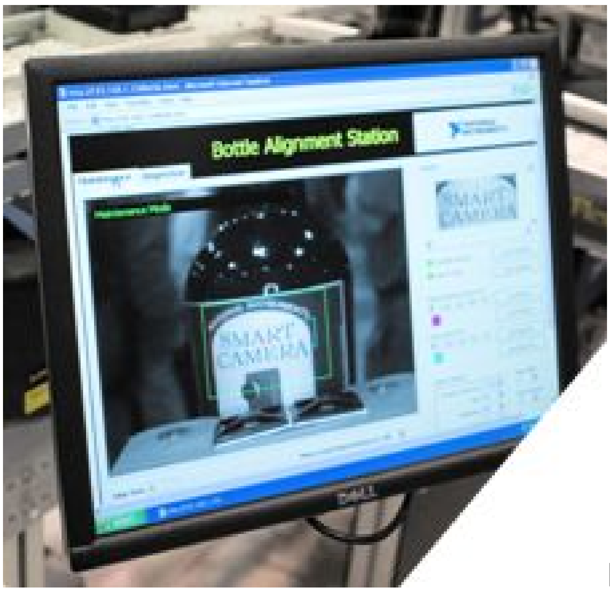
Take a Top-Down View of Industrial Imaging

Jon Titus, Senior Technical Editor

Don't get stuck on details when you plan to implement vision-system hardware and software.

by Jon Titus, Senior Technical Editor

Before you think about using an industrial imaging system to solve an inspection problem, you must exactly define the problem or problems you need to solve. To start, John Keating, product marketing manager at Cognex breaks vision tasks into four categories: inspection that detects flaws, gauging that performs a dimensional measurement, identification that reads information, and guidance that directs movement. "As engineers begin to tackle a vision problem, they must concentrate on the problems and not on the latest types of equipment and software they might buy," said Keating. "They have to take a top-down approach to solve their problems."



Evaluating the requirements for a vision system involves establishing a priority for the various tasks you plan to perform. In the area of defects, for example, suppose in a lot of 1000 assemblies you find 37 missing rivets, 17 misaligned connectors, and one broken LCD. The missing rivets seem like the "killer" defect that you should address first. A Pareto chart of defects plotted by type vs. number helps determine which defects cause the most problems. (See: For further reading.)

"Some people might want to inspect for small surface defects on a product," noted Keating. "Those 'defects' don't affect the product's performance. But inspecting for them can add to the inspection-software overhead and decrease throughput. So, surface inspections might seem nice to have, but usually they become low-priority needs." Keep in mind that human inspectors have 3-D vision, whereas an electronic camera produces a 2-D view of an assembly or product. (Multiple cameras can provide 3-D information, though.)

After you establish the priority for defect types, you can start to contact vendors. "When engineers aren't familiar with vision applications, we recommend they send us images of good and rejected parts and tell us what they want to inspect for and what they want the results to tell them," said Pierantonio Boriero, product line manager at Matrox Imaging. "They might need to check for a missing hole and measure dimensions of other features. When we have this information we can suggest the types of lights, optics and cameras they could use. But we can't say, 'Pick camera XYZ,' because the mechanics of their production line governs things such as the distance to the products and how much a camera can actually see."

"Engineers must remember the quality of the images they obtain from a vision system determines the quality of the inspection results," noted Boriero. "If you have a machined part and you need to measure hole diameters, you need images that lets vision software accurately resolve dimensions to the physical tolerance you expect. Getting good images involves an iterative process of changing camera configurations, adjusting lenses and lighting equipment, and working with vision software, and repeating the process."

"Engineers who lack hands-on experience with vision software often say, 'I have a task to do but I don’t know anything about the software available to help me,'" said Matt Slaughter, a product marketing engineer at National Instruments. "There are comprehensive guides, articles, and books that explain the different types of algorithms that detect edges, make measurements, match patterns, improve contrast, and so on. Most of the time, commercial tools simplify inspection tasks to almost point-and-click operations in a high-level software-development environment. Engineers don't have to write vision code in C or C++."

Before engineers buy vision equipment, they can obtain software tools and use their own photographic images to explore tool and algorithm suitability to a task. Processing test images also helps them determine how long it takes a computer to perform the operations they would like to use. That timing information plays a key role in the design of a vision system.

