

You can convert a 3D waves containing an RGB image into a grayscale image using the **ImageTransform** operation with the `rgb2gray` keyword.

You can convert a number of 2D image waves into a 3D stack using the **ImageTransform** operation with the `stackImages` keyword.

## Loading Sun Raster Files

Sun Raster files are loaded as 2D waves.

If the Sun Raster file includes a color map, Igor creates, in addition to the image wave, a colormap wave, named with the suffix `"_CMap"`.

## Loading Row-Oriented Text Data

All of the built-in text file loaders are column-oriented — they load the columns of data in the file into 1D waves. There is a row-oriented format that is fairly common. In this format, the file represents data for one wave but is written in multiple columns. Here is an example:

350	2.97	1.95	1.00	8.10	2.42
351	3.09	4.08	1.90	7.53	4.87
352	3.18	5.91	1.04	6.90	1.77

In this example, the first column contains X values and the remaining columns contain data values, written in row/column order.

Igor Pro does not have a file-loader extension to handle this format, but there is a WaveMetrics procedure file for it. To use it, use the Load Row Data procedure file in the “WaveMetrics Procedures:File Input Output” folder. It adds a Load Row Data item to the Macros menu. When you choose this item, Igor presents a dialog that offers several options. One of the options treats the first column as X values or as data. If you specify treating the column as X values, Igor will use it to determine the X scaling of the output wave, assuming that the values in the first column are evenly spaced. This is usually the case.

## Loading Excel Files

You can load data from Excel files into Igor using the **XLLoadWave** operation directly or by choosing Data→Load Waves→Load Excel File which displays the Load Excel File dialog.

XLLoadWave loads numeric, text, date, time and date/time data from Excel files into Igor waves. It can load data from `.xls` and `.xlsx` files. It does not support `.xlsb` (binary format for large files) files. It also can not load password-protected Excel files.

On Macintosh, it is possible to have a worksheet open in Excel and to use XLLoadWave to load the worksheet into Igor at the same time. When you do this, Igor loads the most recently saved version of the worksheet. On Windows, you must close the worksheet in Excel before loading it in Igor.

Some programs unfortunately save tab-delimited or other non-Excel type files using the `.xls` extension. If you try to load one of these files, XLLoadWave will tell you that it is not an Excel binary file.

## What XLLoadWave Loads

A worksheet can be very simple, consisting of just a rectangular block of numbers, or it can be very complex, with blocks of numbers, strings, and formulas mixed up in arbitrary ways. XLLoadWave is designed to pick a rectangular block of cells out of a worksheet, converting the columns into Igor waves.

XLLoadWave can load both numeric and text (string) data. An Excel column can contain a mix of numeric and text cells. An Igor wave must be all numeric or all text. When you load an Excel column into an Igor wave, you need to decide whether to load the data into a numeric wave or into a text wave. XLLoadWave can also load date, time, and date/time data into numeric waves.

### Column and Wave Types

XLLoadWave provides the following methods of determining the type of wave that it will create for a given column. These methods are presented in the Load Excel File dialog and are controlled by the /C and /COLT flags of the XLLoadWave command line operation.

#### Treat all columns as numeric

This is the default method. If you have a simple block of numbers that you want to load into waves, this is the method to use, and you can forget about the others.

XLLoadWave creates a numeric wave for each Excel column that you are loading. If the column contains numeric cells, their values are stored in the corresponding point of the wave. If the column contains text cells, XLLoadWave stores NaNs (blanks) in the corresponding point of the wave.

#### Treat all columns as date

This is the same as the preceding method except that XLLoadWave converts the numeric data from Excel date/time format into Igor date/time format. See Excel Date/Time Versus Igor Date/Time for details.

When XLLoadWave creates a numeric wave that is to store dates or times, it always creates a double-precision wave, because double precision is required to accurately store dates. Also, XLLoadWave sets the data units of the wave to "dat". Igor recognizes "dat" as signifying that the wave contains dates and/or times when you use the wave in a graph as the X part of an XY pair.

In this method, when XLLoadWave displays the wave in a table, it uses date/time formatting for the table column. You can change the column format to just date or just time using the ModifyTable operation.

#### Treat all columns as text

XLLoadWave loads all columns into text waves.

If you load a column containing numeric cells into a text wave, Igor converts the numeric cell value into text and stores the resulting text in the wave.

#### Deduce from row

This is a good method to use for loading a mix of columns of different types (numeric and/or date and/or text) into Igor.

You tell XLLoadWave what row to look at. XLLoadWave examines the cells in that row. For a given column, if the cell is numeric then XLLoadWave creates a numeric wave and if the cell is text then XLLoadWave creates a text wave.

If a numeric cell uses an Excel built-in date, time, or date/time format, XLLoadWave converts the numeric data from Excel date/time format into Igor date/time format. XLLoadWave can not deduce date and time formatting for cells that are governed by custom cell formats. In this case, see Excel Date/Time Versus Igor Date/Time for details on manually conversion.

When XLLoadWave deduces the column type using this method, it sets the Igor table column format for date/time waves to either date, time or date/time, depending on the built-in cell format for the corresponding column in the Excel file.

#### Use column type string

Use this method if you have a mix of columns of different types (numeric and/or date and/or text) and the "deduce from row" method does not make the correct deduction. For example, in some files there may be no single row that is suitable for deducing the column type.

In this method, you provide a string that identifies the type of each column to be loaded. For example, the string "1T1D3N" tells XLLoadWave that the first column loaded is to be loaded into a text wave, the next column is to be loaded into a numeric date/time wave, and the next three columns are to be loaded into numeric waves. If you load more columns than are covered by the string, extra columns are loaded as numeric. Also, the string "N" means all columns are numeric, the string "D" means all columns are numeric