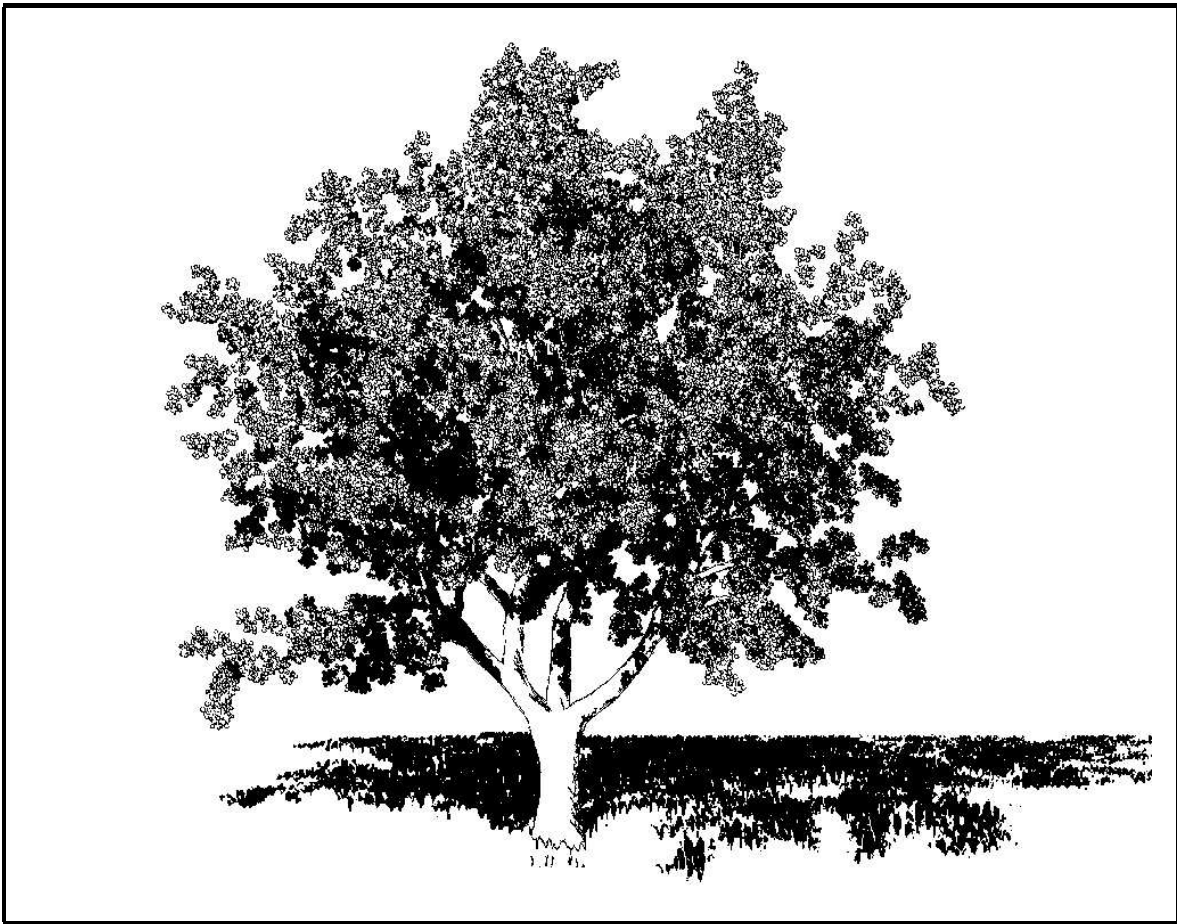




NORSIGD INFO

Nummer 2 2002



NORSK SAMARBEID INNEN GRAFISK DATABEHANDLING

ISSN 0803-8317

Aktivitetskalender

Hva skjer når og hvor?

November 2002

28–29 **SIGRAD 2002**, Norrköping, Sverige. <http://www.sigrad.org/SIGRAD2002/SIGRAD2002.html>.

Desember 2002

8–9 **MM 2002** – 10th ACM International Multimedia Conference, Juan-les-Pins, Frankrike. <http://mm02.eurecom.fr/>.

Februar 2003

3–7 **WSCG 2003** – 11th Int'l Conf. in Central Europe on CG, Visualization and Interactive Digital Media, Plzen, Tsjekkia. <http://wscg.zcu.cz/wscg2003/wscg2003.htm>.

Mars 2003

3–4 **Nasjonalt seminar i Grafisk Databehandling**, Fornebu, Oslo. <mailto:leneha@simula.no>.

6–7 **SIMVIS 2003** – Simulation und Visualisierung, Magdeburg, Tyskland. <http://www.simvis.org>.

April 2003

5–10 **CHI 2003**, Annual ACM SICHI Conference on Human Factors in Computing Systems 2003, Fort Lauderdale, FL, USA. <http://sigchi.org/chi2003/>.

Mai 2003

22–23 **IPT/EGVE 2003**, Joint 7th Immersive Projection Technology Workshop / 9th Eurographics Workshop on Virtual Environments, Zürich, Sveits. <http://www.ipt-egve.ethz.ch/>.

Juli 2003

27–31 **SIGGRAPH 2003** – 30th International Conference on Computer Graphics and Interactive Techniques, San Diego, CA, USA. <http://www.siggraph.org/>.

September 2003

1–6 **EG 2003** – 24th annual conf. of the European Association for Computer Graphics (EUROGRAPHICS). Granada, Spania.. <http://www.eg.org/eg2003/>.

Helwig's Conference Calender

Flere aktiviteter finner du på <http://www.vrvis.at/ConfCal/>.



Om forsiden

Bildet på forsiden viser et tre som er laget med non-photorealistic rendering. Metoden ble presentert på Siggraph 2000 i følgende bidrag: Oliver Deussen, Thomas Strothotte: "Computer-Generated Pen-and-ink Illustration of Trees", Computer Graphics, Vol. 34(4), s. 13–18.

Hilsen fra styret

Kjære medlemmer,

Det skjer mange interessante utviklinger om dagen innen datagrafikk. Vi skal i dette nummeret oppdatere dere på det som skjer i miljøet. Vi presenterer en fersk rapport fra Eurographics 2002, en oversikt over JPEG 2000 standarden, og en oversikt over visual datamining. Dessuten ser vi i grafikkhjørnet nærmere på PCer og spillkonsoller, det nye grafikk-språket Cg, og utviklingsverktøyet RenderMonkey.

3.-4. mars 2003 skal et nasjonalt seminar om datagrafikk gå av stabelen, der Norsigd er aktivt med under forberedelsene. Se annonseringen i dette heftet. Vi håper å se mange av dere på seminaret. Meld dere på med én gang!

Når så mye interessant skjer innen fagmiljøet håper vi at dette nye Norsigd Info kan bringe mer virtuell farge i de ellers så grå høstdagene. Styret i Norsigd ønsker dere en fortsatt god høst.