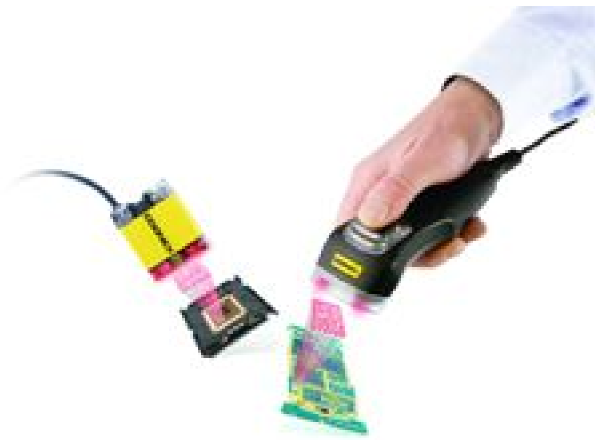

Most production lines have a set rate, so a vision system must keep pace. Suppose you plan to inspect 50 products per second. Electronic cameras can easily capture a complete image every 20 milliseconds, but can your software process each image in real time? The more vision tasks you add for each inspection, the longer the software takes. Take advantage of the priority list or Pareto chart you created to determine which inspection tasks you could eliminate to save time. "Engineers should tell us about their production cycle rate at the start of a discussion about their inspection requirements," stressed Slaughter of National Instruments. "Often, the image-processing time rather than a camera's image-acquisition rate becomes the limiting factor in how fast a vision system can go." When processing time becomes critical, developers can use software to select regions of interest in an image. Then, the software can process only those regions rather than other sections of an image that are of no interest at the time.

In some cases, an inspection task might not require a complete vision system. If engineers simply want something to happen once an object comes by, they don’t need a machine vision camera and software. Many companies make small proximity sensors as well as modules that detect color, read a bar code or Data Matrix code, or sense the presence or absence of features in an image. These devices will instantly give you information without any processing overhead. And they connect directly to programmable logic controllers and other industrial equipment.

The ease of connection with other equipment plays an important role in the choice of vision-system hardware and software. "During early planning stages, talk with your IT people," said Boriero of Matrox. "Ask them what kind of information they need and ensure your system can send it to them. People in the pharmaceutical business, for example, face the broad issue of traceability. They have to keep track of products and log inspection and tracking information. So, you must know their logging requirements." Even microprocessors and modules come with machine-readable codes, characters, and symbols for tracking and inventory control.

You also must determine how your vision equipment will interface with hardware already on a production line. Equipment, such as gates, sorters, and robots use PLCs that should easily connect with a vision system, but some equipment may have nonstandard or specialized control requirements. Along with hardware compatibility, you must ensure your vision software can handle industry-standard and not-so-standard protocols.

Production lines also have characteristics that affect the results of inspections. A production line could use fixtures to hold products in a set orientation, so images look much the same. That arrangement simplifies inspection because software does not have to account for random orientations of products. Think of circuit boards on a fixed frame versus randomly placed circuit boards and you can appreciate the problem.



Because a production line moves a product, you must ensure your camera acquires blur-free images. "From a software point of view, we assume the camera and its related hardware will 'freeze' any motion," said Boriero. "It’s difficult to analyze a blurred image. But the need to stop motion photographically has implications for the image-capture part of a system. It must trigger the camera and strobe lights at the proper moment."

You can adjust camera functions to acquire images during a short exposure time.

But as that time decreases, the camera requires more light. As you open a lens iris to admit more light, you reduce the camera's depth of field--the range of distances over which features remain in focus. In some cases, you can use a bright light source--constant-on or strobed--to illuminate products. Brighter lights let you close the lens iris and increase the depth of field so you can focus on a wider range of feature heights.

Before you get overwhelmed with specifications and tradeoffs, ask vendors for help. "There’s no substitute for having an application engineer next to you on the factory floor," said Keating of Cognex. "They can examine your products, look at good and defective parts, view the dimensions you will measure, and see what your production environment looks like and note ambient conditions. The apps engineer might say, 'A back light would be perfect but you can’t mount it where you plan to put a camera.' So you take it off a list of potential lighting equipment and with the apps engineer's help, look for another type of light source."

"You understand the manufacturing process and the production line and what you can and cannot do there," said Keating. "The apps engineer understand how to get good images and how you plan to use the image-analysis results. So you both can discuss how to inspect products based on firsthand knowledge. You can't get as quick a start by phone or email."

A first-hand look by a vision expert also will help you determine where to place vision equipment on a production line. You might find the chosen location does not provide sufficient space for the types of lights or camera you want to use. And an apps engineer can point out areas that might subject a camera to damage or quickly coat a lens with dirt or dust.

If at this point you feel a bit overwhelmed, relax. Vendors can point you to system integrators who have experience with a variety of products and who have worked with vision equipment in industries like yours. So, instead of shouldering the entire vision-system burden yourself, rely on an outside expert who can handle the entire project. Equipment vendors also can run feasibility studies and connect you with their support people.

Whether you decide to develop your vision application in house or use a system integrator, try to closely replicate production-line conditions during development work. A customer of a company in the vision-equipment business designed and produced a system to inspect labels. The system worked perfectly until installed on a production line. The labels on the production line were wet, as opposed to the dry labels the design team used as the basis for their design. So the customer had a great vision system, but couldn’t figure out how to integrate it into the wet-label machine.

For further reading

Simon, Kerry, "Pareto Chart," iSixSigma.
http://www.isixsigma.com/library/content/c010527a.asp [1].

NI Vision Concepts Manual, National Instruments, 2005.
http://www.ni.com/pdf/manuals/372916e.pdf [2].

Automated Imaging Association white papers:
http://www.machinevisiononline.org/public/articles/articles.cfm?cat=7 [3]