Watch Where You're Going! Motion Control Precision Through Vision

J. Randolph Andrews
Douloi Automation
740 Camden Avenue Suite B
Campbell, CA 95008-4102
(408) 374-6322

Paper Presented at the
1995 Incremental Motion
Control Systems and
Devices Symposium

Abstract

Motion control systems have grown in sophistication and precision. However the precision of an entire machine process, for example an assembly operation, involves factors beyond the movement of the controlled mechanism. The initial location of parts prior to handling is subject to a tolerance often much greater than the placement tolerance. The correct position of a part is relative to the final assembly, not the part handling mechanism.

When precision beyond the motor shaft is required an "outer loop" or "dual loop" approach is often taken to improve machine performance. Measuring and controlling the resultant position of a handling mechanism is more precise than just controlling the motor which drives it. Even better than measuring the resultant position of the mechanism is measuring the position of the part being handled itself. This ultimate outer loop control can be implemented with machine vision to monitor part position and target placement positions resulting in improved placement performance.

Introduction

Although once an area of research, machine vision is now a commercially available component that can augment motion systems and provide higher levels of system performance if properly integrated.

Machine vision can be used in a number of ways. Often vision is contributing high resolution position information to increase placement accuracy in assembly systems. Vision can also do course-positioning of less fixtured items to prepare them for grasping or having information read from a particular section of the part. Vision can be used to read part identification numbers, such as the characters placed on semiconductor wafers and tracked during their manufacturing process. In some cases vision is used for part inspection with no further use of the information positionally than to confirm that all the geometric features that were expected were actually found.

Machine Vision History

An important problem which motivated development of machine vision for robotic motion systems was the "Bin Picking Problem". Manufacturers first using industrial robots placed robots in the work locations of human workers with the expectation that the same environment could be used by the robot. Bins of parts presented for the next manufacturing step often held parts in a stacked, disoriented arrangement. This situation is shown in figure 1.

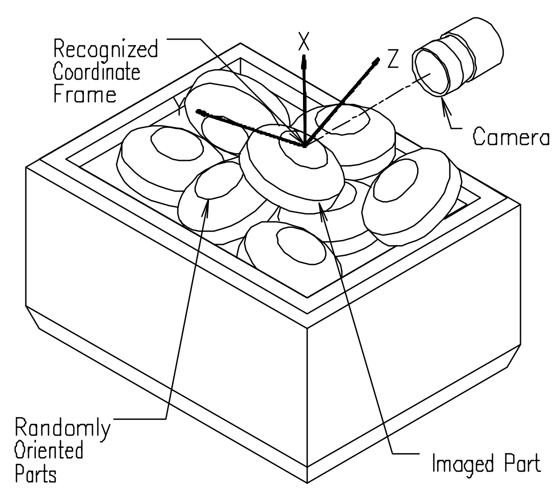

Figure 1. Bin Picking Problem

In an effort to have the robot mimic the behavior of the human, work was done with cameras to view the bin of parts, recognize which part was on top, and to discover it's orientation well enough to grasp and handle the part.

This is not an easy problem. The problem was popular in academic settings and even demonstrated at robot trade shows in it's most general form. Some commercial systems provide this type of capability. However, the practical conclusion to the bin-picking research was that it is best not to throw parts into bins in the first place. It is simplest to retain at least the rough position and orientation of the part in a tote or pallet through the manufacturing process. Nevertheless, machine vision as a technology made progress.

The outcome of research and development directed to this problem has provided tools to handle more common commercial applications. These cases are much more controlled than the general bin picking problem being primarily 2 dimensional in nature, not 3 dimensional, and more restricted with regards to lighting and possible part orientations.

Machine Vision Strategies

There are many ways to use an image and distill information. The two main techniques used in this discussion include model-driven vision and data-driven vision.

Model Driven Vision

In model driven vision a nominal example image is shown to represent the ideal or preferred case. An "averaged" model might be composed by looking at many examples. Subsequent images are then compared to the "model" image. Differences between the model and run-time images can be determined to provide information such as a position and angular offset. A positive aspect of the model approach is the ability to use a complex image with features which might not be particularly distinct. A negative attribute is that the information that can be extracted is relatively limited since the image is never geometrically "understood" by the vision system.

Data Driven Vision

In a data-driven vision system a geometric expectation is described. The vision system then analyses the image to identify geometric features in the image. These features are compared against the expectation to determine whether the image is a match and what positional and rotational differences there might be. This approach is most often used on 2 dimensional images that have high contrast through the use of thoughtfully designed lighting system. The contrasts helps the vision system distinguish edges and features of the observed parts. Data driven images can manage more geometric detail than model driven approaches but require more lighting consideration to produce an image that can be reliably interpreted.