Hilsen,

Wolfgang Leister



NORSIGD Info

– medlemsblad for NORSIGD

Utgitt av: NORSIGD
 Ansvarlig: Wolfgang Leister
 Norsk Regnesentral
 Postboks 114 Blindern
 0314 OSLO

ISSN: 0803-8317

Utgivelser: 2002: 15/4 15/10

Annonsepriser: Helseid kr 5 000
 Halvsid kr 2 500

Layout: Wolfgang Leister
 L^AT_EX₂ ϵ

Ettertrykk tillatt med kildeangivelse

Innhold

Aktivitetskalender	2
Hilsen fra styret	3
Visual Data Mining	4
Grafikkhjørnet	7
Nasjonalt seminar om datagrafikk	8
JPEG 2000	9
EG 2002	13

Visual Data Mining

Wolfgang Müller, University of Applied Sciences, Darmstadt
Heidrun Schumann, University of Rostock

Visual data mining is a novel approach to deal with the growing flood of information. The aim is to combine traditional data mining algorithms with information visualization techniques to utilize the advantages of both approaches. We give a short overview on visual data mining with a focus on information visualization aspects.

The amounts of data collected in corporate and public databases are increasing day by day. Databases with several terabyte of data are not uncommon any more. For example, the K-mart customer database is expected to grow up to 4-5 terabyte by the end of this year, and Envisat, a satellite launched by ESA (European Space Association) to observe Earth's environment at the beginning of this year, exceeded the terabyte mark for data transferred over a satellite channel in only 3 months [Wrol 02].

Data is not information. All the bits and bytes collected in these databases are without value if we cannot explore and analyse this data to extract meaning and to get insight. Standard data management systems do not provide the needed functionality. This problem is targeted in the field of Data Mining. Data mining denotes the analysis of huge amounts of data targeting to extract useful information, utilizing statistical methods such as clustering techniques, Factor Analysis, Multidimensional Scaling, and AI- methods like Kohonen Networks. Data mining techniques are today an integral part of today's data warehouse solutions. Nevertheless, current data mining tools are far from being optimal. In general, the complex parameters of available analysis techniques make it difficult to comprehend and control the mining process; consequently, data mining tools are sometimes awkward to use and the results are difficult to value.

From the very beginning, information visualization techniques have been considered to be a promising alternative to analysis methods based e.g., on statistic and AI techniques. Information visualization exploits the phenomenal abilities of human perception to identify structures by presenting abstract data visually, allowing the user to explore the complex information space to get insight, to draw conclusions and directly interact with the data.

Visual data mining is a novel approach to data mining. It denotes the combination of traditional data mining techniques and information

visualization methods. The utilization of both automatic analysis methods and human perception/understanding promises better and more effective data exploration. Visual data mining techniques have proven to be of high value especially in exploratory data analysis. They have high potential especially

- For exploring large databases,
- When little is known about the data and the exploration goals are vague, and
- When highly inhomogeneous and noisy data is given.

Based on the degree of integration of information visualization and automated data mining techniques we can distinguish three classes of visual data mining solutions:

- No or very limited integration. This corresponds to the application of traditional information visualization or automated data mining techniques on raw data.
- Loose integration of information visualization and traditional data mining. Visualization and automated mining techniques are applied sequentially. The results of each mining operation can be used as an input to the following analysis step.
- Full integration of information visualization and automated techniques. This allows for the application of techniques from both fields in parallel; results can be combined, providing an integrated view on the data mining process and its outcome.

Visualization

Visualization is a key process in Visual Data Mining. Visualization techniques can provide a clearer and more detailed view on different aspects of the data as well as on results of automated mining algorithms. Some of these aspects will be discussed in the following. We will focus on the information content and the information structure, in which the information is organized [Bert 77].

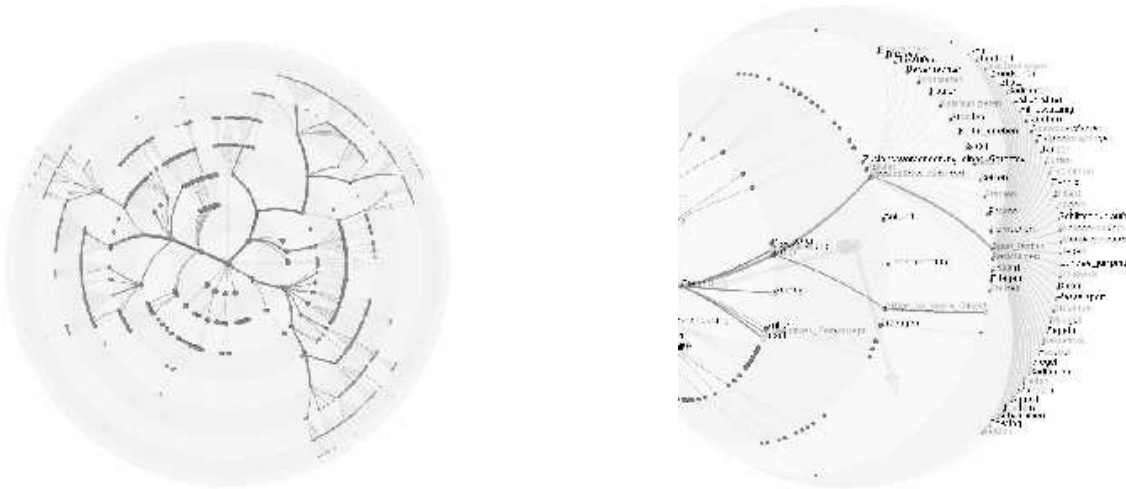


Figure 1: Magic Eye View with different focus areas

Information Structure

The exploration of relationships between several information objects, which represent a selection of the information content, is an important task in visual data mining. Such relations can either be given explicitly - when being specified in the data - or they can be given implicitly, when the relationships are the result of an automated mining process, e.g., based on the similarity of information objects by hierarchical clustering.

A number of customized methods for visualizing an information structure have been developed. Most of them are based on the visualization of hierarchies. Here, we distinguish between space-filling and explicit techniques. Techniques of the first class show relationships between information objects by special arrangements. Popular examples for this approach are Treemap [Shne 92] and Sunburst [StZh 00].

Techniques of the second class represent relationships by edges. A well-known example for the latter approach is the Hyperbolic Viewer [Lamp95]. Its main idea is to use the hyperbolic plane for arranging the nodes of the hierarchy by radial layout and to achieve a focus area in the near of the midpoint by reprojection into the Euclidian space.

The Magic Eye View [KrLS 00] is another example for a explicit layout. Here, the layout is based on a 2D radial layout which is mapped onto a hemisphere. An additional projection is introduced in order to achieve a focus & context display and to enable a smooth transition between these regions. Figure 1 demonstrates this technique.

Information Content

In addition to exploring the information structure, the identification of structures in the data values is an important task in visual data mining. For this, techniques to visualize the qualitative and quantitative properties of information objects are required. In addition, these techniques usually have to deal with large amounts of multivariate data. Standard techniques in this context are panel matrices, parallel coordinates, icon and pixel based techniques.

