

An OpenGL based stereo system for 3D-measurements

Frank Boochs, Anja Gehrhoff, Markus Neifer

i3mainz, Holzstrasse 36, 55130 Mainz, Germany

Univ. of Applied Sciences/Mainz

ABSTRACT

A stereo system designed and used for the measurement of 3D-coordinates within metric stereo image pairs will be presented. First, the motivation for the development is shown, allowing to evaluate stereo images. As the use and availability of metric images of digital type rapidly increases corresponding equipment for the measuring process is needed. Systems which have been developed up to now are either very special ones, founded on high end graphics workstations with an according pricing or simple ones with restricted measuring functionality. A new conception will be shown, avoiding special high end graphics hardware but providing the measuring functionality required. The presented stereo system is based on PC-hardware equipped with a graphic board and uses an object oriented programming technique. The specific needs of a measuring system are shown and the corresponding requirements which have to be met by the system. The key role of OpenGL is described, which supplies some elementary graphic functions, being directly supported by graphic boards and thus provides the performance needed. Further important aspects as modularity and hardware independence and their value for the solution are shown. Finally some sample functions concerned with image display and handling are presented in more detail.

Keywords: OpenGL, wxWindows, Geographic Information Systems, Digital Photogrammetry, Photogrammetric Workstations, 3D-Measurements, Stereoscopy

INTRODUCTION

The stereoscopic perception is one important feature of the human species. It is used for orientation within the 3D-space and allows to securely navigate through the surrounding environment. Since almost 100 years this capability is used to create metric descriptions of any kind of objects of interest. The way from the perception process to the 3D description leads over specialized instruments allowing to evaluate metric images. These instruments have permanently been subject to technological changes finally resulting in purely digital solutions. Together with the development on the field of personal computers it is possible to provide a technological demanding tool based on standard PC-hardware.

In the following article motivation and background for the proposed solution are outlined. Then the design of the system is sketched and some technical details are presented.

1. ASPECTS OF 3D-MEASUREMENTS

1.1. APPLICATIONS

As the world which surrounds us is three dimensional all applications dealing with our world are potential disciplines for descriptions using a certain amount of 3D points. These points then are treated as representatives for an object of our world or the world itself. Using these points one then has the opportunity to work with a digital copy of the object in question instead of handling the real object⁴. Advantages of such an approach are various, because any kind of operations on the object can be realized without the need of handling the real object. The type of manipulations achievable may range from simple geometric calculations, over any kind of physical or metrical simulations up to any sort of visualizations. At the very beginning of such an application the process of generating the 3D-model is to be found. This might be achieved by an editing or construction procedure based on any information describing the object or is has to be done by a surveying step applied to the existing object. The first way is very common in any computer based design or engineering application and in

the virtual reality (VR) community, whereas the second way is prominent, when to deal with the topography of our natural environment.

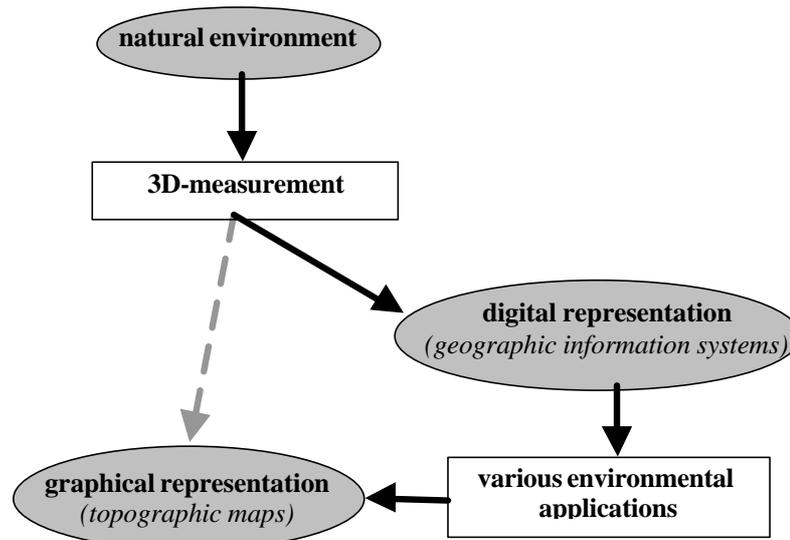


Figure 1: representations based on 3D-measurements

Registrations of our natural environment do have a long history and have found their expression in well known cartographic maps (cf. Fig.1, dotted arrow). Meanwhile the analog description has been replaced by a digital one, which is commonly called geographic information system⁵ (GIS). In such systems the topography is not only expressed by manifold 3D points but also by attributes representing any kind of information needed to describe the nature of the objects collected in the data base. Due to the digital nature of the data numerous applications are possible extending the scope of such a GIS far beyond a simple geometric description of the topographic surface. Furthermore, the technique which is developed for these purposes can be used in other disciplines dealing with geometric descriptions of objects (archeology, architecture, engineering sciences etc.).

