european machine vision association

Machine Vision Fundamentals

Outline and Principles

THIS IS THE FIRST IN A SERIES of articles intended to provide a short introduction to Machine Vision. These articles will focus on quality control on the factory floor, which is one of the most important application areas for Machine Vision, dealing with well-defined objects of interest in a well-defined, controlled environment. This situation is a well-suited background for the explanation of the fundamental design-rules and procedures for successful machine vision solutions. Machine Vision, however, also covers a broad range of outdoor use-cases ranging from guided harvesters chopping crop to open-pit mining. In comparison with applications in industry, much more variation may occur in the scene and has to be tolerated by the machine vision system, such as changing lighting conditions or foreign objects. Likewise, some applications in logistics see a huge variation of parts to be processed by the system, and industrial robot vision poses certain challenges related to the depth of the scene. The fundamental design-rules for successful machine vision, however, are basically the same, whatever the task may be.

First and foremost it is important to accept that machine vision systems have to be carefully chosen with the specific needs of the individual application in mind. Seemingly identical tasks, like reading a barcode, nevertheless may require quite different equipment and sometimes even different algorithms. Printed codes on flat paper may call for a different approach than laser-engraved code on a glass bottle. Characters printed with an inkjet may need additional pre-processing compared to offset-printed characters. Thus, a precise definition of the assignment is always helpful. Many machine vision systems are unique solutions, carefully developed in individual projects and tailored to the specific needs of the application. In such cases, a detailed specification is inevitable, along with requirements and procedures of acceptance agreed upon in advance.

Nevertheless, machine vision is a mature technology. Quite a few applications may nowadays be rated as standard assignments, where standard systems are available to cover the required performance. Several companies have this field in view as their business model, backed up by a team of experienced application engineers who need just a few hours to find out whether their standard systems are suited or not. For well-defined tasks, it may be a good idea to check out one of these solution providers. Although this line of action may not lead to the same level of performance as in a system carefully tailored to the specific assignment, it may well be quite sufficient, thus providing a quick and appropriate solution. Even if an assignment calls for special components, thanks to the high level of standardization already reached in the machine vision industry, a broad range of optics, cameras and even software modules is available which can smoothly be combined to build individual solutions. For such an approach, however, a certain amount of engineering competence and expert knowledge is inevitable.

EMVA
Gran Vía de Carles III 84
08028 Barcelona, SPAIN
info@emva.org
www.emva.org

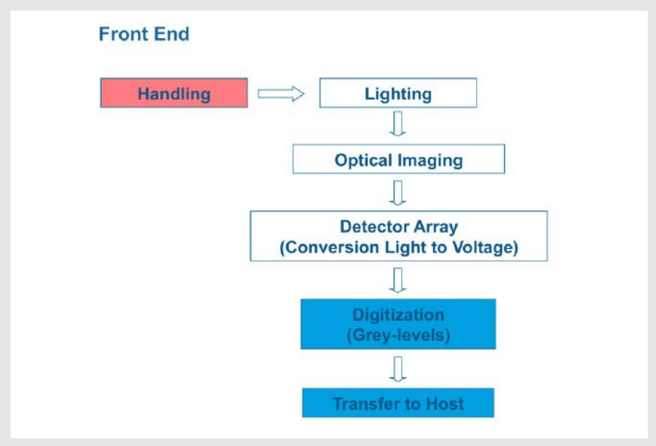

Fig. 1 Front end of the image processing chain.

Machine Vision is a well-proven technology, but some tasks are still demanding and challenging. Sometimes, it is difficult to find out whether the border between a standard assignment, where an off-the-shelf-system will do the job, and a challenging task, necessitating an individual solution, has already been crossed. Deployment will be faster with a standard system and scaling will be easier, but eventually, performance may be more important. Machine vision is not simple, and some tasks are even not well suited for machine vision. Whenever an assignment is repetitive and can be described as a structured decision based on causality ("if this than that"), it may lend itself to automation, and whenever visual inspection, gauging, identification and decoding or guidance is involved, machine vision may be an appropriate solution. Needless to say, economic considerations come into play. High-volume production lines usually are better suited for machine vision than highly individualized handiwork.

Anyway, a certain understanding of the basic principles behind a machine vision solution along with some familiarity with the lines of thought which have developed through the decades in this field, will be helpful to assess whether a certain task may be tackled by machine vision or not. These articles try to provide such a background.

