Preface to the First Edition

CCD cameras have brought a vast new wealth to astronomy. Amateur astronomers and students now undertake observing projects that would have been unimaginable a decade ago: resolving sub-arc-second details on the planets, recording 21st magnitude stars, measuring the motions of nearby stars, uncovering scores of new asteroids, charting the ups and downs of variable stars, and finding supernovae by the dozen. And a vast wealth of magnificent planet, nebula, and galaxy images resides on web sites around the world. No longer is the taking of high-quality data restricted to professional astronomers.

Richard's Remarks

Nine years have passed since *Introduction to Astronomical Image Processing*, the forerunner of this book, appeared. In its preface, I wrote:

"Back in 1965, Mariner 4 sent the first close-up pictures of Mars back to Earth. I was a teenager, and terrifically impressed that pictures could be radioed to Earth from a planet millions of miles away. And, despite the fact that Mariner 4's camera didn't work quite right, over a period of months, scientists at the Jet Propulsion Laboratory discovered that Mars is a cratered world because they computer-processed the images. It seemed almost magical. I longed to get my hands on those images and to learn how to manipulate the brightnesses and contrasts of the originals until the forms of the craters emerged from the murk."

Today, my dream has come true—and that is why I wanted to write this book. This is a book about images and manipulating images—extracting data, seeing the unseen, enhancing detail—until the raw information collected at the telescope has yielded everything there is to see and know. I wanted this book to be a behind-the-scenes look at image processing; not another "drag this and click that" computer book, but an in-depth analysis and exploration of how image processing works. I wanted a book that wasn't afraid to dig into the math and show you the algorithms that image processing software uses to measure and manipulate images.

Why? Because digital imaging has transformed astronomy. To make the most of CCD imaging, you need to have some idea of what's going on inside the camera and inside the software. Just as carpenters must know the tools they use,

so must observers understand the electronic and software tools they use.

Today a single amateur astronomer has more computer power sitting on the desk than JPL's mainframes had during the first Voyager encounter with Jupiter. A typical amateur astronomer's CCD outperforms the most advanced CCDs professional astronomers had on the 5-meter Hale telescope on Palomar Mountain fifteen years ago. And the skills to measure and manipulate images are open to everyone who seriously wants to learn them.

For me, writing the *Introduction to Astronomical Image Processing* was a splendid opportunity to learn the basics of image processing. The book included a tidy little processing program—*AIP*, alias *AstroIP*—that I had written to accompany the book. The earliest versions of the software ran under a BASIC interpreter on an 8MHz PC XT computer with a CGA graphics card. It was so slow that it took ten seconds to draw an image on the screen. But it was a deep thrill for me—I was writing image processing software!

The year after *Introduction to Astronomical Image Processing* was published, Veikko Kanto, an amateur astronomer from Arizona, built a CCD-based autoguider for astrophotography. When he saw my book he changed the guider to take images—and that camera became the prototype of the Cookbook 211 camera.

Over the next two years, Veikko and I, with John Munger, wrote *The CCD Camera Cookbook* for two CCD chips: the TC211 and the TC245. We put a lot of work into simplifying the design so that anyone willing to acquire basic soldering skills and test their work carefully could build an inexpensive CCD camera. Since then, thousands of amateur astronomers have built Cookbook cameras and set them to work making wonderful images.

With my own Cookbook camera, I began to explore CCD imaging and image processing more deeply. I carried out experiments to determine how best to use my camera, and I developed software using *QuickBasic 4.5*, to expand what the camera could do. Techniques like multiple dark frames, dark-frame matching, track-and-stack imaging, histogram shaping, and color imaging transformed the Cookbook camera into a remarkably powerful imaging tool. The resulting software suite—*BatchPIX*, *CB245*, *Multi245*, and *QColor*—soon attracted users all around the world.

