Journal of Industrial Information Integration 15 (2019) 29–41

Contents lists available at ScienceDirect

Journal of Industrial Information Integration

journal homepage: www.elsevier.com/locate/jii

ViDAQ: A computer vision based remote data acquisition system for reading multi-dial gauges

Harsh V.P. Singh[a], Qusay H. Mahmoud[*,b]

[a] Computer, Control and Design Department, Ontario Power Generation, Pickering, ON, Canada
[b] Department of Electrical, Computer and Software Engineering, Ontario Tech University, Oshawa, ON, Canada

ARTICLE INFO

Keywords:
Computer vision
Human machine interface (HMI)
Human factors engineering (HFE)
Expert supervisory system
Nuclear power plant (NPP)
Cyber physical systems (CPS)
Remote monitoring

ABSTRACT

This paper presents and evaluates design improvements to the Visual Data Acquisition (ViDAQ) system for reading multi-dial gauges. ViDAQ in general, is targeted to occupy a niche application for a cost effective and readily deployable solution for non-intrusive and remote acquisition of data from legacy human machine interface (HMI) indicators. Legacy HMI indicators that pose numerous technological hurdles in being digitally monitored, include analogue rotary multi-dial gauges, alarm lamps, switches etc, much like those common to industrial process monitoring systems can benefit from ViDAQ. Furthermore, ViDAQ is poised to assist in realizing an overarching design goal of a generic EYE-on-HMI (**E**xpert supervisor**Y** syst**E**m) framework. As a framework, EYE-on-HMI stands to integrate the burgeoning field of machine learning and computer vision for real-time detection of human-in-the-loop operator errors and gather human performance data in any commercial and/or industrial process control domain for improving operational safety. Operator interaction with HMI is vital to the operational safety of any process control such as in nuclear power plant operation, aviation, public transit vehicles, driverless vehicles, etc. and thus should be monitored actively.

1. Introduction

Many severe industrial accidents, such as in the NPP industry, have brought forward grave lessons in human operator error, lax safety culture and deficient HMI designs. A review of NPP accidents rated at high severity (ranging between 5 and 7) on the IAEA International Nuclear Event Scale [1] include: NRX Chalk River Canada (INES-5) - where multiple failures involving incorrect control rod status indicator lights in control room, mechanical failures and miscommunication between control room personnel led to accidental withdrawal of a safeguard bank of shut-off rods, causing an uncontrolled reactor power excursion over 4 times its design limit in matter of 5 seconds, thus causing severe core damage on December 12, 1952; Three Mile Island (TMI), USA (INES-5) - where poorly designed ambiguous control room indicators introduced operator error to inadvertently override the emergency cooling water supply causing a partial meltdown of the TMI Unit-2 reactor core containment on March 28, 1979; Chernobyl disaster, USSR (INES-7) - where confounding human factors and inherent design flaws led to a catastrophic reactor Unit-4 explosion and release of radioactivity on April 26, 1986. Efficacy of such adverse events bear tidings of compromised command inputs by trained CROs, due to following key accident precursors as evident from post accident reports [2–4]: (1) reduction in situational awareness owing to human factors related deficiencies in legacy HMI design; (2) normalization to deviance to lax nuclear safety culture; (3) information overload (looking-but-not-seeing effects [5]) owing to rapid rate at which information was presented to operators via the control room HMIs (panel indications, annunciations, etc); and (4) incorrect mental model of highly dynamic unit evolutions resulting in cognitive errors, owing to conflicting plant information supplied by failed or faulty sensors.

Over the span of 3 decades since the Chernobyl disaster, several marked improvements in areas of process instrumentation and control automation, computer and software technology, HMI design, operational standards, etc have been engrained in global NPP and other industries alike to reduce HITL errors. For example, currently in most all NPPs a plant wide information (PI) system [7,8] collects selected field transmitter sensory data for trending plant process data. However, the information as actually displayed to operators via control room HMIs is neither captured nor analysed currently. Furthermore, owing to practical feasibility, economic and operator re-training costs associated with replacing legacy HMI equipment with new digital devices that can be connected to a human performance monitoring system is challenging. Hence, human machine interaction continues to play a crucial factor affecting safe operation.

