MacVISTA
Version 1.3.1

30 September 1992

Tim DeBenedictis
229 Commonwealth Av.
Boston MA 02116
e-mail: timmyd@benz.mit.edu

MacVISTA is a program for analysis of astronomical images. It was inspired by VISTA, a professional astronomical image-analysis program developed at Lick Observatory in 1982 for use on SUN, VAX, and other UNIX-based workstations. Originally I wrote MacVISTA as a tool for a research project at M.I.T.'s Center for Space Research. However, now that CCD-based imaging systems are relatively inexpensive, many amateur astronomers posess the same electronic imaging equipment available only at major observatories only a few years ago. MacVISTA is a program to assist those amateurs who do not have access to high-level workstations or state-of-the-art analysis software. All MacVISTA requires is a MacIntosh or compatible computer with a 68020 or higher processor, a math co-processor, and a color monitor. At the time of publication, this would include the Mac II, IIx, IIcx, IIfx, IIci, IIsi (with coprocessor), any Mac Quadra, and the Mac LC or LC II with a math coprocessor installed.

Note that MacVISTA is an image *analysis* program. It assumes you already have images to work with: MacVISTA was not designed to operate the hardware which gathers the image data itself. That must be done by whatever software came with the particular piece of imaging hardware. However, once you have the image data stored as a file on disk, MacVISTA can read, display, and help you analyze it.

Finally, please note that MacVISTA is a shareware program. The author of MacVISTA is a student at M.I.T. who (like most college students) could use every extra bit of cash that rolls in. Feel free to distribute copies of MacVISTA to anyone you want, but if you or your institution regularly use MacVISTA, please send cash, a check, or a money order for $20.00 (US) to the author at the address listed on the cover of this document. For that reason, it's also probably a good idea to copy this manual when distributing copies of MacVISTA. Your shareware donation will not only help pay my tuition, but also make you eligible to recieve a copy of MacVISTA 2.0 when it is completed. (Features under development in version 2.0 include true color reconstruction, image convolution and deconvolution, contour and surface plots, and more...)

MacVISTA FILES

MacVISTA reads FITS files. The FITS (*F*lexible *I*mage *T*ransport *S*ystem) format, widely adopted as a standard in the astronomical community, is a very straightforward way of storing image data. The specifications for FITS format detailed below should allow you to convert images obtained on your own hardware to a format MacVISTA can understand.

FITS files have two parts— the image header, and data itself. The header consists of 2880 bytes of ASCII character data. Read as lines of 80 characters each, a typical FITS header might look like this:

```
SIMPLE  =                    T /
BITPIX  =                   16 /
NAXIS   =                    2 /
NAXIS1  =                  398 /
NAXIS2  =                  292 /
TIME    = 'UT 8:46:54' /
DATE    = '4 Dec 1991' /
FILENAME= 'i911204.087.6' /
COMPRESS= 'linear  ' /
CAMNUM  = '6       ' /
CCD-ID  = 'TI100-12-11' /
EPERADU =                  31. /
FOCALLEN=                 24.3 /
CCDANGLE=                40000 /
PIXELDIM= '22.3 microns' /
BANDPASS=                 2000 /
```

```
FILTER  = 'B+V      ' /
RA      = '4h 26m 4s' /
DEC     = '13d 59m 56s' /
AIRMASS =                 1.16 /
EXPTIME =                   5. /
EPERADU =                  31. /
RNOISE  =                  15. /
TEMP    =                  -60 /
END
```

The format of the header is simple: it consists of lines of text containing keywords and data . FITS header keywords can by anything; however, the header *must* contain the following information in the following order:

`SIMPLE`: whether or not the file conforms to standard FITS format; either 'T' for true or 'F' for false in column 30.
`BITPIX`: the number of bits per pixel in the image data. In the above example, `BITPIX` is 16, meaning that each each pixel is represented by 16 bits (or 2 bytes) of image data. This allows a maximum 32767 different levels of pixel intensity. The only allowed values for `BITPIX` are 8, 16, and 32.
`NAXIS`: the number of axes in the data's coordinate frame. For two-dimensional images, `NAXIS` is 2; for spectra, which only have one row of data, it is 1. Although the FITS format can conceivably store three-, four-, or even higher-dimensional data sets, MacVISTA can only read 2-dimensional data, i.e. images.
`NAXIS1`: the number of pixels along a single row of the CCD, i.e. the number of columns in the image. `NAXIS1` is 398 in the above example.
`NAXIS2`: the number of pixels along a single column of the CCD, i.e. the number of rows in the image. `NAXIS2` is 292 in the above example.
`END`: indicates that there are no more keywords to be read. Anything in the 2880-byte header after the `END` keyword is ignored.

