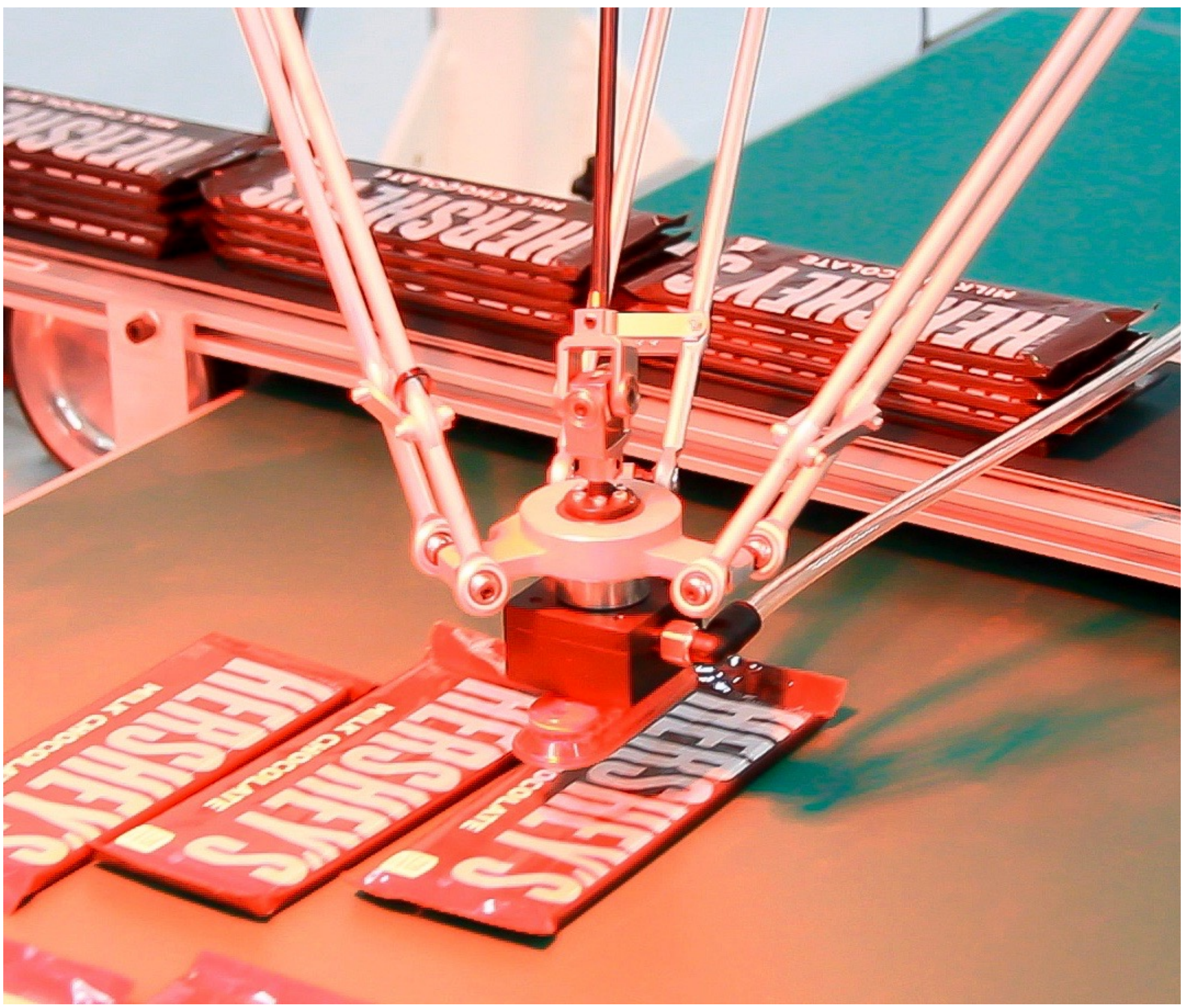
Bringing Sight to Your Production Line With Vision Guided Robotics

Since Henry Ford installed the first moving assembly line in the early 20th century, innovative manufacturers have constantly looked for ways to make their manufacturing processes more time and cost effective. With the development and installation of feedback controllers in the 1930's and 1940's, the concept of automation began to emerge. Today, as equipment becomes more intelligent, automation concepts continue to evolve and the use of robotics has increasingly replaced much of the mundane or unsafe human interaction on manufacturing lines.