* Corresponding author.
E-mail addresses: harsh.singh@uoit.net (H.V.P. Singh), qusay.mahmoud@uoit.ca (Q.H. Mahmoud).

https://doi.org/10.1016/j.jii.2019.05.001
Received 13 December 2018; Received in revised form 6 May 2019; Accepted 17 May 2019
Available online 20 May 2019
2452-414X/

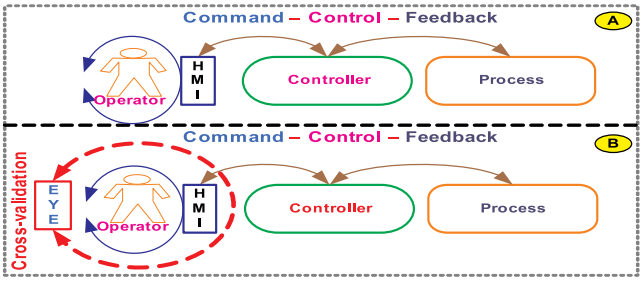


Fig. 1. Architecture of :(A) Operator based command-control-feedback process (B) EYE-on-HMI adds cross-validation to operator command-control-feedback process using ViDAQ. Reprinted from [6].

In order to address above accident precursors, the NPP industry has incorporated frequent training programs stressing human performance by covering several accident scenarios in simulation to prepare NPP CROs. As a result, NPP CROs are trained to rely on manual effort and acquired cognitive skills to overcome the fundamental limitation inherent in the conventional operator based command - control - feedback architecture (Fig. 1-A), vis-á-vis errors injected through human command inputs via HMIs. While rigorous operator training does minimize human command input errors, in reality unit transients continue to fatigue the human mind owing to sensory overload thus, increasing chances of human-in-the-loop cognitive errors.

1.1. Concept EYE-ON-HMI framework

The EYE-on-HMI [9] framework (Fig. 2a) proposes a paradigm shift in real-time monitoring of human performance (situational awareness) in operator based control, by incorporating a closed-loop independent cross-validation on top of the conventional command - control - feedback architecture (Fig. 1-B) found in most CPS. Cross-validation firstly, can be achieved by verifying the information operator is visualizing from the HMI, truly matches the plant process state. Secondly, logging and verifying operator command input patterns in response to the current HMI state are in fact safe and in alignment with the approved procedure(s).

The intent of EYE-on-HMI [9] framework (Fig. 2a) is towards realizing an extensible architecture that exploits computer vision (by proposed ViDAQ [6]) and machine learning techniques to independently capture sequence of events from physical HMI devices such as alarm indicator lights, displays, meters, alarm windows, panel buttons, switches etc via visual data acquisition.

An overview of the software tool flow developed for the EYE-on-HMI [9] framework to digitize arbitrary HMI design that the ViDAQ [6] can recognize is shown in Fig. 2b. Initially, given a HMI panel, a HMI design template file containing co-ordinate locations of various indicators and their outer boundary profile is captured in a standard (xml) text format. This in conjunction with a HMI Equipment Tag list file, captures the HMI instrument type (E.g. dial, lamp, digital indicator, etc.) and associated engineering specification (E.g. indication range, engineering units, etc.). Having two distinct design inputs gives the flexibility of using this tool flow with HMIs having similar layouts but with different indicator devices (common in industrial control rooms).

The digitization process involves converting the HMI design template information into a vector image format using a custom tool. Lastly, another custom tool, EYE-HMI composer engine, combines the input HMI layout vector graphic map and equipment type (HMI Equip. Tag list) with various image processing parameters obtained from a custom EYE-on-HMI widget library, to output the final EYE-on-HMI design file. The latter governs loading of appropriate ViDAQ [6] library routines to successfully recognize a given HMI and extract data from various types of instruments present on the HMI visually in real-time.

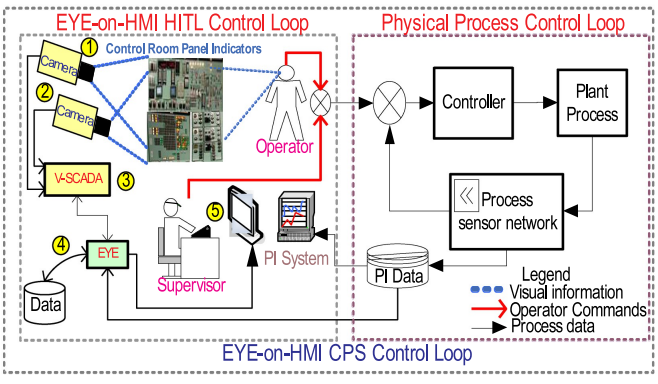
The benefits of the EYE-on-HMI [9] framework include: (a) improved tracking of operator situational awareness; (b) it can aid in providing supervisory oversight on any missed procedural step for post event analysis and lastly; (c) it can hold a longer term contextual memory of all actual or spurious HMI indication events which can help to quickly correlate and diagnose problems that persist over long duration. Such events, otherwise can be missed by CROs over time or subside lower in priority on list of other issues.

