

## How to plan your first vision system

---

### Abstract

Developing a machine vision application for the first time need not be a headache. If you follow a thorough, three-stage process to develop, test and deploy the project, the results should provide an essential tool in product inspection and valuable insight to enhance overall product quality.



---

#### Corporate headquarters:

Canada and U.S.A.  
Matrox Electronic Systems Ltd.  
1055 St. Regis Blvd. Dorval, QC H9P 2T4  
Canada  
Tel: +1 (514) 685-2630  
Fax: +1 (514) 822-6273

# Matrox Imaging White Paper

## Abstract

Developing a machine vision application for the first time need not be a headache. If you follow a thorough, three-stage process to develop, test and deploy the project, the results should provide an essential tool in product inspection and valuable insight to enhance overall product quality.

## How to plan your first vision system

Experts in the machine vision field say the technology often is an afterthought in manufacturing systems. Adding vision is sometimes thought of as an “upgrade,” so like a home renovation, you might have to work within the limitations of your current space. There are dozens of details to consider and a lot of hard work. Even the best-planned project can hit a snag. If you’re not an architect or a contractor, you could be building for a very long time.

This article will help a first-time vision specifier understand the needs of his or her vision system, the first step in the process to develop a successful application. One way to determine the application requirements is to develop the project in three stages.

- Objectives: Sketch out the overall requirements in order to answer some basic questions.
- Experiments: Determine the equipment needed to work on a prototype in the lab or at the test bench. This is the point to use a camera to take sample images.
- Deployment: Look at how the vision system fits into the production process and choose equipment. After building a working prototype, move it to the factory floor to see how it fits.

## Objectives

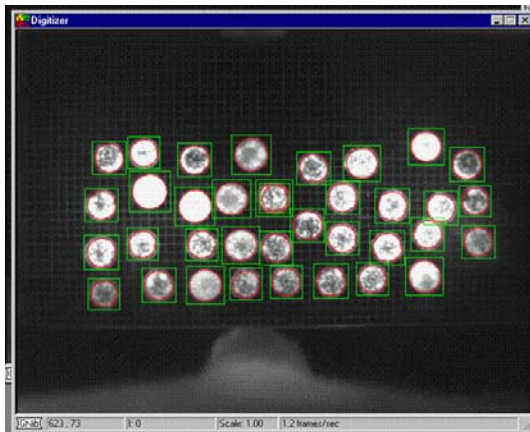
This is the time to establish the parameters and the role of a new vision system. What do you want vision to do? Do you need it to guide? For example, do you need to pass coordinates to a stage, robot, or gantry? Will the system inspect objects? For example, do you need to count pills in a blister pack or measure the dimensions of machined parts? Or, do you need to read text characters or 1D and 2D barcodes? Many applications will perform several functions, so list everything you want the vision system to do.

Determine the vision system’s expected performance in terms of its accuracy, precision, repeatability. In metrological terms, accuracy is defined as the degree to which a given measurement conforms to the standard value for that measurement. Indeed, governments oversee weights and measures to ensure instruments give accurate results. Precision defines the degree of certainty with which a measurement

# Matrox Imaging White Paper

can be stated. Repeatability is the range of variation in repeated measurements. If an object is measured ten times by different people and they get the same result, we can assume that the measurement process is highly repeatable.

But in a vision system, it's the *image* of the object that gets measured. The imaging software will use the pixels (mapped to the real-world coordinate system through calibration) to calculate the measurements of the object. An important rule of metrology (the science of measurement) is that the instrument should be 10 times better than what you want to measure. If the object has a tolerance of  $\pm 0.5$  mm, then the image's pixels must be in the order of 50 microns. Essentially, the relationship between the camera and working plane will influence the optics. And most – if not all – image processing packages offer sub-pixel accuracy, so you'll be sure to get the required precision from your images if you have the right lens.



Vision-guided robots is quite common. The camera acquires an image, and software processes the image. In this cookie sorting application, the robots place different shaped cookies into a blister tray for packaging. The imaging software will help determine the correct shape of the cookie, index it so as not to overfill the tray, and send coordinate information to the robot. Images courtesy of Bosch Packaging Technology.

Your application's expected speed is another factor you need to consider at this point. You need to know the rate your widgets will pass the camera's field of view. With the timings, you'll figure out how much time is available for processing, and your camera vendor (and later, software vendor) will be able to understand your needs.

# Matrox Imaging White Paper

Next, consider the camera and lighting. Where will they go? Determine the physical constraints and environment of the system. Be sure that the camera, and the lighting as well, will fit in the available space. The factory environment is important here too. Environmental variables are temperature, humidity, dust, vibrations, and electro-magnetic noise from DC motors. If the camera's wire/cable is too close to a DC motor or its housing, the motor's electro-magnetic noise could disrupt the transmission and corrupt your image data.

If the system is to be PC-based, determine the proximity of the camera to the computer. The cable length will determine your choices for the camera interface. This is true even for a smart camera. You'll also want to take flexing motions into account if the cable is part of a moving assembly.

Decide how you'll operate the vision system. Will it be deeply embedded or will it have a user interface? If the latter, determine the requirements for the human-machine interface (HMI). Some industries have very strict controls and require product tracking at every step in the manufacturing process. The pharmaceutical industry, for example, requires access permissions and change logs for regulatory compliance.