1.2. REQUIREMENTS

Basically the establishment of information systems needs to have a measuring technique at its disposal allowing to generate precise and reliable 3D-coordinates of a huge quantity of object points representing all the objects within the area of interest. Such a technique needs to be

- fast
- accurate
- intelligent

in order to fulfil the objectives of the data collection process.

The speed is of importance, because of the amount of data required for a morphologic exact description of the objects. Secondly, quickness reduces the effect of time dependent variations of the object during the data collection process. Accuracy is required with respect to the applications planned and the quality of the final products to be prepared. In general, the accuracy of the collected data has to be several times superior to the accuracy of the product. Intelligence is needed because an information system primarily stores semantic data, which are referenced to the object by means of the 3D-points. The geometry therefore serves as a base and has inevitably to be grasped, but a powerful measuring technique additionally should provide all semantic data measurable.

In general there are two approaches ensuring the needs mentioned above:

- direct measurement of the objects
- indirect measurement, using an intermediate storage medium.

Direct measurements are useful for static objects and in cases of areas with moderate size. They are primarily to be found in engineering tasks. Indirect measurements are applied to large objects and in cases, when the object might change during the observation period. As storage medium generally serve metric images allowing to capture the actual state of the object. Later on, the collected images are evaluated.

2. PHOTOGRAMMETRIC 3D-SYSTEMS

2.1. CONCEPTS

Photogrammetry is the science and technology providing tools and tasks for indirect measurement of objects. First solutions have been presented a century ago and since then a continuous development has been going on. The principle is based on the measurement of image coordinates representing projection rays within the framework of a camera system. Using two camera stations a triangle can be formed allowing to determine an object point within the 3D-space. As the rays are expressed with respect to the camera system the triangle only can be formed if the individual images are either referenced to each other or if they are referenced to a given object system (cf. Fig. 2).

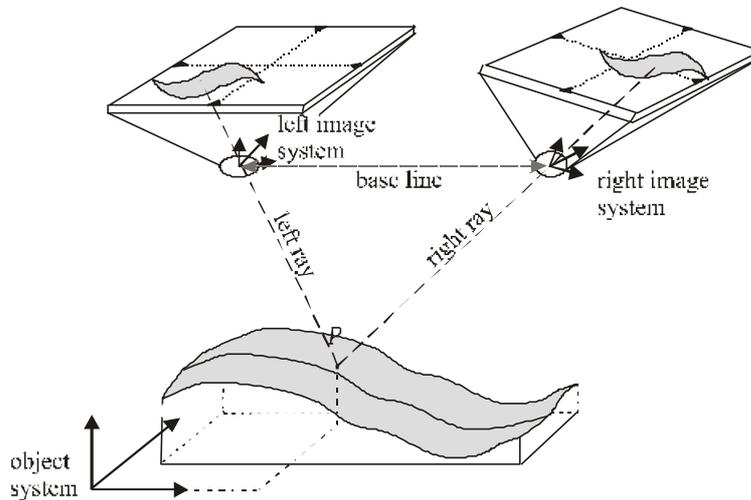


Figure2: ray intersection from metric images

The accuracy achieved with this procedure is strongly dependent on the metric quality of the components participating:

- camera geometry
- image geometry
- image orientation
- positional accuracy of measured point

A operational 3D-measurement system therefore has to assure, that:

- all information needed can be generated or introduced

- the precision of the data collected is as high as necessary
- a virtual pointing device is available (floating mark)

Furthermore aspects of an ergonomic working environment have to be considered.

2.2. SOLUTIONS

Until mid 70ies photogrammetry was performed using purely analog instruments. Then after a technological break computer controlled systems were introduced consisting of an optical-mechanical viewer working under real-time control of a microprocessor. Meanwhile another break is going on, providing purely digital instruments. These are based on metric images, which have to be available in digital form. These images are photogrammetrically processed in an appropriate computer environment.

This computer environment is adopted to the needs of a photogrammetric work flow used for the generation of geographic information systems and is typified by the demands arising from the use of large digital images. Such metric images either come from a digitization process using a photogrammetric scanner or from large airborne or spaceborne push broom scanner. They typically have sizes from 10.000 by 10.000 pixel up to 30.000 by 30.000 pixel each. Accordingly the work load for the computer system is high within all those steps, which are dynamically accessing the image data. This is the case for all measuring tasks, especially for those ones performed under human control.

Due to this situation digital photogrammetric instruments are generally consisting of powerful graphic workstations equipped with special viewer components needed for the stereoscopic perception process. Moreover the distributing companies have generally adopted there software solutions to the special characteristics of these workstations in order to maximize the performance. The capabilities of such instruments range from all necessary preparational steps over human controlled interactive data collection procedures up to automatically driven evaluation tasks.

2.3. CONSEQUENCES

The conception of the offered solutions leads to some disadvantages. The nature of these instruments causes

- high costs
- dependence on the specialized hardware and on the providing company
- low flexibility to technological changes

Most of these items are caused by the proprietary characteristic of the instruments. The developing company has to invest great effort into an own solution which has to be amortized. The customer has to rely on the provider and his capability to keep track with the technological process. Furthermore, third party solutions being available are hardly to integrate, because of the hardware dependence of most proprietary equipment.