The rest of the paper is organized as follows: Section 2 is a review of state-of-the-art. Section 3 outlines ViDAQ design and new improvements. Section 4 discusses the experimental setup for evaluating the ViDAQ prototype. Section 5 provides discussion of results. Finally, Section 6 concludes the paper and provides ideas for future work.

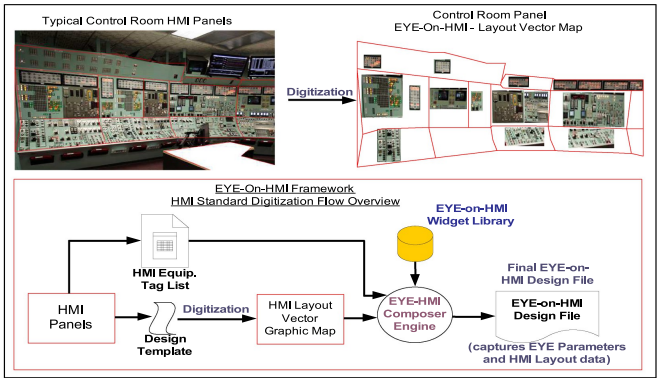
2. Background and related work

Modern industrial establishments represent the apex ecology of several engineering disciples integrated together to achieve complex functional processing lines, that safely produce energy or finished products. Moreover, information is the key intertwining thread that makes such large scale integrations possible. Thus, this area has emerged as an engineering deciple on its own and referred to as Industrial Information Integration Engineering (IIIE) engineering [10].

The proposed ViDAQ [6] system is an example of IIIE that addresses the non-trivial task of remote monitoring of legacy industrial analog HMI indicators (E.g. dials, guages, alarm indication lamps etc.) that are not instrumented (or electronically read). Notably, related area of Machine Vision (MV) represents an active research domain under IIIE which relies on state-of-the-art computer vision algorithms research. Such applications are emerging in industrial robotics [11,12], intelligent transportation systems [13,14], etc. Application of MV via ViDAQ [6] in EYE-on-HMI [9] framework, stands to extend the current



(a)



(b)

Fig. 2. (a) EYE-on-HMI Framework Concept (b) EYE-on-HMI Framework - HMI Layout to Final Design file generation flow. Sub-figure (b) reprinted from [6].

generation of industrial plant monitoring system to use visual data acquisition for reading instruments mounted in control rooms or remote field mounted guages.

Acquiring data using image processing was driven by the need to automate reading of water and electric utility meters for AMRS (Automatic Meter reading System) [15,16] has been shown by previous works such as [17–19] and few patents [20,21] as well. Majority of such works can broadly be categorized in two bins: (1) research works that attempt to do optical digit recognition to acquire meter values directly; (2) others that attempt to interpret the analogue dial needle position to read the meter value indirectly. Both categories of approaches explore unique image processing challenges and present innovative use of image processing algorithms.

2.1. Dial reading - digit recognition

Works such as [18,22] used a 3-layer artificial neural network (ANN), with an input layer of 315 nodes and 10 nodes in hidden and output layer respectively. The ANN is trained off-line in Matlab using back-propagation algorithm to recognize digits in the image training set. Image training set includes cropped binary images for each category of numerical digits $[0-9]$ with resolution $21 \times 15 = 315$ pixels (each pixel maps to an ANN node in the input layer). While, it would be possible to use currently existing OCR (optical character recognition) algorithms to achieve the same functionality, their notable contribution was demonstrating use of an embedded DSP (digital signal processing) system that uses the pre-trained ANN classifier to do in-situ processing and classification of images being captured via a camera. Extracted value is then relayed to a remote concentrator in real-time wirelessly, thereby making the system scalable and readily deployable.

