JOURNAL OF STRUCTURAL BIOLOGY **116,** 30–34 (1996)
ARTICLE NO. 0006

The EM Program Package: A Platform for Image Processing in Biological Electron Microscopy

REINER HEGERL

Max-Planck-Institut für Biochemie, D-82152 Martinsried, Germany

Received May 13, 1995, and in revised form June 20, 1995

EM is a software package developed for image processing in the field of biological electron microscopy. After a brief summary of its history, an outline of the package is given, by describing some characteristic features rather than by a complete report of all facilities. New developments, connected with the introduction of a UNIX version and the advent of automated electron microscopy, are described in more detail. © 1996 Academic Press, Inc.

INTRODUCTION

The very first version of the EM[1] program package was created in the late sixties (Bussler, 1974). Generally, it was the first time people engaged in biological electron microscopy became interested in digital image processing. New approaches, like the 3-D reconstruction of macromolecules or the correction of image distortions due to lens aberrations, could only be realized with the help of computers. Within a group at the former Max-Planck-Institut für Eiweiß- und Lederforschung in Munich, headed by W. Hoppe, some students decided to develop tools for these new tasks (Hoppe *et al.*, 1973, 1976), not only for themselves but also for others not as specialized in computer processing. The resulting program package was named EM simply to indicate its field of application, namely, electron microscopy. The EM software has continued to develop and has been used at the succeeding Max-Planck-Institut für Biochemie (Hegerl and Altbauer, 1982) as well as by other groups working in this field (e.g., Olofsson *et al.*, 1990).

In the development of this computer software, three periods can be recognized. The first period starting about 1970 is characterized by "large" mainframe computers (SIEMENS 4004, IBM 360) operated by a highly specialized computer staff. Users of EM had to submit batch jobs (initially prepared on punched cards) and wait 1 day or more for the result. Pictures were displayed by using plotters or line printers, e.g., by overprinting of the letters. The second period began in 1979, when a VAX780 became available to the group. An entirely new version of EM was created, based on the VMS operating system and using elements of FORTRAN 77. Development and support of the software was facilitated by new system tools (e.g., a debugger) and, for the normal user, interactive applications became possible. Considerable progress came in 1981 due to the installation of the first raster graphic display system (Genisco). New software was added to EM, enabling the display of pictures with various functions. The use of plotters and printers was then relegated to producing documentation. Toward the end of the eighties, EM was installed on new workstations connected by ethernet. This development was accompanied by combining EM with the X-Window software in order to make the image display functions independent of hardware. The third period was begun in 1993 by establishing the first version of EM running on the UNIX operating system. Previously, software coding was done exclusively in FORTRAN, but now the image display routines are written in C, a language more suitable to the UNIX environment. Meanwhile, versions of EM are running on IRIX (SGI), OSF (DEC), and Sun-OS. The present challenge to EM is automatic image recording available with a new generation of electron microscopes. Extremely large data sets—the size of which is still growing—have to be transferred from the microscope to the computer and processed.

Some of this software development is described in more detail in a previous account of EM published in 1982 (Hegerl and Altbauer) which reflects the status of the package at that time. More recently, seven software packages including EM, all dedicated to image processing in electron microscopy, were compared with respect to the various functions and techniques offered by them as well as their installation

[1] Abbreviations used: EM, electron microscopy; MB, megabytes; GB, gigabytes; RAM, random access memory; FTP, file transfer protocol; CCD, charged coupled device; SGI, Silicon Graphics; DEC, Digital Equipment Corporation; 3-D, three-dimensional.

1047-8477/96 $18.00

and data handling capabilities (Hegerl, 1992). The reader is also referred to this 1992 paper for a basic information about the capabilities of EM. In the following program description, only some characteristic features of EM's general design are presented, concentrating more on developments connected with the new UNIX version.

SOME CHARACTERISTIC PROPERTIES OF EM

Today the package is designed to be used mainly on graphics workstations with modest hardware requirements. A memory (RAM) of at least 32 MB is recommended, together with a raster graphic color display, typically with a resolution of 1280 by 1024 pixels and a depth of 8 bit planes. Depending on the application, sufficient disk space should be available, e.g., 2 GB. This equipment is representative of modern workstations at the lower end of the price scale.

Application and Syntax

To the user, EM appears as a command interpreter. The execution of a command can be controlled by arguments or subcommands (in brackets) which again can be followed by arguments. There are four types of arguments: Symbolic addresses identifying pictures in the RAM, numbers, strings, and special characters such as "!" controlling output messages. The commands can be entered interactively or they can be typed in a command input file which is submitted as a batch job. Frequently used sequences of commands can be combined and defined by the user as a new command called a procedure. Symbolic variables make it possible to write very complex and flexible procedures. Procedures are stored in libraries and recalled by their names together with arguments filling the variables with actual values.

