blt* (block-transfer), and therefore flicker is discernible on the screen when *StretchDIBits()* does not succeed to write a new map into the video-memory at the beginning of a new refreshing cycle. For this reason, the next version of program will probably use faster *DirectDraw blt* functions.

The performance of two visualization environments is represented in Figure 4. The animation cycle is measured on single processor PC PentiumPro 166MHz system, with 64MB of RAM and 512kB of L2 cache, running on Windows NT operating system. Most of the BRAINMAP acceleration is deducted from the above mentioned memorizing of visible triangles for each pixel to be color-coded. Thus, we can recommend as a rule of thumb this simple optimization technique during any kind of visualization where viewpoint doesnot change too frequently. It might be interesting to note the performance of our FFT algorithm, too. On the same machine, our algorithm calculates FFT for 16 channel EEG with 512 samples epoch every 0.027s.

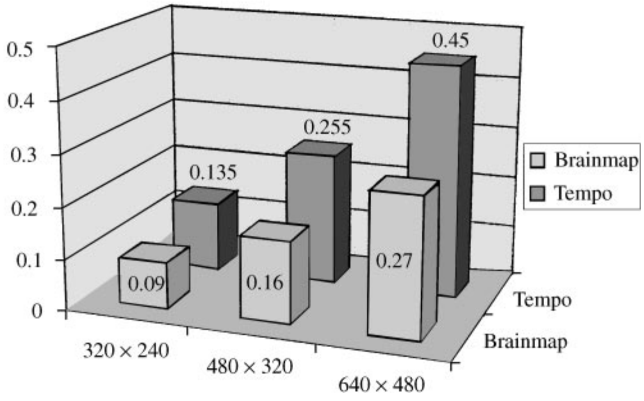

Figure 4. Comparison of TEMPO and BRAINMAP animation cycles according to the image resolution; animation cycle is given in seconds, and image size in pixels.

Discussion

Real-time visualization of brain electrical activity relies on technological and perceptual issues. While technological issues with the present trend of computer performance improvement lose their significance, perceptual features must be carefully observed. For instance, even if we were able to visualize recorded potential in real-time (with few hundred Hz sampling rate), it would have been far from acceptable for a human observer. However, the correct time scaling still preserves relative spatio-temporal patterns of brain electrical activity as presented in Figure 5.

The second important problem is the presence of artifacts in the EEG signal. Our system provides the possibility to manually review digitized polygraph EEG recording for artifacts, drowsiness, normal variance, and other abnormal features of clinical significance. Only then visualization of selected periods (epochs) may have clinical and research significance.

Parallel processing could significantly increase the real-time performance. When applications are executed on multiple processors, the animation cycle would be:

$$T_{ac} = \max(T_{acq}, T_{sc}, T_{vis}) + T'_{do}.$$

Within the same workstation or hospital environment with fast LAN communication T'_{do} is negligible. For telemedical applications based on WAN, significant processes communication time emphasizes load balancing issues. We have been experimenting with Internet-based telemedical application written in Java with three layers:

- acquisition
- web hosting with score calculation
- visualization based on VRML supported browser.

This architecture provides both real-time monitoring and data storing on a Web host, with the choice of other