The fundamental principle and the image processing chain

Some people say that the secret of life is to reduce your troubles to a minimum. Whether you will take such a cynical perspective or not - for machine vision in industrial quality control, this certainly is true. When checking a part in the production line, you will capture at least one camera frame with let's say 1000 x 1000 pixels, each with a grey value within an eight bit range, ending up with about 1 MByte of data. You are not at all interested in all these values. Rather, you would like to know whether the part is OK or not. Thus, the art of machine vision might be described as the task of effectively breaking down 1 MByte of data to a single bit of significant information. The recipe to achieve this is to optimize every step in the processing pipeline in order to enhance those features in the image, which are relevant for your specific task, and to eliminate or damp those features which are not relevant. Taking this as a guideline, you will end up with an image where the relevant features can be extracted and classified with simple and robust methods without the interference from irrelevant structures.

Fortunately, machine vision in industrial applications can in general well be described as a linear procedure stepping through well-defined functional blocks, where one step is linked to the next like the elements of a chain. This structural model, the image processing chain, provides an abstract guideline to define the general outline of an image processing application. Optimizing each functional block of the image processing chain according to the principle of highlighting relevant and eliminating irrelevant features will lead to a proper and well-balanced system design meeting the requirements of the task.

The image processing chain begins with the handling stage, transporting the parts to the inspection unit and ends up with the classification result fed back to the process. It may roughly be divided into the front end, comprising all the steps needed to produce an image in the main memory of the CPU, and the back end, that is the software or algorithmic part of the image processing task. In the following chapters, the functional blocks of the front end and the back end are briefly described. To guide your imagination, it might be helpful to have a simple inspection problem for discrete parts in mind, like counting the balls in a ball bearing or checking the number of pills in a pharmaceutical blister pack. The front end is schematically depicted in fig. 1, the back end in fig. 5. For other machine vision assignments like robot guidance, e.g., the processing chain may be somewhat modified, but the fundamental principles remain the same.

Front End: Handling

The inspection unit will usually be fixed in space. Parts to be inspected have to be brought to the inspection unit and placed within the field of view of the camera system. Alternatively, a camera head may be brought to a well-defined position relative to the component, and sometimes both component and camera undergo synchronized movement. Thus, the first element in the front end of an industrial machine vision system will be a handling device. Parts may be delivered by a human operator, by a conveyor belt, they may fly by under the influence of gravity or be precisely placed by a robot. A trigger unit is helpful, signalling the incoming part as an interrupt to the system in order to activate a strobe flash and to initiate the image capture. Parts may be at rest when the image is taken, or they may move along at significant speed on a conveyor belt or similar. Whenever it is possible to place every part precisely at the same position and with defined orientation in the field of view, the subsequent steps in the image processing chain will be much simpler and the algorithmic image processing will be more stable compared to a situation where parts are stochastically placed and oriented. In the latter case, additional effort may be necessary to cope with position and orientation corrections. For moving parts, the same velocity (speed and direction) for all the parts in the line is desirable, and a minimum distance between adjacent parts should be specified such that always only a single part will be in the field of view of the camera system. When the machine vision unit is mounted on a robot arm or a translation stage, travelling along the part to be inspected, the same general ideas apply.

Needless to say, the ideal handling system does not exist. Usually, certain properties have to be compromised upon to get an appropriate engineering solution in a real-world application. Whenever a close approximation to the ideal situation is feasible, however, you will get rid of a lot of problems which otherwise would consume quite a few hours of engineering effort or processing time in the subsequent steps of the image processing chain.

Front End: Lighting

The next step will be to illuminate the scene. Lighting is one of the most powerful methods to enhance the relevant features in an image and to eliminate irrelevant structures. The main purpose of proper lighting is to provide contrast, since most image processing algorithms rely upon grey level differences. A simple example would be edge detection. In an 8-bit-grey-level image with grey levels between 0 (for black) and 255 (for white), a tiny difference of just 5 grey levels in an edge region usually will invoke a considerable risk for failure in edge detection, whereas a difference of 50 or more grey levels will allow for stable and robust performance, even with considerable signal noise and some inhomogeneity of the illumination across the field of view. Contrast may simply be defined as the grey-level difference between the brightest and the darkest parts in an image. As an example, imagine a data-matrix-code with black dots on a bright background. The illumination should be tuned such that the bright pixels are not yet saturated and the black pixels at the same time are as dark as possible. Under these conditions, it will be quite simple to differentiate between black and white areas in the image by just checking the grey level of every pixel against a global threshold value, valid for the whole image.

