COMPUTING SCIENCE

Computational Photography

Brian Hayes

The digital camera has brought a revolutionary shift in the nature of photography, sweeping aside more than 150 years of technology based on the weird and wonderful photochemistry of silver halide crystals. Curiously, though, the camera itself has come through this transformation with remarkably little change. A digital camera has a silicon sensor where the film used to go, and there's a new display screen on the back, but the lens and shutter and the rest of the optical system all work just as they always have, and so do most of the controls. The images that come out of the camera also look much the same—at least until you examine them microscopically.

But further changes in the art and science of photography may be coming soon. Imaging laboratories are experimenting with cameras that don't merely digitize an image but also perform extensive computations on the image data. Some of the experiments seek to improve or augment current photographic practices, for example by boosting the dynamic range of an image (preserving detail in both the brightest and dimmest areas) or by increasing the depth of field (so that both near and far objects remain in focus). Other innovations would give the photographer control over factors such as motion blur. And the wildest ideas challenge the very notion of the photograph as a realistic representation. Future cameras might allow a photographer to record a scene and then alter the lighting or shift the point of view, or even insert fictitious objects. Or a camera might have a setting that would cause it to render images in the style of watercolors or pen-and-ink drawings.

New cameras don't just capture photons; they compute pictures

Making Pictures

Digital cameras already do more computing than you might think. The image sensor inside the camera is a rectangular array of tiny light-sensitive semiconductor elements called photosites. The image that eventually comes out of the camera is also a rectangular array, made up of colored pixels. You might therefore suppose there's a simple one-to-one mapping between the photosites and the pixels: Each photosite would measure the intensity and the color of the light falling on its surface, assigning these values to the corresponding pixel in the image. But that's not the way it's done.

In most cameras, the sensor array is overlain by a patchwork pattern of red, green and blue filters, so that a photosite receives light in only one band of wavelengths. In the final image, however, every pixel includes all three color components. The pixel colors are calculated by a process called de-mosaicing, in which signals from nearby photosites are interpolated in various ways. A single image pixel might combine information from a dozen adjacent photosites.

In addition to the color filters, most cameras have another optical filter that intentionally blurs the image, suppressing features of very high spatial frequency. If you photograph a distant picket fence, the spacing between pickets in the image might be close to the spacing between photosites in the sensor, leading to disruptive moiré or aliasing effects. The low-pass filter eliminates these artifacts, but the blurring must then be corrected by an algorithmic sharpening operation. Still another computational process adjusts the color balance of the final image.

Given all this post-processing of the image data, it seems a digital camera is not simply a passive recording device. It doesn't *take* pictures; it *makes* them. The sensor array intercepts a pattern of illumination, just as film used to do, but that's only the start of the process that creates the image. In existing digital cameras, all the algorithmic wizardry is directed toward making digital pictures look as much as possible like their wet-chemistry forebears. But once the camera is equipped with an image-processing computer, that device can also run more ambitious or fanciful programs. Images from such a computational camera could capture aspects of reality that other cameras miss.

The Light Field

We live immersed in a field of light. At every point in space, rays of light arrive from every possible direction. Many of the new techniques of computational photography work by extracting more information from this luminous field.

Here's a thought experiment: Remove an image sensor from its camera and mount it facing a flat-panel display screen. Suppose both the sensor and the display are square arrays of size 1,000×1,000; to keep things simple, assume they are monochromatic devices. The pixels on the surface of the panel emit light, with the intensity varying from point to point depending on the pattern displayed. Each pixel's light radiates outward to reach all the photosites of the sensor. Likewise each photosite receives light from all the

Brian Hayes is Senior Writer for American Scientist. *A collection of his columns,* Group Theory in the Bedroom, and Other Mathematical Diversions, *will be published in April by Hill and Wang. Additional material related to the "Computing Science" column appears in Hayes's Weblog at http://bit-player.org. Address: 211 Dacian Avenue, Durham, NC 27701. Internet: bhayes@amsci.org*