It therefore seems very promising to have a solution avoiding hardware dependence. With respect to flexibility it is additionally necessary to have a modular and open software structure, which allows to substitute individual components as there are better methods available. Furthermore one should try to found an approach on available standards and other open resources in order to reduce the effort having to be invested.

Nevertheless, the demanding characteristic of photogrammetric evaluation processes remains and has to be considered. In order to sketch just some of the needs, the following list gives an impression of the features to be met:

- allowing images of any size and depth (typically up to 900 Mpixel, B/W and color)
- fixed floating mark with moving image
- continuous, endless image move with different speeds
- single, double image and 3D-viewing measuring mode with appropriate interactive control features

- registration of all kinds of coordinates together with semantic object information
- ergonomic environment and an agreeable, not tiring stereoscopic viewing system

3. DESIGN OF THE PROPOSED SYSTEM

3.1. GENERAL

As conclusion from the deficiencies of specialized solutions we have tried to find a modular concept based on available standards and the corresponding resources. Fortunately we can profit from the rapid development of the market of PC hardware and applications, which meanwhile even has produced components for a stereoscopic environment. Although this is driven by the game and virtual reality market it allows to choose a solution founded on standard PC components, which should be powerful enough to realize a 3D-measurement application.

In order to keep a maximum of portability, device dependent control components have been avoided. This might result in a slight loss of capacity, but that should be acceptable in comparisons to the cost of device dependency coming with specialized solutions. Stereoscopy and graphic control therefore are based on the VESA and OpenGL standards.

The aspect of portability is extended to the use of other operating systems than MIRCOSOFT™ based ones resulting in the need to have system specific software elements controlling the graphic devices. As it should not be subject to this project to develop system specialities it has been decided to incorporate the wxWindows Graphical User Interface¹. This provides all necessary routines for the installation of graphic solutions within different environments and additionally gives the corresponding look and feel.

Look and feeling is of supplementary value for the acceptance of the graphical interface by the user community. Once used to work on a PC one is in general familiar with MIRCOSOFT™ and the window technique utilized there. The acceptance of products coherently increases if the interface seems to be well-known. The design of the graphical user interface takes these aspects into account and tries to provide simple and easy to use tools.

3.2. HARDWARE

Central part of the hardware is a Pentium III CPU with 500 MHz equipped with 512 MB memory. The CPU provides standard processing speed with respect to the actual PC generation. Of importance is the equipment with sufficient main memory, because the main work load during the measuring process comes from the transportation of image data. Here a large intermediate buffer is necessary, to avoid waiting periods induced by the slow access to the mass storage. The selected amount of memory seems to be a minimum quantity required. Correspondingly the amount of mass storage should not fall

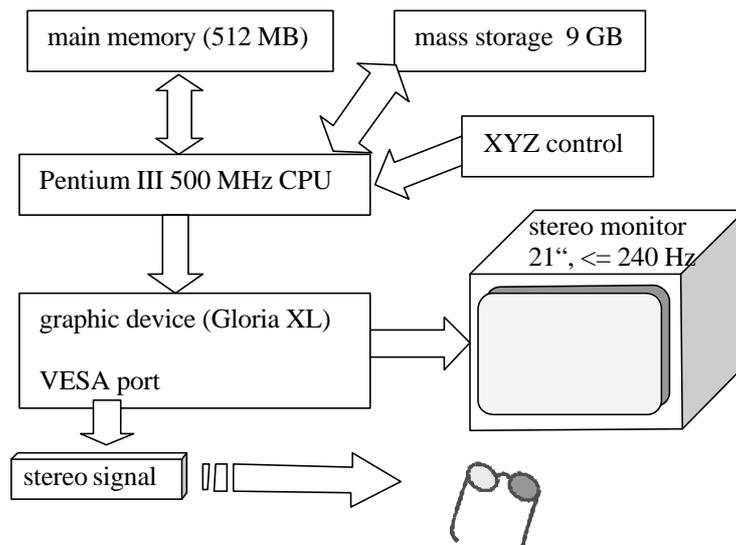


Figure 3: system components

beyond 5GB, as standard projects may require up to 3 GB storage capacity (cf. Fig. 3).

Stereoscopy is based on a powerful graphic device (ELSA Gloria XL) a stereo monitor and stereo glasses (Stereographics® CrystalEyes®) with appropriate control. The graphic device shows up some features, which are important for a 3D-measurement environment:

- high resolution (up to 1280*1024 true color mode)
- VESA interface for direct connection to stereo glasses
- hardware acceleration for several OpenGL-functions
- stereo-in-window technique
- stereo refresh rate up to 120Hz

This graphic card is one representative of a new generation of graphic boards widely supporting OpenGL functionality onboard and being designed for stereoscopic applications. Their appearance will increase in the future allowing to choose between several alternatives if a software is founded on a graphic standard like OpenGL.