Contrast based on the remission-properties of the target may be called radiometric contrast. However, this is only one possibility out of several other, sometimes quite elaborate methods. Scratches for example, will show up as bright spots when the surface is illuminated under grazing incidence with the camera picking up light along the normal of the surface. This usually is called a dark-field configuration. Thus, the geometry of illumination is of importance. Fig. 2 and fig. 3 show some examples. Polarized light may enhance or eliminate reflections, and spectrally well-defined illumination will enhance or reduce contrast between areas with different colours. Projecting patterns onto the part to be inspected will provide texture, which is necessary for some 3D-methods, and the projection of stripes with defined spatial period and intensity distribution allows for very precise and accurate measurements of the topography of a surface. Finally, the time structure of the illumination is crucial in some situations. For objects with linear velocity of a few meters per second, which is not unusual in a production line, a strobe light with well-defined rectangular intensity shape in time will be necessary to control the blur at the edges of objects in the image. For a velocity of 1 m/s, an edge perpendicular to the direction of motion will be blurred by 1 mm for a lighting pulse of 1 ms. This may be quite a lot for some image processing algorithms. When reading a bar code for example, the module width may well be in the order of magnitude of this blur. Usually it is a good idea to reduce motion-blur to less than the pixel width.

Fig. 3 A coin under directional lighting from an LED-ring. Edges are highlighted.

Lighting may therefore be looked upon as a means of pre-processing the image. Proper lighting design for an industrial machine vision task will have dramatic effects on the performance of the application. Everything you can master by proper lighting will have an immediate effect upon every pixel in the image in parallel. This happens on the fly while the image is captured. There is no better way to pre-process your image data. Thus, it is a good strategy to use well-defined, controlled lighting in an industrial application and to shield the device against ambient light from lamps on the ceiling or from the sky. Fortunately, this is indeed possible in most situations on the factory floor, in contrast to outdoor-applications such as agriculture, or toll systems or in open pit mining.

Fig. 2 The same object under different lighting conditions. From the left: Undefined ambient light; bright field lighting; dark field lighting; diffuse light. Contrast between gravure and plain metal can be enhanced or eliminated.

Front End: Optics

The next step will be to provide a suitable image of the illuminated scene on the surface of the detector in the camera. This is the domain of optical elements, usually special lenses optimized for machine vision purposes. Lenses seem to be a commodity in machine vision, but they are not. Rather, they are marvellous elements of technical optics. The choice of a proper lens may be as crucial for system performance as proper lighting is.

Basically, the lens provides a defined field of view, meaning that a specific, well defined part of the scene is "seen" by the sensor in the camera. In addition, the lens defines the magnification, and its f-number defines the depth of field, that is the volume of the scene which will be transformed to a sharp image

Fig. 4 Perspective distortion with a standard lens. Scale varies significantly through the field of view.

on the sensor. A standard lens will provoke perspective distortion in the image. A square in a plane, viewed at under an angle of 30° for example, will not appear as a square in the image anymore because a standard lens is based upon central projection with the projection centre somewhere on the optical axis within the lens. Fig. 4 shows an example. Even if a plane is viewed at along its normal, the central projection must be taken into account. While a square in this configuration will be imaged to a square (more or less, due to optical distortions), the scale in the image will vary from the centre to the edges. This may be a problem or not, depending upon the application. With special lenses you may get rid of the central perspective, telecentric lenses in this case, where the magnification is independent of the working distance. However, you have to trade in something for this advantage. Telecentric lenses are just one example for some very sophisticated imaging optics for special purposes in machine vision. Fortunately, there are several quite experienced companies in the market which have specialized in these areas.

In general, it is necessary to get some insight into technical optics to understand the possibilities of optical elements for machine vision and to find out about the obstacles. The concept of MTF (modulation transfer function) of the lens for example, is quite abstract at first sight, but provides very important data concerning the resolution of an optical system. The layout of a machine vision application always should account for the proper match of the pixel-resolution of the detector array to the optical resolution of the lens. Mismatch may lead to undersampling, which in turn can provoke pseudo-structures in the image signal by aliasing. These structures are artefacts; they do not exist in the optical image formed on the surface of the detector array. They appear in the signal, when the optical image is sampled by the discrete pixel-structure of the detector-array, and it is impossible to get rid of these artefacts once they are embedded in the image signal.