Two floral images are both photographs—given a broad definition of "photograph." The image at left was made with a conventional digital camera; at right, a modified camera recorded the same vase of flowers but then applied edge-recognition algorithms to extract the three-dimensional structure of the scene; the camera then rendered the image in a more painterly style. This method of "non-photorealistic photography" was devised by Ramesh Raskar of the Mitsubishi Electric Research Laboratory and several colleagues. (Images courtesy of Raskar.)

display pixels. With a million emitters and a million receivers, there are 10^{12} interactions. What kind of image does the sensor produce? The answer is: A total blur. The sensor captures a vast amount of information about the energy radiated by the display, but that information is smeared across the entire array and cannot readily be recovered.

Now interpose a pinhole between the display and the sensor. If the aperture is small enough, each display pixel illuminates exactly one sensor photosite, yielding a sharp image. But clarity comes at a price, namely throwing away all but a millionth of the incident light. Instead of having 10^{12} exchanges between pixels and photosites, there are only 10^6.

A lens is less wasteful than a pinhole: It bends light, so that an entire cone of rays emanating from a pixel is made to reconverge on a photosite. But if the lens does its job correctly, it still enforces a one-pixel, one photosite rule. Moreover, objects are in focus only if their distance from the lens is exactly right; rays originating at other distances are focused to a disk rather than a point, causing blur.

Photography with any conventional camera—digital or analog—is an art of compromise. Open the aperture wide, and the lens gathers plenty of light, but it also limits depth of field; you can't get both ends of a horse in focus at once. A slower shutter (or longer exposure time) allows you to stop down the aperture and thereby increase the depth of field; but then the horse comes out unblurred only if it stands still. A fast shutter and a narrow aperture alleviate the problems of depth of field and motion blur, but then the sensor receives so few photons that the image is mottled by random noise.

Computational photography can ease some of these constraints. In particular, capturing additional information about the light field allows focus and depth of field to be corrected after the fact. Other techniques can remove motion blur.

Four-Dimensional Images

A digital camera sensor registers the intensity of light falling on each photosite but tells us nothing about where the light came from. To record the full light field we would need a sensor that measures both the intensity and the direction of every incident light ray. Thus the information recorded at each photosite would be not just a single number (the total intensity) but a complex data structure (giving the intensity in each of many directions). As yet, no sensor chip can accomplish this feat on its own, but the effect can be approximated with extra hardware. The underlying principles were explored in the early 1990s by Edward H. Adelson and John Y. A. Yang of the Massachusetts Institute of Technology, although they did not actually build a working light-field camera.

One approach to recording the light field is to construct a gridlike array of many cameras, each with its own lens and photosensor. The cameras produce multiple images of the same scene, but the images are not quite identical because each camera views the scene from a slightly different perspective. Rays of light coming from the same point in the scene register at a different point on each camera's sensor. By combining information from all the cameras, it's possible to reconstruct the light field. (I'll return below to the question of how this is done.)

Experiments with camera arrays began in the 1990s. In one recent project Bennett Wilburn and several colleagues at Stanford University built a bookcase-size array of 96 video cameras, connected to four computers that digest the high-speed stream of data. The array allows "synthetic aperture photography," analogous to a technique used with radio telescopes and radar antennas.

A rack holding 96 cameras and four computers is not something you'd want to lug along on a family vacation. Ren Ng and another Stanford group (Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz and Pat Hanrahan) implemented a conceptually similar scheme in a much smaller package. Instead of ganging together many separate cameras, they inserted an array of "microlenses" just in front of the sensor chip inside a single camera. The camera is still equipped with its standard main lens, shutter and aperture control. Each microlens focuses an image of the main lens aperture

Focusing an image is something the photographer has traditionally done just before clicking the shutter, but new methods of "light field photography" allow the focus and depth of field to be adjusted after the fact. All of the images above are derived from a single exposure, made with a camera devised by Ren Ng and colleagues at Stanford University. In the first four images from left to right the plane of sharp focus is moved from front to back; the rightmost image is a high-depth-of-field composite where all the figures are in focus. (Images courtesy of Ng.)

onto a region of the sensor chip. Thus instead of one large image, the sensor sees many small images, viewing the scene from slightly different angles.

