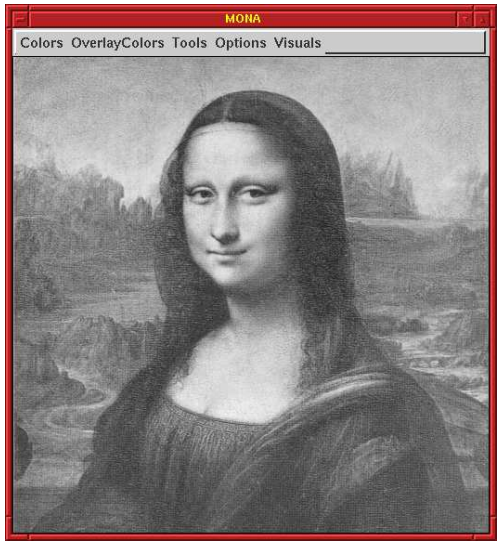
Image Processing Laboratory
Department of Informatics
University of Oslo

Report No. 91



XITE

X-based Image Processing Tools and Environment

System Administrator's Manual

For version 3.3

Svein Bøe

August 1997

Bildebehandlingslaboratoriet
Institutt for informatikk
Universitetet i Oslo

Image Processing Laboratory
Department of Informatics
University of Oslo

Tittel/Title:

XITE System Administrator's Manual

Forfatter(e)/Author(s):

Svein Bøe

Rapport nr./Report no.: **91** ISBN: **82-7476-099-9** Dato/Date: **August 1997**

Resymé/Abstract:

**XITE consists of X11-based display programs with image widget and graphical user interface as well as more than 200 command line programs and 600 subroutines for image processing, all documented on-line.
The display programs work with images of arbitrary size and pixel type on 8-bit PseudoColor and 24-bit DirectColor and TrueColor displays. Images can be zoomed and panned, and colortables can be selected from a menu. The main display program, xshow, gives access to most of the other programs via a menu interface which the user can customize and extend to include local programs. Input images for the menu entries can be selected with the mouse, and output images appear on the display.
This report describes how to install XITE and how to prepare a UNIX account for the software.**

Norske emneord/Indexing terms - Norwegian:

Bildebehandling
Bildebegrep
Bildeprogram
Vindussystemet X
Farger
UNIX
C

Engelske emneord/Indexing terms - English:

Image Processing
Image Concept
Display Program
X Window System
Colors
UNIX
C

Adresse:

Bildebehandlingslaboratoriet
Institutt for informatikk
Universitetet i Oslo
Boks 1080 Blindern
0316 Oslo

epost: blab@ifi.uio.no
tlf: 22 85 24 10

Address:

Image Processing Laboratory
Department of Informatics
University of Oslo
P. O. Box 1080 Blindern
N – 0316 Oslo
NORWAY

email: blab@ifi.uio.no
phone: +47 22 85 24 10

XITE

X-based Image processing Tools and Environment

System Administrator's Manual

For version 3.3

Image Processing Laboratory
Department of Informatics
University of Oslo

Svein Bøe
August 1997

Contents

1 Introduction **1**
1.1 Required disk space . . . 1
1.2 Root privileges . . . 1
1.3 What you need besides XITE . . . 1
1.4 About this report . . . 2

2 Installing XITE **3**
2.1 Installing XITE for multiple architectures . . . 3

3 Customizing the installation **4**
3.1 Building a manual page database . . . 4
3.2 X `app-defaults` resource files . . . 4
3.3 WWW browser as XITE help program . . . 4
3.4 Setup of user accounts . . . 5
3.5 Customization of the xshow menu tree . . . 5
3.6 Compilation of XITE dependent software . . . 5

4 Miscellaneous **6**
4.1 Bug reports . . . 6
4.2 Registration . . . 6
4.3 Copyright . . . 6
4.4 Mailing list . . . 6
4.5 WWW home page for XITE . . . 6
4.6 How to contribute to XITE . . . 6

A Details on the installation procedure **8**
A.1 XITE home directory . . . 8
A.2 Unpacking, building and installing . . . 8
A.2.1 What you need before unpacking . . . 8
A.2.2 Unpacking . . . 9
A.2.3 Checking for errors . . . 10
A.2.4 Rebuilding of parts which failed . . . 10
A.3 Building XITE on a different platform . . . 10

References **12**

1 Introduction

XITE (pronounced *excite*) is an acronym for "X-based Image processing Tools and Environment". The display programs are based on the X Window System, Version 11, currently using release 5 or 6, operating in a UNIX environment. It works with System V as well as BSD versions of UNIX. It is – at least – being used on the following platforms

- Sun sparc (SunOS 4.1.3_U1 and Solaris 2.5)
- Silicon Graphics (IRIX 5.3)
- DEC alpha (OSF1 V4.0)
- Linux
- IBM (RS 6000, AIX)
- DECstation (ULTRIX 4.4)

The XITE distribution consists of C source code, documentation, some images and colortables and installation scripts. During the installation procedure, the XITE administrator may choose where to locate object libraries, executables, manual pages and other documentation.