Data Handling in RAM

EM allocates memory as a working area where images are stored and processed. The area is subdivided into blocks with a size of 4224×4 bytes (Hegerl and Altbauer, 1982). The user can work with a default setup of symbolic addresses assigned to these blocks (letters from A to N) or, if necessary, he can define his own setup, thus adjusting available memory to the picture size. Pictures are utilized by such letter addresses and manipulated by commands referring to these addresses. Usually pictures are kept whole in memory, facilitating fast processing. In order to store results permanently, they have to be written from RAM to disk by a simple WRITE command. The total size of the working area depends on an actual installation parameter. It can be adjusted to available hardware resources by compiling and linking the package. In the case of a large data set that overflows the available memory, special facilities as described below can be used.

Storage of Data on Disk

Inside EM, data files on disk are identified by a name and a number. This makes it possible that a series of data files belonging together, e.g., all projections of a tilt series, can be handled successively by a simple incrementation of the file number. The corresponding name of the file on the operating system is obtained by concatenation of the name and the number.

Image Display

Pictures can be displayed as gray scale images or in false colors using an intensity range from 0 to 127. Only 7 planes from an 8-bit plane system are used for image display. One plane is reserved for overlay graphics which enables the user to mark positions, areas, etc. Image and overlay graphics can be handled independently of each other. There are various functions for the manipulation of contrast and color. In the case of a stack of pictures, the user can inspect the data by displaying them sequentially, as in a movie.

Handling of Three-Dimensional Data

Starting with the second version developed on the VMS operating system, EM was designed for the handling of data sets describing 3-D structures. Projections and rotations can be done in real space, entering three Eulerian angles for the latter. This is very useful for all types of 3-D reconstructions or modeling. In addition, operations not directly involved in 3-D reconstruction can also be done three-dimensionally, e.g., real space filtering. The result of filtering a 3-D structure with a 3-D filter kernel, e.g., with a median filter, is different from that obtained by successive filtering of two-dimensional layers with a corresponding 2-D filter.

NEW FACILITIES IN GENERAL DATA HANDLING

Handling of Large Data Sets

As mentioned above, pictures are normally handled as a whole in memory. But the first programmers of EM had already foreseen the problems associated with large data sets. Therefore, they created a special technique of memory management in which only parts of an image are kept in memory and processed. Today, this option is rarely used because it is very slow. Generally, the problem of large data sets has disappeared because memory has become much cheaper. Modern workstations can be equipped with 128 MB of RAM for an acceptable price. Large data sets may arise in electron crystallography where the crystal patches should be as large as possible. A picture size of 4096×4096 pixels is typical for a more advanced analysis, e.g., by correlation averaging. In single particle averaging, the size of an image showing one molecule may be small, but the number of these small images has been increased up to several thousands (Koster *et al.*, 1994). With EM, a new version of the program system can be created within a few minutes which is able to process images with a size of 4096 × 4096 pixels completely in memory. New facilities were developed in order to handle many small pictures, not separately, but as a stack, for the purpose of single particle averaging. The introduction of this concept was considerably facilitated by the built-in possibility of processing 3-D data.

Data Format and Data Transfer

Since the first version on VMS, a typical record-oriented FORTRAN file format was used as a standard format. It is obtained by an "unformatted write" with variable record length. The first record is a header of 256 bytes in length, containing data type and dimensions, a descriptive comment of 80 characters, and 40 floating point values prepared to store optional imaging parameters, e.g., magnification, defocus, or tilt angles. The image data or, in the case of a three-dimensional file, the data in each layer are handled as one logical record. However, this format has become unsuitable when data have to be transferred between different operating systems. This problem first occurred when a Philips CM20FEG equipped with a CCD camera for automatic image recording was installed (Dierksen *et al.*, 1992). Data were then stored first on the computer controlling the microscope on the basis of the OS9 operating system and had to be transferred to VMS or to UNIX via FTP. Therefore, a new file format was introduced, again "unformatted," but now based on records with a fixed length of 512 bytes. The header was extended to 512 bytes, followed by the data as a steady stream. This format can easily be realized using the FORTRAN option "fixed block" on VMS or OSF and "stream" in IRIX. On a Sun-OS, a special subroutine in C must be written because no equivalent option in the FORTRAN compiler is available. Now files transferred from any of these systems can be read directly into EM. Since the type of operating system used to create an EM file is stored in the header, byte reversal and bit shifts are performed automatically if needed.

Image Representation and Documentation

It is an important function of any image processing system that all graphics displayed on the screen can be stored and documented by printers or similar devices. In its present environment, EM enables the user to save the content of the EM display window as an RGB file (SGI). Such files can be printed directly by modern printers or can be further processed by standard tools available on the operating system.