Keywords must contain no more than 8 characters, followed by an '=' in column 9, followed by the data the keyword represents. Integer data is represented as numerals without a decimal point; floating-point numbers are represented with a decimal point, and character infomation is enclosed between single quotes (') in column 11 and in column 20 or greater. Numerals are left-justified to column 30. Anything after the '/' character is considered a comment and is ignored.

The actual image data is stored after the 2880-byte header in binary format; each row of the image is written out consecutively. A row of image data in the above example would consist of 398 contiguous 2-byte integers. Since there are 292 rows, the total image size would by 2880 + (398)(292)(2) bytes = 235,312 bytes or 230K. Image files can eat up a lot of hard-disk space really fast.

Viewed as character data, the image rows aren't very meaningful:

```
\^ÊèééåèêéåååçäàààãáàâàâàåâàâçãàÑàÜàÑààÜWÑÜÉâà~ÅÜÖÑáÜÖÖÜÜÑÉàÜ
ÜÇâÑÑ ÑáÑÑÉÉÖÑÑÑÑÇÑÑÖÑâÇÇÉâàÇÜÜÇÖÉÉÉÑ~ÑÅÑÉÑÑÉÇÉÑÇÑÜÑÖÑÜáÉ~É..
.
```

But translated from 2-byte binary integer format into arabic numerals, the above image data reads

```
92   94   230  143  142  142  140  143  144  142  140  140
140  141  138  136  136  139  135  136  137  137  136  137
140  137  136  137  141  139  136  132  136  134  136  132
136  136...
```

(To see what the whole image actually looks like, try opening the sample FITS file "Hyades.FITS" distributed with MacVISTA.)

MacVISTA can open any file of type TEXT, FITS, as well as files in MacVISTA's own proprietary format. MacVISTA tries to read TEXT files as if they contained data in FITS format. This feature was included to make it easier to open images obtained on other platforms: simply porting an image over as a text file, without changing any of the file's contents, should produce a TEXT file MacVISTA can understand. (There are no guarantees, however, if you try to open "Letter to Julie" with MacVISTA.)

MacVISTA can save images either as FITS files or as files in its own format. MacVISTA's format represents image pixels as floating-point numbers instead of integers, which allows greater precision but eats up more hard-disk space. MacVISTA files also contain photometric records and other data obtained from the image. FITS files and MacVISTA files have their own distinct Finder icons, so it should be easy to distinguish between them.

MacVISTA can generate 3 other types of files. First, it can save the contents of an open image window as a PICT file, which can be read by any Macintosh graphics application (*MacPaint, Canvas, Adobe Illustrator*, etc. *Canvas* deals with the PICT format particularly well.) How accurately the colors or grays in a MacVISTA PICT file will be displayed depends on your graphics application.

MacVISTA can generate two types of TEXT files. First, it can save a selected region of an open image window as tab-delimited numerals. This effectively makes a "spreadsheet" out of a range of image data, with rows and columns containing numerals describing the pixels in the corresponding rows and columns of the image. This is useful if you wish to import raw image data into a different application (such as Microsoft Excel). Here is a selection from a typical such TEXT file:

```
148.00       151.00       149.00       147.00       148.00
154.00       157.00       157.00       152.00       150.00
164.00       284.00       190.00       163.00       158.00
173.00       330.00       218.00       222.00       210.00
162.00       170.00       185.00       347.00       312.00
160.00       165.00       173.00       180.00       175.00
158.00       162.00       161.00       160.00       162.00
157.00       156.00       156.00       157.00       155.00
```

