Image Processing in a Network Environment

R. Müller, P. Reinartz, G. Kritikos, M. Schroeder

Deutsche Forschungsanstalt für Luft- und Raumfahrt (DLR)
Institut für Optoelektronik
Post Weßling/Obb.
D-8031 Oberpfaffenhofen, Germany

Abstract

XDibias is a new image processing system developed at the DLR Institute of Optoelectronics.
The user interface based on XWindow allows easy use. The communication protocol RPC (Remote Procedure Call) provides the network service and is specified by XDR (External Data Representation). The administration of the objects (images and related data, modules, users) is managed by the database system db_Vista.
The system features are:

- object oriented user interface
- network distributed data and functions
- virtual image display with integrated graphic editor
- image administration in a network
- archive tool with preview possibility

The mentioned system includes well-proven image processing modules for enhancement, geometric and radiometric corrections, filter operations, supervised and unsupervised algorithms, stereo evaluation, image transformation.

Keywords: Image Processing, Remote Sensing, Image Analyses,

1. Introduction

After twelve years of operation the image processing system DIBIAS II, running on a mainframe, had come to its limits. Modern and up-to-date applications require easy interaction techniques, decentralised data distribution, portability and flexibility in regard to different hardware platforms.

The succession system XDibias (Extended Digital Interactive Bavarian Image Analysis System) is a entirely new third generation system with emphasis on network capability and administration of data. It also contains an archive tool. It offers well-tried algorithms and methods to solve different problems within the scope of remote sensing. The technique of interaction is designed in a manner, that even non experts can productively work after a short initial period.

XDibias profits from the 20 years of experience with the two previous systems DIBIAS I and DIBIAS II. Priority in application of both systems was concentrated on environmental research and ecological mapping, especially to develop processing algorithms and evaluation procedures. Databases were raster images obtained from air and satellite borne sensors.

1969 - 1973	first developments	
1973 - 1981	DIBIAS I	FMP Airborne Remote Sensing Program (*BENDIX)*
1979 - 1991	DIBIAS II	thermal mapping, forest desease, water colour (*DAEDALUS, CZCS, TM, AVHRR)*
1991 - 20??	X-DIBIAS	biosphere research, tropical rain forest, stereo (*MOMS-02, ROSIS, TM, SPOT, DAEDALUS)*

At the moment SUN workstations with 8 and 24 Bit Display (SUN 4/390, IPC, SUN 2) are used as hardware platforms. The transport layer of the Ethernet network is TCP/IP.

This paper describes the new image processing system XDibias. Less emphasis is given to the hardware configuration and image processing algorithms.

2. Basic Concept and Realisation

The basic concept of the image processing system XDibias is the modular software structure. It is designed to be very adaptable to fulfil the manifold tasks within the scope of projects in a research environment. New algorithms can easily be integrated in the system. It also contains GIS functions (Geographic Information System) to process vector data, to superimpose and to connect different types of information. A lot of interfaces to other image processing systems and data formats allow the use of the resources of other image processing and Geo Information systems. A table of import and export filters is shown below.

Image Processing Systems **Geo Information Systems**	*ERDAS, SUNVISION, GRASS4.0*
Data Formats	*PBMPLUS Toolkit for converting various images*

The fast development of soft- and hardware requires flexible and portable software. Therefore XDibias is realised using high levels of software standards and tools.

Operating System	*Unix BSD (SunOS 4.1.x)*
Dialogue System	*OPEN LOOK (XView,XLib)*
Compiler	*C (Kernighan-Richie)* *Fortran 77*
Network Service	*RPC, XDR*
Database System	*db_Vista III (network development database)* *XResource Manager*

The interaction technique is based on the concept which is known as 'direct manipulation'. This kind of 'man-machine' communication is very introduced (use of the OPEN LOOK Application Style Guidelines). It minimises the learning phase as well as incorrect user inputs. For the dialogue system the X-Window interface was chosen. This standard guarantees also the portability in regard to different display hardware components.
In addition to the object oriented user interface there is a command driven language similar to the UNIX syntax. This kind of input is necessary to write batch jobs (scripts).

XDibias consists of three thematic parts:

- administration and information system
- image processing modules
- image display and highly interactive processing

2.1 Administration

An important feature is the management of the data in a network environment. The decentralised concept of workstations also requires the distribution of data and functions in a software system. It supports the distribution on object level. Objects, in the sense of XDibias, can be pictorial data, geometric figures, processing history, image attributes, image processing modules and so on. The advantage of the data distribution is

- high efficiency in storing and processing data and
- higher error tolerance and availability.