The summer after the *Cookbook* came out, I set up my computer and Cookbook 245 on the south side of the little pink clubhouse at Stellafane. Lots of people visited and stayed to talk for a few minutes, but one guy stayed and talked for hours. He asked everything about the camera. It got so late, in fact, that his little daughter curled up and fell asleep at our feet. The visitor was Jim Burnell.

Jim went home, built a Cookbook 245 camera; and then to help other members of his astronomy club, he ran classes on building Cookbook cameras. As an electrical engineer, Jim had a solid background in programming, and developed a nifty software package—*Prep245*—for calibrating and displaying images.

Meanwhile, I was trying to convert my programs from QB4.5 to the Windows platform, with very little luck. Those who are not programmers cannot imagine how unfriendly programming for Windows really is. I needed help. Twice I found partners who knew Windows, but the first became ill and the second got a new job and had to drop out of the project. My DOS software was getting more sophisticated—I had astrometry, photometry, FFTs, and Richardson-Lucy deconvolution running under DOS—but DOS could not access enough memory, and it was clear that PC users really wanted graphic-user-interface software.

At that point, I just about gave up. Displaying good-looking grayscale images in Windows seemed impossible—and until I could figure out how to display an image, it wouldn't do me any good to port my software tools to Windows. Meanwhile, Jim had already written *Prep245*, a solid starting point for a bigger, more ambitious project. We exchanged e-mails—and soon hatched our plan to write ***AIP4Win*** and the book you now hold in your hands.

Jim and I have devoted nearly three years to the *Handbook of Astronomical Image Processing* and ***AIP4Win***. We see the book and software as two complementary parts of a package for learning about imaging and image processing. The book provides background and theory, and the software puts powerful image processing tools at your fingertips. The book is not a manual for our ***AIP4Win*** program—you'll find that in the extensive Help file—but an exploration of the measuring tools and enhancement algorithms common to all image-processing software, whatever software package you happen to be running.

Jim's Remarks

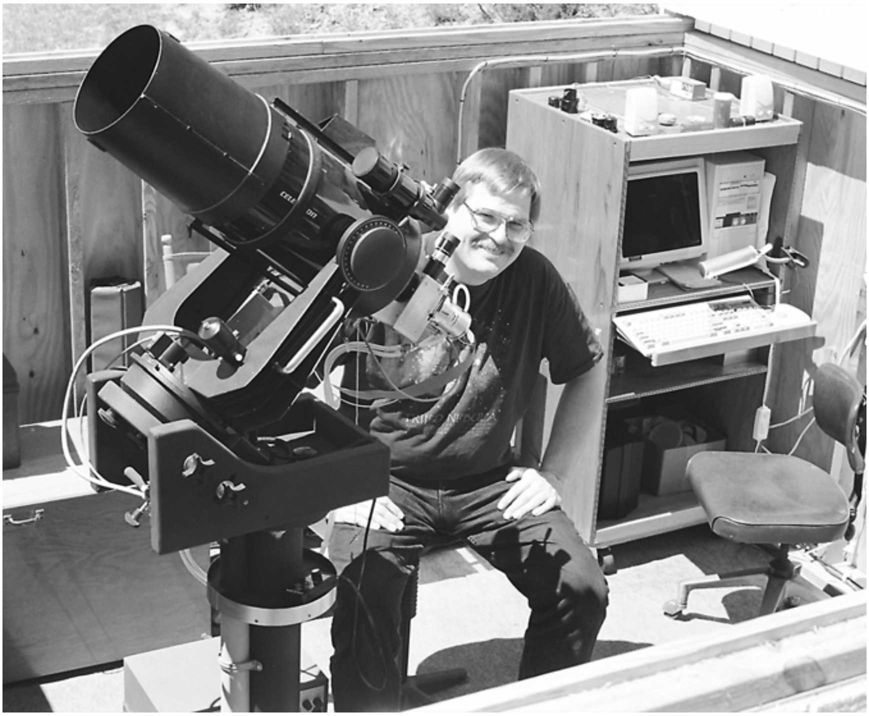
I first became seriously interested in image processing in the late '80s. I was working at Bell Labs at the time and had access to a lot of interesting people and computer resources. I remember playing with the Voyager images as they became available. During lunch and after hours, my buddies and I would experiment to see what we could do with them. I used an image-processing tool called *IMDISP*, written by Ron Baalke, and available via an ftp site at NASA. A bunch of us used to pass the images around on the old USENET, and it was fascinating to see what details we could coax out of the amazing data coming back from Jupiter, Saturn, Uranus and Neptune. This was in the pre-Hubble days, and there was very little deep-sky imagery available over the internet at that time; when someone found a place where you could download images of nebulae and galaxies, it was usually very slow because of all the people downloading the images.