Second, MacVISTA can save the photometric data it has obtained from an image as a TEXT file. TEXT photometry files contain lists of objects found in the image with data on positions, brightnesses, etc. They should be readable by any word-processing program and by most spreadsheets. A typical MacVISTA photometry file looks like this:

```
1 93.93  289.04  0.8004  0.7760  9  2328.00  3  57  13.7  +9   23  27.1  3.13
2 103.05 260.94  0.7509  0.7590  9  2375.00  3  51  53.8  +9   53  6.6   3.04
3 103.95 186.04  0.7914  0.7822  9  1997.00  3  37  18.9  +10  3   5.6   4.12
4 109.16 251.92  0.6941  0.6989  9  3875.00  3  50  11.0  +10  11  51.5  2.54
5 120.10 208.03  0.7371  0.7260  9  2687.00  3  41  41.2  +10  48  22.6  2.87
6 122.00 165.99  0.7934  0.7940  9  1835.00  3  33  24.0  +10  57  45.4  4.20
```

Each row contains information for a separate photometry object. The first column is the ID number of the photometry record. The second and third columns are the object's centroid row and column. The fourth and fifth columns are the row and column standard deviations. (For an explanation of these, see "Image Analysis with MacVISTA", below.) The sixth column is the number of pixels included in the object; the seventh column is the total count of ADU in those pixels. Columns 8-10 and 11-13 contain the object's right ascension and declination in hours, minutes, and seconds (degreees, minutes, and seconds for declination.) If the R.A. and dec. have not been calculated, these columns will read "`0 0 0.0`". The last column contains the object's magnitude. Again, if this has not been calculated, it will be `0.00`.

Note that PICT files generated by MacVISTA cannot be reopened by it. While both types of TEXT files can be opened, MacVISTA cannot understand either, since it interprets TEXT files as FITS files. If you wish to save photometric data for further analysis within MacVISTA, save it in a MacVISTA file, not as a photometry (TEXT) file.

Files of all formats are opened and saved with the commands in the **File** menu:

New… (**N**) does nothing at present but exists to maintain compatibility with Apple's user interface guidelines.

Open… (**O**) allows you to open a FITS, TEXT, or MacVISTA file with the standard "Open File" dialog box.

Close… (**W**) closes the image in the currently selected window.

Save (**S**), which saves the active image in whatever format is currently selected. *Be careful* if the current format is anything other than FITS or MacVISTA format—otherwise you may overwrite an image file with data in an irretrievable format.

Save As… is a safer option: it opens the standard "Save As.." dialog box, which asks you for a filename, and allows you to cancel before saving if you wish.

File Format… opens a dialog box which allows you to pick what kind of file you wish to save.

Page Setup… opens the familiar "Page Setup..." dialog box, and

Print… (**P**) prints the frontmost window.

Rename window… opens a dialog box which allows you to rename the currently selected window.

Edit FITS header… opens a window which allows you to view and/or change the FITS header of the currently selected image window. (Carriage returns are added after each 80 characters to make editing easier. When the header is saved on disk, these are stripped out and replaced with enough spaces (' ') to make each line exactly 80 characters long. Since FITS headers all contain 2880 bytes of data, only the first 36 lines in the edit window, and the first 80 characters in each line, are saved on disk.) Any changes you make are not saved until you choose **Save** or **Save As…**. Be careful when saving changes—you may make the file unreadable if you alter the key header information (`BITPIX, NAXIS1, NAXIS2,` etc.) mentioned above.

Quit (**Q**) quits MacVISTA.

MacVISTA IMAGE DISPLAY

Once an image is opened, it appears in a new window. MacVISTA can open as many images at once as your computer's memory allows for. (To increase the number of images MacVISTA can open at once, increase the size of MacVISTA's memory partition by choosing **Get Info…** from the Finder.) Once you have opened an image, you can control how it is displayed with the following items in the **Display** menu:

Display Settings… (**D**) allows you to set the color-to-brightness scale for an image. This command opens a dialog box which contains two numbers: the display zero, and the display span. The zero is the intensity level at or below which pixels will appear black in greyscale mode. The span is the range of intensities between black and white. For instance, if the zero is set to 140 and the span is set to 20, then pixels at or below 140 will appear black and pixels above 160 will appear white. Pixels with intensities in between will appear as a shade of grey. The exact shade is a linear function of intensity; in the above example, pixels with an intensity of 150 would appear halfway between black and white. In rainbow dispay mode, the color scale runs from red (brightest) through the spectrum to violet (faintest).