The network service is based on the communication protocol RPC (Remote Procedure Call) and is specified using XDR (External Data Representation). Consequently the data are interchangeable in a heterogeneous network (different computers, different operating systems). The data structure is specified to send tables, linked lists and tables of linked lists over the network.

There are two kinds of objects, local and global. Global objects, for example images, are distributed in the network and physically stored on various media. The user does not know the physical location of the global object. A database server supervises all the names of the global objects. The server itself is implemented as a global object but with a predefined address. The contents of the database are:

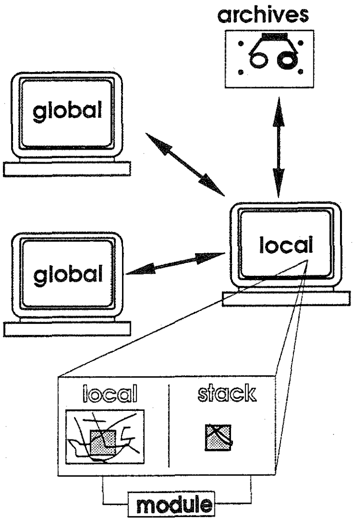
- information on global and archived images
- information on modules
- list of users, accounting
- error text
- look up tables
- user specific data like special sets of module parameters

The principle of addressing global objects in the network is similar to the Internet-Routing.

2.1.2 Working Areas

Every user owns four working areas.

• archives	(external storage media)
• global area	(network resources)
• local area	(user's work space, home directory)
• stack area	(input list for image processing modules)



The global area contains all configured network nodes with the different direct access storage media. In opposition to the data archives, that should be considered for long time storage of image objects, the network area is a "pool" for fast data availability. The advantage of the global area is

- fast image access in the network
- use of the storage capacity of all workstations
- combination of users in a project

The image list of the stack area is the input for image processing modules. It contains only references to the images in the local area. An image processing module "sees" only the given subimage of the input picture, defined in the stack area. The subimage selection is part of the base system.

An archive is necessary because the storage consuming images can occupy all resources after a short time. Different storage media for the archive are supported like optical disc (read/write), streamer tape, Exabyte and 9-track-tape. The image information including the quick-look of the picture is stored in the database and can be retrieved by the information system.

During a session the user always has the possibility to copy global objects to the local workstation and transform them to local objects. This implies better performance of image processing by discharging the network for managing temporary results.

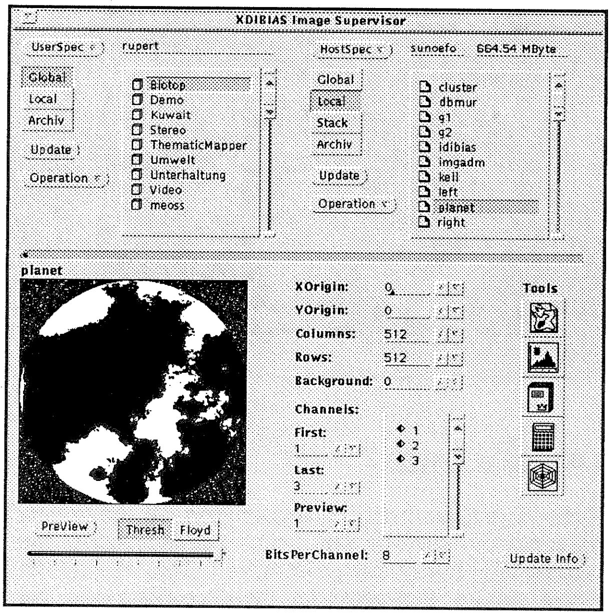
The philosophy of a XDibias session requires the following steps:

1. Copy global image objects to the local workstation
2. Do the desired evaluation of data by image processing methods
3. Distribute the results in the network
4. Delete temporary results on the local workstation.

This scheme guarantees the availability of all data on every workstation.

2.1.3 Information and Selection

XDibias contains a information and selection unit for images and processing modules. Images are assigned to projects and like directories the user can build cascades of sub directories. Information about the actually selected picture of the chosen working area is shown on the screen and subimage definition can be made interactively.