Panel matrices arrange bivariate displays of adjacent attributes in matrix form. A popular visualization technique in this category is the Scatterplot Matrix, where multiple adjacent scatterplots are displayed in one image [Clev 93].

Parallel Coordinates map the n-dimensional space onto a two-dimensional plane. For each attribute a separate coordinate axis is constructed. These axis are positioned aside each other. Each information object is presented as a polygonal line intersecting each of the axes at the point corresponding to the value of the considered attribute.

In icon-based techniques the attribute values of an information object are mapped to the features of an icon. Icons may be defined arbitrarily - for example as little faces in the example of Chernoff Faces [Cher 73] or a colored symbol in the case of Needle Icons [AlKo 01], and Color Icons [Levk 91]. Figure 2 displays an example using morphed faces for the visualization of multivariate data. Pixel-based techniques push the idea of minimizing the size of an attribute in screen space even further to present even larger amounts of data [Keim 00].

Information Objects

Another set of visualization techniques focuses on the presentation of all aspects of information objects, mostly to support their identification, their analysis, or to find relations to other elements. Typical examples are the visualization of DNA sequences, documents, or search results. The transition to information graphics - specifically designed for a given application - is sometimes fluid.

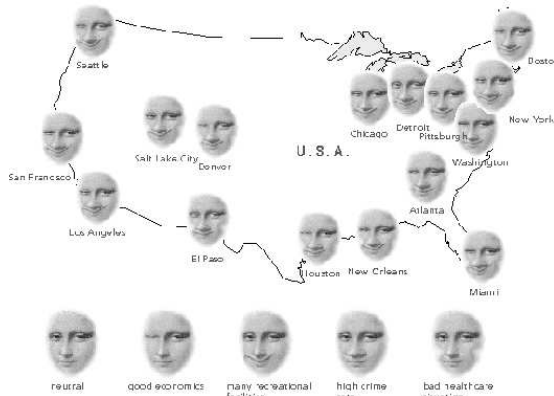


Figure 2: Visualization of demographic information using icons [AlMü 9800]

Interaction Techniques

Interaction is crucial for effective visual data mining. The data analyst to directly must be able to interact with the presented data and to change the visualizations and the mining parameters according to his needs. Also, interaction techniques may applied to enable the user to relate and to combine multiple independent views of the data.

Interaction techniques can be categorized based on the effects they have on the display. Navigation techniques focus on modifying the projection of the data on the screen, using either manual or automated methods. View enhancement methods allow users to adjust the level of detail on the visualization or of parts of it; furthermore, they allow for the modification of the mapping to emphasize some subset of the data. Selection techniques provide users with the ability to isolate a subset of the displayed data for operations such as highlighting, filtering, and quantitative analysis. Selection can be done directly on the visualization (direct manipulation) or via dialog boxes and other query mechanisms (indirect manipulation).

Conclusions

Visual Data Mining is a novel, and efficient approach for exploring large data sets. The combination of automated mining methods and visualization techniques take advantage of both, computational power and the abilities of human perception. Since we could cover only few aspects of these technology see e.g., [KeMS 02] for further information.

References

- [AlMü 98] Alexa, Marc, und Müller, Wolfgang: Visualization by Metamorphosis, In: Craig M. Wittenbrink and Amitabh Varshney (Eds.): IE-EE Visualization 1998 Late Breaking Hot Topics Proceedings, S. 33-36, October 1998.
- [AlKo 01] Abello J.; and Korn, J.: MGv: A system for visualizing massive multi-digraphs, Transactions on Visualization and Computer Graphics, 2001.
- [Bert 77] Bertin, J.: Graphics and Graphic Information Processing, deGruyter Press, Berlin 1977.
- [Cher 73] Chernoff, H.: The use of faces to represent points in k-dimensional space graphically, Journal Amer. Statistical Association, 68:361-368, 1973.
- [Cleve 93] Cleveland, W. S.: Visualizing Data, Hobart Press, New Jersey, 1993.
- [Keim 00] Keim, D.A.: Designing pixel-oriented visualization techniques: Theory and applications. Transactions on Visualization and Computer Graphics, 6(1):59-78, Jan-Mar 2000.
- [KeMS 02] Keim, Daniel A.; Müller, Wolfgang; und Schumann, Heidrun: State-of-the-Art Report Visual Data Mining, State of the Art Report, Eurographics Conference 2002, Saarbrücken, September 2002.
- [KrLS 00] Kreuzeler, M.; Lopez, N.; Schumann, H.: A Scalable Framework for information Visualization, Proceedings InfoVis'2000, Salt Lake City, 2000, pp.27-36.
- [Lamp 95] Lamping, J. et al: A focus+context technique based on hyperbolic geometry for viewing large hierarchies. ACM Proceedings CHI'95, Denver, 1995, S. 401-408.
- [Levk 91] Levkowitz, H.: Color Icons: Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters. Proceedings Visualization'91, IEEE Computer Society Press, Los Alamitos, 1991, pp. 164-170.
- [Shne 92] Shneiderman, B.: Tree Visualization with Treemaps: A 2D Space Filling Approach.

ACM Transactions on Graphics, Vol.11, No. 1, 1992, pp. 92-99.

[ScMü 99] Schumann, Heidrun; und Müller, Wolfgang: Visualisierung - Grundlagen und allgemeine Methoden, Springer Verlag, ISBN 3-540-64944-1, Heidelberg, 1999.

[StZh 00] Stasko, J.; Zhang, E.: Focus+Context Display and Navigation Techniques for Enhanc-

ing Radial, Space-Filling Hierarchy Visualizations, Proc. IEEE Information Visualization 2000, Salt Lake City, UT, Oct. 2000, pp. 57-65.

[Wrol 02] Wrolstadt, Jay: Satellite Smashes Terabyte Data Barrier, NewsFactor Sci.:Tech, <http://sci.newsfactor.com/perl/story/18424.html>, June 2002.

Grafikkhjørnet

Wolfgang Leister, Norsk Regnesentral

Det skjer mye om dagen innen datagrafikk. Tidlig i vår kom flere nye spillkonsoller på markedet som kan vise frem en imponerende grafikkytelse. I sommer kom nVidia med programmeringsspråket Cg, som vi skal se nærmere på.

Dataspill og grafikk

I våren 2002 kom flere produsenter av spillkonsoller ut med nye modeller: Sony Playstation II, Xbox og Gamecube. Disse maskinene brillerer med sin grafikkytelse som er optimalisert for dataspill. Er datagrafikk på spillkonsoller så mye raskere enn på PC?

Siden disse spillkonsollene skal være veldig mye bedre på grafikkensiden enn en vanlig PC, skal vi se nærmere på de forskjellige arkitekturene. Vi tar svaret først: PCen er designet for å være en maskin som er egnet til det meste innen IKT-oppgaver, mens spillkonsollene er optimalisert for spill og grafikkytelse.

Xbox er egentlig et grafikk-kort med påhengt lyd kort og CPU. Alle byggesteinene er loddet sammen på hovedkortet, mens PCen har en arkitektur med kort som settes inn i en standardisert buss-arkitektur. Xbox har dermed ikke den fleksibiliteten til en PC, men dette danner grunnlaget til å kunne optimalisere noe på hastigheten (mindre elektrisk motstand, og dermed høyere hastighet).