Any time the display settings are changed, the image will be redisplayed. This may take take several seconds, since the color of each pixel must be recalculated from scratch. The default display settings for a newly opened image file are for the zero to be half a standard deviation below the image mean, and the span to be two standard deviations. (For speed, these default settings are calculated using only every 64th image pixel. Because of the relatively small sample size, the value of the standard deviation obtained may not be very accurate— but it will produce a decent range of display settings.)

Zoom In (**[**) and **Zoom Out** (**]**) respectively double and halve the image magnification. You can zoom in arbitrarily; but you cannot zoom out further than 1:1.

The scroll bars along the right and bottom edges of and image window indicate where on the image the window is centered. Setting the horizontal scroll bar all the way to the left, for instance, would center the image on its first column, while setting the vertical scroll bar all the way to the bottom would center the image on its last row. A click in the "up" or "down" buttons of either scroll bar moves the centering by one row or column in the appropriate direction. You can automatically recenter the image by pressing the "command" () key while clicking the mouse button on the image. The image will be recentered where the mouse was clicked.

Rainbow Display (**R**) and **Greyscale Display** (**G**) switch between color and black-and-white.

Inverse Colors (**I**) reverses the order of the colors, so that bright pixels appear darker (in greyscale) or closer to the violet end of the spectrum (in rainbow display).

Show Pixel Values, if checked, causes the row, column, R.A. (in h, m, s) Declination (in °, ‘, “), and ADU count of the pixel the mouse is over to be displayed inthe upper left hand corner of the image. Selecting this command toggles this feature off and on. The information is displayed in the following order:

```
(row,col): h m s ° ‘ “ ADU
```

If the image origin, position angle, and scale have not been calculated, the R.A. and Dec. will read 0 0 0.

Show Marked Stars, when checked, puts a white circle 10 pixels in diameter around the centroid of every photometry object in the image (to create photometry obhects, see "Image Analysis with MacVISTA", below.). Objects which are photometric or astrometric standards (see below) are given emphasized circles. If the circle is difficult to pick out against the image background, try switching to rainbow display, or changing the zoom and/or display settings.

Optimize Settings, if checked, automatically resets an image's display settings when you change the value of the image pixels— say, by adding or subtracting something from an image. The zero is set to half a standard deviation below the new image mean, and the span to two standard deviations. Selecting this command toggles this feature off and on.

Blink with allows you to blink-compare two images in the same window. Selecting this command leads to a menu of all the other images that are currently open. The image you select will be "blinked" once a second with the image in the currently selected (frontmost) window. Each image will be shown at its own display settings, centering, and magnification. (You can blink an image with itself, but the result will not be be very informative!)

IMAGE PROCESSING WITH MacVISTA

The most common operations used to process and/or enhance an image involve image arithmentic— subtracting a bias level, dividing by a flatfield, and so on. These kinds of operations are handled by the first four items in the **Processing** menu:

Add, **Subtract**, **Multiply by**, and **Divide by** open submenus which allow you to pick another image to add to, subtract from, multiply, or divide the currently selected image by. You may add, multiply, etc. an image by itself, or by any other image of the same size. You can also add, subtract, multiply, or divide by a constant, in which case a dialog box opens asking you for the value of the constant you wish to use. However, images cannot be divided by zero. When dividing an image by another image, individual pixels which would be divided by zero are instead set to zero.

Log takes the base-10 log of an image (i.e. replaces every pixel by the log of its value)

Nearest Integer does exactly what it says: it rounds the value of every pixel in the image to the nearest integer.

The next group of commands in the **Processing** menu allows you to move, flip, shrink, expand, and rotate images:

Flip rows and **flip columns** flip an image horizontally and vertically, respectively. This is useful if the image was taken with an inverting telescope, and you wish to show it at its true orientation. Note that these commands do not flip the coordinate frame of the image; they only flip the image data itself. Hence row 0, column 0 will no longer correspond to the same R.A. and Dec. in the sky after the image is flipped. To change the orientation of the image's row and column axes relative to the sky, use the **Calculate Positions** command in the **Analysis** menu (see below).

