# Flexible Automatic Visual Inspection based on the Separation of Detection and Analysis

Robert Sablatnig
Department of Pattern Recognition and Image Processing, Technical University Vienna, Treitlstr.3/183-2, A-1040 Vienna, AUSTRIA, e-mail: sab@prip.tuwien.ac.at

## Abstract

*Since the mid-1970s, a large number of visual inspection systems and algorithms for industrial inspection have been developed. To be acceptable in industry, vision systems must be inexpensive, within the speed of the production-line flow, and very accurate. Furthermore the system should be flexible enough to accommodate changes in products. This flexibility can only be achieved by a modular concept that allows a quick and inexpensive adaption of the inspection process to changes in production. This paper presents a concept for visual inspection where the detection of primitives is separated from the model-based analysis process. Existing pattern recognition software is re-used in the detection stage and therefore the use of any detection algorithm is possible without changing the analysis process. The visual inspection of analog display measuring instruments serves as a demonstration of this concept. Results concerning time, accuracy, and reliability for the specific inspection task are given at the end of the paper.*

## 1. Introduction

In mass-production manufacturing facilities, an attempt is made to achieve 100% quality assurance of all products. One difficult task in this process is the inspection that seeks to identify both functional and cosmetic defects [4]. With an emerging requirement for quality control within the manufacturing industry, visual inspection of the product becomes a necessity. The key to solving the problem of flexibility is the development of visual inspection systems which are able to inspect different objects without or only partly changing the analysis algorithm. While visual inspection is high in potential, at present the design and implementation of automatic inspection systems is labour-intensive. Inspection systems have been developed in isolation with no systematic approach, a fact which has led to the design of inflexible customized solutions involving high system engineering costs [3,5,14]. Furthermore there is a lack of software re-use; the computing world consists of developers with similar problems who implement similar solutions, all in isolation. By using libraries of generic code modules, much of the programming work could be avoided [21].

In the design of systems that are easy to install, modify, and support one should distinguish two kinds of application areas in visual inspection: inspection of unstructured images like surface inspection, often based on defect recognition and classification; and inspection of well-structured images resulting from well-structured objects to be inspected to determine whether they variate from idealized geometric properties [17]. Defect recognition problems are usually solved using template matching techniques [5] where systems using knowledge based defect classification were developed [15]. The development of universal inspection systems for the second kind of inspection is difficult since inspection applications deal with different objects with different features to be extracted.

One solution lies in the separation of the application independent detection process from the application-dependent analysis process. For this purpose, existing standard pattern recognition algorithms can be used to detect the primitives in the intensity image, no specific pattern recognition software has to be developed for a specific problem. Several existing algorithms are adapted by adding a standardized interface, tested for the given inspection task and used if they achieve a certain pre-defined detection rate under the constraint. Thus the imaging geometry and the illumination condition are fixed in the final industrial setting.

This paper describes a visual inspection concept introducing the separation of detection and analysis process for inspection tasks (section 2) and explains how existing software can be re-used to solve the problem of the calibration of watermeters; used as a demonstration of the concept (section 3). The paper concludes with experimental results concerning time, accuracy and reliability, showing that the developed visual inspection process can be successfully used. Furthermore the benefits of the separation of detection and analysis process are discussed.

## 2. Separation of detection and analysis

Since the success of feature-based inspection techniques depends on the quality of feature detection, this problem

1015-4651/96 $5.00 © 1996 IEEE
*Proceedings of ICPR '96*

has been the subject of considerable enquiry over the last ten years. Expert systems are continually being developed to solve and refine the feature detection problem on given objects to be inspected [9]. Image segmentation problems belong to the mathematical class of inverse ill-posed problems [16]. There exists no unique and stable transformation function that can build a specific representation of a scene, starting from any kind of observation. To overcome this problem, one has to reduce the number of acceptable solutions by introducing a priori knowledge of the problem space on the solution space and by considering the detection process as broken down into a sequence of sub-problems, that are either well-posed problems or problems for which classical regularization methods exist.

The problem space usually includes all possible instances of the problem. First, all elements to which the inspection algorithm should apply must be described. The primitives of the object form the vocabulary of the description language and relations among the elements constitute the grammar. The model on which the inspection is based is formulated by defining relations between the object primitives and contains the application-specific information for the analysis. The benefit of separating detection and analysis process lies in re-using existing detection routines provided by pattern recognition software libraries.

### 2.1. Detection

In the case of 2D-industrial inspection, objects can be represented by simple geometric forms such as lines, triangles, rectangles, circular arcs, circles, characters and the like. For any of these primitives it is necessary to define their generic parameters in order to be able to define a standard interface for the detection algorithms.