Front End: Camera

The camera sensor is the interface between optics and electronics. The optical image formed by the lens on the surface of the sensor consists of photons entering the bulk of the semiconductor material which are converted to electrons by the internal photoelectric effect. Charges accumulated on each pixel of the sensor over a small time slice are stored in a capacitor and eventually read out, thereby converted to a voltage, amplified, then sampled by an ADC and stored as integers in a memory. The optical intensity pattern of the image is thus converted first to a charge pattern sampled by the pixel-area and then to a digital pattern in the memory of the camera, reflecting the pixel structure of the sensor. The result is a huge, but finite number of integers representing the image instead of the continuous intensity distribution provided by the lens. In this discrete and digital form only, the image information is suited for processing on a digital computing device.

Modern machine vision cameras cover a broad range, from competitively priced general-purpose models to highly sophisticated systems with breath-taking pixel resolution, frame rate, sensitivity and signal-to-noise-ratio. Usually the customer has no idea whatsoever about what is really done to the raw signal in the camera before it is output to the camera interface. Dead pixels are masked and patched based on values from their neighbouring pixels, dark signals are subtracted, photo-response-non-uniformity (PRNU) is accounted for, to name just a few. Machine vision cameras roughly come in two classes: equipped either with a traditional CCD-sensor or with a CMOS-sensor. Nowadays, there is rarely the need to come back to CCDs for machine vision applications. Modern CMOS-sensors have performance parameters matching or even exceeding those of CCDs. This is quite a recent development and has been achieved by systematic efforts along two lines, one line resulting in so-called sCMOS-chips (s for "scientific"). The other approach employed is a combination of CCD and CMOS technology, backed by some ingenious developments in solid-state-engineering, resulting in CMOS-Sensors with outstanding performance, leading SONY to discontinue its production of CCDs, and switching to entirely CMOS technology. Such clear statements on performance, by the way, are only possible due to the enormous effort which went into the development of the EMVAs 1288 standard, which provides the ability to unambiguously characterize camera performance parameters within a linear signal model which is now used by major camera manufacturers as input for their datasheets.

Since there is a large variety of camera models for several purposes in the market, the proper choice for a specific application is not an easy task. Like with other components, the decision should not only be based upon technical considerations, but also take non-technical issues into account, such as support by the vendor or availability over the intended lifetime of the application. Compliance with various standards becomes more and more important, and to push forward and coordinate the activities of those organisations which are active in this field makes a significant contribution to machine vision.

Front End: Interface

The digital image data, acquired in the camera and stored in memory, is now ready for further processing. So-called smart cameras have powerful processors on board, and for some applications, the image processing task can completely be accomplished within the camera. These solutions are very effective, and the camera will only need an interface picking up interrupt signals from the process to trigger the image acquisition, usually including the strobe pulse, and to signal the result of the image processing back to the PLC of the production line. Sometimes, some pre-processing only will be done on board to reduce the data load when the image processing algorithms are handled by a host computer. Dedicated hardware boards between camera and host may be used to speed up certain calculations, such as filter operations or colour-space transformations. These boards usually also perform other tasks, such as picking up and shaping a trigger signal, adding a delay and firing a strobe. In an abstract sense, they may be dubbed as "frame grabbers", and indeed are marketed under this term by some vendors. Anyway, data flow has to be maintained such that processing can keep pace with the speed of the production line constantly delivering new parts and waiting for the OK or not to trigger a sorter. Fortunately, there exist several digital interfaces for machine vision cameras tailored to the various needs of the community. Some of the modern CMOS-cameras can provide megapixel-images at a rate of several hundred frames per second, producing net data rates of several hundred MBytes/s, with a few even providing GBytes/s. This enormous amount of data has to be transported in real-time from the camera to the host memory, and not only in streaming video applications, where the loss of some frames from time to time will be tolerated, but often in applications where every single frame in the data stream must be acquired reliably in a 24/7 environment. Interfaces from the consumer market will sometimes not comply with these strict requirements, at least not in every mode of operation. Specifications of bandwidth for interfaces should thus be scrutinized carefully, taking into account the overhead needed to handle the raw data and the real-time capability of the interface. Fortunately, the machine vision community has quite some experts who dedicate some of their precious time to the painstaking work on international standards for the machine vision industry. Thanks to these people and the international machine vision industry organizations who have established the framework for such activities, we now have a thorough understanding of the importance of these issues, and information on the state of the art is easily available and freely distributed on a regular basis. Have a look at their excellent documentation, and you will get a sound basis for your decision concerning a proper digital interface for your application. Finally, needless to say, there may be fast or slow memory modules in your host computer. Sometimes, memory bandwidth is the restricting factor when it comes to processing time, even if you have a high-performance interface in your system.