Your last step in the Basics phase is straightforward math. You need a budget. Estimate both up-front and recurring costs, and don't forget maintenance costs such as cleaning, lighting replacements, and regulatory compliance updates.

## **Experiment: Set up the lab**

When you know your application's requirements, then shop around and select the components. Once you have your smart camera or camera, frame grabber, PC, and illumination device you get to have some fun. It's time for the photo shoot.

To develop your application's software, you need a clear idea of what the software will "see" in the images. Take pictures—lots of them—to gather a representative set of images that show the full range of situations (i.e., defects) that could occur. This set of images will define how the scene or object can change over time. It's the defects you want the vision system to find. If, for example, you're inspecting machined parts, be sure to acquire images of burrs, parts that are bent, parts with too-small openings and other significant defects.

Examine the images carefully. Take note of shadows (dark regions), reflections (bright spots), or uneven lighting. The human visual system is fine-tuned to spot irregularities in images, but a computer isn't. For example, if the software is looking for edges, an object's shadow might be misinterpreted as an edge. A

# Matrox Imaging White Paper

reflection could be identified as a blob. A picture is only as good as its lighting. Depending on what appears in your images, you might need to tune or reconsider the illumination setup.

Armed with a complete set of images, you will be able to confidently analyze them. You can put your requirements into concrete terms that will help determine the kind of machine vision tools (algorithms) you'll need. Imaging algorithms for machine vision applications generally fall into three categories—for locating, measuring, and reading.

Locating tools include pattern-recognition, pattern-matching, pattern-search algorithms, and blob analysis. They are more examples of the superiority of the human brain—we easily see the object in an image, but a computer needs a little help. A locating algorithm determines the coordinates of an object so other analysis functions have a reference point. Locating algorithms also help speed up the processing for other measuring and reading functions by closing in on an area of interest.

Algorithms used for measuring would be measurement, metrology, edge-and-stripe, and blob analysis (some tools have multiple uses). Measurement tools are quite capable of measuring geometric features and allow you to set tolerances to sort the conforming parts from the defective ones. These tools are indispensable for many applications, especially for machined parts. If you are measuring objects and want results in world units, calibration tools will also find their way into your toolbox. Most machine vision applications make use of a calibrated coordinate system.



The Matrox Imaging Library Metrology module analyzes the image of a machined hook. Based on its features, the metrology results determine that the part falls outside the accepted tolerances. Image courtesy of Matrox Imaging

# Matrox Imaging White Paper

Alpha-numeric characters come to mind for algorithms that perform the reading functions. Machine vision reads characters for two purposes. The first is OCV, optical character verification, which determines the presence or absence of specific printed text such as an expiration date. The second is OCR, optical character recognition, which actually reads the characters and returns them as results. In machine vision, reading also can refer to 1D and 2D codes, or more specifically, both bar and matrix codes.

Machine vision specialists recommend using off-the-shelf tools instead of creating algorithms from scratch. The Matrox Imaging Library (MIL) is just one of several image processing packages available, and the well-known ones are built based on field-proven technology. Developing and maintaining algorithms is extremely time-consuming and expensive. A vendor might have a large team of highly-skilled and experienced developers working on image processing algorithms. If you choose to buy instead of build, you will spend your time developing your application—not creating algorithms. Consider that a particular vision problem typically has more than one solution, and an image processing package will give you many options. Frankly, the algorithm (or algorithms) used in the solution must be designed to catch the anomalies.

## **Deployment: Move it to the factory floor**

This is the time to start building the machine. The prep work is complete and the materials are assembled. Now consider the vision system's role in the manufacturing system, or perhaps in the entire enterprise. What will you do with the imaging results? How must the vision system interact with other equipment? What happens to a part that fails inspection? Will you blast it with an air jet? Will you instruct a robot gripper to pick the object off the line? These mechanical issues will shape the physical design of the system. On the back-end, what will you do with the results gathered from the vision system? Will they be used to make real-time decisions, for example, to activate an ejector? Do you need to keep statistics in order to identify trends? Do you need to archive the images for regulatory compliance? When you're at the point of answering these questions, you're well on the way to building your prototype; the system validation must be done in-process and not just in the lab.

It even might be necessary to take a few steps backward and revisit your camera setup. The camera, optics, lighting and algorithm-selection process is iterative. You might find the chosen algorithm doesn't work properly in your setup. For example, 2D code reading will work best if the minimum element size is three pixels tall and wide so the camera setup needs to resolve to this level.

# Matrox Imaging White Paper

## Is there a Stage 4?

Performing automated inspection with machine vision techniques is accepted across myriad industries. It has great potential to reduce long-term costs and improve the quality control of your product. Remember that vision is not meant to, and will not, fix your product. Its purpose is to ensure the product's quality. Over time, it might help define flaws in the manufacturing process.

## Conclusion

Implementing vision is not a decision to be taken lightly, and the DIY approach requires expertise and time. Are you prepared for the work that's involved? If not, consult a system integrator who specializes in machine vision. These integrators will be able to guide you through the process. They have the experience and foresight to prevent bad choices. Machine vision's complexity can be overwhelming, and working with an expert will ensure a successful deployment.

To learn about Matrox Imaging machine vision products or to locate an Authorized Integrator, visit [www.matroximaging.com](http://www.matroximaging.com).

## References:

DeVries, Warren R. Analysis of material removal process. New York: Springer-Verlag ,1992.