Table of Contents



Preparing for the Factory of the Future

Now, as the rise of smart machines and cyber-physical systems brings us into the fourth Industrial Revolution (or Industry 4.0), the tasks being performed by robots on the line are becoming progressively more complex. One feature now enabling robots to perform even more sophisticated tasks is the addition of vision capabilities. Known as vision guided robotics (VGR), these robotic applications are fitted with one or more cameras that provide feedback signals to more accurately move a robot to a variable target position. The addition of visual guidance instills a new intelligence into modern robots that allows the robots to identify, locate, and handle most material parts. This white paper provides a general overview of key considerations for developing VGR applications and how installing a VGR solution to your production line can bring a multitude of benefits to your organization.

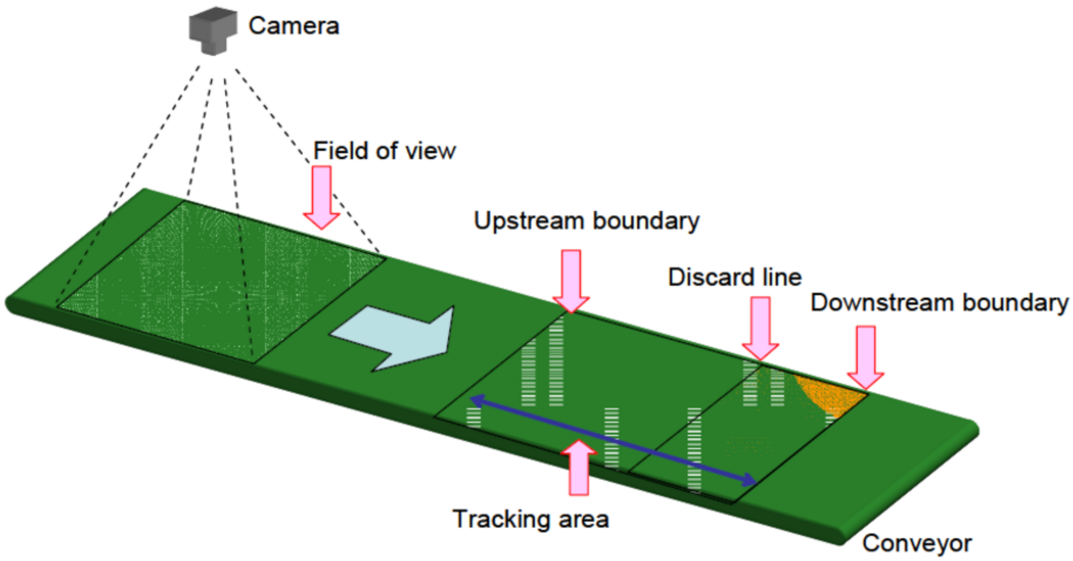
While robots are currently being used to automate a lot of repetitive or potentially hazardous tasks on production lines today, there is typically still a need for human involvement.

Production Prior to Vision Guided Robotics

When requirements on a production line call for multiple tasks, varying part placements, complex motion, or sophisticated interactions, the capabilities of a robot offer distinct advantages over traditional hard tooling. While robots are currently being used to automate a lot of repetitive or potentially hazardous tasks on production lines today, there is typically still a need for human involvement for some tasks, such as ensuring parts are not randomly placed on a conveyor in a way that the robot cannot inspect them. There are also instances where there are too many variables with a part for a standard robot to accurately operate. For example, in a bakery, cookies typically aren't all the same size or positioned the same way when coming out of the oven and down a conveyor for packaging, so a robot cannot consistently and accurately package the cookies. However, a task like this could be automated by adding vision capabilities to the robot.

A High-Level VGR System Overview

Before we jump into the specifics of developing a VGR system, let's start by taking a high-level look at one. This diagram shows a diagram of a typical production line VGR system setup.



Basic set-up and boundaries of a typical VGR system for a manufacturer.

Let's walk through this setup using an example application that a candy bar manufacturer might use. With this setup, the infeed conveyor belt brings the candy bars down the line at a predefined rate. When a candy bar reaches the camera's field of view, the camera captures an image of the candy bar, notes its location on the conveyor belt, and feeds the coordinates of the candy bar's location on the conveyor to the robot. Once the candy bar reaches the upstream boundary of the robot's reach, the robot can identify the target, pick it up, and stack it on the next conveyor belt, advancing it to the packaging machine. If the candy bar is not picked prior to reaching the downstream boundary for any reason, it will be sent to an area where all of the "faulted" candy bars are collected. Let's take a closer look at how to get started with developing a VGR application and the key components and characteristics that will optimize the performance of your VGR application.