Of principal interest for our solution are the direct support of stereo glasses, the high refresh rate guaranteeing flicker free stereo viewing and the stereo-in-window technique. The latter has some advantages compared to the standard above-below stereo solutions, which are splitting the screen in a top and bottom part each containing one of the stereo images. By this the whole screen is affected and can only be operated in stereo or non stereo mode avoiding to simply mix different windows on the screen. Furthermore there is a loss of 50% resolution. Both drawbacks are avoided with the stereo-in-window principle, which is possible due to the stereo mode in OpenGL. This means to have a quad buffered window, where front and back buffer are split into two pieces each referenced to one image of the stereo pair. Finally we have the hardware acceleration provided by the board, what of course is very useful, although software solutions would do their task likewise.

3.3. SOFTWARE

3.3.1. DESIGN

The software (cf. Fig. 4) has to fulfil all functional needs, but has additionally to consider requirements coming from the wish to avoid hardware dependence, to be flexible to the technological progress, to be able to exchange components and to incorporate new ones and to reduce the effort for maintenance. There is only one way to support all these wishes, that is to use an object oriented programming approach. With the final design in mind this moreover allows to realize the system step by step, starting with a simple prototype which will be extended as more elements are available. Selected language is C++,

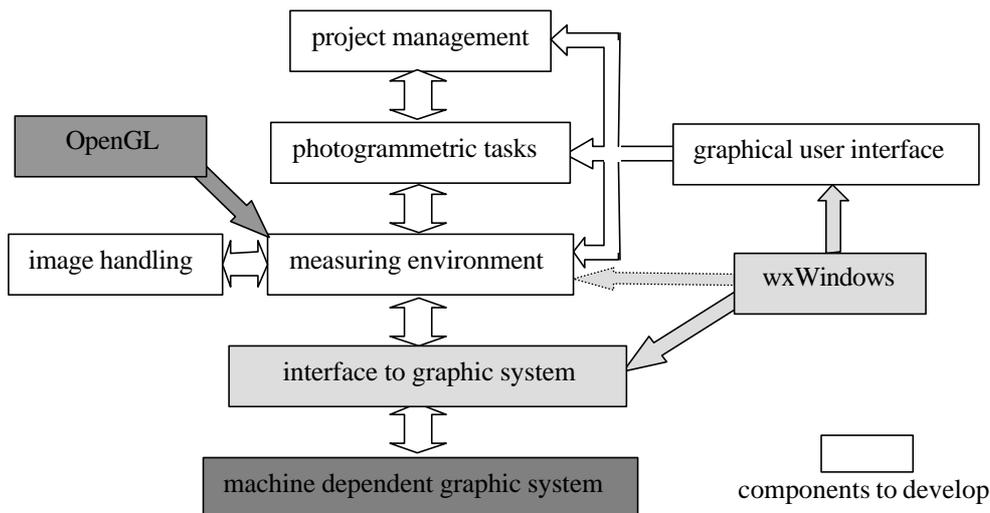


Figure 4: structure of software

because of its popular character permitting to incorporate existing solutions, which are widely available.

One very important already existing component is the graphical user interface wxWindows. It is an object oriented software solution written in C++, which is useful for generation of graphical elements and has the advantage to provide all necessary libraries needed to transfer a solution to the graphic environment given (Windows, Motif, etc.). Choosing wxWindows allows to make use of three important features: object orientation, system independence and integration of existing resources.

3.3.2. PHOTOGRAMMETRY

Photogrammetric tasks have to take care of the metric aspects within the framework of this solution. Four fields can be identified with important photogrammetric contributions:

- preparation of the stereo model
- geometric calculations during evaluation
- user support during evaluation
- support by automatic evaluation steps

Furthermore some components within the measuring environment are affected by the needs of a photogrammetric workplace. For example, photogrammetric operators expect to have certain control elements allowing to move through the 3D-space. They await to find the floating mark pointing to the object details resting fixed in the middle of the viewer. This makes it necessary to permanently move through the image space, what means to shift the image data continuously through the viewer – a challenging aspect, having the amount of image data in mind, which has to be handled.

3.3.3. GRAPHIC

As shown in Fig. 4 almost all components have a relation to graphics. Simple ones are those providing the user with appropriate tools to introduce data, to control the course of work or to be informed about results, actual process stages or any events.

The more demanding elements can be found in the context of the measuring process. Here, graphic means handling of large amounts of image data either in the context of displaying information within a viewer or with respect to manage buffers of image information, which have to keep track of the progress of activities applied to the image data. Fortunately a lot of functions are available from OpenGL, relieving from the need to deal with elementary graphic functions. In addition, the possible hardware support may accelerate activities considerably.

Just to mention some of the photogrammetry related graphic tasks it should be noticed that there different display modes necessary:

- single image and double image within preparation steps
- stereoscopic mode for 3D-evaluation
- combined display of image information and evaluation results for control and visualization purposes

All image related display modes need to have appropriate support to move through the image, to improve the display characteristics and to control the image zoom.

