

# Digital Image Processing

*Presented by:*

Dr. Moe Moe Myint

moemoemyint@moemyanmar.ml

Information Technology Department

Technological University (Kyaukse), Myanmar

- Only Original Owner has full rights reserved for copied images.
- This PPT is only for fair academic use.



How to read and display image

How to enhance an image



How to write MATLAB code

How to study Image Processing Toolbox with MATLAB




Let's use MATLAB programming software

# MATLAB®

*The Language of Technical Computing*



 The MathWorks

Copyright 1984-2006, The MathWorks, Inc.  
Protected by U.S. patents. See [www.mathworks.com/patents](http://www.mathworks.com/patents).

# What is MATLAB?



MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects.



# Overview of MATLAB

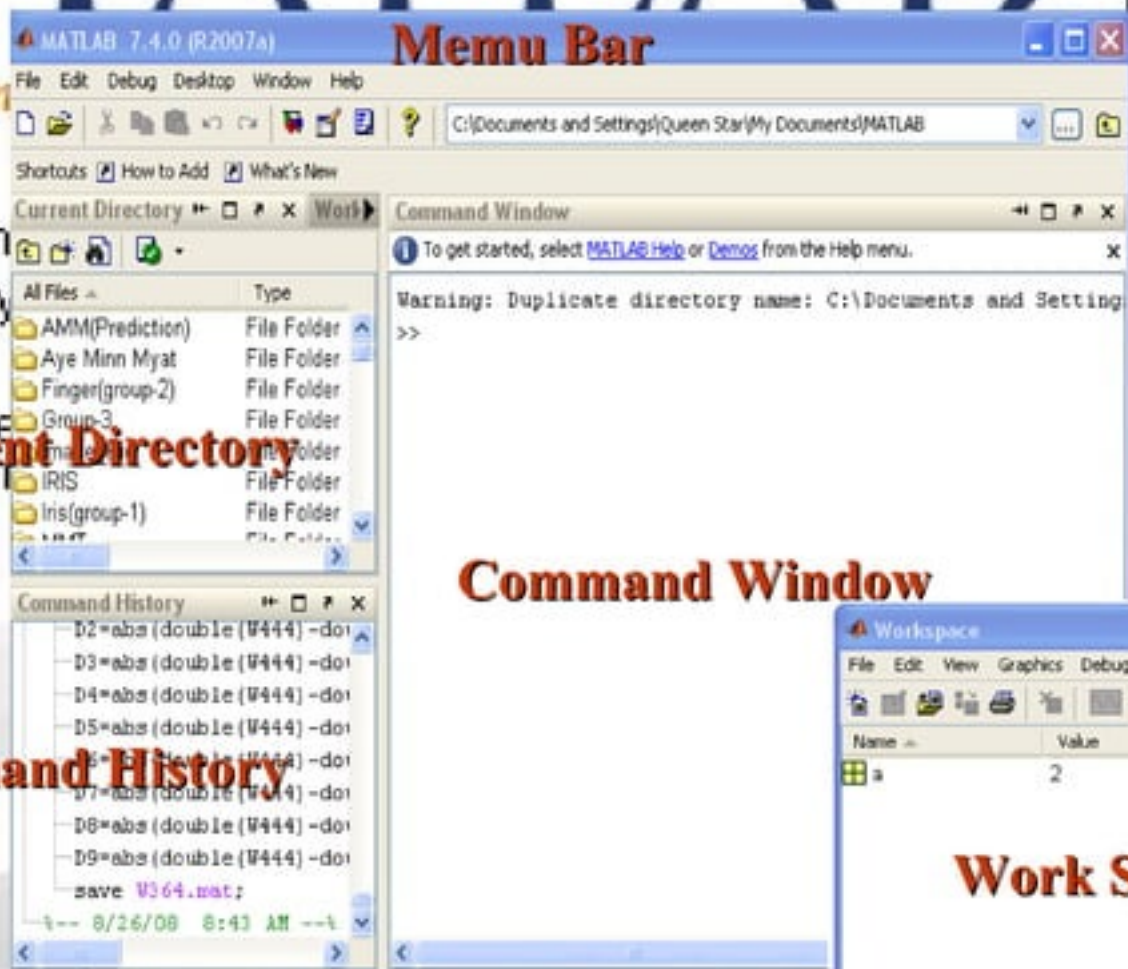
MATLAB is a **high-performance language for technical computing**. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