In contrast [23] relied on dual threshold (whole and partial) constants to generate binary images from grayscale that offer better contrast results. Binary image conversion and segmentation is critical to visual data acquisition to reduce false positives and improve data acquisition precision. They separate out individual digit by locating the digit boundaries using a vertical projection histogram (i.e. counting the number of white pixels along the vertical lines). Digit recognition involves using a heuristic method based on locating closed curve shapes on a grid E.g. if a closed curve shape is on top quadrant, the shape is likely to be digits categorized as [8,9] etc.

2.2. Dial reading - position detection

The latter category of works [15,19,24–26] attempt to interpret dial positions. These are of particular interest to the current work on ViDAQ [6] multi-dial reading component. The general motivation of previous research works is to develop an advanced metering infrastructure (AMI) [24] by using cameras to read legacy electromechanical power utility meter dials and transmitting this data for internet cloud based tracking of power consumption per household.

Image data extraction presented in [24] involves using a static thresholding constant, which is experimentally derived to generate a binary image with suitable contrast quality for feature extraction and segmentation steps. Pointer extraction algorithm involves use of morphological operators (dilation and erosion) to fill any holes or deficiencies in dial pointer area. Followed by, superimposition of a fixed size box on the cropped binary images of each sub-dials. The intersection of the box and dial shape edges allows locating the center of the dial and the angle the dial makes with respect to the square edges which is finally used to extract the meter reading. This approach while claimed to work well is quite specific to the type of meter model.

In contrast, the methodology adopted in [19] is towards a more general approach. They used SIFT (Scale-Invariant Feature Transform) to robustly match features of a reference image of the sub-dials to the actual sub-dials in the input image irrespective of orientation, resolution, illumination and other substantial range of a affine distortion that may be present in the source. As a result, image segmentation is improved as target sub-dials region-of-interest (ROI) can easily be validated and cropped out for downstream processing.

Binary image conversion, an important pre-processing step in any image processing application, in [19] is done by applying Otsu's Thresholding [27] method, that dynamically selects an optimal threshold based on discriminant criterion to improve contrast quality, which also improves effectiveness of morphological operators such as dilation (thickening of ROI shape boundaries) and erosion (thinning of ROI shape boundaries) to remove any noise, edge discontinuities and emphasizes separation of neighbouring shape boundaries in ROI.

Dial tip resolution technique used by Ocampo-Vega et al. [19] involves using a structuring element (kernel) to perform a morphological opening operation - this works by applying erosion followed by dilation using a common structuring kernel, which has the effect of slightly eroding the dial shape boundaries to be more curtailed toward larger dial lobe, while allowing thinner boundaries connecting neighbouring unwanted shapes to un-merge (since thresholding operation can create unwanted neighbouring shape boundaries to merge and distort the original dial shape and dial tip). Next, the centroid of the dial ROI shape is calculated using first and second order sequence of image moments $(\bar{x}, \bar{y}) = (M_{10}/M_{00}, M_{01}/M_{00})$. Once the center is located the tip of the dial is found by finding the pixel in dial ROI that maximizes the Eucledian distance from centroid. Finally, the angle is computed using the coordinates of the centroid and tip point using trigonometric functions.

The results presented in [19] includes comparison of two approaches: Combination of first using HARRIS (corner detection) algorithm for detecting key points followed by, using BRIEF (Binary Robust Independent Elementary Features) algorithm for extracting the descriptors, versus using just the SIFT (Scale-invariant feature transform) algorithm that accomplishes both tasks in unison. They showed, SIFT runs faster in detecting and segmenting the dial ROIs overall compared to HARRIS/BRIEF algorithms. In addition, distance at which the dial image is captured from also affects the run time of HARRIS/BRIEF more than SIFT algorithm. Merit of the technique presented above addresses the image orientation issue and robust dial ROI selection using advanced image processing algorithms such as SIFT, BRIEF as shown in [19] and ORB [28](Oriented FAST and rotated BRIEF) as shown in [25]. Though this approach is well demonstrated for single dial per ROI, it does not address meters with multiple dials in the same ROI such as in a clock. Furthermore, use of SIFT and BRIEF algorithms are computationally intensive affecting runtime latency and are an overkill for only detecting dial shapes.