4. IMPORTANT FEATURES

4.1. PORTABILITY

Portability means for this application to allow installations on various platforms. Although the primary objective is to use a standard PC environment, other configurations may be useful for graphic applications as well and it therefore should be

possible to additionally amortise the development by installations on other systems (LINUX, UNIXTM, MAC, OS/2, ...). Trying this one primarily has to handle hardware and system dependent differences. In the context of this solution this is mainly the graphics environment. Even the use of some standard graphics software components like OpenGL doesn't prevent, that there remain discrepancies in the control of graphic devices (UNIX: X11, PC: WIN32). The problem might be simplified by an object oriented design of the software, allowing to separate between system and application dependent components. But even then one needs to have the appropriate control functions for the graphic devices. In this context it is extremely helpful to have software tools bringing in the device dependent components, which only have to be exchanged if a system change is intended. Exactly this performance is introduced by the wxWindows Graphical User Interface (GUI).

This framework provides an easy-to-use set of classes to develop platform independent applications in C++. It is used by a wide variety of users in commercial, education and research. It uses the native controls whenever possible. Controls not available are emulated such as tree controls under UNIX. Another main goal is that you get the full source which makes quick bug fixing possible. It's design is close to that of the Microsoft Foundation Classes (MFC) and the event handling mechanism using event tables is equal.

4.1.2. CLASSES IN WXWINDOWS

It has all GUI classes needed to build a modern ergonomic user interface. For example classes for frames, controls, toolbars and menus are included. But wxWindows does not only provide GUI classes. Various helpful classes are available for handling configuration files, display a OpenGL canvas and printing preview to name only a few. To save a program configuration the native methods are used. That is the registry under Windows and plain text files under Unix. These capabilities are useful for handling project files too.

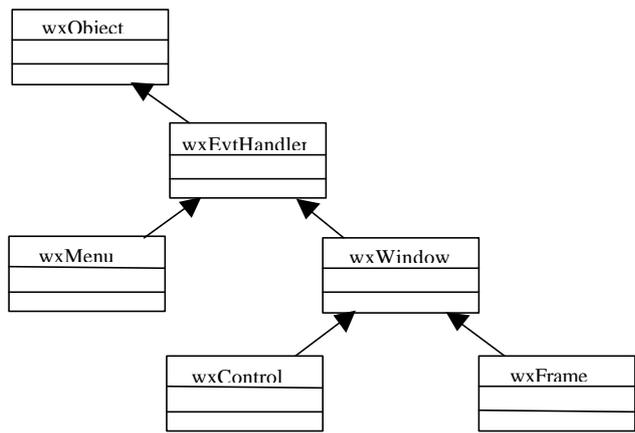


Figure 5: Simple UML showing some wxWindows classes

4.1.2. WXWINDOWS SUPPORT

The wxWindows Framework has an excellent support by mailing lists where you have direct access to the wxWindows developer team and share your problems with other users. Questions are usually answered within the day you mailed them to the list. The whole source code base is handled via GNU Concurrent Versions System (CVS) and readonly accessible for everybody. If you're interested in submitting corrections or additional features you may ask for write access.

4.2. USE OF OPENGL

OpenGL serves as the underlying graphic system for the proposed application. It handles all graphic output and the main image processing. We take full advantage of the functionality OpenGL provides but currently don't use all features. It supplies much more than just the low level pixel operations we're using here. For example texturing, lighting, fog, antialiasing and clipping are interesting features that can be used when modeling 3D virtual worlds. Maybe we use some of this advanced features in a later version of our application with extended functionality. To increase the performance of OpenGL and the application using it all this is controlled by state variables. Therefore OpenGL is often referred to as a state

machine. One doesn't need to set options every time you perform a function call. Options are set one time and the given state is used for all necessary calculations.

4.3. HANDLING OF LARGE DATA SETS

This is one of the most important features, because of the possible size of the images and the fact, that a photogrammetric operator expects to have a fix floating mark and a continuously moving image. In cases of analog instruments this does not