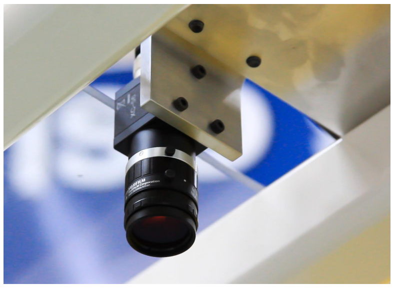
Development of a VGR system requires experience in both vision systems and robotics. Selecting an expert system integrator to develop your VGR application can save you a lot of time and money in the long run.

Getting Started With VGR: Consider All Your Variables

When developing a VGR application, precision is key. Therefore, it is essential that you consider all of your variables from the very beginning to ensure you have the most efficient and effective system. Getting it right the first time can save you a lot of headaches and debugging later. It is also important to recognize that the development of a VGR system requires experience in both vision systems and robotics. Selecting an expert system integrator to develop your VGR application can also save you a lot of time and money in the long run. A good system integrator will want to develop a thorough understanding of your application up front and should challenge you with the following questions right away:

1. What is your part rate (parts/minute) and your desired throughput for this system?
2. What is the size, shape, and weight of the part the robot will need to handle?
3. Will the parts be given to the robot in a randomized order or systematically?
4. What level of inspection beyond simply identifying the part will be required (e.g. defects, color, shape).
5. What will the part be placed into?
6. What do we do with the parts downstream that the robot can't pick up?

The answers to these questions will dictate the key components and specs of your VGR system, and while your system integrator will guide you through the selection and configuration of the system, it is important for you to understand the process as well. For example, the throughput will determine the rate at which the camera needs to acquire images and how quickly the robot needs to react. Which leads to one of the most important considerations for developing a VGR application—camera selection.



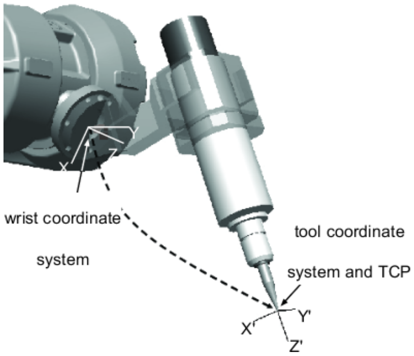
This area scan camera is used to capture images of candy bars moving down a packaging line.

Help Your Robot See More Clearly: Lighting Considerations and Camera Requirements

Lighting is critical for any VGR system and should be discussed early on in the design process. Any change in lighting can skew the robot's vision, especially if the camera is performing color identification tasks. Various types of lighting that can be used on the system include ring lights, diffuse lighting, diffuse-on-access lighting, low-angle dark field illumination, or even infrared lighting under the conveyor. The lighting in the room where this system will be located, including any natural light, also needs to be evaluated because this light can skew the way the camera sees the parts as well. Additionally, when you think about lighting, the color of the conveyor belt needs to be taken into account, especially if color subtraction will be performed. For example, if you have an application where the robot needs to pick up and move all of the blue parts to a certain location, you cannot use a blue conveyor belt because the robot will not be able to distinguish between the blue parts and the blue conveyor belt.

When it comes to selecting the camera itself, beyond the lighting, there are a variety of considerations for selecting the most appropriate camera for your VGR application. First, and probably most obvious, is that you need to know the size of the space you plan to insert the camera in and if there are any mechanical restrictions. Next, determine the required field of view for the camera and the type of inspection you will be performing on the parts. This will help you decide if an area or line scan camera is the best fit. Area scan cameras are used to inspect discrete components while line scan cameras are best for inspecting single lines to build a 2D image (e.g. long rolls of paper or fabric). Or, if your application needs to pick up parts randomly oriented in more than two-axis of position (such as selecting parts from a bin), you may need to consider using a 3D camera.

Resolution is also a key consideration for your camera. First of all, don't make the mistake of assuming you need the highest resolution camera available. Take some time to analyze what will be best for your application. To start, calculating the resolution requires



Teaching the TCP

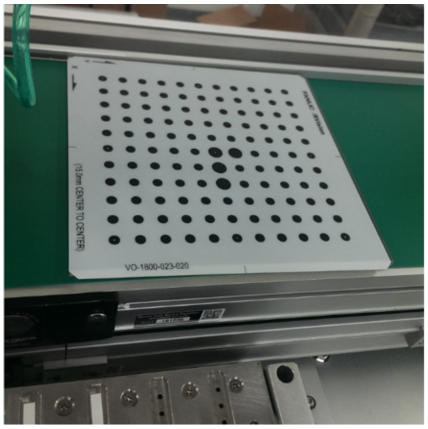
knowing the size of the field of view and knowing the smallest feature, which could be an item that is supposed to be located on the product or a defect that is not supposed to be there, that needs to be detected. A good rule to follow is to have 3 pixels to cover the area of the feature. For example, to detect a feature as small as .010 of an inch, choose a pixel size three times smaller than that, or .003 of an inch. Depending on your application, camera resolution can be as low as 640 x 480 to 1280 x 1024 (1080P) or even higher.