Whereas a normal photograph is two-dimensional, a light field has at least four dimensions. For each element of the field, two coordinates specify position in the picture plane and another two coordinates represent direction (perhaps as angles in elevation and azimuth). Even though the sensor in the microlens camera is merely a planar array, the partitioning of its surface into subimages allows the two extra dimensions of directional information to be recovered. One demonstration of this fact appears in the light-field photograph of a sheaf of crayons reproduced below. The image was made at close range, and so there are substantial angular differences across the area of the camera's sensor. Selecting one subimage or another changes the point of view. Note that these shifts in perspective are not merely geometric transformations such as the scalings or warpings that can be applied to an ordinary photograph. The views present different information; for example, some objects are occluded in one view but not in another.

Staying Focused

Shifts in point of view are another option in images made with Ng's light-field camera. All three images come from a single exposure, but the camera seems to move laterally, and in the rightmost panel is brought closer to the subject. (Images courtesy of Ng.)

Shifting the point of view is one of the simpler operations made possible by a light-field camera; less obvious is the ability to adjust focus and depth of field.

When the image of an object is out of focus, light that ought to be concentrated on one photosite is spread out over several neighboring sites—covering an area known rather poignantly as the circle of confusion. The extent of the spreading depends on the object's distance from the camera, compared with the ideal distance determined by the focal setting of the lens. If the actual distance is known, then the size of the circle of confusion can be calculated, and the blurring can be undone algorithmically. In essence, light is subtracted from the pixels it has diffused into and is restored to its correct place. Mathematically, the process is called deconvolution.

To put this scheme into action, we need to know the distance from the camera to each point in the scene—the point's *depth*. For a conventional photograph, depth cues are hard to come by, but the light-field camera encodes a depth map within the image data. The key is parallax: an object's apparent shift in position when the viewer moves. In general, an object will occupy a slightly different set of pixels in each of the subimages of the microlens camera; the magnitude and direction of these displacements will depend on the object's depth within the scene. The process of recovering the depth information is much like that in a stereoscopic camera, but it can draw on data from many images instead of just two.

Recording a four-dimensional light field allows for more than just fixing a misfocused image. With appropriate software for viewing the stored data set, the photographer can move the point of focus back and forth through the scene, or can create a composite image with high depth of field, where all planes are in focus. This capability takes focus out of the category of things you have to get right when you click the shutter and places it among parameters (such as color and contrast) that can be adjusted after the fact.

The microlens array is not the only approach to computing focus and depth of field. Anat Levin, Rob Fergus, Frédo Durand and William T. Freeman of M.I.T. have recently described another technique, based on a "coded aperture." Again the idea is to modify a normal camera, but instead of inserting microlenses near the sensor, a patterned mask or filter is placed in the aperture of the main lens. The pattern consists of irregular opaque and transparent areas. The simplest mask is a half-disk that blocks half the aperture. You might think such a screen would merely cast a shadow over half the image, but because the filter is in the aperture of the lens, that's not what happens. Although it's true that half the light is blocked, rays from the entire scene reach the entire sensor area by passing through the open half of the lens. But the half-occluded aperture *does* alter the blurring of out-of-focus objects, making it asymmetrical. Detecting this asymmetry provides a tool for correcting the focus. The ideal mask is not a simple half-disk but a pattern with openings of various sizes, shapes and orientations.

The Flutter Shutter

Patterns encoded in a different dimension—time rather than space—provide a strategy for coping with motion blur. In principle, the fuzzy or streaky appearance of objects that move while the shutter is open can be corrected in much the same way that focusing errors are removed. In this case, though, what you need to know is not the object's distance from the camera but its velocity vector. A camera that can collect velocity information was recently described by Ramesh Raskar and Amit Agrawal of the Mitsubishi Electric Research Laboratory and Jack Tumblin of Northwestern University.