XDIBIAS Image Supervisor

UserSpec: rupert

HostSpec: sunoefo 664.54 MByte

planet

XOrigin: 0

YOrigin: 0

Columns: 512

Rows: 512

Background: 0

Channels:

First: 1

Last: 3

Preview: 1

BitsPerChannel: 8

The following information about the actual selected image is displayed:

- image geometry (rows, columns, channels, pixel quantization)
- quick look of image (preview)
- image attributes (geometric figures, vector data, look up tables, training areas, control points).
- processing history of the image
- image characterising values (histogram, minimum-maximum,entropy,anistropy)

The image processing modules are arranged in a thematic list and are displayed on the screen.

An on-line help function is integrated and a detailed description of every dialogue component and image processing module is provided for the user on request.

2.2 Image Processing

The basic component of XDibias, the image supervisor or image administrator, is the starting point of image processing. It allows image selection, subimage definition, module selection, module activation and so on. There are two kinds of image processing within the system:

- image presentation and highly interactive image processing
- algorithms or image processing modules.

2.2.1 Image Processing Modules

The image processing programs are independent tasks (applications). The connection to other system components is realised using high level functions and inter process communication facilities. The advantages of a modular structure are on the one hand an ease of implementation of new algorithms and on the other hand the capability of batch operation jobs (including all the control mechanism and replacement features of shell-scripts).

Modules are running under control of the base system, therefore the status of an active program is displayed (how much percentage of an image is written or read), the error handling is standardised and the termination of the program is supervised.

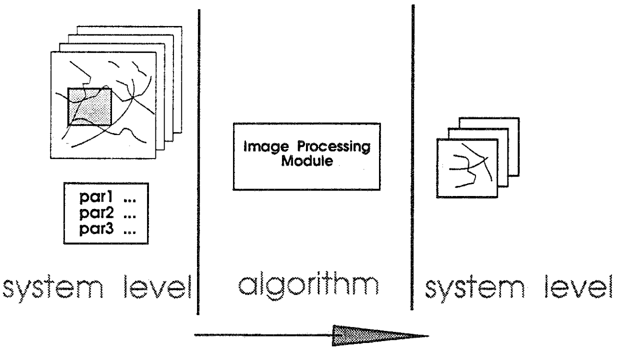
At the moment there are 70 image processing algorithms available. A table of themes and examples is given below.

input and output of images	input from tape, hard copy, 3-d plot, stereo display of images
basic operations	arithmetic connections, merge, logical operations
geometric corrections	rectification of locally distorted images, affine transformation, rotation, panorama correction
radiometric correction and transformations	calibration, atmospheric correction, radiometric Off-Nadir correction
multispectral classification	Maximum-Likelihood, K-Means, Minimum-Distance, statistical cluster analyses, unsupervised classification
image transformation	principal components, Fourier transformation, texture parameters
filter operation	different filters in the frequency and spatial domain
stereo processing	image correlation
neuronal network	associative memory, synergetic
special programs	artificial images, annotation and decoration of images

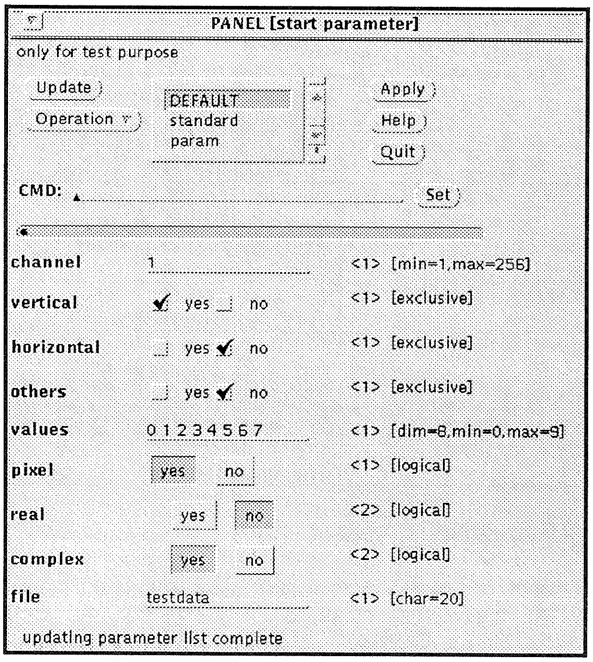
2.2.2 Program Interface

A main feature is the ease of implementation of new image processing modules. The system contains interfaces for C and FORTRAN 77 languages. To program a module a proper algorithm has to be written without parameter input (they are handed over in a Common-Block or C-Structure), and a formal parameter description has to be entered into the database. High level functions fetch the module parameter, open and create images or read and write images (an automatic transformation of pixel quantization and subimage selection is done by high level functions).