**Typical uses** include

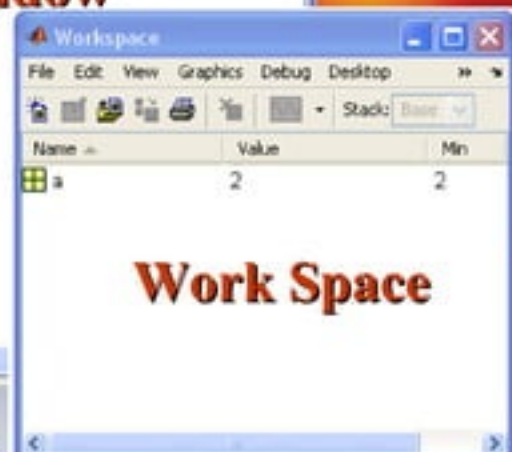
- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

# MATLAB®

R2007a



**Command Window**

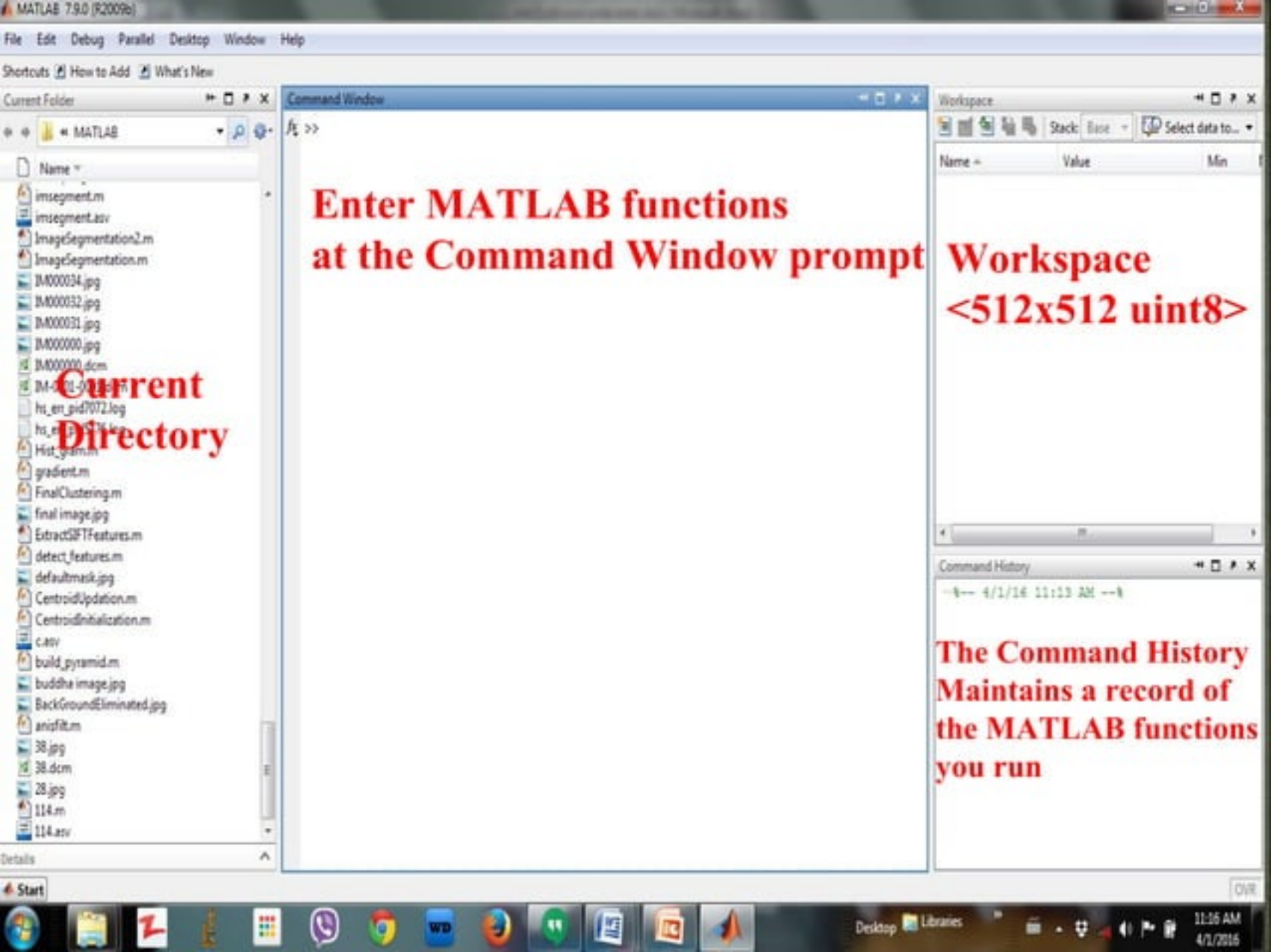
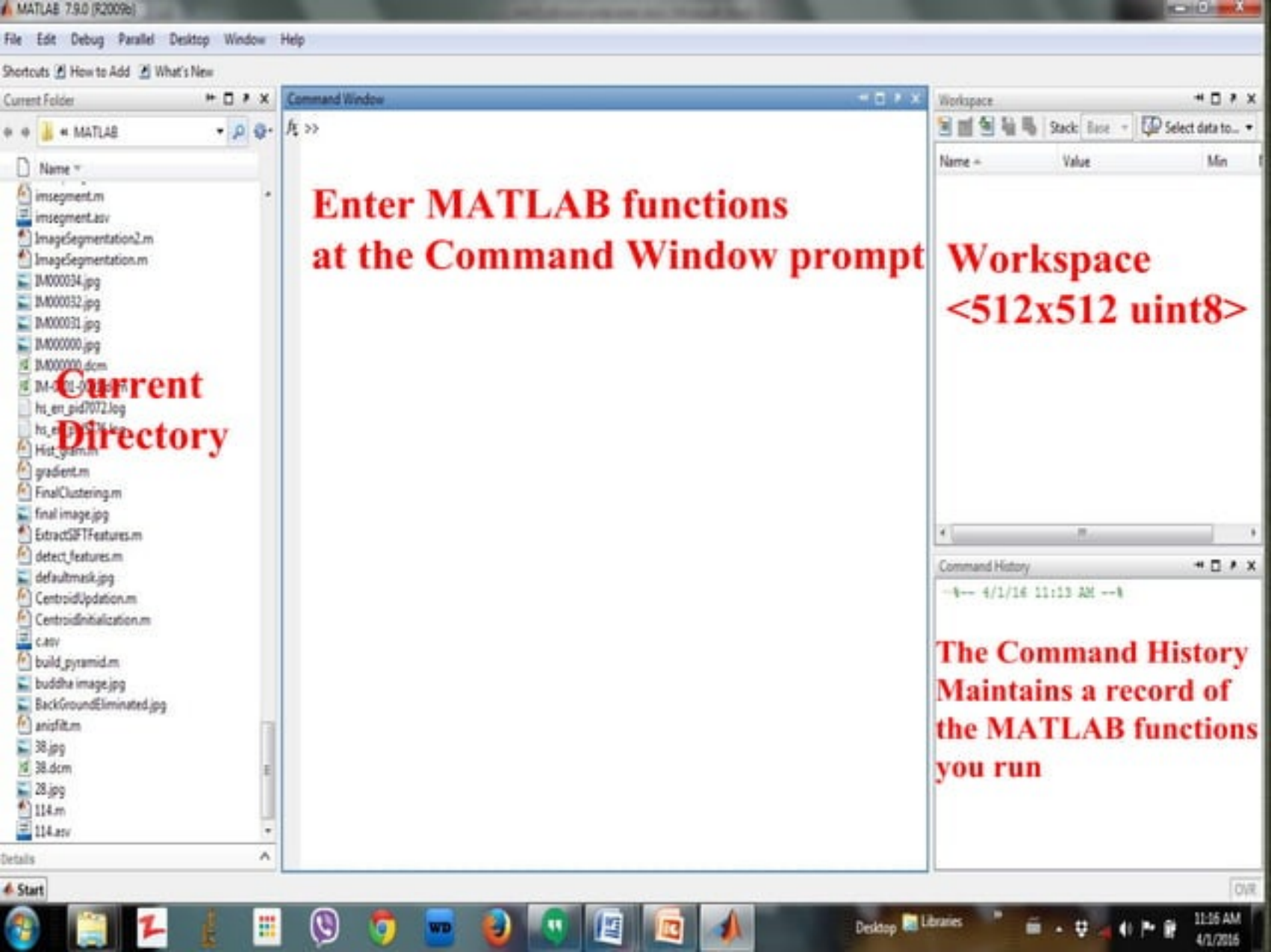