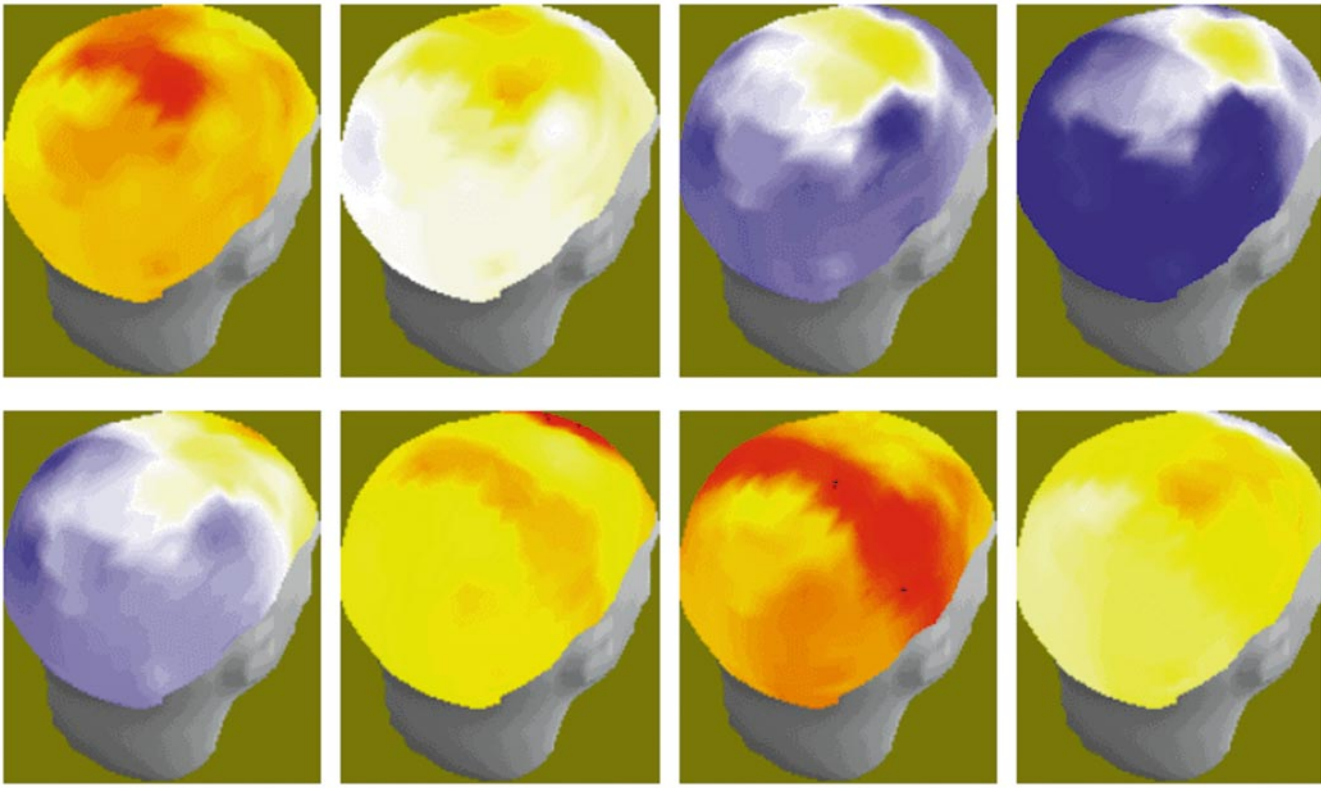

Figure 5. An example of epileptic activity visualization; absolute EEG power; Sequence time $t = 8.961$–9.016 s (from upper left to bottom right).

parameters visualized later on. However, one has to bear in mind the communication cost of distributed computing. Transferring of large chunks of EEG data to a specialized processor may diminish the faster processing pay-off. Therefore, processors have to reside close to the information sources. For instance, we should take advantage of DSP based A/D converter board in a data acquisition system. For most scores it will significantly reduce traffic on a system bus. In a distributed environment, the best performance is achieved when the first two layers are located on the same machine, and only scores are transferred through the network. Consequently, reduced data communication requirements facilitate the real-time implementation in distributed environment.

Conclusion

Visualization of brain electrical represents a complex multidisciplinary problem and is of significance both for research and everyday clinical practice. This paper presents an effort to highlight its complexity and describes engineering problems for the real-time visualization. As with many other engineering disciplines, when carefully chosen, the application design is required to make the best use of the technology available. Although standards for EEG visualization are still not defined, we suggest experimenting with different scores and methods to encounter perceptually appropriate visualization for a given application. Then we will be able to extract the relevant diagnostic information features.

Our system provides methods for assessing genuine dynamics of brain electrical activity, and perception of inherent spatio-temporal patterns of brain electrical activity. Further work will investigate possible integration of different imaging modalities, such as MRI and PET, into a single visualization environment. Another possible option would be the representation of signal sources (dipoles) on the volume head model [17,18].