The speed, or imaging rate, of the camera is another important factor. The camera needs to acquire images at a rate that is at least equal to the number of components that will be passing it, but, it's best to select a camera that has a slightly higher acquisition rate to allow for a slight margin of error. Additionally, if color detection isn't a requirement for your application, a monochrome camera could be a good money-saving option.

Ensuring Accuracy With Proper Calibration

Perhaps one of the most important aspects of the entire system is the way the camera and the robot communicate. Typically, this bridge between the vision system and the robot is a shared coordinate system where the camera identifies a target and converts the position of that target from camera coordinates (pixels) to real-world coordinates for the robot. Because this communication needs to be performed with micro-millimeter accuracy, proper calibration of the entire system is crucial. For example, when using FANUC's iR PickTool and iR Vision in your application, follow the following five steps to calibrate the system:

1. Teach the tool
2. Teach the robot the tracking frame
3. Teach the camera the vision frame
4. Teach the reference position
5. Fine tune the system



An example of a calibration grid placed on top of the infeed conveyor.

Teach the Tool

Before you start calibrating the robot and camera, you need to “teach” the tool you will be using on the end of the robot’s arm by defining the tool center point (TCP). This is critical because the TCP is the point from which all of the robot’s positioning is determined. Defining the TCP is done by placing a teach pointer on the end of the robot’s arm that will stay in one location while the robot’s arm moves around through its various positions.

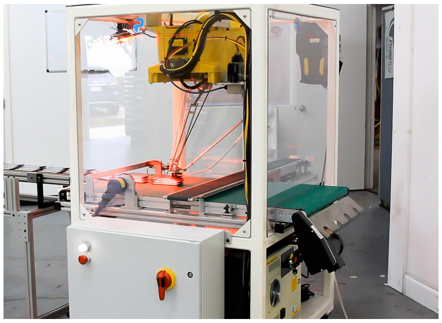
Teach the Robot the Tracking Frame

Prior to calibrating the robot’s tracking frame, you need to install an encoder that tracks the movement of the conveyor belt. Typically, the encoder measures the number of axis turns of the belt, and this number is then converted into a real-world measurement. For example, 120 turns of the belt measured by the encoder might be equal to a 5 mm movement of the conveyor belt. This conversion is critical for the robot to track distance on the conveyor belt.

Next, the robot itself must be calibrated to the working surface. Using a calibration wizard, such as the one in FANUC’s iR Vision, start by laying down the calibration grid on the working surface under the robot. Next, have the robot touch the origin point on the grid, then, systematically move through the grid from a positive X to a positive Y positions, as far as the robot can reach. Going through this for the entire area where the robot can reach produces the tracking frame for the robot.

Teaching the Vision Frame

To teach the vision frame, with the inspection/camera light on, and iR Vision providing a live image, place the same calibration grid that was used for the robot’s tracking frame on top of the infeed conveyor, directly below the camera. Manually position the grid so the center is lined up with the center of the crosshairs in iR Vision. The camera will snap a photo of the grid and you will need to use iR Vision to develop the connection between what the camera sees and what the robot interprets.



An example of a full VGR system being used to inspect candy bars prior to packaging.