1.1 Required disk space

Estimated disk space requirements

Source code:	$\approx$ 8 Mb
Manual pages:	$\approx$ 3.5 Mb
Hypertext doc:	$\approx$ 2 Mb
Scripts, images etc:	$\approx$ 4 Mb
Libraries and executables:	20–34 Mb

The space required for the libraries and executables is strongly dependent on computer architecture. For ULTRIX, 75 Mb is needed.

1.2 Root privileges

Depending on how you choose to install XITE, you may need `root` privileges for the following

- Creation of XITE home directory.
- Creation of directories for XITE C header files, libraries, executables, man pages, hypertext reference manual and other documentation.

1.3 What you need besides XITE

If you want to be able to convert images between the XITE file format BIFF (Blab Image File Format) and other popular formats such as TIFF, `pnm` (`pbm`/`pgm`/`ppm`) and MATLAB's `.mat`-files, you will need the `libtiff` TIFF library, `netpbm` or `pbmplus` and MATLAB.

To build the TIFF-to-BIFF converters you need an ANSI C compiler, such as `gcc`.

The XITE display programs need the X Window System, Version 11, release 5 or 6, a Pseudo-Color 8-plane (preferrably) or 24-plane DirectColor display. It should also work with a TrueColor 24-plane display, although the colortables can not be manipulated on such a display. Modern workstations often offer all these color modes in a single display.

1.4 About this report

This report describes how to

- build the object libraries and executables
- build hypertext and man page documentation
- hook a WWW browser to the *Help* button in the main display program
- prepare user accounts for access to XITE

Refer your XITE users to the User's Manual [2] for information on how to use XITE. The Programmer's Manual [1] describes how to write XITE dependent functions and programs, including the use of the X toolkit ximage, supplied with XITE.

XITE has also been described in [3, 4, 5].

2 Installing XITE

Go through the following three steps in order to install XITE.

1. Make an XITE home directory and set the environment variable

   `XITE_HOME`

   to reflect your choice. If you need more information about this, please refer to appendix A.1.

2. Move the script `install_xite` and the file `xite.tar.gz` (or `xite.tar.Z` or `xite.tar`) into the XITE home directory.

3. From the XITE home directory, issue the command (where `%` means the shell prompt)

   ```
   % sh install_xite
   ```

   If you need more information about the install script and how to check for installation errors, please refer to appendix A.2.

2.1 Installing XITE for multiple architectures

If you decide to make a complete XITE installation on a different platform, then rerun the script `install_xite` as explained in section 2.

If you simply want to install binary executables, object libraries and possibly include files, manual pages and hypertext documentation for a different platform in the same networked file system, while not duplicating the source files and scripts, there is no need to unpack XITE again. Instead, use the command

```
% sh install_xite -d
```

You may want to check out the `-p` option. If you need more information on this, please refer to appendix A.3.

3 Customizing the installation

After having installed XITE successfully, there are a few remaining steps which you need to go through manually, in order to make XITE operate at its best.

3.1 Building a manual page database

The XITE manual pages have been installed in a directory which you chose during the unpacking procedure. Depending on the Unix system you're on, you should probably go to this directory and build a `whatis` or `windex` database for the `man` pages. This will enable keyword search with `man` to work. On my local systems I use one of the two commands

```
% makewhatis $XITE_HOME/man
% catman -M $XITE_HOME/man
```

This may be different on your system.

3.2 X `app-defaults` resource files

The XITE display programs will not work properly without the `app-defaults` files which are found in the directory

```
$XITE_HOME/etc/app-defaults
```

A warning will be printed if the display programs can't find these files.

Each display program sets the environment variable `XAPPLRESDIR`[1] so that the `app-defaults` files can be found by the X Window System. The files are not installed into the standard location[2].

If you already have a previous version of XITE, it may be that the resource files for a previous XITE version were installed in the standard location[2]. You should remove those files.