Xbox og andre TV-spill leverer lavere oppløsning og oppdateringsfrekvens enn det man forventer fra en PC. Spillkonsollene brukes vanligvis i forbindelse med en TV på 640 × 480 piksler på 60 hhv. 50 Hz. Brukerne av spillkonsollene er vant til TV-oppløsning, samtidig som disse ofte brukes i en hjemme-omgivelse.

På en PC er ingen lenger fornøyd med mindre enn 1024 × 768 piksler på 75 Hz. Dermed må en PC levere grafikk med en båndbredde som tilsvarende 7.7 ganger så mye som på en Xbox. Dermed er altså kravene for en PC mye større

enn for en spillkonsoll, selv om en PC ikke er optimalisert for spillanvendelser.

Gamecube har en noe annen tilnærming. Den er optimalisert med henblikk på kostnader. Fordyrende elementer som DVD spiller er ikke inkludert i konseptet.

Playstation II har kanskje det mest ekstravagante design av spillboksene. På innsiden finner vi den såkalte "Emotion Engine" som er i prinsippet en innebygd Playstation I. Videre finner vi fpu (floating point unit), vector processing unit og en egen bildeprosessor. Denne image processing unit sørger bl.a. for MPEG dekoding.

Utsalgsprisen på spillkonsollene i forhold til PC kan være misvisende! Ofte er hardware sponset gjennom salg av spill. Ofte finnes det tekniske og juridiske hindringer mot å bruke konsollene til andre formål enn tiltenkt.

Denne korte oversikten er ikke ment til å beskrive fordelene eller ulempene de forskjellige spillkonsollene seg imellom. Du finner mer informasjon på Web sidene til de ulike leverandørene.

C for graphics: Cg

I sommer nVidia med en nyhet: et nytt programmeringsspråk skal gjøre det lettere å programmere grafiske applikasjoner: "C for graphics" eller kort **Cg**. nVidia har utviklet språket sammen med Microsoft. Cg kode kan kjøres sammen med OpenGL 1.3 eller DirectX 8.0 vertex shaders. Konseptet inkluderer en toolkit med bl.a. compiler og runtime omgivelse.

Målet med Cg er å gjøre det lettere å utvikle grafiske applikasjoner som kjøres på kraftige grafikk-kort. Hittil har det vært nødvendig med

assembler-kode for å kunne utnytte de mulighetene som grafikk-kortene tilbyr i applikasjoner som f.eks. spill. For å oppnå høye hastigheter må mest mulig av spesialiserte grafikk-operasjonene kjøres på grafikk-kortet uten at CPU er innblandet i dette. Med Cg skal lette det å skrive kode som kan kjøres uten endringer på flere plattformer og hardware.

Cg bygger i utgangspunktet på programmeringsspråket C, men også en del andre programmeringsspråk som Java, C++, RenderMan og Stanford shading language hadde innflytelse på utviklingen. Cg har en del spesielle datatyper og operatører for vektorer og matriser. Det finnes også en del restriksjoner i forhold til C, delvis for å kunne lettere tilpasse seg hardware-behovene. Så finnes det et krav om at løkker formulert i Cg skal kunne ruller ut ved kompileringstid; dvs. at løkker heller betraktes som kraftige makroer.

Andre restriksjoner har etter mitt skjønn ingen spesiell grunn, og det ser ut til at utviklerne prøver å rydde opp i anakronismer i C, som de fleste utviklerne har vennet seg til. Det at Cg ikke støtter operatører som `goto`, `break`,

`continue`, `switch`, osv. kan gjøre det lettere for kompilator, men jeg ville heller valgt å legge til et optimaliseringssteg i kompilatoren.

En fordel ved Cg er at man kan skrive datatyper og operatører i Cg kortere og mer oversiktlig enn i C. Så finnes det korte skrivemåter for å definere arrayer og matriser, samt en “swizzle”-operatør.

Cg bygger på shader filosofien, som også andre moderne arkitekturer bruker. En *shader* er et program som implementerer en grafisk effekt på en mengde piksler eller vertices. Cg tilbyr en plattform for å konstruere plattformuavhengige shaders.

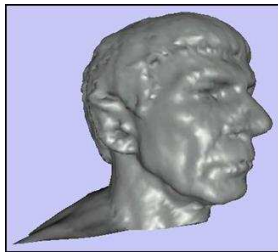
Cg er et forsøk for å normere programmering med datagrafikk, denne gangen ved å bruke C som utgangspunkt. Cg trenger nok noe modning, og jeg er sikker på at dette er bare det første steget i en større utvikling. Det finnes også rykter om at en Cg kompilator skal utvikles etter open source modellen.

Mer informasjon om Cg finner du på <http://developer.nvidia.com/Cg> og på <http://www.cgshaders.org>.

Annonsering:

Nasjonalt seminar i Grafisk Databehandling

Institutt for informatikk, Universitetet i Oslo, Simula Research Laboratory AS i samarbeid med Norsigd inviterer til seminar i grafisk databehandling. Tidspunktet er satt til 3.–4. mars 2003 og arrangementet vil bli avholdt hos Simula Research Laboratory AS på Fornebu. På <http://www.simula.no/about.php> finner dere en beskrivelse av hvordan finne veien til Simula.



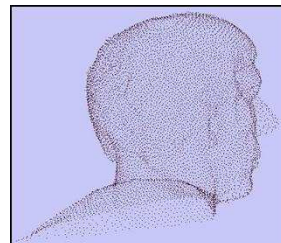
Den videre annonsering av seminaret vil foregå per epost og dersom du ønsker å være oppdatert send en epost til Lene Hansen (leneha@simula.no) og merk denne GD2003.

Seminaret skal dreie seg om grafisk databehandling i Norge og vi er åpne for alle forskningsbidrag som sorterer under dette begrepet, inkludert anvendelser innen områder som spill, DAK, GIS og medisinsk IT.

Professor Tom Lyche (tom@ifi.uio.no) og professor Morten Dæhlen (morten@simula.no) er ansvarlig for programmet. Dersom du ønsker

å holde et innlegg på seminaret, send “abstract” på inntil én side til Morten Dæhlen (morten@simula.no) innen 20. januar 2003. “Abstract” skal skrives på engelsk. Endelig program vil foreligge kort tid etter og fristen for påmelding til seminaret er satt til 10. februar 2003.

Vi planlegger å starte med en sesjon før lunch på mandag 3. mars og avslutte til lunch dagen etter — tirsdag 4. mars. Det blir middag på kvelden. Seminarets omfang og form vil avhenge av antall bidrag.



JPEG 2000 – En standard for bilder

Wolfgang Leister, Norsk Regnesentral

I dette bidraget presenteres JPEG 2000 standarden. På NR har vi tatt en nærmere titt på standarden og implementert deler av den.