**Work Space**

# MATLAB Introduction

- MATLAB : Long term calls MATrix LABoratory.
- It is not like that C,C++,FORTRAN Programming language.(that is MATLAB is BASIC)
- Everybody can afford to study this MATLAB who have the basic of programming knowledge.
- Educated persons and Engineers in every field use MATLAB that it is also used to write simple program to simulation program.
- MATLAB is an interactive program for scientific and engineering calculation.
- For example, differential equation...
- If it is difficult to solve it by analytical methods, we use MATLAB solvers ,Symbolic math toolbox and Simulink in MATLAB.





**Enter MATLAB functions  
at the Command Window prompt**

**Workspace  
<512x512 uint8>**

**Current  
Directory**

**The Command History  
Maintains a record of  
the MATLAB functions  
you run**

# Digital Images and Pixels

- **Digital image:** discrete samples  $f[x,y]$  representing continuous image  $f(x,y)$
- Each element of the 2-d array  $f[x,y]$  is called a **pixel** or **pel** (from “picture element”)



200x200



100x100



50x50



25x25

Dr. Moe Moe Myint  
Information Technology Department  
Technological University (Kyaukse)

# Reading and Writing Images (Lab 1)

---

- Introduces some basic image processing concepts
- starts by reading an image into the MATLAB workspace
- performs some contrast adjustment on the image
- Writes the adjusted image to a file

# Step 1: Read and Display Image

- To clear the MATLAB workspace of any variables and close open figure windows  
`close all, clear all, clc`
- To read an image, use the `imread` command  
`I=imread('pout.tif');`
- To display the image, use the `imshow` command  
`imshow(I);`
- `title('Grayscale Image')`
- `imtool(I)`



## Step 2: Check How the Image Appears in the Workspace

- To see how the **imread** function stores the image data in the workspace
- Return the image data in the variable **I**, which is a 291-by-240 element array of **uint8** data
- To get information about variables in the workspace by calling the **whos** command

Name	Size	Bytes	Class	Attributes
<b>I</b>	<b>291x240</b>	<b>69840</b>	<b>uint8</b>	

## Step 3: Improve Image Contrast

- To see the distribution of intensities in pout.tif, use the **imhist** function

```
figure, imhist(I)
```

Notice how the intensity range is rather narrow.

- To improve the contrast in an image, do the histogram equalization

```
I2 = histeq(I);
```

- Display the new equalized image, I2, in a new figure window

```
figure, imshow(I2)
```

```
figure, imhist(I2)
```

## Step 4: Write the Image to a Disk File

- To write the newly adjusted Image I2 to a disk file, use the **imwrite** function

```
imwrite(I2, 'pout2.png');
```

## Step 5: Check the Contents of the Newly Written File

- To see what `imwrite` wrote to the disk file, use the `imfinfo` function

`imfinfo('pout2.png')`

(The `imfinfo` function returns information about the image in the file, such as its format, size, width, and height.)