Shift.... creates a copy of the active image, but with the image data physically shifted. As with the last two commands, this does not shift the image's coordinate system; it only moves the image data itself. Selecting this command opens a dialog box which asks you how many pixels right and down you wish to shift the currently selected image. For instance, shifting an image down by 3 pixels and right by 4 pixels would cause the pixel at row 100, column 100 to move to row 103, column 104 in the shifted copy. To shift an image left or up, type a negative number in the appropriate field of the dialog box. Shifting an image 0 rows and 0 columns will produce an identical copy. Regions of th shifted copy that correspond to pixels outside the original image are set to 0.0 ADU.

Expand... creates an enlarged or reduced copy of the active image. Choosing this command opens a dialog box which asks you how much you wish to enlarge the image; type a number less than 1 to shrink it. The copy is expanded about the origin (row 0, column 0) by the factor you typed. Outlying regions of the original image will be left out of enlarged copies; regions of reduced copies corresponding to regions outside the original will be filled with 0.0. Once again, only the image data, not the coordinate frame, is moved.

Rotate... creates a copy of the active image which is rotated. Since images are rectangular, and not circular, some of the image data may be rotated outside of the image window (and therefore lost.) Similarly, regions of the copy which correspond to regions outside the original image are set to zero. Again, only the image data, not the coordinate system, is rotated.

The next group of commands in the **Processing** menu deals with more complex image operations:

Clip... opens a dialog box which allows the user to reset the values of pixels in the selected region of the image. Pixels in the region brighter than a user-specified maximum or below a user-specified minimum are reset to the values you pick. This is useful, for instance, for removing pixels which are obviously defective on the CCD.

Sky Background... generates an image which approximates the sky background of the active image window. The original image is divided up into rectangular regions of user-specified width. The mean value in each region is computed, excluding pixels above or below a set of user-specified limits. The sky background is assumed to equal this mean. You can set the size of the region, and the minimum/maximum pixel values to be included in the background calculation, in the dialog box which opens after selecting **Sky Background.**

Median filter... is a method of computing the background of an image and/or filtering out "bad" pixels and noise. This is significantly slower than **Sky Background**, but it may give a more precise representation of the true sky background. **Median filter...** generates an image in which every pixel is replaced by the median of all of pixels within a user-specified range of the corresponding pixel on the original image. That range is defined in the dialog box which opens when **Median filter...** is chosen. The default is a box 3 rows by 3 columns centered on the pixel. Also, as with **Sky Background**, a minimum and maximum can be set. Pixels centered on regions of the image with medians above the maximum or below the minimum will be set to the maximum or minimum (respectively). This is to help prevent objects from being picked up on an image which supposedly consists of sky background only. This may be difficult to avoid, however, for espcially large objects (e.g. galaxies) or regions with very high star densities.

The next three commands generate mathematical best fits of various kinds to an image. Choosing any of these will first open up a dialog box asking for information to help compute the fit. Instead of including every image pixel in the fit, you can specify that only pixels in, say, every 10th row and 5th column (or whatever) will be included. Also, you can specify a range of intensities outside which pixels will be discarded from the fit. Again, this is to avoid including stars, defective pixels, etc. in the fit calculation. The default settings are to include pixels in every 10th row and column, and to ignore pixels with intensities outside the range specified in the image's current display settings. In the

Fit Plane... computes the plane which best approximates an image:

$$z = Ax + By + C$$

Hitting the **Calculate** button computes and displays the best-fit coefficients A, B, and C, along with their standard deviations. (Here x and y are column and row on the CCD.) Hitting the **Draw** button will create a new window in which the best fitting plane surface is drawn. You must calculate the fit before you can draw it!

Fit Paraboloid... computes the best fitting paraboloid surface:

$$z = A\left(x^2 + y^2\right) + Bx + Cy + D$$

This works the same way as **Fit Plane**.

Fit Image... computes the linear coefficients which will best fit the active image to another image:

$$image_1 = A\left(image_2\right) + B$$

i.e., the value of each pixel in $image_1$ is a linear function of the same pixel in $image_2$. $image_1$ is the currently selected image; you choose $image_2$. from a pop-up menu in the dialog box. Since there is nothing to draw in an image-to-image fit, the **Draw** button is disabled.

The intitial display settings for images which are generated from another image (median frames, paraboloids, etc.) are the same as the original image's.