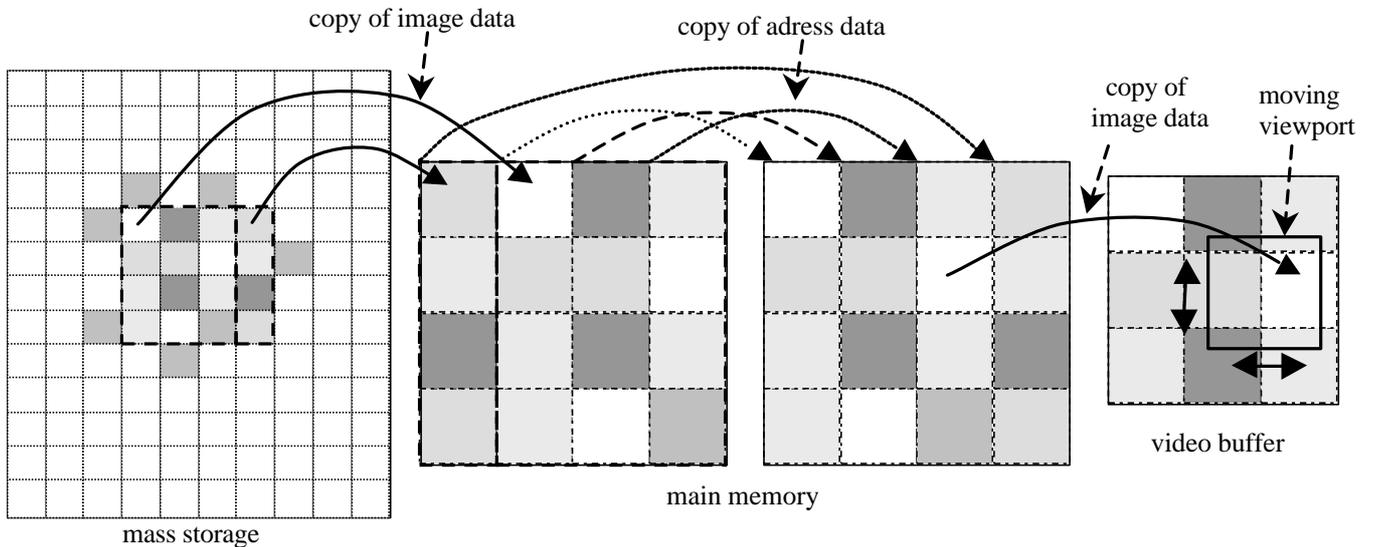


Figure 6: Three stage handling of image data

produce any major problem, but in digital systems this requires an effective handling of large data quantities.

The amount of data prohibits to think about a complete storage of the image data within the video buffer and even the main memory is much too small to take all of the image information. Consequently the mass storage is the space where to keep the data while measuring in the images. However, the transfer speed of mass storage is in-acceptable for a dynamic viewing process. It therefore has been designed a three stage solution to combine capacity with reasonable access rates (cf. Fig. 6).

To achieve speed and flexibility, the image is separated into tiles, which will dynamically be copied into appropriate memory locations. The content of this memory covers the actually needed image fragment and some additional buffer space. The memory is partly reloaded as soon as new information in the border region is required. The topology of the memory therefore doesn't correspond to the image data. This topology is reconstructed using a second memory, which serves as base for the video buffer. In order to avoid unnecessary image transfer, the content of the second memory space is created simply by copying addresses. Finally the video memory gets its data from the second buffer.

This concept allows to keep most of the image data in the mass storage and restricts the data transfer to the inevitable minimum.

5. SOLUTION DETAILS

We've already implemented some essential image handling techniques such as pan, zoom and stereoscopic display. The OpenGL Application Programming Interface (API) provides some very helpful functionality to implement these techniques. Because today's graphics cards often have built-in OpenGL support and there are some graphics cards with hardware acceleration available it makes sense to use this API for such a kind of application like we're doing here.

5.1. IMAGE PAN

With OpenGL it's possible to display bitmaps or images by using the OpenGL API function `glDrawPixels()`. Therefore this function plays a major role for image pan. It uses raw pixel data as input and writes this to a specified position in the

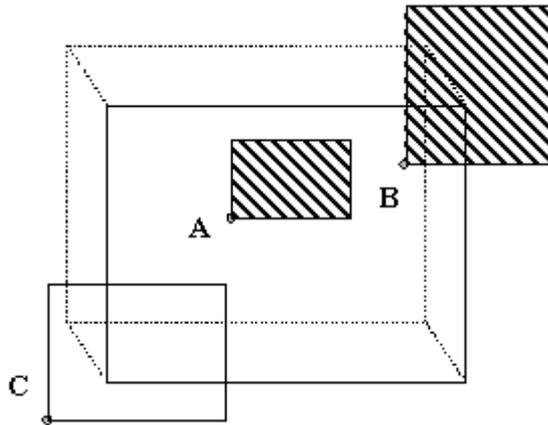


Figure 7: Various raster position scenarios¹

OpenGL virtual universe. The idea behind is first to display the image at a given position and if the user wants to move the image, the position has to be updated and the image will be displayed at the new position again. To display an image it would be enough to have one buffer. But in cases of animation like the image pan here one needs to have two buffers.