A moving object has its image smeared along the direction of motion as projected onto the picture plane. Undoing this defect would seem to be easier than correcting focus because the blur is essentially one dimensional. You just gather up the pixels along the trajectory and apply a suitable deconvolution to separate the stationary background from the elements in motion. Sometimes this program works well, but ambiguities can spoil the results. When an object is greatly elongated by motion blur, the image may offer few clues to the object's true length or shape. Guessing wrong about these properties introduces unsightly artifacts.

A well-known trick for avoiding motion blur is stroboscopic lighting: a brief flash that freezes the action. Firing a rapid series of flashes gives information about the successive positions of a moving object. The trouble is, stroboscopic equipment is not always available or appropriate. Raskar and his colleagues have turned the technique inside out. Instead of flashing the light, they flutter the shutter. The camera's shutter is opened and closed several times in rapid succession, with the total amount of open time calculated to give the correct overall exposure. This technique turns one long smeared image into a sequence of several shorter blurs. The boundaries of the separate images provide useful landmarks in deconvolution.

A further refinement is to make the flutter pattern nonuniform. Blinking the shutter at a fixed rate would create markers are regular intervals in the image, or in other words at just one spatial frequency. For inferring true velocity the most useful signal is one that maximizes the number of *distinct* spatial frequencies. Identifying shutter-flutter patterns that have this property is an interesting mathematical challenge; Raskar and his colleagues have found some that perform well in practice.

Many recent digital cameras are equipped with an image stabilizer designed to suppress a particular kind of motion blur—that caused by shaking of the camera itself. Most of these devices are optical and mechanical rather than computational; they physically shift the lens or the sensor to compensate for camera movement. The shutter-flutter mechanism could handle this task as well.

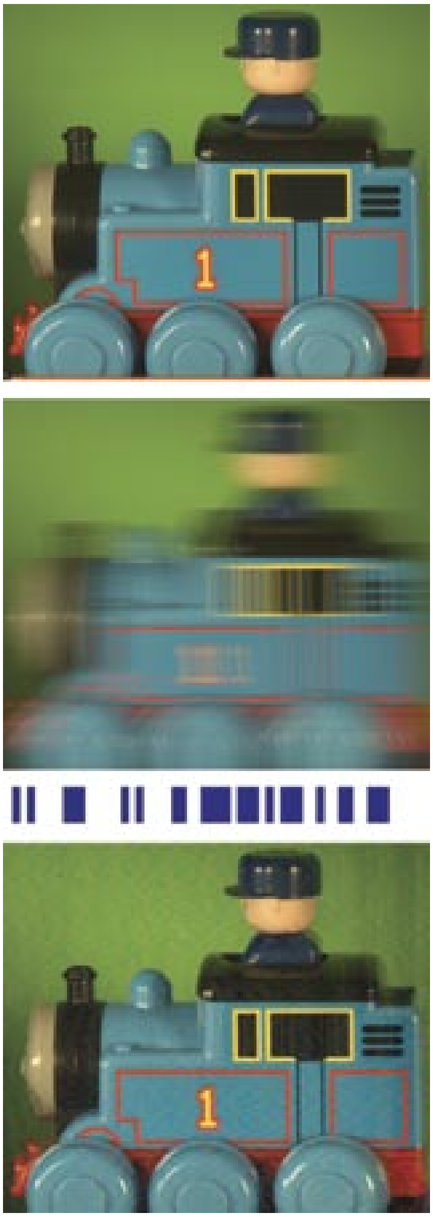
Beyond Photorealism

Computational photography is currently a hot topic in computer graphics, and there's more going on than I have room to report. (A special issue of *Computer* was devoted to the subject in 2006.) Here I want to mention just two more adventurous ideas.