IMAGE ANALYSIS WITH MacVISTA

MacVISTA provides a collection of image analysis tools. To use these, you must first select a region of the image by clicking and dragging the mouse over the image window. (Or choose **Select All** (**A**) from the **Edit** menu to select the entire image.) Then choose an item from the **Analysis** menu:

Sum, Min, Max... (**1**) will find the sum of all pixels enclosed in the selection rectangle, and the values and locations of the minimum and maximum pixels in that rectange.

Mean & Std. Dev.... (**2**) calculates the mean value and standard deviation about the mean of all the pixels enclosed in the selection rectangle.

Median... (**3**) computes the median of the selected pixels; see **Percentile**.

Mode... (**4**) finds the most common pixel value in the region enclosed by the selection rectangle. *Note— this function is not yet implemented on MacVISTA.*

Percentile... (**5**) firsts prompts you for a percentile rank to calculate. It then calculates the value of that percentile rank in the selected image region. (Calculating the median is the same as calulating the 50th precentile.) **Percentile...** uses an iterative algorithm which stops when it is within 1.0 of the correct value, so for image regions with very small brightness ranges, it may not give very accurate results. Try multiplying the image by a large constant (say, 1000) and then dividing the result of the **Percentile...** calculation by that constant to find a more precise answer.

Centroid... (**6**)computes the centroid of the region selected and the brightness-weighted standard deviation of pixels about the centroid. The centroid is the average position (row and column) of all the selected pixels where values of individual pixels have been weighted by their intensities. Individual pixels are similarly weighted by their intensities in the calculation of the standard deviations:

$$N = \sum_{i=1}^{n} p_i$$

$$x_c = \frac{1}{N} \sum_{i-1}^{n} p_i x_i$$

$$y_c = \frac{1}{N} \sum_{i-1}^{n} p_i y_i$$

$$\sigma_x = \sqrt{\frac{1}{N} \sum_{i=1}^{n} p_i \left(x_i - x_c\right)^2}$$

$$\sigma_y = \sqrt{\frac{1}{N} \sum_{i=1}^{n} p_i \left(y_i - y_c\right)^2}$$

where N is the total brightness of the n selected pixels p_i, x is the image column, y is the image row, x_c and y_c are the column and row centroids, and σ_x and σ_y are row and column standard deviations. Note that the standard deviations are a measure of the *distribution* of pixels in the selected region, not a measure of the accuracy of the centroid position information.

Histogram... (**7**) creates a histogram of the pixels in the currently selected image region in a new dialog box. The height of a bar on the histogram is proportional to the number of pixels in that bar's brightness range. Initially, the range of brightnesses on the histogram is defined by the image's display settings. However, you can change the range displayed— as well as set the vertical scale— by clicking on the **Axes...** button at the bottom of the dialog box. When you are satisifed, you can save a histogram as a PICT file by clicking on the **Save As...** button in the dialog box, or print it by hitting the **Print...** button.

Slice Profile... (**8**) allows you to take a cross section through an image. After choosing **Slice Profile...**, use the mouse to select a slice by clicking and dragging through the image. (The selection rectangle temporarily changes to a line while selecting a slice.) After you release the mouse button, a dialog box will open up, and a graph of pixel intensities along the slice will be drawn. The range along the intensity (y) axis is initially defined by the window's display settings. As with histograms, though, you can change the scale, as well as the beginning and end points of the slice, by clicking on the **Axes...** button. Again, you can save the slice's graph as a PICT file and/or print it by clicking the appropriate buttons in the dialog box.

Separation... (**9**) allows you to compute the angular sepration and position angle between two points on the image. After selcting this command, click and drag the mouse to pick the points on the image. (The selection box will temporarily change to a line.) When you release the mouse, a dialog box will open, showing the celestial coordinates of the two points, their angular separation in both degrees and arcsecnds, and the position angle between them. Computing this information requires that you know the coordinates of the image center, the image scale, and orientaion of the image. MacVISTA will try to extract this information from the FITS header. If it can't find it there, it will open up a dialog box for you to enter it into. (You can open this dialog box at any time by selecting **Calculate Positions...** from the **Analysis** menu— see below.)

The next group of commands in the **Analysis** menu deals with photometric and astrometric analysis of objects in an image. Objects are stored in memory as photometry records, which can be saved as part of a MacVISTA file, or separately as a TEXT file which can be read by word-processing or spreadsheet applications.