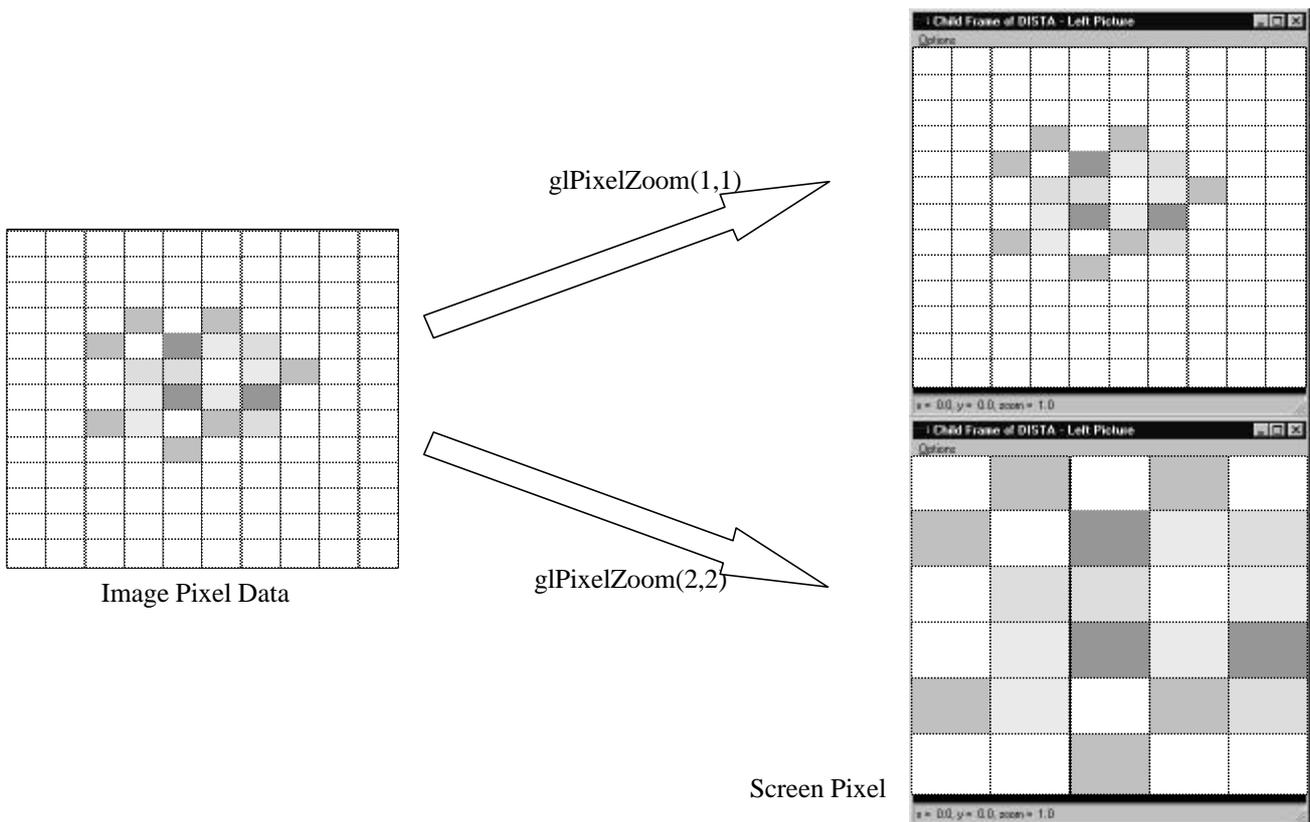


Figure 8: Image Zoom

OpenGL calls them front and back buffer. The front buffer is displayed at the screen. The back buffer is written in the meantime and afterwards front and back buffer have to be swapped. The content of the back buffer is then displayed. If the user now presses one specified key to move the image, the position has to be updated, the image then is to draw at the new position into the back buffer and finally the buffer are to swap again. Then image is updated.

Figure 7 shows some various raster position scenarios. A: raster position is within the view frustum and the image is totally within the viewport. B: raster position is within the view frustum but the image is only partially within the viewport; still fragments are generated outside the viewport. C: raster position is invalid (due to being placed outside the view frustum); no pixels are rasterized¹.

To avoid the invalid raster positions the following technique could be used¹: First setup the raster position to the origin with a call to `glRasterPos*()`. Secondly display an zero-width and zero-height bitmap at the current raster position with a call to `glBitmap()`. Within this call it's possible to set an offset with relative coordinates for the new raster position. This new raster position may now lay outside the view frustum but an image drawn at this position will render completely.

5.2. IMAGE ZOOM

The zoom functionality is implemented using the OpenGL API function `glPixelZoom()`. If images are displayed using `glDrawPixel()` one pixel of the given pixel data corresponds to one pixel on the screen. The `glPixelZoom()` function now lets to specify a factor for the x and y direction to change this correspondence between data pixels and screen pixels. For example one may specify a factor of 2 for both directions: Now one data pixel is drawn using four pixels on the screen.

With this API function call it's possible to avoid calculating pyramid layers of different zoom states increasing the amount of data necessary to display the image.