Alternate technique of using a dynamic sliding window for dial needle detection and image segmentation is presented by Sownthar and Manikandababu [15]. Post thresholding (conversion to binary image), a (rectangular) window of fixed width is used to systematically search for the location of dial needle, which terminates once the dial arm is inside this window. They claim this may not be suitable for real-time processing and propose the Angular Detection Algorithm (ADA) as the alternative to sliding window. ADA detects the dials in the following manner: it tries to find vertical growing lines in the image (details of which are not adequately covered in [15]) and selects the longest one, as that would correspond to the dial needle of interest. Upon successfully locating the dial needle, the image is segmented with ROI containing only the dial needle. Needle angle is then computed using start and end points of needle. ADS is certainly shown as an improvement over the former, however vertical growing lines use case is applicable to only dials where the needle pivot is at the bottom. In cases where the needle has full 360° of motion on the meter face the notion of vertical growth lines may be not be adequate for image segmentation.

Lastly, other widely used technique by few works is to rely on Hough Transform (HT) [29] to detect straight lines in binary images, as demonstrated in [26]. The standard Hough Transform algorithm detects and interpolates straight edges in binary images. It does so by means of transforming image coordinate space into polar coordinate space. A line

in cartesian space $y = mx + b$, can be transformed to polar coordinate space by $\rho = xcos\theta + ysin\theta$, where ρ is perpendicular distance between the line and origin and θ is the angle between x-axis and ρ vector). Computational benefits of this transformation is that it limits the values of θ to $0 - 2\pi$ radians, thereby making it possible to represent any line in the ROI with tuple (ρ, θ). The algorithm checks for each white ($value = 255$) pixel/point in the ROI, and computes and records tuple (ρ, θ) of the line that passes through the point. It continues to tally votes for each tuple (ρ, θ) in an accumulator matrix. Once all white points in ROI are visited, it checks which tuple (ρ, θ) has the highest votes, which corresponds to the straight line segment where majority of points are found to lie along (line). Moreover, Xu et al. [26] also demonstrated an improvement of the HT by limiting ρ to only those ranges the dial needle is expected to travel. We believe this is a promising approach and can be extended to reading multi-dial meters, as shown to address by the ViDAQ [6] framework.

Extrapolating the commonality revealed in the above related works, we generalize the essential pre-processing steps required to prepare images as inputs to the ViDAQ. These include: (1) image thresholding for conversion to binary image using Otsu's thresholding, which adaptively generates binary images that are clearly segmented with respect to foreground and background; (2) smart application of fundamental image morphological operators (dilation, erosion); (3) image segmentation to isolate the ROI containing only the dials. While all such proven and general steps have been considered in the ViDAQ framework as deemed required, our focus is mainly to improve dial tip localization in the ROI to extract the measurement value of interest with sufficient precision. ViDAQ uses a new approach for reading multi-dial meters using image contours and convex hull edges as discussed in following section.

3. ViDAQ system design

This section provides details of the ViDAQ [6] processing framework (Fig. 3a) that is used to visually acquire reading indicated by an analogue dial gauges. Motivation behind development of ViDAQ [6] was to accurately and precisely acquire values from a typical multi-dial meter, such as a clock with at least two (hours and minutes) dial arms (needles). As, in a typical industrial control room, there are usually rotary dials for indicating various process parameters. Initially, rotary dials with circular faces was addressed which is now relaxed to recognize variety of dial form factors then extended to a more general non circular dial outline detection technique owing to new design improvements described below (Section 3.5.3).

ViDAQ [6] (Fig. 3a) essentially entails image acquisition followed by an image processing pipeline, output of which facilitates the required feature extraction and measurement extraction logic. Following description provides details of each stage.

3.1. Image acquisition

Image source for ViDAQ [6] is selectable between a fixed source (E.g. directory archive) or a streaming source (E.g. live camera video stream). When a streaming source is selected, only frames that vary by a certain threshold compared to previously processed frame, are forwarded to the image processing pipeline. Image delta is determined by performing an absolute (pixel) colour (RGB) difference (using image histograms), with the pixel values in region of interest (ROI) being assigned higher weights than those outside the ROI area. This is used to optimize dial movement detection routine and effectively avoid unnecessary processing of similar content frames.