The JPEG-2000 Standard published by ISO under ISO/IEC 15444. The standard consists of ten parts. Some parts of the standard have not been released yet.

Parts of the JPEG 2000 Standard

The standard consists of the following parts:

Part 1 – Core Coding system: Part 1 is published as ISO/IEC 15444-1:2000. It consists of an overview and eleven annexes, which actually describe the technical details. Part 1 of the standard is the most relevant for our current project. Part 1 describes syntax of code stream (Annex A), the JP2 file format (Annex I), coding methods, etc.

Part 2 – Extensions: Part 2 describes extensions to part 1. The extensions are possibly patented, and might be non-free in terms of royalties. The extensions include compression, variable offset, quantization, visual masking, decomposition of tile-components, transformations, and the JPX file format (described in Annexes L and M).

Part 3 – Motion JPEG 2000: Specifies the use of the JPEG 2000 Codec for the coding of timed sequences of images. This part of the standard is intended for use of capturing (e.g., digital cameras) and film production. The standard extends the JP2 definition, and includes audio tracks from the MPEG-4 standard.

Part 4 – Conformance testing: This part specifies the framework, concepts, and methodology for testing and criteria to be achieved to claim conformance to the standard. It provides a framework for specifying abstract test suites (ATS) and for defining the procedures to be followed during conformance testing. This part could be of interest within our project later, when comparing our implementation toward the standard.

Part 5 – Reference software: Part 5 contains the reference software itself, in source code form. The reference software includes a C implementation (JasPer software found at <http://www.imagepower.com/>), and a Java implementation (JJ2000 found at <http://jj2000.epfl.ch/>).

Part 6 – Compound image file format:

Part 6 describes a file format for storing compound images. The file format builds on the JP2 and JPX file formats. Especially it specifies a container and mechanism for combining multiple images into a single compound image, including grouping of images in a hierarchy. It allows also compression methods other than JPEG 2000.

Parts 7 to 10: These parts are not available yet. They include definitions of JPSEC, JPIP, and JP3D.

JPEG 2000 Part 1

Part 1 consists of a short overview and eleven annexes, which actually describe the technical details.

Annex A – Codestream Syntax: Consists of the definition of marks and marker segments. The marker segments contain the image data and information on image characteristics.

Annex B – Data ordering: Describes how to compose the images and their organization in the codestream into components, tiles, sub-bands, and their divisions.

Annex C – Arithmetic entropy coding: Defines the lossless arithmetic entropy encoding (MQ-coder).

Annex D – Coefficient bit modeling: Defines the modeling of the transform coefficient bits, and how coefficients are arranged into code-blocks, bit-planes and coding passes.

Annex E – Quantization: Specifies the forms of quantization used for encoding (also: dequantization for decoding).

Annex F – Discrete wavelet transformation of tile components: Describes the forward (and inverse) discrete wavelet transformation applied to one tile component.

Annex G – DC level shifting and component transformations: Specifies how signed values resulting from the decoding process are converted into the proper reconstructed samples.

Annex H – Coding of images with Regions of Interest: Describes how to specify parts of the image that are to be coded before other parts. We do not need this Annex for our purposes.

Annex I – JP2 file format syntax: Describes how JP2 images are built.

Annex J – Examples and Guidelines: Includes a number of examples intended to indicate how the encoding process works.

Annex K – Bibliography.

Encoding a JPEG 2000 image

Most compression algorithms for photographic images or video frames are built on the following three phases:

1. Use a transform on the image to reorder the information content in a more “ordered” way, reducing the entropy of the elements. Quantization may also be applied during the transform to further reduce the entropy.
2. Apply an entropy based compression scheme on these elements to find a shorter way of expressing the same content.
3. Package and output the information in a useful and robust format.

The JPEG 200 standard is no exception to this “rule”. In order to encode a JPEG 2000 image, the following steps must be performed (in order of encoding task):

- DC level component transform (Annex G).

- Image decomposition into components, tiles, sub-bands, and their divisions (Annex B).
- Discrete Forward Wavelet Transform (Annex F.3).
- Quantization (Annex E).
- Coefficient bit modeling (Annex D).
- Arithmetic coding (Annexes C.1 and C.2).
- Code Stream Syntax (Annex A).
- Encoding the JP2 File Format (Annex I).

DC level component transform

For details on DC level shifting we refer to Annex G of the JPEG 2000 standard.

DC level shifting is performed on samples of components that are unsigned only. A constant value defined in Annex G.1.1 is subtracted, so that the component contains both positive and negative sample values in the transformed image.

In order to improve compression efficiency component transforms are applied to the samples. The reversible component transformation (RCT) in Annex G.2.1 is used only together with the 5-3 reversible wavelet transform. The RCT is applied to the three first components of an image as a decorrelating transformation. The irreversible component transformation (ICT) in Annex G.3.1 is used together with irreversible wavelet transform. The ICT is applied to the three first components of an image as a decorrelating transformation. Note: When the first components are Red, Green and Blue, then the Forward ICT can be seen as an approximation of a YCbCr transformation.

Image Decomposition

For details on image decomposition we refer to Annex B of the JPEG 2000 standard.

An image consists of several **components**. A component is a two-dimensional array of samples. A colour image typically consists of several components from a specified colour space, for instance representing the different colour channels. The different components can have different reference grid sizes, e.g., when using different sampling factors.

All components are defined with respect to a high resolution grid. The reference grid is a rectangular grid of data points with indices from (0,0) to (Xsiz-1, Ysiz-1). Note, that the coordinate (0,0) is in the upper left corner, i.e. the image is defined in the 4th quadrant. Within the grid

an image area can be defined with offset (XO-siz,YOsiz). The image area reaches from (XO-siz,YOsiz) to (Xsiz-1,Ysiz-1).

The reference grid is partitioned into a regular sized rectangular array of **tiles**. The reference grid is defined in Annex B.2. The tile size and the tiling offset are defined on the reference grid by offset and tile width/height. It is not necessary that the whole tile is filled with image information, e.g., in case the image size is not a multiplum of the tile size. The tile index number is row by row for tiles that are within the image area.

The use of image areas within the reference grid, and the choice of the tile grid can be used to optimise some applications, e.g., for fast access of the image in other aspect ratios than the original image. The standardization document shows an example in Annex B.3.

Each tile contains several **precincts**, which are defined in a grid of width ($2^{PPx}, 2^{PPy}$), and which has the upper left corner at the coordinates (0,0). The values for PPx and PPy are specific for each tile, and are coded into the code stream. The definition of the precincts is given in Annex B.5.

The tile components are subdivided into $N_L + 1$ **sub-bands** after the wavelet transformation (as described in Annex F), representing the resolutions of the wavelet transform. The borders of the sub-bands are defined by the equations in Annex B.4. The sub-bands show the position of the information in this representation of the image component **after** the wavelet transformation.

Each sub band is divided into **code blocks** which are defined by the parameters xcb and ycb. These values are encoded into the code stream. The size of the code blocks is defined by (n^{xcb}, n^{ycb}), and is the same for all code blocks of a tile at the same resolution. The code block grid is anchored at the coordinates (0,0). Code blocks are defined in Annex B.6.