Vision System Architectures

How is vision processing and analysis accomplished? Vision processing has followed the same technology trend as motion control for many of the same reasons. Some vision systems are separate computers in their own enclosures, similar in architecture to a stand-alone motion controller. The stand-alone approach has been followed by board-level vision co-processors which plug directly into the host computer, such as an IBM PC. This approach is comparable to a motion-coprocessor board. This approach can be more cost effective than the stand-alone approach and takes advantage of the host computer's open architecture. The host then communicates to the board to gather information about the analyzed image.

A more recent step being taken in commercial vision systems is the use of low-cost "frame-grabber" boards which simply collect the image and make the image available to the host computer. Vision algorithms running on the host computer itself process the image and present the results to the requesting program, also running on the host. This is similar to motion controller architectures which use the host computer for all of the motion system processing. The benefit of this approach includes high value since the "interface" hardware is less expensive than co-processor hardware, high performance since PC platform capabilities continue to increase, and simple communication since the software often connects through standard vehicles such as Windows DLL libraries. However a suitable pre-emptive real-time operating system must be present to properly share the host computing resource.

Motion Benefits of Machine Vision

Using machine vision allows a shift in perspective from moving a mechanism to moving a part. This perspective acknowledges tolerance issues in part presentation and assembly fixturing. The technique provides a way to compensate for those tolerance buildups.

Placement precision of a non-vision assembly machine is influenced by the part presentation fixtures. Mechanical fixturing of parts has a number of limitations. Some part presentation techniques, such as bowl or tube feeding, require clearance in the part presentation tooling to allow the part to easily slide into position. This prevents the tool from providing consistent part position, particularly if it is a vibrating feeder.

Some parts being presented are quite fragile. For example certain QFPs, Quad Flat Package electronic parts, must be handled in trays to protect the fragile leads from touching anything that might cause them to bend. These trays do not provide enough mechanical registration to precisely place the part, and the trays themselves are subject to tooling tolerances that influence knowing precisely where the tray itself is located.

In some cases, the desired feature is not even mechanically available. In one application example, an optical component handled in an assembly application contained an important optical feature with great relative precision that had a poor tolerance to the physical outside edge of the substrate. That precise optical feature had to be placed on another part with a tolerance of +/- 2 microns. Using mechanical features on the outside of the optical part was not an effective approach because the feature-to-outside tolerance was much greater than the assembly tolerance. The important optical feature had to be imaged directly. That information was used to guide the placement.

In some cases, the part presentation tooling can be precise but its own position might not be well known. Some machines have part presentation “magazines” which swap into position quickly when the previous magazine becomes depleted. In this case the part presentation fixture has a compound tolerance, the tolerance of the part in the fixture, and the tolerance of the fixture with respect to the machine itself. These two tolerances combine and must both be held to achieve part presentation tolerance without vision.. Total tool cost can be expensive because of the large number of part presentation magazines on an assembly machine and the need to have even more off the machine getting prepped for loading.

The Vision to Motion Connection

To benefit from the information coming from the vision system there needs to be a vision system to motion control connection. This connection involves the following issues.

Communication

There has to be a communication link between the vision and motion system. Generally the vision system is not being used in a real-time manner although the vision information is influencing placement positions. Typical vision analysis times range in speed from 50 milliseconds to a second depending on the nature of the algorithms, size of image, and computation engine. Because the analysis times are relatively large compared to motion controller sample rates, high speed communication, in the motion control sense, is usually not required. In general, information coming from the vision system can be as simple as a position offset and rotational offset. This is a small amount of data and does not require high speed communication.

Serial ports are a ubiquitous resource available for communicating to a stand-alone style vision system. Some data-driven vision systems require a large amount of information to describe the shape of a part and benefit from higher speed communication than the typical serial port.

Microsoft Windows is becoming more popular as a tool for solving automation problems. Third parties have contributed hard real-time extensions to Windows 3.1 that provide the pre-emption necessary for machine control. As well, newer versions of Windows have pre-emptive architectures that are more appropriate for automation than previous versions.

One reason for Windows popularity is the convenience of using Windows Dynamic Link Libraries to communicate to automation components such as vision systems and motion systems. These libraries contain a list of commands that the component will respond to. All of the details of communicating between the application program and the component are accomplished by the library provided by the component manufacturer in conjunction with the Windows operating system. An example connection to a DLL is shown below in figure 2.