It is possible to compile a large library of operators and then to envision the building of a primitiva detection application as the generation of a program, through selection, parameter adjustment and linking of operators [13]. Since multiple algorithms for each processing algorithm are possible, control rules taking image characteristics into account assist in the determination of an appropriate solution and in determining appropriate parameter values for the operations. If processing costs are available for each operation, then a minimum cost function can be taken into consideration. The steps in the plan can either be executed interactively with the user [6] or using a knowledge base which is able to solve this kind of problem [7,8]. If primitives can be formulated by parametric models, generic algorithms can be used to robustly extract the primitives [20].

There are pattern recognition and image processing packages like KHOROS, KBVision, Ad Oculos, to name a few, which offer a variety of different algorithms used for detection. For every given primitive there should exist two or more different detection algorithms for detection. Each of the methods has to be tested within the industrial environment where the inspection system is to be used with regard to the constraint of unchanging imaging geometry and illumination in the final inspection system. If a detection algorithm reaches a certain threshold of recognition rate in an extensive testing phase (typically more than 98%) it can be used for the final inspection system.

### 2.2. Analysis

The generic parameters of the primitives together with the relations among the primitives form the model of the object to be inspected. This model is an organized representation of the primitives extracted from the object providing adequate descriptions and information. This representation can be generalized as a model consisting of a graph structure in which nodes represent the primitives and the arcs in the graph the relations between the primitives. A priori information concerning the quality standard (e.g. manufacturing and detection tolerances [11]) can also be a part of the model. From the point of view of the description language, the modeling can be interpreted as a syntactic pattern recognition approach in which the primitives are transformed into the vocabulary and the relations are transformed into a string grammar [10]. The most important relations among primitives are spatial relations like distances and angles. For any application, rules necessary for correctly inspecting the object can be added.

## 3. Visual inspection of watermeters

The visual inspection of analog watermeters serves as a demonstration of the proposed separation of detection and analysis process. The selection, parameter adjustment and linking of primitiva detection algorithms were performed manually. In industrial settings, especially in the presence of fluids, analog measuring instruments are still used instead of digitally-monitorable instruments. These instruments have to be calibrated by human workers after the manufacturing process. In the calibration of a watermeter, the instrument is read, a given amount of water is conducted through it, and then it is read once more. Reading a measuring instrument means detecting the position of scales and pointers in the intensity image in order to determine the value displayed by the measuring instrument. An automated inspection provides faster execution and a possibility for quality assurance in the watermeter production. In some literature this inspection is also referred as testing [14] but since the inspection is based on the detection of positions of primitives and very similar to "classical" visual inspection, we regard this calibration as visual inspection.

### 3.1. Primitives, relations, and constraints

The first step in the design of the inspection system is the determination of primitives and the relations between the primitives as a basis for the analysis process and the resulting detection stage. Three primitives describe watermeters: pointers, scales, and lettering elements (see Fig. 1).

A **pointer** can have any symmetric shape with an detectable medial axis such as a triangle, a rectangle or a combination of both. The shape is defined by a bitmap, containing half of the shape and the medial axis. The shape of a **scale** depends on the motion of the pointer. The shape of the scale for a rotating pointer is a **circle** or a **circular arc**. Pointers moving straight have rectangular scales. There are other layout elements of a measuring unit that carry information about the measurement and the global orientation, classified as **lettering**. This includes writings such as company name and firm's logo.

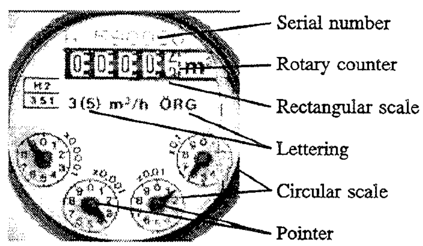


*Fig. 1 Primitives of a water meter*

Following the description of the primitive, a graph containing the generic parameters of the primitives and their relations is formed. Fig. 2 shows the graph designed for the calibration of watermeters. The measurement space defines which unit and which value the **measuring unit**, consisting of a scale and a pointer, displays and at which sample rate the displayed value can be read. The measuring space also defines the absolute measuring value which the **measuring instrument** displays as a combination of all measuring units.