The coded data of each code-block is distributed across one or more layers in the code-stream. Each **layer** consists of some number of consecutive bit-plane coding passes from each code-block in the tile. Each layer successively and monotonically improves the image quality. Layers are defined in Annex B.7.

The data representing a specific tile, layer, component, resolution and precinct appears in the codestream in a contiguous segment called a **packet**. Packet data is aligned at 8-bit boundaries. Within each sub-band, the code-block

contributions appear in raster order. In resolution $r = 0$ the packets only contain the LL band, while at resolution $r > 0$ the packets only contain HL, LH and HH bands. Packets are defined in Annex B.8.

The definition of the packet headers is given in Annex B.9.

Progression orders are defined in Annex B.11. The progression order defines in which order the data of the image is coded, i.e. the order in which packets are included into the stream.

Discrete Forward Wavelet Transform

For details on the Discrete Forward Wavelet Transform we refer to Annex F.3 of the JPEG 2000 standard. The forward discrete wavelet transformation (FDWT) is applied to one tile component at a time. To perform the FDWT a one-dimensional sub-band decomposition of a one-dimensional set of samples into low-pass coefficients is used. It produces a downsampled low-resolution version of the original set, and three high-pass coefficients representing a downsampled residual version of the original set.

Each tile component is transformed into a set of two-dimensional sub-band signals (called **sub-bands**), at various spatial resolutions. The different number of levels of spatial resolution N_L is called the number of decomposition levels.

In Annex F.3 the placement of the values, and the procedure for the FDWT is explained. The procedure is defined by flow charts. The one-dimensional filtering is explained in Annex F.3.8. The standard defines reversible (Annex F.3.8.1) and irreversible (Annex F.3.8.2) 1D-filtering.

Quantization

For details on quantization we refer to Annex E of the JPEG 2000 standard. Quantization is described in Annex E.2, while some equations from Annex E.1 are used. Basically each of the transform coefficients of a sub-band is divided by factor for the sub-band, and treated with the floor function.

Coefficient bit modeling

For details on Coefficient bit modeling we refer to Annex D of the JPEG 2000 standard. It describes how the coefficients are arranged into code-blocks, bit-planes, and coding passes. The

coefficients from the wavelet transform are arranged into rectangular blocks within each sub-band, called **code blocks**. These code blocks are coded a bit-plane at a time starting from the most significant bit-plane with a non-zero element to the least significant bit-plane.

Each bit-plane in a code-block is coded according to a code-block scan pattern in three coding passes called significance propagation, magnitude refinement, and cleanup. For each pass contexts CX are created which are provided to the arithmetic coder, along with the bit stream D. The arithmetic coder is reset according to selected rules.

The rules for bit modelling and arithmetic coding ensures that the information in the code-block that contributes most to the visible features of the image always is output before less important refining information. For bitarate control the output codestream can be truncated after each pass. Trunkating on a bit-plane boundary can be considered equivalent with quantization by a power of 2, except that the quantization is performed only on a codeblock a time, not on the output of the wavelet transform on the entire tile.

Arithmetic entropy coding

For details on arithmetic coding we refer to Annexes C.1 and C.2 of the JPEG 2000 standard. The arithmetic coding is compatible with the arithmetic coder defined in ISO/IEC 14492 [?]. The definition in the standard is normative in the sense that the given flow charts and tables define the output that the encoder shall follow nevertheless how it is produced.

The binary adaptive arithmetic encoder takes a decision D and a context CX as input, and produces compressed data output CD. Both D and CX are provided the coefficient bit modeling step. The recursive probability interval subdivision of Elias coding is the basis for the binary arithmetic coding process. With each binary decision the probability interval is subdivided into a sub-interval for the more probable symbols (MPS) and a sub-interval for the less probable symbol (LPS). The algorithm uses an approximation, which in some cases makes a correction step necessary.

The algorithm for the arithmetic encoder is described in Annex C2 using flow charts and tables. The algorithm is named once by its name MQ-coder within Annex C.2.

Codestream Syntax

For details on the codestream syntax we refer to Annex A of the JPEG 2000 standard.

In the codestream markers and marker segments are used to delimit and signal characteristics of the content. Each marker is 2 bytes long: one byte 0xFF and an opcode 0x01–0xFE. Markers that have parameters form a marker segment, that includes the marker and the associated parameters. Marker segments include a length-field.

Headers are collections of markers and marker segments. There are two types of headers: The Main header is found at the beginning of the codestream. The tile-part headers are found at the beginning of each tile part. Some markers and marker segments are restricted to only one of the two types of headers, while others can be found in either. A list of marker segments can be found in Table A-2 of Annex A.2.

The codestream starts always with the SOC marker followed by the SIZ marker segment. Several markers follow within the main header. Then one or more tile parts follow. The codestream is finished with the EOC marker at the end.

A tile part starts with the SOT marker followed by several marker segments. The tile part header is concluded by a SOD marker, and the bit stream of the tile follows. Note, that there are marker segments for the packets, that are included into the tile part header (see Annex A.7).

There are markers for marking positions in the bitstream, e.g., to signal the start of a packet. These markers are intended for error resilience.

Coding into the JP2 File Format

For details on the JP2 file format we refer to Annex I of the JPEG 2000 standard.

A JP2 file represents a collection of boxes. The binary structure of a JP2 file is a contiguous sequence of boxes, where the start of the first box shall be the first byte of the file, and the last byte of the last box shall be the last byte of the file. The standards speaks of simple boxes containing payload and superboxes containing other boxes. Some box types are optional, other box types are mandatory.

The binary syntax of the boxes is defined in Annex I.6. The codestream where the image is encoded is contained in a contiguous codestream box, defined in Annex I.7.4.

Eurographics 2002

Wolfgang Leister, Norsk Regnesentral

NORSIGDs fagansvarlig har besøkt Eurographics 2002 i Saarbrücken. Her gjengis det noen inntrykk fra konferansen.

Eurographics konferansen 2002 fant sted i byen Saarbrücken sørvest i Tyskland. Siden det i vår ble mulig til å ta relativt rimelige flyturer fra Torp til Hahn, var det økonomisk sett lettere å delta på EG 2002 konferansen uten at budsjettet sprakk.¹

Undertegnede har altså tatt turen til Saarbrücken for å se hva som er aktuelt innen grafisk databehandling. Konferansen var delt inn i to dager med tutorials og tre dager med presentasjoner av papers og prosjekter, oversiktsforedrag, industriseminarer og lab-presentasjoner. Selv om det var et bredt spekter av tutorials på programmet, deltok jeg ikke på disse.

Konferansen

Konferansedelen varte i tre dager, fra onsdag til fredag. Hver dag inneholdt et invitert foredrag med aktuelle temaer, tre sesjoner med presentasjoner (papers, industriseminar, state-of-the-art rapport), og lab-presentasjoner. Totalt sett ble det tilbudt tretten paper-sesjoner, fem STAR (state of the art) reports, tre industriseminarer og en industri paneldiskusjon. I tillegg kom det en presentasjon av short papers og ni sesjoner med lab-presentasjoner. Blant disse programpunktene kunne deltagerne velge, men det var dessverre planlagt at flere interessante ting skjedde samtidig.