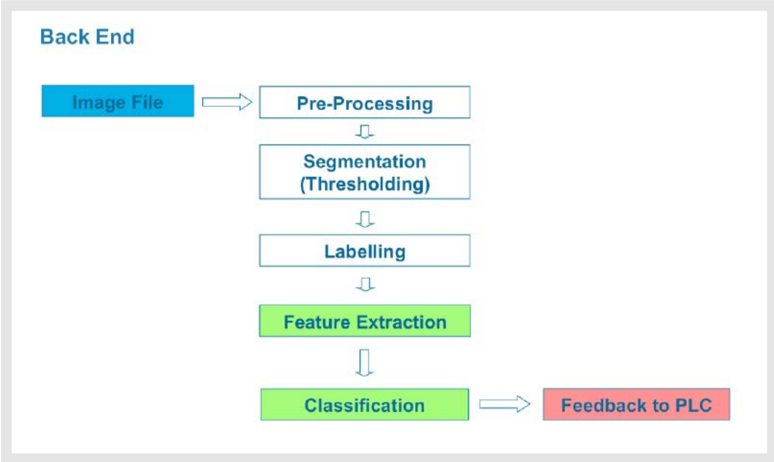

Fig. 5 Back end of the image processing chain.

With this step in the image processing chain, the image is stored in memory, waiting for further processing. We are now leaving the front end and enter the back end of the processing chain. From now on, the whole problem seems to be a topic within the field of discreet mathematics and software engineering. However, reality rather than the cosy world of abstract algorithms, soon takes over again, since everything has to be implemented and run on hardware, coupled back to the machine on the factory floor. If you look back at the front end, it is immediately clear that a huge amount of the data reduction which is the basic assignment of machine vision already has been dealt with by lighting, optics, detector and electronics in the camera. And once you have realized that machine vision always means to come up with reliable quantitative data about features in an image, it becomes clear that every bit of information lost at the front end never ever can be retrieved by software downstream, whatever clever algorithm you may apply. Once lost in the measurement process, the information is lost for ever. Machine vision should be regarded as a measurement technique rather than a means to mimic the perception capabilities of the human visual system. Thus, the proper design of the front end is of utmost importance for successful machine vision applications in industry.

Back End: Pre-Processing

Usually, the first step in image evaluation is some sort of pre-processing. Filter operations to reduce signal noise, geometrical transformations to compensate for perspective distortion, correction of optical distortions from the lens, shading-correction to reduce the influence of inhomogeneous lighting or scaling the intensity to a common mean value are some of the techniques used at this stage of the image processing chain. Some of these steps may be avoided by careful design of the front end, but others may be necessary due to mechanical constraints at the production line. This step is highly application-specific and cannot be generalized. Pre-processing often is performed in the camera or on dedicated hardware boards.

A common method in this stage is to choose one or more areas of interest or regions of interest (ROI) in the image. Further evaluation may be restricted to these ROI, ignoring all the other content of the image. Algorithms can be more stable on pre-defined areas, and when complicated algorithms are inevitable, processing time will be much shorter when these methods have to be applied to some small parts of the image only. Needless to say, some previous knowledge about where the relevant areas of the image are is necessary. In industrial production lines, however, the setting usually is well defined. Sometimes deviations are compensated by detecting edges, lines or fiducials as references to shift the ROI or by linear and rotational transformation of the image back to the standard situation. ROI usually, but not necessarily are rectangles with edges parallel to the edges of the image. Some CMOS-cameras allow for the definition of ROI directly in the camera.

In some applications, the grey-level image is transformed to an edge image, where edges and their vicinity appear bright on a black background. Several machine vision algorithms use edges only and ignore regions with more or less constant grey levels. Filter operations which enhance edges are well understood and have been studied in great detail in image processing. Fig. 6 shows an example. They are standard tools in image processing libraries. Basically, they compute the gradient, that is the derivative of the grey-level function along the x- and y-axis of the image, which is a simple and straightforward procedure. Edge filters, however, have to deal with every single pixel and in addition, read out and process grey-values within a mask around every pixel, such as a 5x5-square, which consumes a lot of precious processing time. Since processing time is generally scarce in machine vision, filtering is one of the image processing steps well suited for pre-processing on FPGA-boards.

Fig. 6 Result of edge-filtering. Original image (upper left), pre-processing by erosion (upper right), and two different methods of edge extraction, morphological gradient (lower left) and Sobel edge detector (lower right).