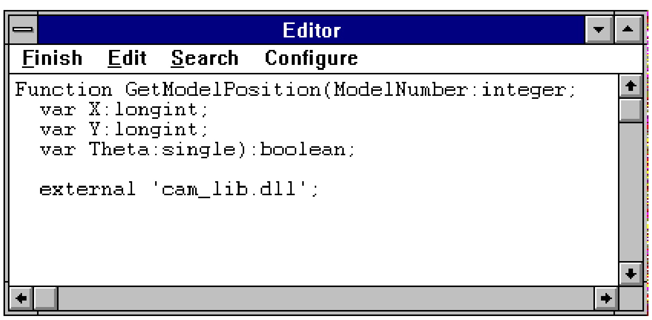

Figure 2. Connecting to a DLL

The name of the function is "GetModelPosition". We provide a model number as a parameter and receive back an X, Y, and Theta offset. The function also returns a boolean condition indicating if the function executed correctly. The connection from this description to the actual software which performs this function is accomplished with the "external" keyword. This keyword indicates that the function is in a separate place, a library name "cam_lib.dll".

Additional functions are needed to initialize the vision system and perform training. They connect in a similar manner. Using a DLL can be much simpler than getting a serial port configuration correct.

Windows DDE is another communication link available with Microsoft Windows allowing an automation application to communicate to a concurrently active vision application which is managing the vision system.

Kinematics

In this discussion, kinematics effectively means hand-eye coordination. A motion system must understand what the spatial consequences are to information that is seen by the vision system.

Vision augmented motion systems sometimes use multiple cameras. Some cameras are stationary, mounted to the base of a machine. Other cameras can be moving, mounted alongside an end-effector and travelling with the moving mechanism. Figure 3 shows an example mechanism that might have such a configuration.

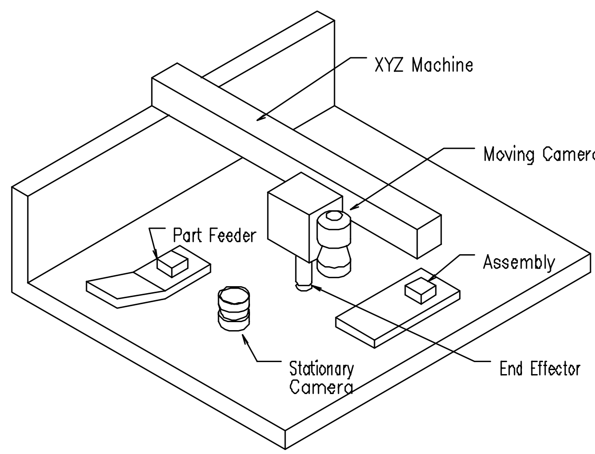

Figure 3. Robot with Multiple Cameras

In order to represent the various elements of the system coordinate frames will need to be described. The following typical coordinate frames are often required to describe how the different parts of the machine relate to one another.

Machine Frame

The machine frame is the most obvious coordinate frame. The X, Y, Z movement of the machine is controlled through coordinate commands to the motion controller. Not all mechanisms that use vision are linear XYZ mechanisms.

However it is common to resolve and describe movement in Cartesian terms so as to have a common spatial language for relating the different coordinate frames to each other.

Stationary Camera Frame

The stationary camera has a coordinate frame, most natively expressed in units of "pixels". Usually a stationary camera is looking upward at the bottom of a part that might be held by an XYZ pick and place mechanism. This frame has a different scale factor than the machine frame since pixels do not directly correspond to encoder counts on the mechanism. The camera frame probably does not have the same scale factor in both directions since camera pixels are generally rectangular in shape rather than square. The camera may also be rotated with respect to the machine frame.

Moving Camera Frame

A moving camera in this type of machine might be pointing downward so as to view features of the assembly base which parts are being placed on. This moving camera frame is concatenated onto the end of the machine frame and may have its own scale factors, offset, and rotation.

Assembly Frame

The assembly base itself may not be precisely located. Because the system has a downward camera and can see the assembly base it is possible to adjust part placement coordinates to compensate for an offset or rotated base. This requires describing an assembly frame, relative to the machine frame, which is used to compensate for this offset and rotation.

It is very helpful if the motion system allows you to describe these frames and describe movement of different objects with respect to these frames. Most motion controllers provide very good ways of describing movement of the mechanism itself. An example command might be

```
Robot.MoveToVector(PartPickup);
```

It is very useful to use commands such as:

```
DownwardCamera.MoveToVector(PartPickup);
```

This type of command is as easy to use as native motion commands, but the details such as offsets and rotations are taken care of for you.