No efforts have been undertaken to develop tools for volume rendering and the corresponding display of three-dimensional structures as a perspective view. Most applications of EM occur in an environment in which adequate program packages are available and can be shared between groups. The new file format described above is well suited to export data from EM into these packages.

NEW TECHNIQUES

Electron Tomography

The advent of electron microscopes controlled by microprocessors and equipped with CCD cameras for automatic image recording has triggered significant new software development (Dierksen *et al.*, 1992, 1993). Apart from the problem of convenient data transfer mentioned above, a further problem arises from the increasing amount of data. In electron tomography, tilt series are now recorded which consist of up to 60 projection views each represented by 1024 × 1024 pixels with a size of 2 bytes, 120 MB in total. Though, in principle, all operations needed for a complete tomographic reconstruction are already available, new procedures must be developed in order to adapt the whole approach to the amount of data and the new techniques of image recording. The most critical step is the alignment of the projection views to one another, which can be done either automatically or with the option of visual control. It is important to avoid the autocorrelation of substructures superimposed on all projections by the CCD camera. The result, a pair of shift parameters for each projection, is stored in a tiny data file. A subsequent 3-D reconstruction, normally based on filtered back-projection (e.g., Hoppe and Hegerl, 1982), can then be performed within minutes using only the original projection data and the file containing the shift parameters. No intermediate results must be stored. At present, our efforts are concentrated on the development of refinement procedures in order to improve the alignment of projections with respect to translation and orientation.

Random Conical Tilt

The random conical tilt approach (Radermacher, 1988) was already incorporated into EM under VMS (Hegerl *et al.*, 1991). In the context of new microscopes, new techniques for image recording became possible, e.g., the spot scan illumination in conjunction with a beam shift along the tilt axis (Phipps *et al.*, 1993). Many more particles become accessible to processing and, as a consequence, new procedures must be developed for the mutual alignment of the spot scan images. As with tomographic reconstruction, new refinement procedures are being designed.

Focus Series

Automatic electron microscopy has also renewed interest in the old idea of reconstructing the complex object function (phase and amplitude contrast) from a series of exposures taken from the same specimen area but with adequately varying defocus (Schiske, 1968). Again, old tools were available in EM and could be used for the determination of defocus parameters and for the calculation of the complex object function. We have found that the alignment must be refined by correcting for a change in magnification between the pictures of a focus series (Typke *et al.*, 1992).

Image Classification

Image classification was introduced in 1981 to biological electron microscopy (van Heel and Frank) and has become an indispensable standard tool. In EM, classification was performed for several years by singular value decomposition of the covariance matrix in conjunction with a cluster analysis based on a hierarchical descendant application of the k-means approach (Therrien, 1989). For the calculation of eigenvalues and eigenfactors, standard libraries commercially available were initially used that had to be linked to EM. However, incompatible upgrades and licensing problems made it necessary to develop new in-house routines, now based on van Mises' iteration approach in connection with Hotelling's matrix deflation (e.g., Zurmühl, 1961). The advantage is that, instead of calculating a covariance matrix, the pictures can be entered simply as a stack of arbitrary depth and the eigenimages are obtained directly as a stack, too.

As an alternative, a new concept of image classification was recently introduced to EM, following the proposals made by Marabini and Carazo (1994). It is based on Kohonen's neural net approach (Kohonen, 1984), which allows a fully automated implementation. The user no longer has to think about the number of eigenimages and classes to be calculated. Since the parameters controlling the self-organization of the neural net have to be optimized, these routines are still in an experimental state.

OUTLOOK AND CONCLUSION

Most scientists are fascinated by the rapid development of computers and their almost incredible increase in the speed of computations. Years ago programmers were proud to find special tricks which made their software faster. The subsequent improvement of hardware rendered good program readability and the organization of software far more important than programming tricks. Parallel processing is a promising concept for further increases in speed and performance. Though array processors were available several years ago, they were rarely used for program packages like EM. Most programmers were afraid of the potential for reaching a dead end resulting from special requirements in programming which would prevent portability of their software. Meanwhile, we can hope for modern workstations equipped with facilities for parallel processing. In cooperation with intelligent compilers, further improvement should be possible without essential changes in software coding. It is important that these changes are not specific for the hardware or the operating system. New concepts in CPU design, e.g., the R8000 processor available with a Power Indigo 2, may be seen as a step in this direction. Recent tests with EM using such a machine have yielded a speedup for some operations, e.g., image classification, by a factor of 2.4 when compared to the presently fastest Indy (R4400-CPU, 175 MHz). This result was obtained simply by recompilation and relinking the software used so far on standard workstations.

One might also mention algorithms which have previously been considered as unfeasible because of their extreme computing time demands. One example is the three-dimensional averaging of many particle structures obtained independently by electron tomography. Earlier approaches using EM have been reported (Knauer *et al.*, 1983); however, they required weeks of computing time alternating with visual supervision. Automatic scanning of different orientations based on simple and easy-to-understand principles within a reasonable time are now possible and can be applied as standard tools.