3.2. Image processing pipeline

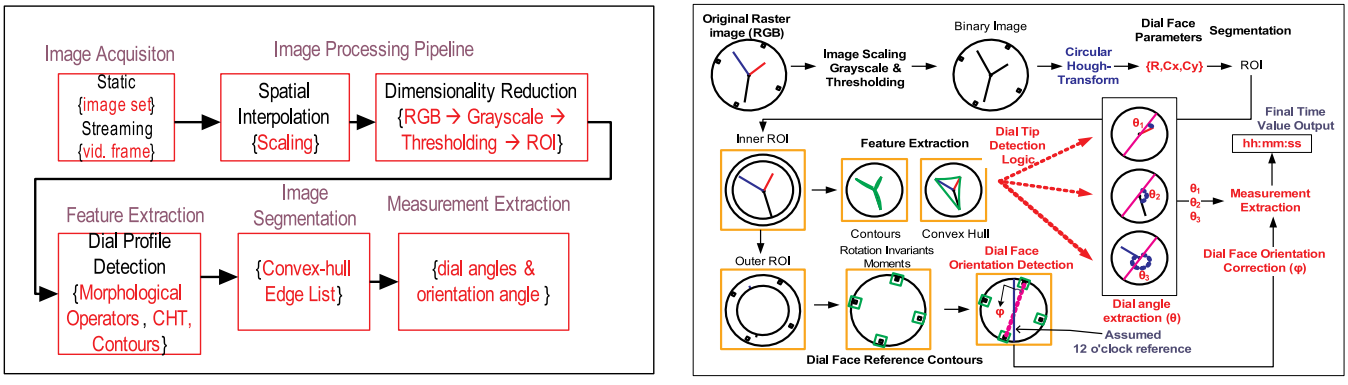
This stage pipelines the required pre-processing steps required to prepare the acquired raster image data for inputs to subsequent feature extraction stages. The pipeline starts with dynamic image scaling followed by conversion to grayscale and binary image formats.

3.2.1. Spatial interpolation

It is essential to address requirement of spatial interpolation (scaling) of input raster images to a required specific resolution (*width* × *height*) or pixel count, prior to processing. Without this step, the input image source may be assumed to have various quality levels (E.g. lossy type JPEG and non-lossy PNG) and captured at high camera sensor resolutions (E.g. 1.5, 2, 12 mega pixels). That is, image scaling is required to bound both the run-time and error rate of the feature and measurement extraction algorithms. While, the run-time directly depends on image resolution (workload) size, the error (false feature detection) rate is inversely dependent on image resolution. For example, certain edge detection algorithms (Laplacian and Sobel) perform worst [30] (higher false positive errors) when edges in the input image are blurry, as in lower resolution images, compared to sharper edges in a higher resolution image. Naive geometric scaling causes pixelation or artifacts (high frequency noise). Either Bicubic or Lanczos [31] based scaling for image downs-sampling showed acceptable results and has been utilized in ViDAQ [6] processing framework (Fig. 3a).

3.2.2. Dimensionality reduction

Image processing pipeline begins with image dimensionality reduction. This typically is achieved by converting a standard 3 channel (R,G,B) 24-bit per pixel image (*Width* × *Height* × 3), into a single channel 8-bit per pixel grayscale image (*Width* × *Height*), which results in an image with only shades of gray. This sufficiently preserves the required visual features (edges and contours) with required luminance levels minus the chromatic noise. Colour channels carry the same image feature information with varying chrominance intensities, which may



(a) (b)

Fig. 3. (a) ViDAQ Processing Framework (b) ViDAQ Processing Stages. Reprinted from [6].

be useful in situation where the subject of ViDAQ [6] involves acquiring measurements from physical indicators (dials) that convey colour encoded information, where conversion to HSB (Hue, Saturation and Brightness) space would be more useful than grayscale. The ViDAQ [6] prototype currently assumes dial indicator are of different shapes and lengths, therefore grayscale reduction is suitable.

There are several RGB to grayscale conversion techniques, however for this prototype using the standard linear transformation produced acceptable results. Linear transformation uses a weighted sum of normalized (between 0 and 1) values of R,G,B intensities per pixel to yield a normalized gray (Y) value: $Y = k*R + l*G + m*B$ (where $k + l + m = 1$ and $k, l, m \neq 0$), that is finally scaled up to [0,255] range. The channel specific non-zero weighing constants (k,l,m) are usually available in image processing literatures for example, $Y = 0.2989*R + 0.5870*G + 0.1140*B$

Conversion to (black/white) binary image is also required to completely suppress unwanted information from the image. Binary images are usually obtained by applying thresholding to a grayscale image, which essentially replaces all pixel values greater than the threshold to ON (white [value = 255]) and others to OFF (black [value = 0]. Significance of Otsu's thresholding [27] for generating binary images is briefly discussed in previous section (2.2) and is also adopted in ViDAQ [6] processing framework (Fig. 3a).