At first glance the kinematic problem appears fairly simple being only 2 dimensional. However a number of attributes can make frame management confusing. The camera frames generally do not have the same x and y scale. Upward cameras see rotations in the opposite direction as the downward cameras because of the view point It is important to be able to explicitly describe these frames and have the motion software know how to relate them to each other.

Calibration

Even though a motion system may have the ability to represent different coordinate frames in a machine, the machine will not be very accurate until the exact numeric values representing those frames is discovered. The process of discovering the physical relationships between frames is referred to here as frame calibration.

Although it may be possible to theoretically calculate distances between camera mounts and end effectors etc., it is almost always better to use the motion system to have the machine calibrate itself. Mechanical tolerances, assembly precision, and optical adjustments make the theoretical approach perform less effectively than direct measurement.

The following calibration example illustrates how to find the scale factor between stationary camera pixels and machine frame user units. The robot presents an object to the stationary camera which notes where the object appears to be. The motion system then moves the object, well within the field of view, a specific distance in X and Y. The stationary camera then views the object again to see where it is relative to it's own camera frame.

Causing a physical effect with the motion system, and observing the effect with the camera system enables establishing a relationship between the two. This is the basis of most vision to motion system calibration.

Calibration becomes more complicated when a mechanism's effector cannot be directly seen. For example, how can the offset of the moving camera be found relative to a rotating end effector in the robot? This unknown offset is shown in figure 4.

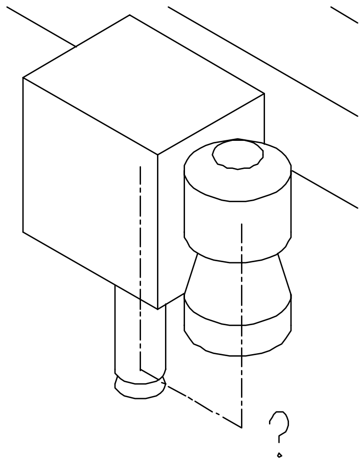

Figure 4. Unknown Camera to End Effector Offset

One approach is the following procedure. Place a disk with a visual feature underneath the downward camera and note it’s location relative to the camera. Then use the robot to move over the disk, pick it up, rotate it 180 degrees, place it back down, and go back to the position where it first viewed the disk. Because the disk was spun 180 degrees, the actual axis of rotation of the robot head is indicated as the average position between the first image and second disk image. The difference between this average position and the center of the camera frame is the correction required to accurately know the camera to head offset. This particular procedure can be used to gain X and Y offset information. Additional measurements would be required for scale and rotation.

Calibration tricks such as these are very mechanism and sensor specific. It is a good idea to consider how calibration will be performed before the machine design is complete to insure that calibration is possible. It is also helpful for any required calibration tooling to be designed and most preferably included in “docks” on the machine itself so that the mechanism can retrieve the tooling and calibrate itself without requiring operator or maintenance intervention.

Synchronization

Another important aspect of the vision system to motion system connection relates to synchronization. In general, the motion system is responsible for presenting a part to a vision system. Once the snapshot has been taken of the part, the motion system is free to move the part while the vision system is performing the analysis. It is valuable for the motion system and the vision system to be able to act independently so that they can both perform work in parallel and save time. This parallel activity can be done even if both vision and motion are running on the same processor if a suitable real-time system is being used. Sometimes the final destination of a move cannot be known until the completion of vision analysis, however the mechanism can at least start moving to the correct “ballpark” area with correction-on-the-fly as the information is disclosed by the vision processing.

Additional Vision Requirements

Beyond issues that relate the vision system to the motion system, vision systems themselves have additional requirements.

One of the most important issues is lighting. Although remarkable things can be done with image analysis software, it’s always better to provide quality, unambiguous information to start with. Proper lighting can be the most important contribution to a quality image and consistent, reliable operation of the machine. Many special lighting “tricks” and techniques are available to emphasize important features and to help separate the relevant from the irrelevant. Ring light assemblies and appropriate incident angles can make subtle features, such as small ball-grid array bumps, become striking and visually distinct. The vision vendor and their support staff is the most valuable resource in finding good lighting.

Some vision vendors provide their own integrated lighting systems. In some cases, the vision algorithms themselves manipulate and alter the lighting to enhance the image in a way the algorithm can detect as valuable. These controlled lighting systems add expense to the vision system but can provide a higher level of performance which may be necessary to meet a specification.