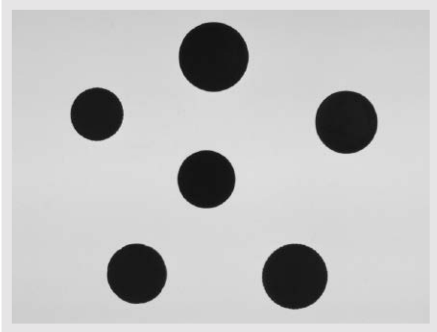

Fig. 7a Grey-level image of coins with very good contrast.

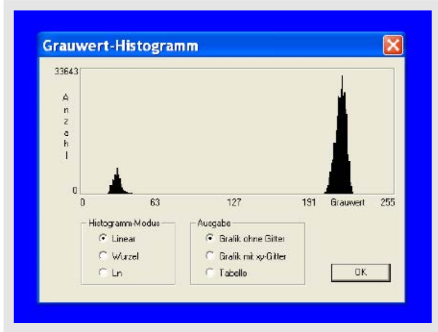

Fig. 7b Grey-level histogram for the image in fig. 7a. Grey-level values are plotted on the x-axis, number of pixels with respective grey-level are plotted on the y-axes. Grey-levels are bimodal, they fall into two distinct clusters for dark pixels (coins) and light pixels (background lighting).

Back End: Segmentation

Segmentation is a crucial step in the image processing chain, where different areas of the image are separated from each other, usually the foreground from the background. Criteria for differentiating between a pixel belonging to the foreground rather than to the background vary quite a lot depending upon the specific application. A simple, but powerful method for segmentation utilizes grey-level contrast. For every pixel in the image file, the grey-level is compared to a fixed threshold value and classified as background, when above the threshold, that is when it is a bright pixel, and as foreground or object when equal to or below the threshold, that is when it is a dark pixel (or vice versa, when looking for bright objects on a dark background). The threshold is usually fixed beforehand as an empirical value, but may also be dynamically calculated for every frame on the fly based on the grey-level distribution of the actual image. With a carefully designed front end, the objects of interest will show up with good contrast against the background, and the simple thresholding described above will yield very good results in segmentation. In these cases, the grey-levels in the histogram fall into two distinct, well separated groups or clusters with small bandwidth, one around small grey-levels and a second around high grey-levels. Fig. 7 shows an example. Thresholding in images with such bimodal histograms allow for robust, stable segmentation with a single threshold value valid for the whole image, thereby enormously simplifying the image evaluation algorithm. The result is a binary image where every pixel has been converted to a black or a white pixel, black belonging to and encoding the foreground and white for the background or vice versa.

Segmentation can also be based on colour, texture or shape, to list just a few possibilities, rather than on intensity. Whenever the procedures on the front end were successful in enhancing relevant features and in eliminating or reducing irrelevant structures based on intensity, binarisation with a global threshold is the prime method of choice for segmentation. A huge number of machine vision assignments in industry are solved by means of this simple method. Segmentation, however, should not be underestimated. It is a critical step with severe consequences. Once a pixel is classified as background, it will usually be ignored in further steps along the image processing chain. If a group of pixels is significant for a defect, but has been put aside as background during segmentation, this defect gets lost in image evaluation and will not be detected. If a group of pixels, on the other hand, is classified as a significant object although it in fact is irrelevant, this additional pseudo-object may spoil the classification at the end of the image processing chain and end up as a false defect, resulting in a reject for the part under inspection although everything is fine.

Back End: Labelling

The next step along the image processing chain is the analysis of the pixels in the foreground to separate different objects from each other. Pixels which are connected to each other are assumed to belong to the same object. These agglomerates of pixels are called blobs or connected components. The process of allocating pixels to their common blobs is called labelling. This comes from the habit of coding all the pixels belonging to a certain blob with a distinct grey-value, which acts as a label for theses pixels. The result is a new image, the label-image, where the background is encoded with the grey-level zero, the pixels belonging to the first blob found with the grey-level 1, the second with grey-level 2 and so on. Every object of interest in the image will thus be "coloured" with its specific grey-level, which is a nice feature for visualization as well as for further processing. Based on the label-image, it is easy to blank out all other objects but a single, specific blob, resulting in an image with a single object only. This is quite convenient for some algorithms, since all the complications which may arise due to several different objects in the image can be ignored. Labelling on binary images is a standard procedure of image evaluation and several algorithms for labelling are published and well documented, as well as implemented in commercial and open source image processing libraries. Labelling nowadays is a commodity. Fig. 8 shows an example.