An image processing program (module) generally needs as input an image, parameters for the algorithm and the name of an output image to store the result. These input parameters are set on system level. The image processing module only contains the inner loop of the algorithm and parameter input is done by the base system in a standardised way.



Module parameters are optional and pre-set by default values. Sets of user specific values can be stored in the database for later use. Module parameters are specific to the module, but XDibias provides standardised input presentation and rules for manipulation of these parameters. The time consuming work to write a user interface for the module parameters is no longer necessary. A generator automatically produces an input panel for the module parameters using a formal description (the programmer need not know anything about XView or other related libraries).



PANEL [start parameter]

only for test purpose

CMD:

channel 1 <1> [min=1,max=256]

vertical ✓ yes □ no <1> [exclusive]

horizontal □ yes ✓ no <1> [exclusive]

others □ yes ✓ no <1> [exclusive]

values 0 1 2 3 4 5 6 7 <1> [dim=8,min=0,max=9]

pixel [✓ yes] no <1> [logical]

real yes [✓ no] <2> [logical]

complex [✓ yes] no <2> [logical]

file testdata <1> [char=20]

updating parameter list complete

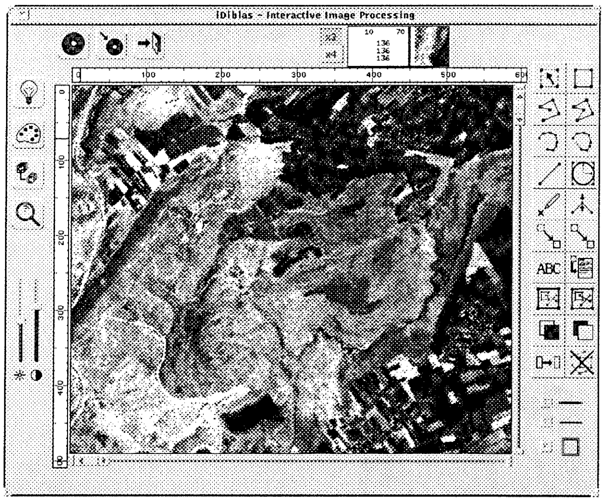
All parameters are transferred to the algorithm after checking the rules and ranges given by the module programmer. Different data types of module parameters are availible like logical, logical alternative, integer, real, integer vector, real vector and character string ones.

The library contains functions to read subimages (defined on system level) and write the result to disc. The pixel quantization of the input image is automatically translated to the modules requested quantization depth. The pixel quantization of the output image (normally the same as the input image) is provided by high level functions. Pixels with 8, 16, 32 and 64 bits are supported.

2.3 Image Display and Interactive processing

Every image processing system needs features like

- display of images
- measure tools and definition of graphical objects
- manipulation of images



2.3.1 Image Display

Pictures greater than the screen can be loaded for presentation whereby the actual display area of the picture can continuously be moved. That is why the whole picture is present for measurements and the original spatial resolution is kept. It is no need for reloading subimages. To improve the performance, especially for 24-Bit display systems, special coding was applied (for example: shared memories), but to remain independent from the hardware high level functions and X-Extensions are used. At the moment frame buffers with 8, 24+1 and 24+8 bit planes are supported.

2.3.2 Graphic Objects

A image consists of pictorial information, textual descriptive attributes and related graphic objects. Graphic objects can be drawn and edited and multistage, textual information can be attached to them. The following graphic objects are supported

- rectangle
- open and closed polygons
- circles
- lines
- points

The graphic editor contains functions like

- move
- copy
- change
- group
- delete
- ungroup

The use of graphic objects depends on the special evaluation of data, for example

- training areas for classification
- control points for geometric rectification
- calculation of statistics
- points for image matching
- overlay of vector data

All the graphic is stored as vector data connected to the pictorial information. Images are extensible. this means that a lot of data types can be attached to the image (stored in ASCII representation) using XResource manager and XDibias high level routines.

2.3.3. Image Manipulation

The interactive work with images requires basic functions to manipulate displayed pictures, to perform transformations or calculations. Some functions are listed below:

- gamma correction
- channel assignment to RGB display
- zoom
- look-up table editor
- pseudo colour manipulation
- image freeze
- plot of displayed image
- image annotation
- arrangement of images
- 3d display (DTM)
- signatures of points or areas

3. Acknowledgements

XDibias was designed and realised at the DLR, Institute of Optoelectronics, together with the software company Max Fruth. We would like to thank Max Fruth and Axel Bauer for the good team work. We would also like to thank Manfred Lehner and Jürgen Schulz for implementation of modules.