Automatic Reading of Analog Display Instruments*

Robert Sablatnig and Walter G. Kropatsch

Institute for Automation, Department of Pattern Recognition and Image Processing,
Technical University Vienna, Treitlstr3/1832, A-1040 Vienna, e-mail: sab@prip.tuwien.ac.at

Abstract

A general design strategy based on experiences of a successful application project concerning the reading of analog devices in order to automate the reading process is presented. The problem space is defined and a description language is introduced which controls the interpretation of specific types of instruments and decouples recognition of primitiva and formation of an aspired result. Furthermore, the interaction between Hough transform or line detection, the user-controlled computation of the measurement and the extension to other measuring instruments is described.

1 Introduction

One industrial problem lies in the reading of existing, analog measuring instruments in production lines. An automatic reading mechanism for these devices would be both safer and cheaper. Reading a measuring instrument means detecting the position of scales and pointers in the intensity image determining the value, the measuring instrument displays. This automatic reading process should work for a large variety types of display instruments in order to cover many applications. In principle there are two different approaches to solving the problem:
A bottom-up design provides a quick solution for a specific problem. The central problem with this quick solution lies in the fact that it has been developed to solve only this specific problem. If for example, another measuring instrument is used, or the pointers have different colors, or the illumination conditions differ, or the measuring instrument is rotated or other changes occur, the bottom-up designed reading process will fail or a redesign will be necessary.
A top-down design starts with the definition of the problem space in which the specific problem is embedded. 'Legal' changes in the input data are specific aspects of the problem space. The problem space can be described by an abstract language which covers both the possible inputs and the possible outputs. The analysis refines the abstraction until operators can be applied to data.

A change in the layout of the measuring instrument only requires a new description. Both the problem space and the detection methods remain the same. The paper describes a top-down design for a given application by defining the problem space (section 2) based on primitiva and relation between the primitiva. The identification of the primitives parses the description grammar (section 3), abstract concepts are refined down step by step to well-established operations on images (section 4), controlled by constraints imposed by the problem space. The evaluated results are converted back into the application language, for example the value displayed on the measuring instrument. Finally, results of the implementation of the algorithm and the extension to another measuring instrument are discussed.

2 The problem space

The problem space includes all possible instances of the problem. First, all elements to which the algorithm should apply must be described. The primitiva of the object form the vocabulary of the description language. Relations among the elements play an important role in the design of the description language.

2.1 Primitiva

Analog measuring instruments consist of three primitiva: pointers, scales and lettering elements (Fig. 1).

A **pointer** can have any symmetric shape with an easy detectable medial axis such as a triangle, a rectangle or a combination of both. In addition, pointers that rotate have a circle at their center of rotation. The shape is defined by a bitmap, containing one half of the shape and the medial axis.

The shape of a **scale** depends on the motion of the pointer. The shape of a rotating pointer is a **circle** or a **circular arc**. Pointers moving straight usually have rectangular scales. In our case we suppose that the scales are

* This work was supported by the Austrian National Fonds für die gewerbliche Wirtschaft under grant Zl3/9934.

1051-4651/94 $04.00 © 1994 IEEE

symmetric in relation to their orientation. Scale captions are considered as part of the scale.

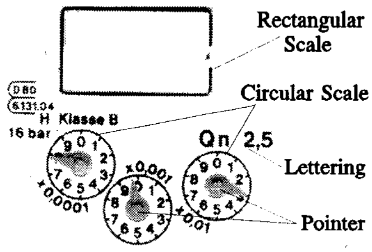

Fig. 1 Primitiva

Other layout elements of a measuring unit that carry information about the measurement and the global orientation are classified as **lettering**. This includes all writing such as unit information, company name, firm's symbol, maker's emblem (e.g. Qn 2,5 in Fig.1).

2.2 Measuring space

One constituent is not contained in an intensity image, but implicitly given with any measuring instrument - the measurement space. It defines what unit and what value the **measuring unit**, consisting of a scale and a pointer, displays and at what sample rate the displayed value can be read. The measuring space also defines the absolute measuring value that the **measuring instrument** displays, as a combination of all measuring units. Note that we assume that one measuring instrument can only measure one physical unit.

2.3 Vocabulary

The elements of the description language are the primitiva of the object. Their generic parameters, all of which are defined in an object-centered coordinate system with respect to the origin in image coordinates, are the following:

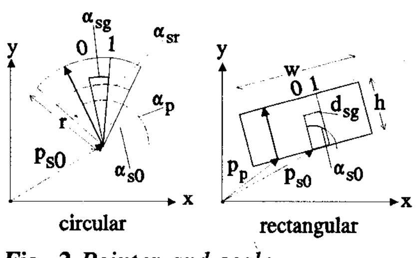

Fig. 2 Pointer and scale

Note that explicit notation is used for the parameters used in Fig. 2 and in relation formulas.

Measuring instrument: type, shape, origin, size, number of measuring units (n), absolute measurement value (m_v).