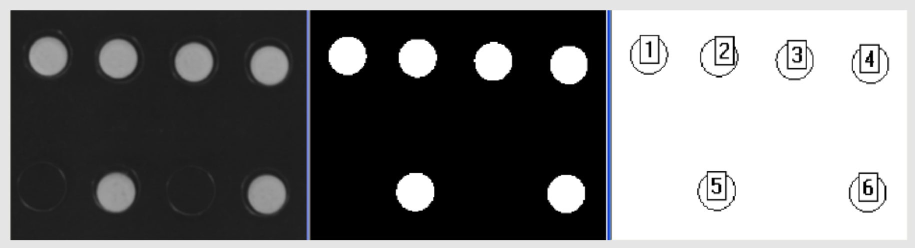

Fig. 8 An example for labelling. Original grey-level image (left), binary image after thresholding (centre), label image with blobs depicted by their contour and label with number (right).

Back End: Blob Analysis

Labelling isolates objects of interest in the image. The next step in image evaluation deals with quantitative measurements of certain properties of these objects such as area, diameter, shape, position or orientation. This step is called blob analysis and the properties are called features. There is a huge number of blob features which may be used for characterization of the objects. The centre of gravity for example, used as a measure of the position of the object in the image, is well suited for pick-and-place-applications or for metrology, when the distance between two circular bore-holes in a part has to be precisely measured. There exist features characterizing the degree of similarity of the object with a circle, features giving the angle between the main axis of elongation of an object with reference to the x-axis of the image file, or features quantifying the convexity of an object by giving the ratio between the area of the object and the area of the smallest object that can be calculated from said object by complementing all concavities, the so-called convex hull. The label-image may also be used as a mask for the original grey-level image, thus giving access to statistical features of the object area in the original image such as mean grey-value, standard deviation of grey-values, entropy and so on. Choosing the appropriate features suitable to support a stable solution for a specific machine vision task is sometimes straight-forward, but sometimes is not an easy task. As always in engineering, experience is the key.

Blob analysis is restricted to binary images. Although this method is a powerful tool and will be successful in a huge number of machine vision applications, there are other approaches to evaluate images. Some methods are entirely based upon grey-level distributions, without the need for labelling or even segmentation. Searching for patterns by grey-level correlation is one example for this kind of procedures, as well as methods for finding the pose of an object based on edges in the image. Another class of algorithms leaves the classical realm of the image plane entirely behind and transforms it into a different space by looking at the Fourier-transform image, that represents spatial frequencies rather than at relationships between pixels. Even in segmented binary images more elaborate methods are quite common, such as looking for straight lines, circles or other geometrical forms by so-called Hough-transformation. The straight-forward blob analysis, however, is quite well suited for many machine vision assignments in industry, and whenever this approach works, it makes sound sense to follow this line of action.

Back End: Classification and Feedback

Near the end of the image processing chain, we now have a set of numbers in hand quantifying several features for all the objects in the image. These features are the basis for a decision leading to an OK or not for the part being checked. In a mechanical component machined from metal there may be some bore holes, and their diameter and relative position on a circle of defined radius for instance, may be checked against the specification. The appropriate features which have to be obtained by image processing are obvious in this case. Other assignments are more complicated, such as checking numbers. Each character hopefully has been isolated by labelling, but what are the features which will lead to a stable differentiation between the numbers 8 and 9 for example. In this instance the number of holes will do the job. But what happens, when the characters are printed by an inkjet, looking more like a swarm of points than like a carefully stencilled number? In this case, at some point along the image processing chain a filter will be applied which fills in the gaps between the points. This might make sense after segmentation, when the foreground has been separated from the background and the image already is a binary image. Nevertheless, optical character recognition is not straight-forward. Just imagine to differentiate between 6 and 9 or between 3 and E. It can be done, of course, but choosing the proper features calls for some thinking.

In classification for industrial machine vision, it is a good idea to base the decision upon rules: if this, then that. Such a decision tree is a transparent, well defined procedure which is open for tolerance analysis. This is a very important feature for industrial applications, since the robustness of a method is a highranking criterion for quality control in the production line. When conditions change slightly, such as the level of ambient light, the precision of the conveyor, the transport velocity, the colour of the parts or their surface roughness and so on, these fluctuations should not affect the result for inspection of a bore hole. With rule-based classification, specifications and tolerances can unambiguously be checked. Nevertheless, statistics comes in as well, since every measurement has uncertainties which have to be carefully evaluated. It is a law of nature that there is always a non-vanishing probability for false acceptance or false rejection in a classification. The rate may be small, but in a high-volume production line, every possibility will come up once in a while, you just have to wait.