3.3. Feature extraction

ViDAQ [6] processing framework (Fig. 3a) relies on the following feature extraction. Firstly, the profile of a circular dial face must be detected to allow selection of the inner ROI for processing. This is followed by contour and convex-hull detection to detect the extremities of the envelope subtended by dial hands (Fig. 3b).

3.3.1. Dial profile detection

In order to detect a dial face in the image, Circle Hough-Transform (CHT) [29] is utilized (Fig. 3a, b) to detect areas in the image that resemble circular dial boundaries. CHT is a specialized algorithm based on general linear Hough-Transform algorithm. Latter is a well known image feature extraction technique for detecting straight line edges in images and is also briefly discussed in previous section (2.2). CHT also relies on a voting procedure to bin points that accumulate high votes when they fall along the profile of a standard analytical shape (lines/circles/ellipse), that is being detected. For CHT[29], the transform parameter space is three dimensional owing to the three parameters involved: radius (r) and center coordinates (cx, cy) describing a circle $(x - cx)^2 + (x - cy)^2 = r^2$. The output of the CHT is the list of parameter tuples for each detected circle: $\{(r_1, cx_1, cy_1), ...,(r_n, cx_n, cy_n)\}$.

Once the dial face radius R and center coordinates (C_x, C_y) are known, rectangular ROI just enclosing the dials arms is found and used for further processing (Fig. 3b). ROIs, allows partitioning the image into smaller computational workloads to take advantage of parallelization and allow simultaneous reading of variety of HMI devices in real-time. Specifically, Inner and Outer ROIs are found (Fig. 3b). While, Inner ROI only includes the central dial face region containing only the dial needles, outer ROI only contains the shapes located along the periphery of the dial such as digits and other markings. Outer ROI is essential to detect dial face orientation (Fig. 3b) by identifying a reference mark (E.g. locating '12' on a clock), the orientation correction angle (ϕ) is found. Reference mark identification is accomplished by comparing the expected values of rotation invariant image moments (I_1) [32].

ViDAQ's terminal feature extraction step is contour generation [33]. A standard implementation of Canny edge detector based contours generator [34] has been utilized in ViDAQ [6] framework. Canny edge detector is a multi-stage adaptive algorithm that produces a set of points representing the edges of the dial needle shape present in the inner ROI (Fig 3b) with excellent signal to noise ratio (Gaussian smoothing removes high-frequency noise) with hysteresis thresholding to reduce false positive edges. Several closed graphs [34] can be generated using the set of edge points and contour is one such largest graph that covers the entire shape.

3.3.2. Image segmentation

A convex-hull is a smallest polygon that encloses a group of objects. Image segmentation using convex-hull entails finding a set of straight edge approximation of a closed graph (contour) covering the shapes with similar colour intensity. Convex-hull is usually found using binary images or output of edge detected images [34]. Canny edge detection algorithm is used to generate contour polygon that are then used to find the convex-hull using Ramer–Douglas–Peucker (RDP) algorithm [35], to find a minimal cover set of points that describes the contour (cartography). The convex-hull is essential in retaining only the extremities of the contour polygon bounding the shape in the image such as dial arms in the inner ROI. In ViDAQ [6] prototype (Fig. 3b) convex-hull edge (CvE) list facilitates resolving the dial needle end points (tips).

3.4. Measurement extraction

Measurement extraction entails three steps: (1) identify the dial arm tips (end points) accurately; (2) determine the clockwise angle each dial needle makes with a reference segment (such as the 12 o'clock diameter segment) (3) convert the measured angle to required measurement quantity - for example in case of clocks, the longest dial arm represents seconds dial and shortest dial as hour.

ViDAQ algorithm (Algorithm 1) takes the list of end points of the of convex-hull edges (CvE) bounding the dial shape and then sorts CvEs in descending order of the CvE lengths (line segment connecting a pair of convex-hull vertices). Sorting ensures longer edges that bound the dial extremity points (dial tips) are selected first while ignoring other extremities, that often occur in watches where the dials arms (or needles) have tails (Fig. 4).