Totalt hadde ca. 450 deltagere funnet veien til konferansen, flere enn i de årene før. Av 233 innsendte bidrag ble 42 valgt ut. Disse ble presentert i konferansens hoveddel. I tillegg ble en del artikler presentert som poster-sesjon.

Tre andre deltagere fra Norge (Universitetet i Oslo og Simula Senteret) var også tilstede på konferansen. Ellers var deltagere fra Norden nokså fraværende, og det ser nesten ut som om hele Norden har meldt seg ut av datagrafikkforskningen. Tendensen fra årene før ser ut til å fortsette.

Short papers ble i år organisert som poster sesjon, dvs. at hver bidragsyter laget noen folier som ble klistret på en plakavegg. I pausene

og spesielt på onsdag ettermiddag kunne man da lese bidragene og snakke med de som hadde laget presentasjonene. Jeg satt med inntrykket av at disse short papers ikke fikk den oppmerksomheten de hadde fortjent.

Temaer

Temaene i konferansen angir ofte hva som er aktuelt for tiden innen datagrafikk-forskningen. Fortsatt er det et meget bredt spekter av temaer som blir dekket av konferansen. Kanskje har temaene blitt mere spesialiserte de siste årene, og man kan se en trend over til mer applikasjonsorienterte temaer. Blant temaene finner vi surface meshes og flater, non-photorealistic rendering, simulering av klær og mennesker, teksturer, bilbasert rendering, og interaktiv rendering.

Det var mange bra bidrag med i konferansen, men ikke alltid klarte foredragsholderne å presentere innholdet på en forståelig måte. I mange tilfeller kunne foredragsholderne forbedret presentasjonen dersom de hadde øvd noe foran speilet. I noen tilfeller kom det også til språkproblemer der foredragsholderen ikke forsto spørsmålene fra publikum.

Men selvsagt var det også meget bra presentasjoner blant foredragene. Spesielt skal arbeidet til Michael Wand og Wolfgang Straßer nevnes, som fikk en pris for beste studentarbeid med temaet "Multi-Resolution Rendering of Complex Animated Scenes". Det ble også demonstrert en meget imponerende animasjon som viste resultatene fra arbeidet.

Prisen for beste artikkel gikk til Gabriel Taubin, Laurent Balmelli, og Fausture Bernardini (Space-Optimized Texture Maps). Andre og tredje priser gikk til Mark Meyer, Mathieu Desbrun og Pierre Alliez (Intrinsic Parametrizations of Surface Meshes) hhv. Ming C. Lin, Jörg Haber, Vincent Scheib og Hans-Peter Seidel (Efficient Fitting and Rendering of Large Scattered Data Sets using Subdivision Surfaces).

¹For de som planlegger reiser til Tyskland skal det sies at flyplassen Hahn ligger noe utenfor allfarvei, ca. halvannen timer med buss fra Frankfurt. Da det bare går ett fly om dagen kl. 10:30 må hjemreisen til Oslo planlegges. Enten man overnatter i nærheten av Hahn eller man tar en buss ved 7-tiden fra Frankfurt eller Mainz. Å reise fra andre destinasjoner kan derfor være noe vanskelig eller kostbart.

Alle artikler finnes samlet i konferanse-utgaven til Computer Graphics Forum. EG-medlemmer får elektronisk tilgang til artiklene på <http://www.eg.org>. Short papers og STAR reports er presentert i separate bind på papir, men er også tilgjengelig elektronisk.

Inviterte foredrag

Hver dag ble det holdt et invitert foredrag med et aktuelt tema. Først ut var **Demetri Terzopoulos** fra New York University med temaet "Artificial Animals and Humans: From Physics to Intelligence". Han presenterte mange anvendelser og åpne problemstillinger innen forskning på dette området. Dette fagfeltet omfatter at objektene behersker delområdene biomekanikk, sansing, oppførsel, læring og "cognition". Dette fører til at man kan bruke animasjoner som et slags kino for en kunstig generert natur. I løpet av foredraget ble det også presentert meget overbevisende filmeksempler der de forskjellige delaspektene ble vist.

På den andre dagen var det **Thomas Ertl** fra Universitat Stuttgart sin tur med "Interactive Visualization with Programmable Hardware". Hovedvekten lå på bruken av programmerbar grafikk-hardware i forskjellige visualiseringsalgoritmer for volum-data, terrengmodeller, NPR rendering etc.

På den siste dagen, rett før sluttseksjonen, fortalte **Roberto Scopigno** fra Consiglio Nazionale delle Ricerche i Pisa om "3D Scanning Technology: Capabilities and Issues". Hans hovedanliggende er å skanne og bearbeide historiske objekter (f.eks. statuer), slik at disse foreligger som data for ettertiden. Han gjennomgikk de forskjellige metodene og skanner-typer, bl.a. passive og aktive skanner-teknologier. Han beskrev også hvilke oppgaver som må gjennomføres for å gjenskape et objekt i datamaskinen. I løpet av foredraget ble det også vist mange eksempler fra arbeider som ble gjennomført ved instituttet i Pisa.

Presentasjoner

Det ble avholdt fem STAR sesjoner, der forskningsfronten og uløste problemstillinger for aktuelle temaer blir presentert. Undertegnede deltok på sesjonen om "Visual Data Mining" og "Tone Reproduction and Physically Based Spectral Rendering". Andre temaer var "flow fields", "interactive rendering" og "global illumination".

Hver dag ble det satt av en time, der forskjellige forskningsinstitusjoner kunne presentere sitt virke. Dessverre ble disse presentasjonene plassert i middagspausen i et ellers tett pakket program, slik at dette ikke fant så mye publikum som man kunne ønske.

På konferansen ble det også avholdt tre industrisesjoner og en panel-debatt. Undertegnede deltok på en av industrisesjonene med temaet "Computer Graphics for Games and Entertainment". Denne sesjonen ble heller noe skuffende: En av foredragsholderne dukket ikke opp, mens det andre foredraget var nok litt for generelt for at man kunne lære noe nytt. Den tredje foredragsholderen fra University of Abertay beskrev samarbeidet mellom universitet og industrien. Han gikk også inn på en del grunnleggende regler som er viktige når man designer et spill. Jeg satt litt med inntrykket av at jeg hadde hørt det meste allerede for et år siden i Manchester.

Det sosiale programmet

Det sosiale programmet er en viktig ingrediens for slike konferanser. Her knytter man nye kontakter og man vedlikeholder de gamle. Eurographics konferansen 2002 bød på mange muligheter. Allerede kvelden før konferansen fant **welcome reception** sted i slottet i Saarbrücken. Barockslottet i Saarbrücken er idag en blanding av barock og modern arkitektur pga. dens historie. Ett av innslagene var et fiktivt foredrag fra slottets opprinnelige arkitekt, Friedrich Stengel som levde fra 1694 til 1787. Dessuten var det flere taler, kanap er og en musikalsk ramme. Dessverre for musikerne var ikke publikums oppmerksomhet rettet mot musikken, da alle var mere innstilt på å utveksle nyheter med kolleger som man ikke hadde sett på lenge.