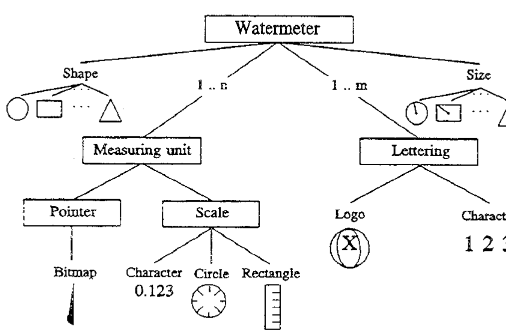


*Fig. 2 Analysis model for watermeters*

The relations among primitives are defined in the grammar of the description language. The most important relations are the spatial relations and in the case of analog instruments the semantic relations, indicating the interdependency among the different primitives forming one measurement value (for details see [18]).

Besides the primitives and the relations between primitives, several constraints are given by the industrial environment. The most important of these are **computation time**, **accuracy**, **extendibility** to other instruments, and **cost**. After defining the primitives to be searched for, pattern recognition algorithms for detecting primitives in the intensity image have to be tested.

## 3.2. Image acquisition

In order to ensure an accurate analysis, the image of the object should have high contrast and there should not be any shadows or reflections in the image. We used a commercial quality monochrome CCD camera in connection with a framegrabber board that can handle both the camera resolution and frame rate fixed in a standard PC 486/66.

Like most measuring instruments, a watermeter is covered with a glass plate. As a result there are specific problems with specular light and shadows. Specular light does not permit reading of the pointers, because a highlight in the area of a pointer makes the pointer invisible and therefore unreadable. A homogeneous, diffuse illumination of the measuring instrument is necessary to obtain a good basis for further computation. A tube with a diffuse, specular inner surface is affixed to the measuring instrument. The lighting itself consists of using a cold light source in combination with a fibre optics cable in order to avoid hazardous voltage in the illumination device. To reduce the specular component of the reflection on the glass plate a polarizer - analyzer filter system is used. The polarizer is in front of the light source so that there is only one lightwave direction; the analyzer (a polarizing filter) is in front of the camera lens in order to filter out all light directions other than the illumination light direction. This setup shown in Fig. 3 ensures that neither highlights nor shadows disturb image acquisition.

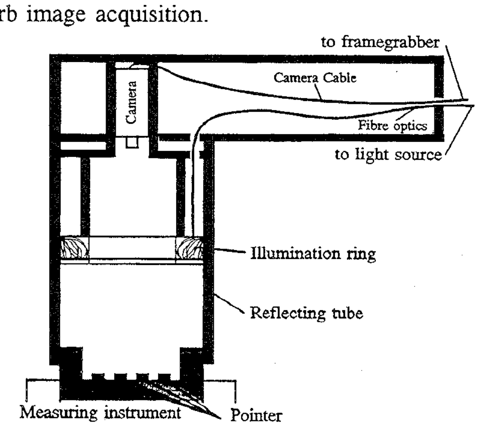


*Fig. 3 Attachment device including illumination*

## 3.3. Detection of watermeter primitives

For the inspection of watermeters, the primitives to be detected are circles, rectangles, pointers, and lettering elements. Therefore, algorithms for detecting these shapes were selected and tested in order to read the value displayed by the measuring instrument.

To detect circular, arc-shaped scales in the intensity image, we used a circular arc detection method based on the Hough transformation [12]. The result of circle detection is a set of potential candidates for arcs and circles. The circular Hough transformation estimates the position and the radius of the circular arc or circle. To verify if the candidate is a valid circle, the edge points of the supposed circle are counted along the circle. If the density of detected edge points on a sector of a certain angle is more than 70% of the circle points in the sector, this sector is accepted as a circular arc; if 70% of the circle sectors are accepted, the supposed center is a center of a circle.

Rectangular scale detection is based on the grouping of four straight lines, which we can detect by using an approach by Burns et.al. [2]. An orientation histogram is computed and lines with 90° orientation difference are combined to form rectangles.

Pointers in the intensity image are detected using a matched filter technique. The shape of the pointer is taken as the filter kernel, where pixels near the medial axis have higher weights than pixels at the border of the pointer. At the center of the circle the filtermask is rotated by steps of 2 degrees and the corresponding pixel values of the intensity images are added into a weighted sum. Since pointers are either darker or brighter than the background, the minimum (respectively maximum) response of the filter defines the position of the pointer.

Lettering elements are used to check and verify the type and the orientation of the measuring instrument. In this case the lettering element is postulated in a certain position and orientation in the image and checked by computing the correlation coefficient between the window in the intensity image and the bitmap of the lettering element by a template matching algorithm [1]. The computed correlation coefficient defines the similarity between the detected area and the generic area, giving the probability of a correct match.