See documentation on the X Window System for more information about the above environment variables.

3.3 WWW browser as XITE help program

xshow tries to load an external helper application when the *Help* button in the Control window is pushed. The default behaviour is to start the WWW browser `netscape`, pointing to the XITE Reference Manual (which was built during the installation procedure). If `netscape` is not available, xshow will usually try to launch another program. The exact behaviour is determined by the X resources `xiteHelper` and `xiteHelperOption` or the environment variables `XITE_HELPER` and `XITE_HELPER_OPTION`.

If you want to customize the behaviour, refer to XITE User's Manual [2] or the manual page for xshow.

[1] possibly also `XUSERFILESEARCHPATH`

[2] `/usr/lib/X11/app-defaults` (or `/usr/openwin/lib/app-defaults` under the Sun Solaris operating system).

3.4 Setup of user accounts

The installation procedure also produced setup files for XITE users. The easiest way for a user to get access to XITE is to include the following line in his/her `.cshrc` file

```
source $XITE_HOME/etc/xite_cshrc
```

(where `$XITE_HOME` is replaced by the actual XITE home directory).

As the XITE administrator you should check that the file `xite_cshrc` reflects the choices made during installation.

3.5 Customization of the xshow menu tree

The main display program xshow can launch most of the other programs supplied with XITE. This is done via a menu hierarchy which is defined in an ascii menu file. One of the entries in the main menu is *Site-specific programs...*. If you want all your users to get access to locally developed programs which operate on images in the XITE file format BIFF, these programs may appear in a submenu tree below this entry.

For information on how to add local programs to the menu hierarchy, refer to the manual page for xshow.

3.6 Compilation of XITE dependent software

A script, cxite, is supplied for compilation and linking of new programs which use XITE routines. It was installed in the XITE executable directory you chose during installation. The script was automatically edited by the install script and should be able to supply the necessary options to find XITE header files, the XITE library as well as X Window System header files and libraries.

The command

```
% cxite -help
```

will give more information about this script.

4 Miscellaneous

4.1 Bug reports

If (or when) you encounter bugs, whether during installation or execution, please send bug reports to the XITE developers. You'll find a bug report template in the file

```
$XITE_HOME/etc/.install/bugReport
```

Mail the bug report to *xite-bugs@ifi.uio.no*.

4.2 Registration

If you install and use XITE, please let us know by filling in the registration form in the file

```
$XITE_HOME/etc/.install/REGISTER
```

Mail it to *xite-register@ifi.uio.no*. This will help us get a general idea of what people find useful, what platforms XITE is most widely used on and whether the users are in universities, research institutions etc.

The use of XITE is free, but we would be happy to know that you find it useful.

4.3 Copyright

XITE comes with a mild copyright notice. You'll find a copy of this in the file

```
$XITE_HOME/etc/.install/COPYRIGHT
```

4.4 Mailing list

An international mailing list exists for discussions about XITE. If you would like to subscribe to this list, send a mail to *xite-request@ifi.uio.no*. The list name is

xite@ifi.uio.no

4.5 WWW home page for XITE

If you have a World Wide Web browser, such as Mosaic, netscape or lynx, you may find more information about XITE in the XITE WWW home page with URL

```
http://www.ifi.uio.no/~blab/Software/Xite/
```

4.6 How to contribute to XITE

If any of your users writes software which she would like to contribute to XITE, refer her to the Programmer's manual [1] supplied with the XITE distribution. Please email *blab@ifi.uio.no* if further instructions are needed.

If anyone at your site makes changes or fixes bugs, please let us know. Refer to section 4.1 for details on sending a bugreport.

A Details on the installation procedure

If you ran into problems during the installation procedure which was briefly outlined in section 2, you may find help here.

A.1 XITE home directory

All the C source code and installation scripts will be placed below the XITE home directory. Typical locations are `/usr/local/xite`, `/usr/lib/xite` or the user `~xite`.

The object library, executables, manual pages and other documentation do not need to reside below the XITE home directory. This can be chosen during the installation procedure.

Set the environment variable `XITE_HOME` equal to the chosen XITE home directory. If you are using `csh` or `tcsh`, this can be done with the command

```
% setenv XITE_HOME /usr/local/xite
```

(if you chose `/usr/local/xite` as the XITE home directory). If you are using the `sh` shell, use the command

```
% XITE_HOME=/usr/local/xite; export XITE_HOME
```

Throughout this document the string `$XITE_HOME` represents the chosen XITE home directory.