Proper selection of optics is also very important. Preliminary, and perhaps even satisfactory results can be achieved with a casual lens and optics selection. However, certain optical distortion effects can occur if the lens system is not thoughtfully chosen. For example, if the field of view is large and the camera too close, a "fish-eye" effect can occur. Even if the effect is not visually noticeable, it can impact the precision of the vision information. Some vision systems can compensate for distortion effects by modelling the distortion with additional calibration.

However, as with the issue of lighting, although it is possible to compensate in software there's a great simplification that comes with having the original image be as spatially accurate and useful as possible. Sometimes the problem can be solved by simply having more distance between the camera and imaged part if room is available. It is best to consider mechanical requirements of the optics early in the machine design. Focal distances and camera mounting schemes are influenced by the machine layout and design dramatically. It's much harder to find an extra 6 inches for an improved camera position after the machine is built than before it's designed.

Tips

In many ways vision companies and motion control companies have the same history from a product use viewpoint. In the past the vision or motion function was regarded as an independent component and the details of communicating to that component are left in the hands of the customer trying to use it. At this point in time, for both motion system and vision systems, the supplier should be able to offer clean and simple ways to communicate to their products with little or no effort on the part of the user.

Beware of vendors who are so generous as to give you the source code of communication solutions that have worked in the past for relating to their products. It should not be necessary to dig through source code to find a solution. Look for a clean, documented connection such as Windows DLL, Windows DDE with command set, or a serial port driver and command set. In 1992, not many vision companies had pre-assembled communication tools. In 1995, many more have this problem solved and packaged for ease of use.

Make sure that the vision vendor you are considering is available to provide on-sight support as well as a product. Because of the lighting and machine issues that can impact vision systems it is hard to replicate your situation in the vendor's lab. The insights and suggestions of on-sight support can dramatically benefit system integration and make or break a project schedule.

Summary

Machine vision has come a long way since the academic root from which it started. For certain classes of applications, limitations of mechanical fixturing and machine precision can be overcome with the benefit of vision based sensing and feedback. With the appropriate software "connectors", commercially available vision systems can be incorporated into PC based motion systems without the cost penalty often associated with proprietary combined motion/vision systems.

Bibliography

1) Andrews, J. Randolph: "A Software Component Library for Motion Control", In *Proceedings of the Twenty Second Annual Symposium on Incremental Motion Control Systems and Devices, San Jose, CA,* 1993.

2) Andrews, J. Randolph: "An Advanced Motion Control System Architecture Based on a 386 PC", In *Proceedings of the Twenty First Annual Symposium on Incremental Motion Control Systems and Devices, San Jose, CA,* 1992.

3) Andrews, J. Randolph: "A Servo Application Development Environment for Microsoft Windows", In *Proceedings of the Twenty First Annual Symposium on Incremental Motion Control Systems and Devices, San Jose, CA,* 1992.

4) Cox, Brad: *Object Oriented Programming: An Evolutionary Approach*, Addison-Wesley Publishing, 1986, 1991

5) W. Brogan, *Modern Control Theory*, Prentice Hall,

Englewood, New Jersey, 1991

6) Budd, Timothy:*An Introduction to Object-Oriented Programming* Addison-Wesley, Massachusetts, 1991

7) Coad, Peter & Yourdon, Edward: *Object-Oriented Analysis*. Yourdon Press, New Jersey, 1990

8) Ellis, George: *Control System Design Guide*, Academic Press, San Diego, 1991

9) Franklin, Gene & Powell, David: *Digital Control of Dynamic Systems*, Addison-Wesley, Massachusetts, 1981

10) Meyer, Bertrand: *Object-oriented Software Construction*, Prentice Hall, New York, 1988

11) Peters, Lawrence: *Advanced Structured Analysis and Design*, Prentice-Hall, New Jersey, 1987

About the Author

J. Randolph Andrews received his B.S. M.E. in 1981, B.S. E.E. in 1981 and M.S.M.E. in 1983 from the Massachusetts Institute of Technology.

Andrews spent 4 years at Hewlett Packard's corporate research laboratory in the Applied Physics Research Center as well as the Manufacturing Research Center.

The following 4 year period was spent with Galil Motion Control.

In July '91 Andrews founded Douloi Automation to provide motion control hardware and software systems for use with Microsoft Windows. Douloi Automation also provides turn-key automation solutions for advanced automation applications.

Professional interests include motion control, software/ electrical/mechanical system design trade-offs, high abstraction programming and visual programming techniques and tools.