I used a UNIX mainframe to do most of my initial experimenting, as early PCs were just too slow and limited in memory capacity even to consider many of the more involved imaging tasks. It wasn't until the 486-based machines came out that I even bothered building a PC for use at home.

I first ran into Richard at Stellafane in the early '90s. At the time I was planning to build an observatory, and I happened to encounter Richard on the trail on the way to the Friday night tent talks and struck up a conversation on the subject. At the time Richard was editing *Telescope Making* as a side project to his main work on *Astronomy*. There had been some coverage of small observatories both in *TM* and *Astronomy*, and I had a few questions about the mechanics of building a roll-off roof. As we chatted, Richard mentioned a CCD camera project that he and some others were working on—something that he hoped would be made available as a kit. This really captured my interest, as I had been playing around with emulsion-based astrophotography and was frustrated by the long exposure times.

The following year my 9-year old daughter and I were at Stellafane for her very first time. Saturday night after the talks we wandered up to the pink clubhouse on Breezy Hill and there was Richard with the prototype for the Cookbook 245 CCD camera. He had a 50 mm camera lens attached to it and was taking wide-field images of the Milky Way with the rig mounted on a small camera tripod. I joined in and spent the next several hours helping to find objects while Richard demonstrated the operation of the camera. I was hooked.

The next spring my buddy Neil and I built our first Cookbook 245 CCD cameras. We started taking a lot of images and soon realized that we needed something more than the photo editing software that was currently available. I bought a copy of Richard's *CB245* package to process my images, but I really yearned for a Windows-based package to do the job. I had been programming in the *C* language for the previous 16 years and decided to experiment with some of the new Windows-based development tools that were coming out. I wrote a package called *Prep245* using *Visual Basic* to create the GUI and using *Visual C++* to do the number crunching. I spent a few hours on the phone with Richard, from time to time, ask-



ing questions on image processing techniques. His book, *Introduction to Astronomical Image Processing*, had been a really helpful reference and got me started on a lot of the more advanced capabilities.

Prep245 had been out for about two years, and I was thinking about making some upgrades to it. I started corresponding with Richard and floated the idea that we might collaborate on a Windows version of his popular DOS software to accompany an image processing book that he had been wanting to write. It has taken three years to complete the job, and the result you now hold in your hands.

This software has been written by people who use it regularly to calibrate and process their own CCD images. Our approach has been to analyze the way we work with our images and to write software that facilitates that work flow. Realizing that not everyone has the same work style, we recruited a number of serious amateur CCD imagers, each working on different imaging projects, who pounded on the program and gave the features a real workout. We owe a real debt of gratitude to them for the years of testing each and every feature, retesting each update as we added new functionality and fixed the inevitable bugs. The end result is an intuitive, easy-to-use package that provides a complete solution to just about any astronomical project one would undertake using a CCD camera, whether it is supernova hunting, generating light curves of variable stars, tracking the changing details of Jupiter's atmosphere, recording the spectrum of a planetary nebula, creating accurate color images, or taking pretty pictures of favorite deep-sky objects.

Acknowledgments

We have used ***AIP4Win*** to process images during the program's whole development cycle. We have processed thousands of images, tested routines over and over, carried out astrometry, photometry, spectroscopy, shaped countless histograms, blinked for supernovae, and made CCD movies. All of the images in the book were prepared using ***AIP4Win***. But we could not have done it alone. We wish to acknowledge the generous help of many friends and associates.