Before unpacking, building and installing XITE, the home directory must be created. If you have chosen `/usr/local/xite` as the XITE home directory, issue the commands

```
% cd /usr/local
% mkdir xite
```

You will probably need `root` privileges to be allowed to create the directory. You could, of course, alternatively have chosen to install XITE below your own home directory, in which case you would not need any special privileges at this stage.

A.2 Unpacking, building and installing

A.2.1 What you need before unpacking

Before you start to build and install XITE, make sure that you have everything you need. You should have the following files

- `install_xite`
- One of `xite.tar.gz`, `xite.tar.Z`, `xite.tar`

If you lack any of these files, you will find them with a WWW browser pointed to XITE's WWW home page, in

```
http://www.ifi.uio.no/~blab/Software/Xite/
```

You can also grab the files with ftp from `ftp.ifi.uio.no`, in the directory `pub/blab/xite`.

Move the above files into `$XITE_HOME` and change to that directory,

```
% cd $XITE_HOME
```

Do **not** unpack the `tar` file manually.

A.2.2 Unpacking

The script `install_xite` will unpack, build and install XITE. To get a list of the various options supported by this script, give the command

```
% sh install_xite -h
```

Note in particular the options `-d`, `-f` and `-o`. You may be using these options if you choose to install XITE for multiple computer architectures.

The easiest way to unpack, build and install XITE is to simply issue the command

```
% sh install_xite
```

and give the information requested by the script as it runs. Be prepared to supply the following

- Directory for storage of XITE C header files
- Directory for storage of XITE object libraries
- Directory for storage of XITE executables
- Directory for storage of XITE manual pages
- Directory for storage of XITE hypertext documentation

The default values suggested by the installation script for the above are all below the XITE home directory.

For the sake of the display programs in XITE, the following information is also needed

- Location of header files for the X Window System
- Location of object libraries for the X Window System

Depending on whether you have MATLAB, the `libtiff` TIFF object library or the `pbm` conversion utilities installed on your system, you may also want to supply the location of

- `libtiff` TIFF header files
- `libtiff` TIFF object library
- `pnm` header files
- `pnm` object libraries
- MATLAB header files

- MATLAB object libraries
- MATLAB mex files (for storage)

At some point during the installation procedure you will be asked whether you want to continue building the library, executables and documentation. This means that unpacking of the XITE distribution has been completed. What remains is the actual compilation, linking and documentation production. This is handled by a separate script, makexite, which is located in `$XITE_HOME/etc`. If the installation so far has gone by without problems, just type `y` in response to this question, and makexite will be started automatically.

A.2.3 Checking for errors

All of the compilation errors and warnings found while the makexite script is run, will be stored in a file with filename starting with the string `xite_log`, and with filename extension given by the computer architecture you are installing XITE on. You should check this file when the installation is finished, looking especially for error messages. A quick way of doing this is with a command such as

```
% grep -i error xite_log.IRIX | grep -v ignored
```

(if you are installing XITE on a computer running the IRIX operating system). This will tell you whether any errors were found. Search through the file to see what the errors were, before sending a bug report to the XITE developers (see section 4.1 on how to send a bug report).

A.2.4 Rebuilding of parts which failed

If the building of XITE libraries or executables failed, you may try to rebuild the system using a different compiler, different compiler options or change other information which was acquired during the unpacking phase. Rebuilding of XITE can be done without unpacking the distribution once again. Go through the following steps

- From the log file, try to locate what part of the build process failed. In which source directory did it occur? Did the compilation of library functions fail, or compilation of executables? Or perhaps there was an error during the linking phase?
- Do you think that it would help making changes to the `Makefile` for the particular architecture you are installing on? You will find the corresponding makefile in `$XITE_HOME/etc`, with an appropriate filename extension. Perhaps you need to change the `CLFAGS` variable (options given to the compiler), or perhaps you want to use a different compiler (the `CC` variable).
- Rebuild the parts of the system which failed, by using the script

  ```
  $XITE_HOME/etc/makexite
  ```

  Give the option `-h` to get help on how to use this script.

A.3 Building XITE on a different platform

If you ran into problems when following the instructions in section 2.1, read this section.