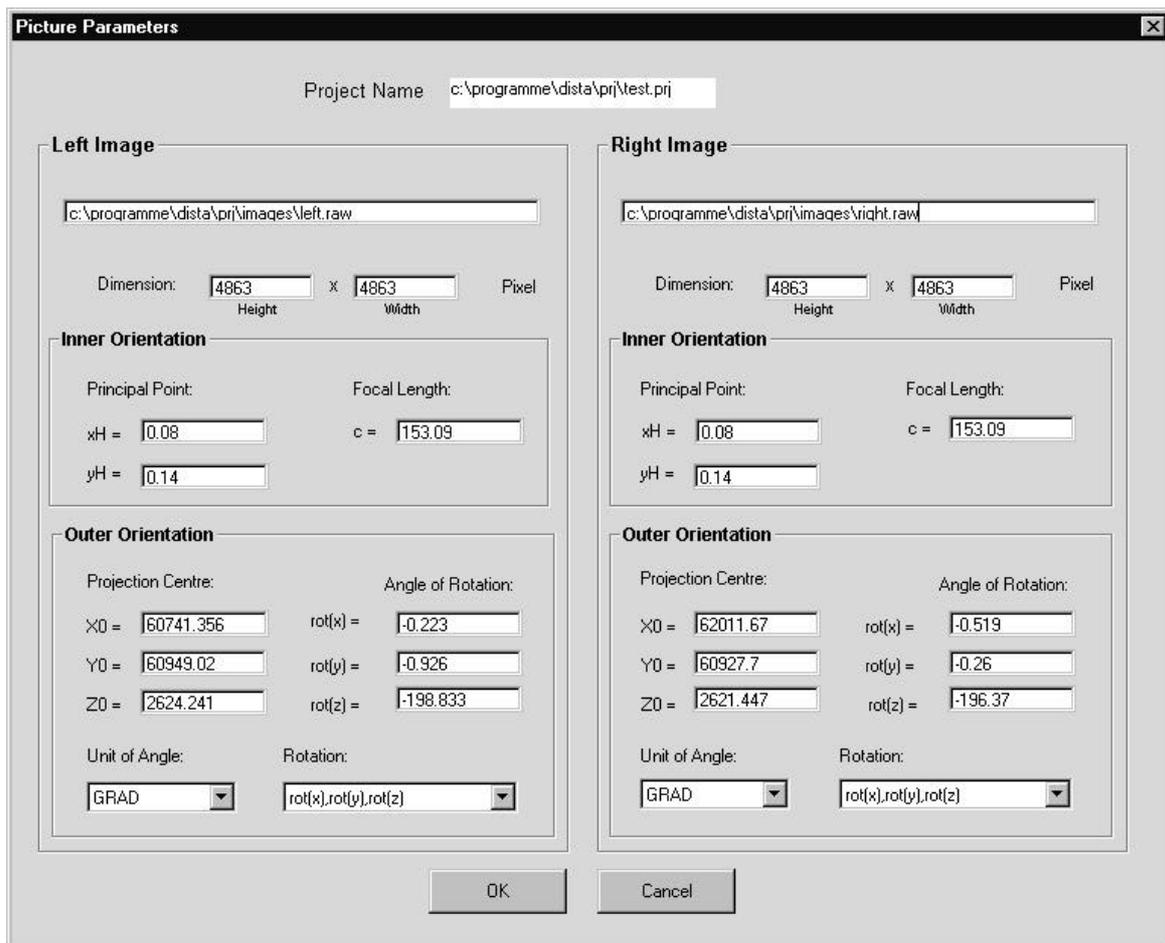


Figure 9: Picture Parameters Dialog

5.3. STEREOSCOPIC DISPLAY

Because OpenGL already comes with build-in stereo support it's relative straightforward to use this in an applications. In chapter 'Image Pan' is already mentioned how to move an image using front and back buffer. In addition to these usually two buffers in stereo mode are needed four buffers. Front buffer left and right and back buffer left and right. To use this buffer constellation one has to request a pixel display format that supports stereo buffers. Not every pixel display format supports this four buffers. It depends on the graphics card memory and graphics card driver on what resolutions and refresh rates one is able to request stereo buffers in an application. Another aspect is to take care of the offset between the two images that creates the stereo effect.

5.4. PROJECT HANDLING

The handling of the project is modular and driven by the data objects existing and the activities applied to them. The very basic data objects are the digital images, which simply have to be available. This is assured if there exists a valuable data set. Next, these objects have to be transformed to metric ones, then the metric ones have to be referenced to the object space. The way from one phase to the next is accompanied by introduction of reference data and by some measuring processes, allowing to calculate the geometric parameters required. In this context, single and double image viewer are necessary in order to get image measurements. The geometry parameters calculated are stored as attribute values and can be checked interactively (cf. Fig. 9). In addition, the windows used to present the results can also be used to directly introduce the parameters if they are known beforehand.

When all necessary geometry parameters are available, the 3D-measurement process starts. This is organized in object oriented manner too. Each object has an identity specified by a unique number allowing unambiguously to access the object. Beside of the co-ordinates, each object possesses attribute values characterizing it and permitting to specify its type. The objects are organized in categories, giving the opportunity to group the data collected and to qualitatively access the data.

REFERENCES

1. J. Smart, *wxWindows Documentation*, 1999.
2. Mark J. Kilgard, *Avoiding 16 Common OpenGL Pitfalls*, 1999.
3. T. Davis, J. Neider, M. Woo, *OpenGL Programming Guide*, Addison-Wesley, New York, 1993.
4. Blau E., Boochs F., Schulz B.S.: *Digital Landscape Model for Europe (DLME)*, OEEPE, Official Publication No. 34, 1997
5. Boochs F.: *Photogrammetry, Script of corresponding lecture*, 1998