Measuring unit: origin, normalized relative measurement(c) measurement digits(e), number of lettering elements(m), unit(u), offset(o).

Scale: type(circular or rectangular), size (radius(r) or height(h)) origin(p_{s0}), orientation(α_{s0}), graduation(α_{sg} or d_{sg}), range (α_{sr} or width(w)).

Pointer: type(circular or rectangular), shape(bitmap), origin, position(α_p or p_p).

Lettering: origin(p_l), content(Bitmap or string), area.

2.4 Relations and constraints

Following relations among primitiva are defined in the grammar of the description language:

At the level of a measuring unit:

R1: type(pointer) = type(scale)

R2: For circular scales only:
origin(pointer) = origin (scale)

R3: (unit independency)

$$c = \frac{\alpha_p - \alpha_{s0}}{\alpha_{sg}} \text{ or } c = \frac{p_p - p_{s0}}{d_{sg}}$$

R4: (range constraint)

$$|\alpha_p - \alpha_{s0}| \leq \frac{\alpha_{sr}}{2} \text{ or } |p_p - p_{s0}| \leq \frac{w}{2}$$

At the level of a measuring instrument:

R5: Two measuring units are related through the position.

R6: (measurement digits) $e_i = c_i * u_i + o_i$

where $i = 0..n$ denotes the measuring unit

The following rule is only used for measuring instruments that have more than one measuring scale:

R7: (measurement) $m_v = \sum_{i=0}^{n} \frac{\lfloor e_i \rfloor}{\prod_{j=0}^{i} u_j}$

where $u_0 = 1$.

2.5 Imaging Parameters

To simplify the analysis process we assume that the imaging instrument has quadratic pixels. Therefore two imaging parameters are defined: image size (x,y), and sampling accuracy (a). There is only one relation for determining the sampling accuracy:

R8: $$a^2 = \frac{size(measuring\ instrument)}{imagesize(measuring\ instrument)}$$

3 The Detection Strategy

There are two different strategies for relating objects in the image coordinate system to the object-centered

coordinate system:
M1: Independent detection and localization of primitiva in the image followed by a verification of the constraints.
M2: Detection and localization of reliable objects in the image; construction of hypotheses based on imposed constraints of the detected objects; verification of hypothesized objects. Both of the principles are used in the analysis process (see Fig. 3):

3.1 Analysis process

In order to simplify the diagram, Fig. 3 does not show the interaction of all processing steps with the description and error treatment. If a failure occurs, the analysis process is stopped.

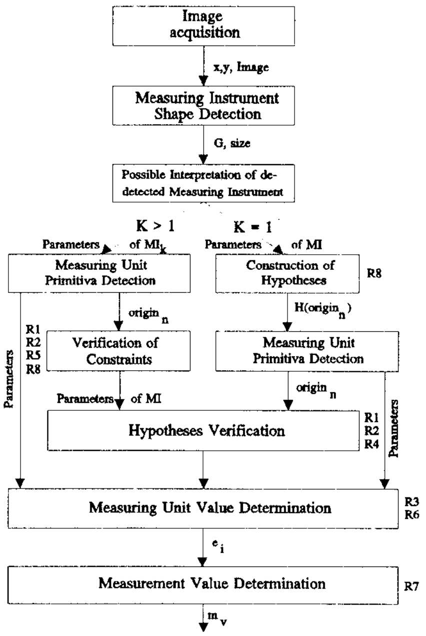

Fig. 3 Analysis process (schematic)

Image acquisition: To simplify the analysis process, we have introduced the restriction that the image contains only a single measuring instrument.
Measuring instrument shape detection: Because shape, position and size of the measuring instrument in the image are not known they have to be detected. All of the shapes of the measuring instruments defined in the description language are looked for in the image with regard to topological, radiometrical and geometrical features of the instrument shape given in the description. **Possible interpretation of detected measuring instrument (*K*):** The detected shape of the measuring instrument defines a set of possible measuring instrument types. The number *K* of measuring instruments satisfying the detected shape in the description is used to decide the further analysis strategy:
K = 1: Construction of hypotheses: Hypotheses about the position of the measuring unit's primitiva are constructed with the help of R8 and the description. These hypothetical positions have to be verified and looked for in a limited area of the image.
Measuring unit primitiva detection: Scales are looked for first because the search space for the corresponding pointer can then be restricted by the region defined by the detected scale. This analysis step supplies the specific origin, type, range, size, α_p and p_p for all primitiva.
K > 1: Measuring unit primitiva detection: If there are different interpretations of the detected measuring instrument, the area of the detected shape is used for detection and localization of measuring unit primitiva. The detection strategy and the result are the same as for K = 1.
Verification of constraints: The aim of the verification is to find out which of the detected primitiva are elements of a measuring unit by applying R1, R2, R5 and R8.
Hypothesis verification: For both analysis strategies a verification of the generated hypotheses is necessary. The identified type is verified by checking the supposed position of lettering elements (p_{ls}, $s = 0..lm_k$) in the image as defined by the description. Furthermore, the type of the measuring instrument is checked by applying R1, R2, R4.
Measuring unit value determination: The value e_i for each measuring unit is determined using R3 and R6.
Measurement value determination: R7 computes m_v.