Our beta test team deserves particular credit: Scott Berfield, Lew Cook, Allen Gilchrist, Veikko Kanto, Al Kelly, Errol Mustafa, Neil McMickle, Tim Puckett, Andy Saulietis, Chuck Shaw, Stuart Warmink, Jan Wisniewski, and Rob West. Tim, Neil, and Rob deserve special mention for demanding the most from us, and we are thankful for the opportunity to make ***AIP4Win*** meet their needs. William Greenlee, Jeff Gunn, Al Nagler, John Rogers, Arne Hendon, Brian Manning, Brian Skiff, and Harold Suiter read and commented on draft chapters; and all of the beta testers read and commented on the entire manuscript.

Special thanks go to Neil McMickle for endless hours spent testing and providing feedback, and for his perpetual readiness to try out new routines, acquire badly needed test images, and retest the program at each new update.

We are grateful to Jack Newton and Don Parker, who shared their ideas and practical imaging experience during the early development of Richard's DOS software, and to Dennis di Cicco for sharing images and ideas. Ed Grafton, Al Kelly, and Chuck Shaw were inspirational in the development of Richard's DOS software for RGB and CMY imaging, which evolved into the first version of the color tools. Brian Manning deserves special thanks for teaching us astrometry. Phil Kuebler got us going on photometry, and Gary Frey challenged us to try even harder. Finally, our thanks to Dave Monet of the U. S. Naval Observatory for generously sending a copy of the USNO A2.0 astrometric database.

We also want to acknowledge many other contributors whom we have not mentioned. We have been participants on the ccd@wwa.com listserve—over the years reading thousands of e-mails—and want to thank every member of the list. The roots of this book go back more than a decade, and it is certain that we have overlooked many who helped, advised, and encouraged us. Please accept our apologies—even though we have not mentioned you by name, we are deeply grateful for your help.

Finally, special thanks in advance to our readers. In the end, you are the most important of all to us. No lengthy book, and no large program, can ever be entirely free of errors. We welcome your comments on the book and your bug reports on the program. Please send them to us through Willmann-Bell so that we can properly log them and act on them. With your help, future editions of the book and software might even approach perfection.

RICHARD BERRY

JAMES BURNELL

Preface to the Second Edition

We are proud to present a new edition of the *Handbook of Astronomical Image Processing* to our readers. As anyone who has participated in the rapidly evolving field of astronomical imaging already knows, everything has changed. New hardware and new software tools have emerged, computers are faster, and the Internet more inclusive. Yet underneath it all, the fundamentals remain the same. On clear nights we wait as a trickle of photons enters the telescope, strikes the detector, and accumulates into an image of a distant place in the cosmos.

Jim's Remarks

A lot has happened in the four years since the first edition of this book was published. Astronomical CCDs have gotten much bigger, much quieter and much more sensitive. WebCams have come into their own as imaging devices for the budding, as well as serious, amateur astronomer to capture high-resolution pictures of the Moon and planets. Also digital single-lens reflex (DSLR) cameras have come way down in price and are rapidly becoming a very popular way for the beginner to "get his feet wet" in astroimaging as well as for the established astroimager to easily take wide-field images of the brighter deep-sky objects.

New, high-quality filters are now available at a reasonable cost, that allow us not only to separate the colors found in celestial objects in order to create color images, but filters that isolate individual spectral lines are also reasonably priced and readily available. These filters provide the ability to isolate fine detail that would otherwise be lost in the background light.

The use of autoguiders is now commonplace, with the result that many astroimagers engage in marathon imaging sessions over multiple nights, creating images with many hours of accumulated exposure time. As a result, amateurs are going deeper than ever before in their quest for ever fainter objects.