All of the algorithms mentioned above were used because the source code of the programs already existed and had already been tested and used by other users. These algorithms also passed our laboratory test with a recognition rate close to 100%. Following the pattern recognition software re-use and testing stage, the analysis process for performing the reading of the watermeter can be carried out.

## 4. Results

The watermeter inspection process was tested comprehensively with our PC486/66 configuration, both under laboratory and industrial conditions. Three constraints had to be considered for this process: the time for reading the pointers has to be max. 4 seconds/image while accuracy and reliability have to be close to 100% to ensure correct calibration and monitoring.

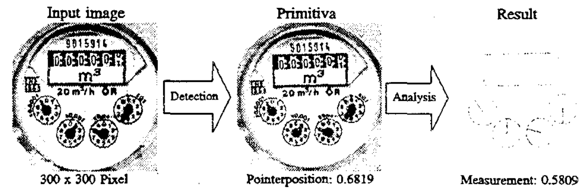


*Fig. 4 Detection of primitives indicates the pointer positions 0.6819, relations between pointers give the correct result*

In our laboratory test-series performed with 200 frames, the positions of **all scales and pointers** were determined in the **requested time** (app. 2 sec.), with the requested **accuracy** (100%) and **reliability** (99%). The reliability of a little more than 99% is due to slight changes in the illumination during acquisition and air bubbles inside the watermeter that can influence the pointer and scale detection. In such a case the image is left for manual visual inspection.

Fig. 4 illustrates the two different stages of detection and analysis during an inspection. The intensity image shown in Fig. 4 on the left is the input for the detection stage. The result of this first step, shown in Fig. 4 in the middle, are the detected scale and pointer positions. These parameters are checked by the analysis process, resulting in a correction of the pointer positions due to the relations defined in the description. The description provides the relations for the so-called coupled pointer computation method [19], resulting in the fact that pointers which have not passed a value on a scale completely are considered to display the previous value. The result is the correct numerical measurement value (0.5809) and the positions of the primitives illustrated in Fig. 4 on the right.

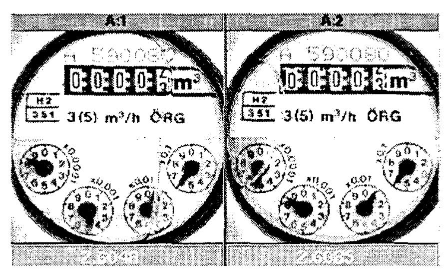


*Fig. 5 Consecutive frames with numerical result*

To find out if the quality of the laboratory results can also be achieved in the final application, a test series (800 frames) was conducted under industrial conditions. This test resulted in a rejection rate of app. 1% (6 frames were not computed because there were air bubbles that influenced the detection of the circular scales and hence the pointers). Although there was a rejection rate of 1%, the accuracy of the reading process was 100%, because the error detection worked properly and non-readable images were marked unreadable and stored for visual inspection by the operator. All of the computed measurements were correct. Fig. 5 shows two consecutive images of a calibration run, denoted A1 and A2, during this test series. These are the control images the user has on his monitor; A1 displays the initial measurement value, A2 the measurement value at the end of the calibration run. In this particular case, 3.7 liters where sent through the meter and the inspection process computed the initial and final measurement values shown at the bottom of the figure. In this figure one can see the disturbances of the air bubbles during this test causing the rejection rate.

## 5. Discussion and conclusion

To prove that the proposed visual inspection concept is easily adaptable and extendible, so that the solution of the specific problem of watermeters also works with other measuring instruments, we tried to read the time on a clock with the inspection process for watermeters [18]. With the description of the clock (3 circular scales and 3 different pointers) the algorithm produced correct results. It was possible to extend the existing inspection process including the model-based description of analog display instruments to another problem by changing the description, while both the detection algorithms and the inspection process remain the same. Note that the detection algorithms can easily be changed if necessary because there exist defined interfaces for detection algorithms.

In this paper a concept for visual inspection was presented where the detection of primitives was separated from the model-based analysis process. Existing pattern recognition software was re-used in the detection stage and therefore the use of any detection algorithm is possible without changing the analysis process. The specific application in the field of visual inspection of analog watermeters served as a demonstration of the concept. The inspection concept starts with dividing the measuring instrument into its primitives and defining a priori known generic parameters of the primitives: shape, relative position, and size. This description of the instrument formed the description language used for detection and analysis. According to the shape of the primitives, existing pattern recognition software was used and tested in order to detect these primitives in the intensity image. After testing and refining the detection, a general analysis strategy was developed to solve the problem of reading the measuring instrument automatically.