Mark Star... (**M**) creates a new photometry record out of the region enclosed by the selection rectangle. The total ADU count, centroid, and standard deviations of the region selected are computed and displayed in a dialog box which opens up. Hitting **Mark** saves the information as a new photometry record; **Cancel** discards the information without creating a new record. If you mark the star, a white circle will be drawn around its centroid on the image. Double-clicking inside that circle will bring up the dialog box showing its ADU count, position, etc., again.

Mark All Stars... attempts to find objects automatically. Selecting this command will open a dialog box prompting the user for a minimum object detection threshold, and a photometry box width. After hitting **OK**, the program will then go through the currently selected image region pixel by pixel, looking for objects. MacVISTA first looks for pixels above the specified threshold. If it finds pixels above the threshold, it then checks to see if they are the brightest in a box, centered on that pixel, of the width you specified. If the pixel meets both criteria, a new photometry object is added to the object list. It will contain the total ADU count, centroid, and standard deviation of pixels in the box.

Clear Marked Stars discards all of an image's existing photometry records.

Double-clicking on any marked photometry object in an image window will open a dialog box displaying the object's photometric information. If the position and magnitude of the object are known, they can be entered into the appropriate text fields. If the **Magnitude standard** box is checked, that object will be used as a standard by which to calibrate photometric measurements for other objects. Similarly, if the **Position Standard** box is checked, the object will be used to calibrate astrometric calculations. The **Find Magnitudes...** and **Find Positions...** commands in the **Analysis** menu do the dirty work of calculating magnitudes and positions.

Find Magnitudes... will first compute the best-fit coefficients for calculating stellar magnitudes from object ADU counts. These are computed using all the objects whose **Magnitude standard** box has been checked. The best fit coefficients are displayed, then magnitudes for all objects whose **Magnitude standard** boxes haven't been checked are computed. Magnitudes are calculated using the following equation:

$$m = A\log_{10} i + B$$

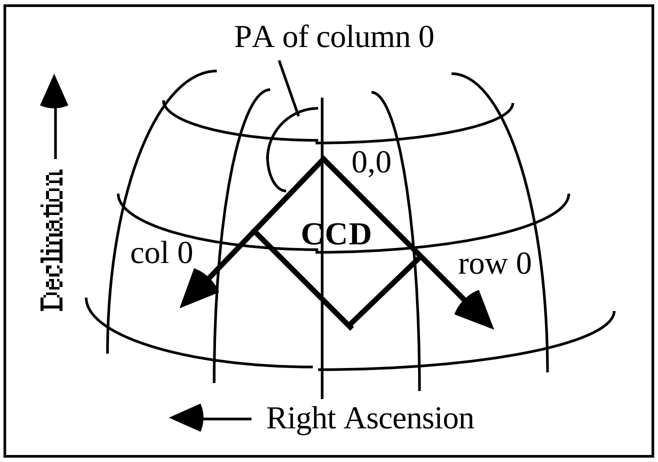
where m is the magnitude and i is the total intensity (ADU count) of the object. The coefficients A and B are computed by a linear least-squares best fit; ideally A should be -2.5.

The **Find Positions...** command works similarly. MacVISTA uses the objects you have designated as position standards to find the coordinates of the image center, the image scale, and the image orientation. It displays these in a dialog box, where you can change them if desired. These are then used to compute the positions of all other photometry objects. At least three position standards are required to calculate the image center, orientation, and scale. It is better, of course, to include a larger number.