The imaging hardware is not the only thing that has changed. Computer hardware, following Moore's Law, has steadily improved its capabilities, where an 80-gigabyte hard drive and 500 megabytes of RAM are found even on entry level computer systems. This gives the software developer much more freedom to implement numerically intensive image processing routines with the problem of them taking all night to execute.

Photograph by Kathleen Burnell.

The art of image processing has undergone continuous growth as well, with new techniques, such as wavelets, that remove noise and noiselessly enhance the details in CCD images. New ways of dealing with color give the user more control than ever in producing color images from a CCD camera or DLSR.

With all the growth in the field of astronomical imaging, it was time to update the text and write new software to take advantage of the new equipment and techniques that have become available. As a result, significant portions of the text have been revised, and chapters added. New techniques and methods are covered along with the use of some of the new equipment that has become available.

The software, originally written with a 486 PC having 16MB or memory and an 800x600 display as a minimum system, has been totally rewritten from the ground up to take advantage of the current crop of entry level PCs with multi-GHz CPUs and hundreds of megabytes of main memory. This new crop of PCs allows routines to be written which were just too computation-intensive to be implemented before. More experience in using the tools in the hands of so many talented imagers has provided tremendous feedback on the techniques and methods implemented in the first version of ***AIP4Win***. These suggestions, provided by the user community, formed the basis for the feature list of the new software. The second version, while keeping as much of the original user interface as was feasible, is an entirely different creature under the hood. Gone are the 32-bit integers that were needed in order to implement processing in a speedy fashion. The use of floating-point math on the current crop of fast machines has facilitated the implementation of more powerful image processing techniques.

On top of the original book and software, nearly a year of writing, coding and testing have gone into the product you now hold in your hands. Enjoy!

Richard's Remarks

In the years since the publication of the *Handbook* and ***AIP4Win***, we have seen three major groups of readers and users emerge:

- Imagers who produce beautiful visions of the heavens,
- observers doing photometry, astrometry, and science, and
- educators teaching and students learning astronomy.

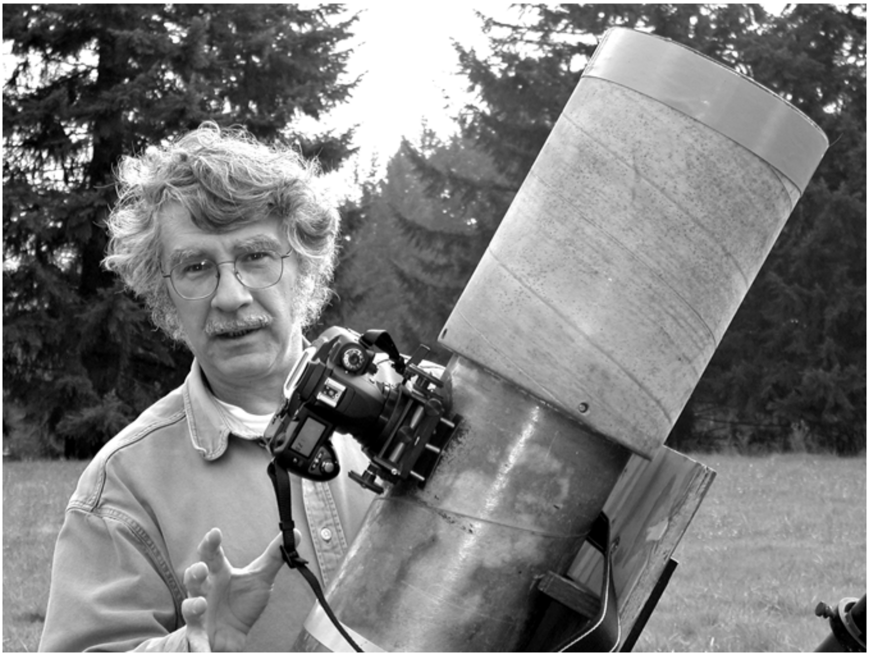
With the second edition of the *Handbook* and the release of ***AIP4Win*** *2.0*, we are pledged to continue serving these constituencies. To that end, we have focused our energies in two major areas: noise reduction and color imaging.