Format	Variants
BMP	1-bit, 4-bit, 8-bit, 16-bit, 24-bit, and 32-bit uncompressed images; 4-bit and 8-bit run-length encoded (RLE) images
CUR	1-bit, 4-bit, and 8-bit uncompressed images
HDF	8-bit raster image datasets, with or without an associated colormap; 24-bit raster image datasets
ICO	1-bit, 4-bit, and 8-bit uncompressed images
<b>JPG</b>	Any baseline JPEG image; JPEG images with some commonly used extensions
PBM	Any 1-bit PBM image; raw (binary) or ASCII (plain) encoded
PCX	1-bit, 8-bit, and 24-bit images
PGM	Any standard PGM image; ASCII (plain) encoded with arbitrary color depth; raw (binary) encoded with up to 16 bits per gray value
<b>PNG</b>	Any PNG image, including 1-bit, 2-bit, 4-bit, 8-bit, and 16-bit grayscale images; 8-bit and 16-bit indexed images; 24-bit and 48-bit RGB images
PPM	Any PPM image; ASCII (plain) encoded with arbitrary color depth; raw (binary) encoded with up to 16 bits per color component
RAS	Any RAS image, including 1-bit bitmap, 8-bit indexed, 24-bit truecolor and 32-bit truecolor with alpha
<b>TIFF</b>	Any baseline TIFF image, including 1-bit, 8-bit, and 24-bit uncompressed images; 1-bit, 8-bit, and 24-bit images with packbits compression; 1-bit images with CCITT compression; also 16-bit grayscale, 16-bit indexed, and 48-bit RGB images
XWD	1-bit and 8-bit ZPixmap; XYBitmaps; 1-bit XYPixmap

Format	Full Name	Variants
'bmp'	Windows Bitmap (BMP)	1-bit, 8-bit, and 24-bit uncompressed images
'gif'	<a href="#">Graphics Interchange Format (GIF)</a>	8-bit images
'hdf'	<a href="#">Hierarchical Data Format (HDF4)</a>	8-bit raster image data sets, with or without associated colormap, 24-bit raster image data sets; uncompressed or with RLE or JPEG compression
'jpg' or 'jpeg'	<a href="#">Joint Photographic Experts Group (JPEG)</a>	8-bit, 12-bit, and 16-bit Baseline JPEG images <div> <b>Note</b> <code>imwrite</code> converts indexed images to RGB before writing data to JPEG files, because the JPEG format does not support indexed images. </div>
'pbm'	Portable Bitmap (PBM)	Any 1-bit PBM image, ASCII (plain) or raw (binary) encoding
'pcx'	Windows Paintbrush (PCX)	8-bit images
'pgm'	Portable Graymap (PGM)	Any standard PGM image; ASCII (plain) encoded with arbitrary color depth; raw (binary) encoded with up to 16 bits per gray value
'png'	<a href="#">Portable Network Graphics (PNG)</a>	1-bit, 2-bit, 4-bit, 8-bit, and 16-bit grayscale images; 8-bit and 16-bit grayscale images with alpha channels; 1-bit, 2-bit, 4-bit, and 8-bit indexed images; 24-bit and 48-bit truecolor images; 24-bit and 48-bit truecolor images with alpha channels
'pnm'	<a href="#">Portable Anymap (PNM)</a>	Any of the PPM/PGM/PBM formats, chosen automatically
'ppm'	Portable Pixmap (PPM)	Any standard PPM image. ASCII (plain) encoded with arbitrary color depth; raw (binary) encoded with up to 16 bits per color component
'ras'	<a href="#">Sun Raster (RAS)</a>	Any RAS image, including 1-bit bitmap, 8-bit indexed, 24-bit truecolor and 32-bit truecolor with alpha
'tif' or 'tiff'	<a href="#">Tagged Image File Format (TIFF)</a>	Baseline TIFF images, including 1-bit, 8-bit, 16-bit, and 24-bit uncompressed images, images with packbits compression, images with LZW compression, and images with Deflate compression; 1-bit images with CCITT 1D, Group 3, and Group 4 compression; CIELAB, ICCLAB, and CMYK images
'xwd'	X Windows Dump (XWD)	8-bit ZPixmap

Questions ???