Etter den første dagen ble det arrangert vinsmaking, samtidig som bidragene som ble stilt ut i poster sesjon fikk noe oppmerksomhet. Høydepunktet på det sosiale programmet var omvisningen på Völklinger Hütte, som er et gammelt jernverk. Jernverket står på UNESCOs liste over verneverdige kulturminner. Omvisningen ble foretatt i flere grupper, der de kunnskapsrike omvisere ga oss en historisk og teknisk overblikk over jernproduksjon. Rett etter omvisningen fikk fant konferansemiddagen sted i den gamle turbinhallen.

Eurographics 2002 har vært en lærerik og vellykket konferanse også i år. Den neste konferansen finner sted 1.-6. september 2003 i Granada, Spania.

Hva er NORSIGD?

NORSIGD – Norsk samarbeid innen grafisk databehandling – ble stiftet 10. januar 1974. NORSIGD er en ikke-kommersiell forening med formål å fremme bruken av, øke interessen for, og øke kunnskapen om grafisk databehandling i Norge.

Foreningen er åpen for alle enkeltpersoner, bedrifter og institusjoner som har interesse for grafisk databehandling. NORSIGD har per januar 2002 28 institusjons- og 42 personlige og EG-medlemmer. Medlemskontingenten er 1.000 kr per år for institusjoner. Institusjonsmedlemmene er stemmeberettiget på foreningens årsmøte, og kan derigjennom påvirke bruken av foreningens midler.

Personlig medlemskap koster 250 kr per år. Personlige medlemmer får tilsendt medlemsbladet *NORSIGD Info*. Kontingenten er redusert til 150 kr ved samtidig medlemskap i vår europeiske samarbeidsorganisasjon *Eurographics*.

Alle medlemmer får tilsendt medlemsbladet *NORSIGD Info* 2-4 ganger per år. NORSIGD har tilrettelagt informasjon om foreningen på Internett på adressen <http://www.norsigd.no>. Der finnes det også informasjon om GPGS, samt tidligere utgaver av *NORSIGD Info*.

Interesseområder

NORSIGD er et forum for alle som er opptatt av grafiske brukergrensesnitt og grafisk presentasjon, uavhengig av om basisen er *The X window System*, *Microsoft Windows* eller andre systemer. NORSIGD arrangerer møter og seminarer, formidler informasjon fra internasjonale fora og distribuerer fritt tilgjengelig programvare. I tillegg formidles kontakt mellom brukere og kommersielle programvareleverandører.

NORSIGD har lang tradisjon for å støtte opp om bruk av datagrafikk. Foreningen bidrar til spredning av

informasjon ved å arrangere møter, seminarer og kurs for brukere og systemutviklere.

GPGS

GPGS er en 2D- og 3D grafisk subrutinepakke. GPGS er maskin- og utstyrsuavhengig. Det vil si at et program utviklet for et operativsystem med f.eks. bruk av plotter, kan flyttes til en annen maskin hvor plotteren er erstattet av en grafisk skjerm uten endringer i de grafiske rutinekallene. Det er definert grensesnitt for bruk av GPGS fra FORTRAN og C.

Det finnes versjoner av GPGS for en rekke forskjellige maskinplattformer, fra stormaskiner til Unix arbeidsstasjoner og PC. GPGS har drivere for over femti forskjellige typer utsyr (plottere, skjermer o.l.). GPGS støtter mange grafikkstandarder slik som Postscript, HPGL/2 og CGM. GPGS er fortsatt under utvikling og støtter stadig nye standarder.

GPGS eies av NORSIGD, og leies ut til foreningens medlemmer.

Eurographics

NORSIGD samarbeider med Eurographics. Personlige medlemmer i NORSIGD får 20 SFr rabatt på medlemskap i Eurographics, og vi formidler informasjon om aktuelle aktiviteter og arrangementer som avholdes i Eurographics-regi. Tilsvarende får Eurographics medlemmer kr 100 i rabatt på medlemskap i NORSIGD.

Eurographics ble grunnlagt i 1981 og har medlemmer over hele verden. Organisasjonen utgir et av verdens fremste fagtidsskrifter innen grafisk databehandling, *Computer Graphics Forum*. *Forum* sendes medlemmene annen hver måned. Eurographics konferansen arrangeres årlig med seminarer, utstilling, kurs og arbeidsgrupper.

NORSIGD
v/ Reidar Rekdal
Postboks 290
1301 Sandvika

Returadresse:
 NORSIGD v/ Reidar Rekdal
 Postboks 290
 1301 Sandvika

Styret i NORSIGD 2002

Funksjon	Adresse	Telefon	email
Leder	Ketil Aamnes Telenor Business Solutions PB 16517 Etterstad 0606 OSLO	22 78 05 56 (direkte) 23 16 64 70 (fax)	Ketil.Aamnes @telenor.com
Fagansvarlig	Wolfgang Leister Norsk Regnesentral Postboks 114 Blindern 0314 OSLO	22 85 25 78 (direkte) 22 85 25 00 (sentralbord) 22 69 76 60 (fax)	leister@online.no
Sekretær	Reidar Rekdal Norsigd Postboks 290 1301 Sandvika	67 57 73 18 (direkte) 67 57 72 50 (sentralbord) 67 57 72 72 (fax)	reidar.rekdal @dnv.com
Styremedlem	Gisle Fiksdal LODIC AS Pirsenteret 7462 TRONDHEIM	73 83 17 34 (direkte) 73 83 17 35 (fax)	Gisle.Fiksdal @lodic.no
Varamedlem	Svein Taksdal Norges Vassdrags- og Energiselskap Hydrologisk Avdeling, Seksjon data Postboks 5091, Majorstua 0301 OSLO	22 95 92 86 (direkte) 22 95 92 01 (fax)	svein.taksdal @nve.no
Varamedlem	Magnar Granhaug ProxyCom AS Kløbuavn. 194 7037 Trondheim	73 95 25 00 97 72 76 98 (mobil) 73 95 25 09 (fax)	Magnar.Granhaug @proxycm.no

<p>Svarkupong</p> <p><input type="radio"/> Innmelding – institusjonsmedlem (Kr 1000)</p> <p><input type="radio"/> Innmelding – personlig medlem (Kr 250)</p> <p><input type="radio"/> Innmelding – Eurographics medlem (Kr 150)</p> <p><input type="radio"/> Ny kontaktperson</p> <p><input type="radio"/> Adresseforandring</p>	<p>Navn:</p> <p>Firma:</p> <p>Gateadresse:</p> <p>.....</p> <p>Postadresse:</p> <p>.....</p> <p>Postnummer/sted:</p> <p>.....</p> <p>Telefon:</p> <p>Telefaks:</p> <p>email:</p>
---	--