If the new platform is unknown to XITE (install_xite does not accept the name of your desired platform), go through the following steps

1. Edit the script `$XITE_HOME/etc/.install/testPlatform` to allow this extra platform.

2. Among the platforms mentioned in the testPlatform script, decide which one is closer to the new platform. Check to see whether there is a corresponding Makefile or Makefile template (extension `.tmpl`) in `$XITE_HOME/etc`. If you can't find one, generate it with the command

   ```
   % $XITE_HOME/etc/.install/makeMakefile -p <platform name>
   ```

   Make a copy with filename extension equal to the new platform name.

3. Incorporate any necessary changes in the Makefile, particularly to `CFLAGS`, possibly also to `LFLAGS` and `CC`.

4. Check the files `includes.h` and `xite_*.h` in the XITE header file directory to see that the appropriate header files will be included with the particular platform. You may need to add extra #ifdef directives to treat your platform appropriately.

5. Use the command

   ```
   % sh install_xite -d
   ```

   or

   ```
   % sh install_xite -d -p
   ```

6. Send an email to *blab@ifi.uio.no* describing this new platform and any changes or additions you made to XITE in order to make it run.

References

[1] Svein Bøe. XITE – X-based Image Processing Tools and Environment – Programmer's Manual for version 3.3. Report 92, Image Processing Laboratory, Department of Informatics, University of Oslo, P. O. Box 1080 Blindern, 0316 Oslo, Norway, August 1997, `http://www.ifi.uio.no/~blab/Software/Xite/ProgrammersManual/`.

[2] Svein Bøe, Tor Lønnestad, and Otto Milvang. XITE – X-based Image Processing Tools and Environment – User's Manual for version 3.3. Report 56, Image Processing Laboratory, Department of Informatics, University of Oslo, P. O. Box 1080 Blindern, 0316 Oslo, Norway, August 1997, `http://www.ifi.uio.no/~blab/Software/Xite/UsersManual/`.

[3] Tor Lønnestad and Otto Milvang. *XITE – X-based Image Processing Tools and Environment*, pages 63–88. World Scientific, Singapore, 1994. ISBN 981-02-1510-X.

[4] Otto Milvang and Tor Lønnestad. An image widget for image processing. *The X-Journal*, 1(2), November/December 1991.

[5] Otto Milvang and Tor Lønnestad. An object oriented image display system. In *Proceedings, 11th International Conference on Pattern Recognition*, pages D218–221, The Hague, The Netherlands, August/September 1992.

REPORTS FROM THE IMAGE PROCESSING LABORATORY

78 Maurycy Szmurlo :
"A comparative study of statistically classifiable features
used within the area of Optical Character Recognition"
Cand.Scient. (Master) thesis May 1995.

79 Ingvil Hovig :
"Verktøy og metoder for komprimering av MR bilder"
Dr.Scient. (Ph.D) thesis August 1995.

80 Øyvind Akerhaugen :
"Automatisk plassering av navn på kart ved hjelp av simulated annealing"
Cand.Scient. (Master) thesis February 1996.

81 Marius Midtvik :
"Reversibel komprimering av MR bilder basert på statisk kildemodellering"
Cand.Scient. (Master) thesis May 1996.

82 Tor Øyvind Didriksen :
"Linjefinning og klassifikasjon med Random Hough-transform
— en eksperimentell studie"
Cand.Scient. (Master) thesis June 1996.

83 Edward Allen Smith :
"Image Processing Techniques on DNA Fingerprint Images
and its Application to Genetic Similarity Analysis"
Cand.Scient. (Master) thesis August 1996.

84 Parviz Heydari :
"Line Following in Digitized Map"
Cand.Scient. (Master) thesis November 1996.

85 Ramin Gordjianfar :
"Vectorization of the Cartographic Data"
Cand.Scient. (Master) thesis November 1996.

86 Luren Yang and Torfinn Taxt :
"Robust Methods for Sonar Bottom Detection"
February 1997.

87 Christian Wladimir Hansson :
"Strukturgrammatikk — en høyeredimensjonal grammatikk
for syntaktisk mønstergjenkjenning av 3D objekter i bilder"
Cand.Scient. (Master) thesis December 1996.

88 Sverre H. Huseby :
"Video on the World Wide Web —
Accessing Video from WWW Browsers"
Cand.Scient. (Master) thesis February 1997.

89 Håvard Lauritzen :
"Raster til vektor konvertering ved simulert størkning"
Cand.Scient. (Master) thesis May 1997.

90 Irene Rødsten :
"Texture Segmnetation using Moment based Features
obtained by Locally Adaptive Thresholding"
Cand.Scient. (Master) thesis May 1997.