Acknowledgement

We would like to thank Vlada Radivojevic M.D., from the Institute for Mental Health for medical support and insightful comments during realization of the visualization environment.

References

1. Nuwer, M.R. (1990) The Development of EEG Brain Mapping. *Journal of Clinical Neurophysiology* **7**(4): 459–471
2. Rémond, A. (1972) *Handbook of EEG and Clinical Neurophysiology*. Amsterdam, Holland: Elsevier.
3. Nunez, P.L. (1995) *Neocortical Dynamics and Human EEG Rhythms*. Oxford: Oxford University Press.
4. Kikins, R., *et al.* (1996) *A Digital Brain Atlas for Surgical Planning, Model-Driven Segmentation, and Teaching. IEEE Transactions on Visualization and Computer Graphics* **2**(3): 232–241
5. Duffy, F.H. (1986) *Topographic Mapping of Brain Electrical Activity*. Boston: Butterworths
6. Jovanov, E., Raković, D., Radivojević, V. & Kušić, D. (1995) *Band Power Envelope Analysis — A New Method in Quantitative EEG*. Seventeenth Annual Int'l Conf. IEEE Engineering in Medicine and Biology, Montreal, Canada, pp. 53–54
7. Samardzic, A., Jovanov, E., & Starcevic, D. (1996) *3D Visualisation of Brain Electrical Activity*. Proceedings of 18th Annual Int'l Conf. IEEE/EMBS, Amsterdam, October 1996.
8. Klymenko, V. & Coggins, J.M. (1990) *Visual Information Processing of Computed Topographic Electrical Activity Brain Maps. Journal of Clinical Neurophysiology* **7**(4): 484–497
9. Haykin, S., Racine, R.J., Xu, Y. & Chapman, C.A. (1996) Monitoring Neuronal Oscillations and Signal Transmission Between Cortical Regions Using Time-Frequency Analysis of Electroencephalographic Activity". *Proceedings of the IEEE* **84** (9): 1295–1301.
10. Williams, W.J. (1996) Reduced Interference Distributions: Biological Applications and Interpreteations. *Proceedings of the IEEE* **84**(9): 1265–1280.
11. Pradhan, N. & Dutt, D.N. (1993) Use of running fractal dimension for the analysis of changing patterns in electroencephalogram. *Computing in Biology and Medicine* **23**(5): 381–388.
12. Pradhan, N. & Dutt, D.N. (1993) A nonlinear perspective in understanding the neurodynamics of EEG. *Computing Biology and Medicine* **23**(6): 425–442.
13. Samardzic, A. (1996) *The Visualisation of Brain Electrical Activity*. Master Thesis, School of Electrical Engineering, University of Belgrade. Also *http://www.eb.uah.edu/~jovanov/projects/tempo/tempo.htm*
14. *Standard Specification for E1467-94 Transferring Digital Neurophysiological Data Between Independent Computer Systems*. (1997) American Society for Testing and Materials. Can be found at: *http://www.astm.org*
15. Soong, A.C., Lind, J.C., Shaw, G.R. & Koles, Z.J. (1993) Systematic Comparisons of Interpolation Techniques in Topographic Brain Mapping *Electroencephalography and clinical Neurophysiology* **87**: 185–195.
16. Foley, J.D. & van Dam A. (1990) *Computer Graphics — Principles and Practice, 2nd ed.* Reading, MA: Addison-Wesley
17. VanVeen, B.D., van Drongelen, W., Yuchtman, M. & Suzuki, A. (1997) *Localization of Brain Electrical Activity via Linearly Constrained Minimum Variance Spatial Filtering. IEEE Transactions on Biomedical Engineering* **44**(9): 867–880.
18. Williams, J.W., Zaveri, H.P. & Sackellares, J.C. (1995) Time-Frequency Analysis of Electrophysiology Signals in Epilepsy. *IEEE EMBS* **14**(2): 133–143.