Recently, a different approach has gained a lot of interest, namely the concept of deep learning in the context of neural networks. These procedures are by no means new, but the development in computing power pushes neural networks to new frontiers compared to the performance in earlier decades. Basically, a neural network is a huge grid of equal interconnected computing components organized in several parallel planes. This grid narrows down the grey levels of the pixels in an image, the input plane, to a bit-pattern in the output plane by weighing and combining the reaction of the "neurons" to their counterparts in the level below. The input-image thus can be broken down to a small number of bits in the output-plane, even to a single bit. This is exactly the paradigm of machine vision, at least concerning the pixel-crunching part. But how about the optimization of every component to enhance the relevant and to suppress the irrelevant structures in the image? Compared to traditional image processing, deep learning is entirely different: the weighing and connecting of the "neurons" is not based on a model and on "if this then that", but rather on a training process, where a huge number of already classified images is presented to the system as input and the connections are determined by an iterative procedure based on the known output bit-pattern for every image. This is described as "learning", but the better phrase is "training". Training is a cumbersome process, since for some applications several thousand, even millions of images have to be used to cover all use cases – which all have to be properly classified beforehand, sometimes by hand. It is obvious that the structure of the network depends heavily upon the number and quality of the images in the learning set. In addition, the calculations done in the training process take a lot of time: minutes, hours, sometimes days. The most critical point, however, is the statistical nature of the classification model. Training of a neural network is based on correlation rather than on causality. Consequently, there is no such thing as a model describing the system and allowing the classification to be structured along well-defined reasoning. This does not mean that the classification is stochastic; the same image presented to the network, will always lead to the same resulting bit pattern at the output plane. The logic behind, however, is deterministic, but it will be quite difficult to put any meaning into the paths taken by the network when an image is classified by the system. In other words, the decision process cannot be understood any more in a traditional sense, rendering classical debugging impossible, and system analysis with regard to failure modes or tolerances is only possible on an empirical basis. For this reason, some vision experts feel somewhat uneasy when classification is based on deep learning for systems where very low failure rates have to be guaranteed. Nevertheless, neural networks and other related approaches are already widely and successfully used even in traditional image processing such as optical character recognition. Their true potential, however, is in applications where the use cases are so diverse and manifold that classical methods will not work, and at the same time, failure of the classification has no severe consequences. Whenever there is a second chance or an alternative path without any harm done, such a strategy might be appropriate.

Finally, the result of the classification is signalled back, and the PLC of the production line takes over. The inspection stage now is clear for the next part, waiting for a trigger pulse signalling the arrival and to initiate the acquisition of the next image. Some documentation may be necessary, such as storing images for reject parts, or the visualization of process trends for some features to aid the supervising personnel may also be implemented within the machine vision system, but basically the unit is a slave within the production line waiting to go ahead and signalling back accomplishment.

The key question, however, is about risk. Risk is the probability times damage associated with the event. If risk can be handled, everything is fine. But handling risk is not an engineering task, it is a management or entrepreneurial task. In an ideal world, it is up to the engineer to quantify the probability, and it is up to the manager to decide. Real life, however, is more complex.

Summary and Outlook

The image processing chain is a simplified procedure model, but it may be a helpful tool to envision the general structure of an industrial machine vision solution. For a specific application, additional steps may come in, others may be unnecessary. Not every machine vision assignment can simply be solved by thresholding, labelling and blob analysis, but for a huge number of problems this is the method of choice. Anyway, the basic strategy holds: optimizing every step in the image processing chain, from lighting to classification, such that the relevant features are enhanced and the irrelevant structures are eliminated as best as possible.

Further articles in this series will focus on some special topics rather than deal with all the aspects of the image processing chain in detail. Articles are not the proper format for the latter task. Machine vision is engineering, and getting in touch with this field means to gain a general overview and some insight first, which is what these articles are aiming for, but hands-on experience and in-depth education on a thorough theoretical background are inevitable to really get going.

The author

Prof. Dr. Christoph Heckenkamp
Darmstadt University of Applied Sciences
Department of Optical Engineering
and Machine Vision (OBV)
Darmstadt, Germany
heckenkamp@h-da.de
www.fbmn.h-da.de