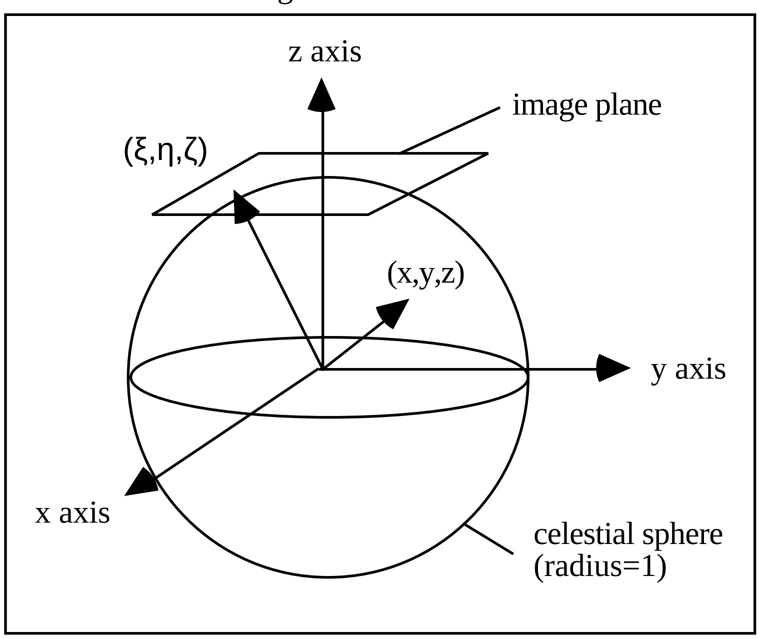
The image scale, L, is the angular size of an image pixel, i.e. radians per pixel. Since most CCD images contain only a tiny part of the celestial sphere, L is effectively uniform across the image. L is determined by taking the ratio of angular separation in radians (θ) to separation in pixels on the CCD image (d), averaged for i paris of position standards:

$$L = \frac{\sum_i \theta_i}{\sum_i d_i}$$

The coordinates of the image origin are the right ascension and declination of the point at the upper left-hand corner of the image (row 0 and column 0). The position angle of the first column (column 0) describes the orientation of the CCD. It is the angle between north in the sky and the direction of the first column. Remember that the positive column direction is downward, not upward, and that position angle is positive to the east (counterclockwise):



MacVISTA uses a rotation matrix to transform coordinates from the frame of the sky to the coordinate frame of the image. It does so by defining positions both on the sky and on the CCD as 3-dimensional rectangular coordinate vectors:



(x,y,z) is the point on the unit sphere corresponding to the spherical coordinates (α,δ):

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos(\alpha)\cos(\delta) \\ \sin(\alpha)\cos(\delta) \\ \sin(\delta) \end{bmatrix}$$

(ξ,η,ζ) is defined by assuming the image is a plane tangent at the image center to the unit sphere at the north pole. Then positions on the image can be converted to points in 3-dimensional space by multiplying them by the image scale and attaching a z coordinate of 1:

$$\begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} L & col \\ L & row \\ & 1 \end{bmatrix}$$

Vectors are transformed from one coordinate frame to the other using a rotation matrix. The matrix coefficients A,B,C, etc. are determined making a least-squares fit to the position standards, for which we know both (x,y,z) and (ξ,η,ζ):

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} A & B & C \\ D & E & F \\ G & H & I \end{bmatrix} \begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix}$$

MacVISTA determines the origin coordinates and orientation of the image precisely from the rotation matrix. It then opens a dialog box and displays them (you may change them at this point if you wish.) Finally, it uses these values to calculate the positions of the other stars which are not position standards.

VERSION HISTORY

MacVISTA 1.0 was created on 21 July 1992.
MacVISTA 1.0.1 corrected a bug in calculating centroids.
MacVISTA 1.1 added slice profiles and histograms.
MacVISTA 1.2 added flipping, shifting, expanding, and rotating images.
MacVISTA 1.3 fixed several bugs in version 1.2, and added mouse celestial coordinates, window renaming, and FITS header editing capabilities. The method used to calculate positions was improved, and substantial parts of the code were rewritten.
MacVISTA 1.3.1 improved handling of the FITS format.

ACKNOWLEDGEMENTS

Many thanks to Roland Vanderspek for his encouragement and to Mike Bradshaw for his endless knowledge of the Mac toolbox.

VISTA was originally written by Richard J. Stover and Tod R. Lauer at Lick Observatory in 1982. VISTA is distributed electronically at the following Internet addresses: vista@lowel.edu and vista@helios.ucsc.edu. As of 23 July 1992, the current version was 4.2. *Note*— the authors of VISTA are in no way responsible for MacVISTA. Please direct all questions or comments concerning MacVISTA to myself at timmyd@athena.mit.edu or timmyd@benz.mit.edu.

MacVISTA was written in THINK C version 5.0 on a MacIntosh LC II with a math coprocessor. This manual was written with Microsoft Word version 5.0.