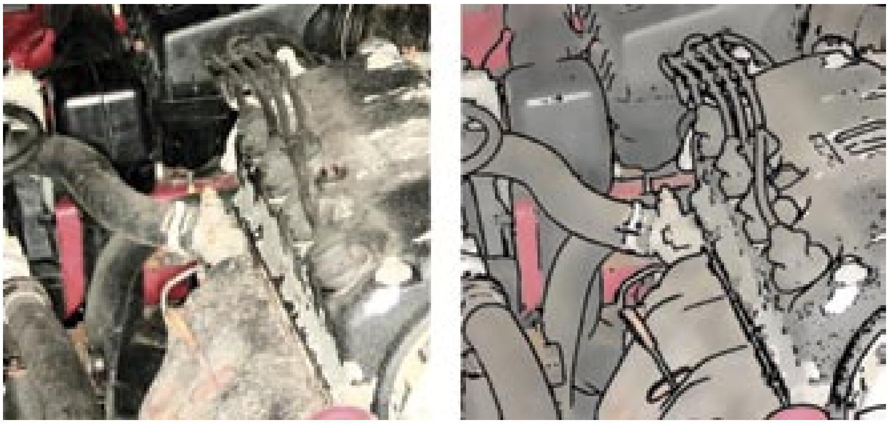
One project comes from Raskar and another group of his colleagues (Kar-Han Tan of Mitsubishi, Rogerio Feris and Matthew Turk of the University of California, Santa Barbara, and Jingyi Yu of M.I.T.). They are experimenting with "non-photorealistic photography"—pictures that come out of the camera looking like drawings, diagrams or paintings.

For some purposes a hand-rendered illustration can be clearer and more informative than a photograph, but creating such artwork requires much labor, not to mention talent. Raskar's camera attempts to automate the process by detecting and emphasizing the features that give a scene its basic three-dimensional structure, most notably the edges of objects. Detecting edges is not always easy. Changes in color or texture can be mistaken for physical boundaries; a wallpaper pattern can look to the computer like a hole in the wall. To resolve this visual ambiguity Raskar *et al.* exploit the fact that only physical edges cast shadows. They have equipped a camera with four flash units surrounding the lens. The flash units are fired separately, producing four images in which shadows delineate changes in contour. Software then accentuates these features, while other areas of the image are flattened and smoothed to suppress distracting detail. The result is reminiscent of a watercolor painting, or in some cases a drawing with ink and wash.

Another wild idea, called dual photography, comes from Hendrik P. A.



Motion blur is another photographic problem being tackled by computational means. A "flutter shutter" camera created by Raskar, Amit Agrawal and Jack Tumblin opens and closes the shutter repeatedly in a quasi-random pattern in order to gather the information needed to correct blur. At top is a conventional photograph of a toy at rest; next is the uncorrected flutter-shutter image, along with the pattern of open and closed intervals represented by white and blue bars; at bottom is the version with blur removed. (Images courtesy of Raskar.)



A diagrammatic style of rendering *(right)* can make it easier to distinguish parts against a busy background than a more conventional photograph *(left)*. By outlining edges and smoothing or flattening broad areas of color, the non-photorealistic camera of Raskar *et al.* emphasizes three-dimensional geometric structures. (Images courtesy of Raskar.)

Lensch, now of the Max-Planck-Institut für Informatik in Saarbrucken, working with Stephen R. Marschner of Cornell University and Pradeep Sen, Billy Chen, Gaurav Garg, Mark Horowitz and Marc Levoy of Stanford. Here's the setup: A camera is focused on a scene, which is illuminated from another angle by a single light source. Obviously, a photograph made in this configuration shows the scene from the camera's point of view. Remarkably, though, a little computation can also produce an image of the scene as it would appear if the camera and the light source swapped places. In other words, the camera creates a photograph that seems to be taken from a place where there is no camera.

It sounds like magic, or like seeing around corners, but the underlying principle is simple: Reflection is symmetrical. If the light rays proceeding from the source to the scene to the camera were reversed, they would follow exactly the same paths in the opposite direction and return to their point of origin. Thus if a camera can figure out where a ray came from, it can also calculate where the reversed ray would wind up.

Sadly, this research is not likely to produce a camera you can take outdoors to photograph a landscape as seen from the sun. For the trick to work, the light source has to be rather special, with individually addressable pixels. Lensch *et al.* adapt a digital projector of the kind used for PowerPoint presentations. In the simplest algorithm, the projector's pixels are turned on one at a time, in order to measure the brightness of that pixel in "reversed light." Thus we return to the thought experiment where each of a million pixels in a display shines on each of a million photosites in a sensor. But now the experiment is done with hardware and software rather than thoughtware.