We began our assault on noise by rewriting ***AIP4Win*** to use floating-point mathematics throughout to eliminate mathematical round-off error. Our old *AIP32* code did many computations using floating-point math, but it stored data in 32-bit integers. Although fine for most users, storing results in integer format produces round-off errors. Furthermore, floating-point math is more accurate (with a "granularity" of ~30 parts per billion) and far more flexible and powerful.

AIP4Win *2.0* supports "astronomical" pixel values ranging from 1×10^{32} to -1×10^{32} and microscopic pixel values as small as 1×10^{-32}. This wide dynamic range means that you can sum images and easily work with pixel counts in the millions and billions, or average and work on images that express pixel values to three or four decimal places. Those decimals mean more dependable dark matching, more accurate flat-fielding, and more precise histogram shaping. In addition, ***AIP4Win*** fully supports floating-point FITS formats, so everything is saved and nothing lost when you save an image to disk. Every user will benefit from ***AIP4Win***'s dedication to robust low-noise internal number-crunching.

In this new edition of the *Handbook*, we treat noise theory more fully than we did in the first edition. The current generation of CCDs perform so well that the random arrival times of photons—shot noise—is the primary limitation in their performance. In a new chapter titled "Counting Photons," we explore what Poisson noise means to astronomy. Serious imagers, educators, and students need to be familiar with this fundamental noise source, and those interested in making beautiful images will understand more fully how and why the long integration times that capture millions of photons yield the most beautiful images.

I became fascinated with noise three summers ago, while teaching a course in digital imaging at Portland State University. One practical benefit for the observers, teachers, and students is that ***AIP4Win***'s stellar photometry tools now perform a noise analysis based on the count of photons detected and pixels measured, all reported as a routine part of measuring stellar brightness. This analysis will alert observers to the precision they have attained in measuring stellar magnitude,



and we trust that the feedback will assist them in making even more precise measurements. For students and educators, it's an object lesson that noise statistics impose fundamental limits on what we can learn about the Universe.

Inevitably, however, one studies noise in search of ways to eliminate it. It was clear that spatial methods (such as pixel averaging) always blur image detail, and frequency methods fail because random noise is random in frequency space. However, a hybrid spatial-and-frequency analysis called the wavelet transform allows one to determine whether a pixel or cluster of pixels is probably real or whether it is just random. ***AIP4Win***'s wavelet tools grew out of this, as I boiled theory into practice. Because they are totally noise free, wavelet-processed images look strange to noise-accustomed eyes—so smooth and silky—yet every detail of a noisy image is faithfully replicated. If you were raised on a diet of film grain and random noise, wavelet processed images may not look right to you; but teachers and students are sure to find wavelets a powerful tool for the analysis of severely noise-limited images.

For the image-makers, we developed a noise-averaging spatial filter called the Smooth Background Tool. This tool smooths noise by averaging pixels in the dark sky background parts of an image, but it leaves bright areas untouched. Instead of hiding noise by making sky backgrounds jet black, imagers can lighten the sky to show the faint outer parts of galaxies and nebulae that are lost when the sky is solid black.

Another feature of the new ***AIP4Win*** is a suite of tools for creating synthetic images. These are intended for the educators and students to explore image constituents—bias, dark current, vignetting, image, and noise—that make up images. Using these tools, you can build images with precisely known constituent parts and use them to explore the relationships between the sky background, stars, objects, and detector characteristics.

The other significant addition is enhanced coverage of color imaging. Five years ago, when we developed ***AIP4Win***'s first set of color tools, color imaging was quite new. Using spectral class G2V solar-analog stars to attain accurate color balance had just caught on, and luminance-overlay color (LRGB) was a cutting edge technique. After a lot of reading, thinking, and experimenting, we took the somewhat scary step of processing color images with ***AIP4Win*** in a luminance/chrominance (LCh) color space rather than the traditional RGB color space. For astronomy, where luminance varies over a million-to-one range (black sky to brilliant star), the choice of LCh color space offers significant advantages over the limited scale of RGB. There are now two chapters in the *Handbook* devoted solely to color—and processing color images is now "native" to ***AIP4Win***.