Over the years, EM reflects not only the general development of computer technique as far as available at the MPI für Biochemie but also the scientific activities of the research groups working with this package. Originally created for some special tasks, EM has become a powerful instrument, offering plenty of operations needed in image processing. Finally, it should be noted that the software package is freely distributed to everybody engaged in noncommercial research. Unfortunately, only very restricted support can be given to outside users, primarily because of time and personnel limitations.

I am very grateful to Dr. George Harauz for critical reading of the manuscript.

REFERENCES

Bussler, P. (1974) Thesis, Techn. Univ. München.

Dierksen, K., Typke, D., Hegerl, R., Koster, A. J., and Baumeister, W. (1992) Towards automatic electron tomography, *Ultramicroscopy* **40,** 71–87.

Dierksen, K., Typke, D., Hegerl, R., and Baumeister, W. (1993) Towards automatic electron tomography: II. Implementation of autofocus and low-dose procedures, *Ultramicroscopy* **49,** 109–120.

Hegerl, R. (1992) A brief survey of software packages for image processing in biological electron microscopy, *Ultramicroscopy* **46,** 417–423.

Hegerl, R., and Altbauer, A. (1982) The "EM" program system, *Ultramicroscopy* **9,** 109–116.

Hegerl, R., Pfeifer, G., Pühler, G., Dahlmann, B., and Baumeister, W. (1991) The three-dimensional structure of proteasomes from Thermoplasma acidophilum as determined by electron microscopy using random conical tilting, *FEBS Lett.* **283,** 117–121.

Hoppe, W., Bussler, P., Feltynowski, A., Hunsmann, N., and Hirt, A. (1973) Some experience with computerized image reconstruction methods, *in* Hawkes, P. W. (Ed.), Image Processing and Computer-Aided Design in Electron Optics, pp. 92–126, Academic Press, New York.

Hoppe, W., Schramm, H. J., Sturm, M., Hunsmann, N., and Gaßmann, J. (1976) Three-dimensional electron microscopy of individual biological objects, Part I, Methods, *Z. Naturforsch.* **31a,** 645–655.

Hoppe, W., and Hegerl, R. (1980) Three-dimensional structure determination by electron microscopy (nonperiodic specimen), *in* Hawkes, P. W. (Ed.), Topics in Current Physics: Computer Processing of Electron Microscope Images, pp. 127–185, Springer-Verlag, Berlin.

Knauer, V., Hegerl, R., and Hoppe, W. (1983) Three-dimensional reconstruction and averaging of 30S ribosomal subunits of *Escherichia coli* from electron micrographs, *J. Mol. Biol.* **163,** 409–430.

Kohonen, T. (1984) Self-Organization and Associative Memory, Series in Information Sciences, Vol. 8, Springer-Verlag, Heidelberg.

Koster, A. J., Walz, J., Kleinz, J., Typke, D., and Baumeister, W. (1994) CCD Spot-Scan Imaging of Biological Macromolecules, Proc. 13th ICEM, Paris, pp. 423–424.

Marabini, R., and Carazo, J. M. (1994) Pattern Recognition and Classification of Images of Biological Macromolecules Using Neural Networks, Proc. 13th ICEM, Paris, Vol. 1, pp. 501–502.

Olofsson, A., Kavéus, U., Hacksell, I., Thelestam, M., and Hebert, H. (1990) Crystalline layers and three-dimensional structure of *Staphylococcus aureus* α-toxin, *J. Mol. Biol.* **214,** 299–306.

Phipps, B. M., Typke, D., Hegerl, R., Volker, S., Hoffmann, A., Stetter, K. O., and Baumeister, W. (1993) Structure of a molecular chaperone from a thermophilic archaebacterium, *Nature* **361,** 475–477.

Radermacher, M. (1988) Three-dimensional reconstruction of single particles from random and nonrandom tilt series, *J. Electron Microsc. Tech.* **9,** 359–394.

Schiske, P. (1968) Zur Frage der Bildrekonstruktion durch Fokusreihen, Proc. 4th EUREM, Rome, pp. 145–146.

Therrien, C. W. (1989) Decision Estimation and Classification, Wiley, New York.

Typke, D., Hegerl, R., and Kleinz, J. (1992) Image restoration for biological objects using external TEM control and electronic image recording, *Ultramicroscopy* **46,** 157–173.

van Heel, M., and Frank, J. (1981) Use of multivariate statistics in analysing the images of biological macromolecules, *Ultramicroscopy* **6,** 187–194.

Zurmühl, R. (1961) Matrizen, 3rd ed., Springer-Verlag, Berlin.