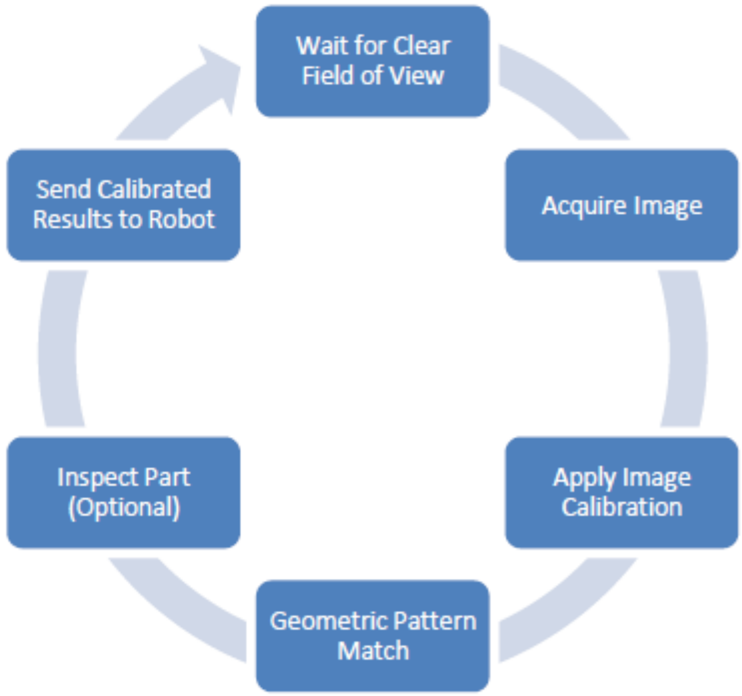
Teaching the Reference Position

After you calibrate the camera and robot, you then need to teach the robot all about the part that needs to be picked up. Using our candy bar example, if the robot needs to pick up and move the candy bar from one place to another, to start, you need to place the candy bar in the robot's arm exactly how you want it to be picked up every time, and you need to store that position as the robots reference point. This will teach the robot that every time a candy bar comes down the conveyor belt, it needs to adjust to its reference position to properly pick-up the candy bar without damaging it.

To teach the reference position, turn off the conveyor belt and place your subject, such as a candy bar, under the camera. This is the reference part that will be used to teach the reference position in the iR Vision software. Take a photo of the part, and if it's successful, you will see the coordinate data popup. Next, move the part downstream until it reaches the robot's upstream boundary without touching it. Once it's in the robot's working envelope, touch the candy bar with the robot's end-of-arm tool, but do not pick it up. This becomes the reference position for the robot that is added into the robot's code. Essentially, this is how you teach the robot to see.

Until this point, all of the calibration steps have been performed in a stop-and-go manner, but you will want to fine tune the system at a more realistic pace, yet using a method that still allows you to make adjustments. To do this, adjust the robot's code to add in a wait code that will pause the part under the robot. This gives you a chance to manually adjust the robot and ensure you have the most accurate pick possible.

PrimeTest Automation focuses on having the highest level of expertise with the products it uses in its VGR applications. Their expertise does not stop when the development and deployment of your application does. They provide comprehensive user manuals with all systems and install remote access devices into their equipment to help troubleshoot issues quickly.



A diagram of the flow of the VGR inspection process.

If your application is moving parts to an outgoing conveyor belt, you will go through the same procedure for putting the parts down, except you may not need to use a camera on the outbound belt, unless you are filling a tray or box.

Success Starts With Your System Integrator

Because of the complexities involved in developing VGR applications, you will likely see the best results by working with an expert system integrator to design, develop, and deploy your VGR application. When selecting a system integrator for your VGR application, it is important to ensure the integrator not only possesses a high level of expertise with vision, robotic, and automation applications, but that they will continue to support you once your application is deployed. An integrator such as PrimeTest Automation, who offers comprehensive machine vision solutions and integrated robotic systems, is an excellent choice because they have an in-house engineering team that includes mechanical, electrical, and software engineers.

PrimeTest Automation also focuses on having the highest level of expertise with the products it uses in its VGR applications. They are a Fanuc Robotics Approved Systems Integrator, a National Instruments Alliance Partner Program member and a worldwide VAR+, and have Certified LabVIEW Architects on staff. But their expertise does not stop when the development and deployment of your application does. They provide comprehensive user manuals with all systems and install remote access devices into their equipment so they can help troubleshoot issues quickly.

The Business Benefits of Bringing VGR to Your Production Line

As vision and robotic control technology continue to advance, VGR applications are performing increasingly complex tasks and becoming even more sophisticated, delivering measurable ROI for manufactures, typically in less than 18 months. Thanks to more sophisticated industrial automation of manufacturing with VGR, the factory of the future will continue to become increasingly efficient, more cost-effective to run, and a safer environment for employees.

Learn more about VGR by watching a video of the candy bar Inspection system in action at https://youtu.be/E_6vv4Qfwx0

About PrimeTest

PrimeTest Automation is a full service systems integration company with a talented in-house engineering team featuring mechanical, electrical, and software engineers. All systems are modeled using the latest in 3D design software, thoroughly reviewed with the customer, and manufactured in our facility located in Boca Raton, Florida. PrimeTest Automation offers a complete set of electrical, mechanical, and pneumatic drawings with each system as well as detailed operations and maintenance manual. On-site training services are also available.