The Computational Eye

Some of the innovations described here may never get out of the laboratory, and others are likely to be taken up only by Hollywood cinematographers. But a number of these ideas seem eminently practical. For example, the flutter shutter could be incorporated into a camera without extravagant expense. In the case of the microlens array for recording light fields, Ng is actively working to commercialize the technology. (See refocusimaging.com.)

If some of these techniques do catch on, I wonder how they will change the way we think about photography. "The camera never lies" was always a lie; and yet, despite a long history of airbrush fakery followed by Photoshop fraud, photography retains a special status as a documentary art, different from painting and other more obviously subjective and interpretive forms of visual expression. At the very least, people tend to assume that every photograph is a photograph *of* something—that it refers to some real-world scene.

Digital imagery has already altered the perception of photography. In the age of silver emulsions, one could think of a photograph as a continuous image with a continuous range of tones or hues, but a digital image is a finite array of pixels, each displaying a color drawn from a discrete spectrum. It follows that a digital camera can produce only a finite number of distinguishable images. That number is enormous (perhaps 10^{10^6}), so you needn't worry about running out; your new camera will not be forced to repeat itself. Still, the mere thought that images are a finite resource can bring about a change in attitude.

Acknowledging that a photograph is a computed object—a product of algorithms—may work a further change. It takes us another step away from the naive notion of a photograph as frozen photons, caught in mid-flight. There's more to it. Neuroscientists have recognized that the faculty of vision resides more in the brain than in the eye; what we "see" is not a pattern on the retina but a world constructed through elaborate processing of such patterns. It seems the camera is evolving in the same direction, that the key elements are not photons and electrons, or even pixels, but higher-level structures that convey the meaning of an image.

Bibliography

Adelson, Edward H., and John Y. A. Wang. 1992. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14(2):99–106.

Bimber, Oliver. 2006. Computational photography—The next big step. (Introduction to special issue on computational photography.) *Computer* 39(8):28–29.

Gortler, Steven J., Radek Grzeszczuk, Richard Szeliski and Michael F. Cohen. 1996. The lumigraph. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996*, pp. 43–54.

Levin, Anat, Rob Fergus, Frédo Durand and William T. Freeman. 2007. Image and depth from a conventional camera with a coded aperture. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2007*, article No. 70.

Levoy, Marc, and Pat Hanrahan. 1996. Light field rendering. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996*, pp. 31–42.

Moreno-Noguer, Francesc, Peter N. Belhumeur and Shree K. Nayar. 2007. Active refocusing of images and videos. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2007*, article No. 67.

Ng, Ren. 2005. Fourier slice photography. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2005*, pp. 735–744.

Ng, Ren, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz and Pat Hanrahan. 2005. Light field photography with a handheld plenoptic camera. Stanford University

Computer Science Tech Report CSTR 2005-02. http://graphics.stanford.edu/papers/lfcamera/lfcamera-150dpi.pdf

Raskar, Ramesh, Kar-Han Tan, Rogerio Feris, Jingyi Yu and Matthew Turk. 2004. Non-photorealistic camera: Depth edge detection and stylized rendering using multi-flash imaging. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2004*, pp. 679–688.

Raskar, Ramesh, Amit Agrawal and Jack Tumblin. 2006. Coded exposure photography: Motion deblurring using fluttered shutter. *ACM Transactions on Graphics* 25:795-804.

Sen, Pradeep, Billy Chen, Gaurav Garg, Stephen R. Marschner, Mark Horowitz, Marc Levoy and Hendrik P. A. Lensch. 2005. Dual photography. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2005*, pp. 745–755.

Wilburn, Bennett, Neel Joshi, Vaibhav Vaish, Eino-Ville, Talvala Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz and Marc Levoy. 2005. High performance imaging using large camera arrays. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2005*, pp. 765–776.