4 Primitiva detection & results

The kind of elements to be looked for is determined in the description.

To detect circular, arc-shaped scales in the intensity image, we used a circular arc detection method based on the Hough transformation [7,6]. The Hough method [5] was extended to circle detection by Duda and Hart [4] and extended by Ballard and Brown by using the gradients [2].

Rectangular scale detection is based on the grouping of 4 straight lines, which we can detect by using an approach by Burns et. al. [1]. Three features given by the Burns algorithm are used, only long straight lines remain in the line set. An orientation histogram is computed and lines with 90° orientation difference are combined to form rectangles.

Pointers in the intensity image are detected using gray level profiles along curves in the image plane. This method works for pointers in rectangular scales (straight line) as well as in circular scales (circle). The intensity values are summarized along an axis and the center is supposed to be the location of the pointer.

Lettering elements are taken to check and verify the hypothesis, the computed correlation coefficient defines the similarity between the detected area and the generic area and gives a probability value for the possible match.

The algorithm was implemented and tested with a PC configuration. The measuring instrument had 4 circular scales with 4 coupled pointers and 1 rectangular scale. The position of **all elements** except the lettering was determined in the **requested time** and with the **requested accuracy**. Fig. 4 shows the intensity image of the measuring instrument on the left side and the detected primitives on the right side. Note that pointers which have not completely passed a value on a scale are considered to display the previous value.

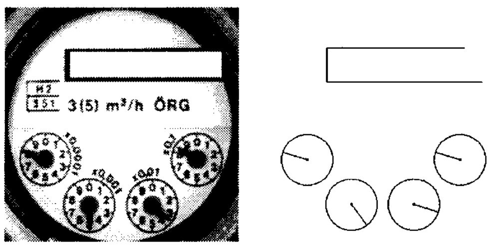

Fig. 4 Result: special measuring instrument

A test series (400 frames) was conducted under industrial conditions without special illumination, causing a rejection rate of 5% (20 frames were not computed because there was too little contrast in the image). Although there was a rejection rate of 5%, the reliability of the reading process was 100%, because non-readable images were marked unreadable and stored for visual inspection. All of the computed measurements were correct.

6 Is this approach suitable for other measuring instruments?

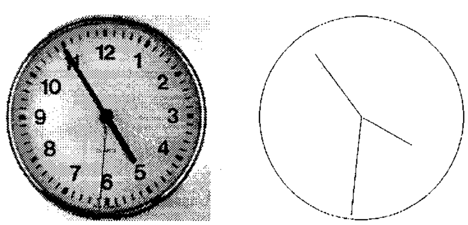

Fig. 5 Result for a watch: 4h 54min 31sec

To prove that the top-down design of the solution of the special problem also works with other measuring instruments, we tested our general analysis process to read the time on a watch. With the description of the watch (3 circular scales and 3 different pointers) the algorithm produced the result shown in Fig. 5. In our test series (20 samples without the special case of pointers being completely overlayed) all images were computed exactly.

7 Conclusion

In this paper we presented a coarse-to-fine design in the problem space to solve one specific application problem. One of the benefits of this solution is, that it is extendable to other applications without redesigning the analysis process. This coarse-to-fine concept can be summarized in 2 steps:

I. General approach to get a large variety of possible solutions: In the first step the problem space and the objects to work with are analyzed. Every object is divided into primitiva and relations between the primitiva, forming the description language for the object in combination with a priori knowledge about the object. The computation and analysis of the requested features is done by parsing the description language in the intensity image after segmenting primitiva.

II. Adapting the general approach to the specific problem: After the general approach is designed and possible algorithms to segment the elements of the description language are tested, the coarse analysis process is refined by adding constraints generated by the specific application. The extension of one special solution to another problem can be done without redesigning the analysis process.

References

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

[2] Ballard, Brown, "Generalizing the Hough transform to detect arbitrary shapes", *Pattern Recognition* Vol. 13(2), pp. 111-122, 1981.

[3] P.E. Danielsson, O. Seger, "Rotation Invariance in Gradient and Higher Order Derivate Detectors", *CVGIP*, Vol. 49, pp. 198-221, 1990.

[4] R.O. Duda, P.E. Hart, "Use of the Hough transform to detect lines and curves in pictures", *Com. of ACM*, Vol. 18(9), pp. 509-517, 1975.

[5] P.V.C. Hough, "Methods and means for recognizing complex patterns", U.S. Patent 3,069,654, 1962.

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

[7] H.K. Yuen, J. Princen, J. Illingworth, J. Kittler, "Comparative Study of Hough Transform Methods for Circle Finding", *Image and Vision Computing*, Vol. 8(1), pp. 71-77, 1990.