The extendibility of the concept was shown by testing the analysis process with the description of a clock, which was performed without changing either the detection algorithms or the analysis process. Since the detection stage is performed with a designed interface for primitives, a change of the detection algorithm is also possible without changing the analysis process. The separation of detection and analysis therefore ensures that any existing pattern recognition software can be re-used in order to solve the detection problem in inspection tasks.

## 6. References


[1] P. Aschwanden, W. Guggenbühl, "Experimental Results from a Comparative Study on Correlation-Type Registration Algorithms", *Robust Computer Vision*, Wichman, Karlsruhe 1992.

[2] J.B. Burns, A.R. Hanson, E.M. Riseman, "Extracting Straight Lines", *IEEE PAMI*, Vol.8, No.4, pp.425-455, 1986.

[3] R.T. Chin, C.A. Harlow, "Automated visual inspection - a survey", *IEEE PAMI*, Vol.4, No.6, pp.557-573, 1982.

[4] R.T. Chin, "Algorithms and Techniques for Automated Visual Inspection", in: T.Y. Young, K.S. Fu (eds.), "*Handbook of Pattern Recognition and Image Processing*", Academic Press, London, pp.587-612, 1986.

[5] R.T. Chin, "Automated visual inspection - 1981-87", *CVGIP*, Vol.-41, pp.346-381, 1988.

[6] M.E. Clarkson, "Intelligent User Interface for the Detection of Arbitrary Shapes by Mathematical Morphology", *Proc. of SPIE*, Vol.-1769, pp.82-93, 1992.

[7] V. Clement, M. Thonnat, "A Knowledge-Based Approach to Integration of Image Processing Procedures", *CVGIP: Image Understanding*, Vol.57, No.2, pp.164-184, 1993.

[8] R. Clouard, C. Porquet, A. Elmoataz, M. Revenu, "Why building Knowledge-Based Image Segmentation Systems is so difficult", *Proc. of 1st Int. Workshop on Knowledge-Based Systems for the (Re)Use of Program Libraries*, pp.138-148, Sophia Antipolis, France, 1995.

[9] D. Crevier, "Expert Systems as Design Aids for Artificial Vision Systems: A Survey", *Proc. of SPIE*, Vol.2055, pp.84-96, 1993.

[10] K.S. Fu, "*Syntactic Pattern Recognition and Applications*", Prentice- Hall, New Jersey, 1982.

[11] P.M. Griffin, J.R. Villalobos, "Process Capability of Automated Visual Inspection Systems", IEEE Trans. Syst., Man, Cybern., Vol.22, No.3, pp.441-448, 1992.

[12] P. Kierkegaard, "A Method for Detection of Circular Arcs Based on the Hough Transform", *Machine Vision and Applications*, Vol.5, pp.249-263, 1992.

[13] T. Matsuyama, "Expert Systems for Image Processing: Knowledge-Based Composition of Image Analysis Processes", *CVGIP*, Vol.48, pp.22-49, 1989.

[14] T.S. Newman, A.K. Jain, "A Survey of Automated Visual Inspection", *CVGIP*, Vol.61, No.2, pp.231-262, 1995.

[15] P. Perner, "A Knowledge-Based Image-Inspection System for Automatic Defect Recognition, Classification, and Process Diagnosis", *Machine Vision and Applications*, Vol.7, pp.135-147, 1994.

[16] T. Poggio, C. Koch, V. Torre, "Computational Vision and Regularization Theory", *Nature*, Vol.317, pp.314-319, 1985.

[17] M.G. Rodd, F. Deravi, Q.M. Wu, J. Powrie, "Knowledge Based Computer Vision Systems for Industrial Control", Proc. of the *11th World Congress of the Int. Fed. of Automatic Control*, pp.153-158, Tallinn, USSR, 1990.

[18] R. Sablatnig, W.G. Kropatsch, "Automatic Reading of Analog Display Instruments", Proc. of the 12th *ICPR94*, Jerusalem, Israel, pp.794-797, 1994.

[19] R. Sablatnig, W.G. Kropatsch, "Application Constraints in the Design of an Automatic Reading Device for Analog Display Instruments", Proc. of the *2nd Workshop on Applications of Computer Vision*, pp.205-212, Sarasota, 1994.

[20] M. Stricker, A. Leonardis, "ExSel++: A General Framework to Extract Parametric Models", Proc. of the *6th International Conf. on Computer Analysis of Images and Patterns*, pp.90-97, Prague, Czech Republic, 1995.

[21] J.D. Todd, "Advanced Vision Systems for Computer-Integrated Manufacture - Part 1", *Computer Integrated Manufacturing Systems*, Vol. 1, No. 3, pp. 143-154, 1988.