Three factors appear to be driving the widespread interest in color imaging: the availability of really good color-separation filters for monochrome CCD cameras, a new generation of high-quality Bayer-array CCD cameras, and the appearance of excellent digital single-lens reflex (DSLR) cameras. We developed new routines and procedures for loading, decoding, and displaying images from these cameras. I bought a Nikon D70 to learn the ins and outs of DSLR imaging and Jim and I have both experimented with a Canon 10D to get everything possible from the images they capture, in the processing making thousands of raw images, dark frames, and flat-field frames.

For the CCD imagers, we revised and updated ***AIP4Win***'s Join Colors Tool. The software code behind this tool translates your filtered monochrome plus luminance images into color. Working in LCh, RGB, and Lab color spaces, we sought new ways to build images with bright, clean colors and crisp, clear detail. We also added a new Color Effects Tool that operates directly on color images from DSLR cameras *and* astronomical CCD cameras.

Finally, to prevent even the slightest loss of color information, ***AIP4Win*** can save color images in a special 96-bit FITS file (32-bit floating-point data in each color channel) as well as the traditional 48-bit TIFF format with 16-bits integer data in each color channel. If you save your images in this format, no information is lost, so you can always pick up and continue processing where you left off.

In addition to color tweaking, every one of ***AIP4Win***'s tools can access and process the luminance component of an image *without* disturbing its color balance or saturation, and the suite of color effects tools can alter chrominance while leaving image brightness untouched. By cleanly splitting image chrominance from image luminance, ***AIP4Win*** allows you to apply deconvolutions, correct brightness gradients, fix uneven sky backgrounds, replace bad star images, smooth sky backgrounds, and otherwise process any image, color or black and white, with any tool.

Richard Berry and James Burnell

Acknowledgments

Once again we acknowledge the many contributors whom we have not mentioned. Shortly after the first edition went on sale, an independent Yahoo group sprang up to discuss ***AIP4Win*** under the leadership of Greg Crawford. Although we do not own or control it, aip4win@*yahoogroups.com* has nonetheless served as a rich source of ideas and inspiration to us. We wish to thank the members of that list-serve for thousands of intriguing questions and stimulating discussions.

However, your *direct* line to ***AIP4Win*** with ideas and suggestions is through our publisher, Willmann-Bell. Send your email comments, suggestions, bug-reports, and requests to us at ***aip4win*@willbell.com**. Messages to this address is automatically distributed to Richard, Jim, and publisher Perry Remaklus.

To our beta testers we owe another big "thank you." Beta testing software can be a thankless task, yet our beta-test team persisted through mysterious and frustrating crashes, tools that did not work, and more mysterious crashes. Beta testing is an important part of software creation because source code that functions perfectly with standard test images and the right input values will run amok when a beta tester plugs in a wild value or tries to process an image containing zeros and negative numbers. We learn from that, and build better software. It falls on the beta tester to give us the unwelcome news that "the old tool was better," or "clicking 'New Image' made a copy of my old image!" When you look under the hood, software is a complex machine composed of bits and bytes instead of gears and pulleys—and the beta testers tell us when that machine is running smooth and purring like a big contented cat.

Finally, we thank our past and future readers. Your comments and suggestions have been and continue to be important of us. As we said in the preface to the first edition, "no lengthy book, and no large program, can ever be entirely free of errors." As time passed, however, we found some typos but gratifyingly few significant errors, and, with each subsequent printing of the book, we have corrected all known errors. We do welcome your feedback and comments, your suggestions and ideas (email: aip4win@willbell.com) for the next edition of the *Handbook* and ***AIP4Win***.

RICHARD BERRY

JAMES BURNELL