The sorted list of CvE typically contains two types of edges as illustrated in Fig. 4 : radial type - edges that may extend from tip of a dial to a point closer to the center of the dial $C_{x,y}$; secant type - edges that extend between tips of one dial to the other dial needle. In order to accurately recognize and capture the dial tips, the custom algorithm discerns the given edge as either a radial or secant type (Fig. 4), followed by, creating a unique list of dial tip candidate coordinates based on the rule(κ).

Rule κ*Given a point of reference $C_{x,y}$, If Convex-hull Edge (CvE) $E_i(s, e)$ is radial type, then only include the far point (either s-start or e-end farthest from $C_{x,y}$) (if not included previously), Else If E_i is secant type, then add both start and end points of the CvE (if not added previously).*

Finally, time value measurement for each dial (hour, minute and second) is done by extracting the clockwise angle the dial needle makes with respect to the 12 o'clock reference segment. The angle is computed using the dot product rule Eq. (1) while the sign of the determinant Eq. (2) between two 2-D vectors helps determine the angle being measured is either clockwise or anti-clockwise angle (E.g. if $det(\vec{v}, \vec{w}) > 0$ then $\vec{v}$ is immediately clockwise of $\vec{w}$)

$$\Theta = cos^{-1}\left(\frac{\vec{v}\cdot\vec{w}}{||v||||w||}\right) \tag{1}$$

$$det(\vec{v}, \vec{w}) = v_x.\, w_y - v_y.\, w_x \tag{2}$$

3.5. Assumptions and improvements

Several design assumptions and notable improvements in the initial implementation of ViDAQ [6] are discussed below. Two design improvements to further increase the accuracy and performance of the ViDAQ [6] design includes: (1) Using Perimeter Approximated Polygon (PAP) instead of RamerVDouglasVPeucker (RDP) [35] algorithm to

procedure FINDDIALTIPS($\tau, R_d, C_{x,y}$) ▷ τ: Convex-hull Edge(CvE) selection parameter, Dial face Radius and Center: R_d, $C_{x,y}$
Initialisation:
    $dial[] \leftarrow \{\}$ ▷ init. list of selected CvE as dial arm candidates
    $DT[] \leftarrow \{\}$ ▷ init. list of DialTip coordinates.
    $hull[] \leftarrow \{E_1, E_2, \ldots, E_n\}$ ▷ get list of CvE(s), where $E_i\{s, e\}$ has start/end cord. tuple.

Convex-Hull Edge Selection:
    for all $(E_i \in hull[])$ **do**
        $d \leftarrow Lenght(E_i\{s\}, E_i\{e\})$ ▷ Compute the length of CvE E_i
        if $(\tau * R_d \leqslant d < R_d)$ **then** ▷ criteria for good dial candidates
            $dial[] \leftarrow \{d, E_i\{s, e\}\}$ ▷ Select this CvE E_i and its length d
        end if
    end for

Sort dial[] List:
    $ReverseSort(dial[], \{d\})$ ▷ sort hull[] in descending order of each tuple $\{d, E_i\{s, e\}\}$ using length d

Dial Tip Selection Logic as per rule κ:
    for all $(D_i\{d, E_i\} \in dial[])$ **do**
        if $\Big(RadialEdge\big(E_i\{s, e\}, C_{x,y}\big)\Big)$ **then** ▷ Checks if E_i is Radial CvE given $C_{x,y}$
            $DT[] \leftarrow UniqueFarPoint\big(DT[], E_i\{s, e\}, C_{x,y}\big)$ ▷ returns farthest point of CvE ($E_i\{s\}$ or $E_i\{e\}$) w.r.t given center $C_{x,y}$, that is $\notin hull[]$
        else if $\Big(SecantEdge\big(E_i\{s, e\}, C_{x,y}\big)\Big)$ **then** ▷ Checks if E_i is Secant CvE given $C_{x,y}$
            $DT[] \leftarrow SelectTips(E_i\{s, e\})$ ▷ Select points of CvE ($E_i\{s\}$ and/or $E_i\{e\}$) that are $\notin hull[]$
        else
            $DT[] \leftarrow null$ ▷ Reject E_i, as its neither Radial nor Secant type
        end if
    end for
    return $DT[]$ ▷ Final list of Dial Tips
end procedure

Algorithm 1. Dial Tip Detection Routine (Reprinted from [6]).

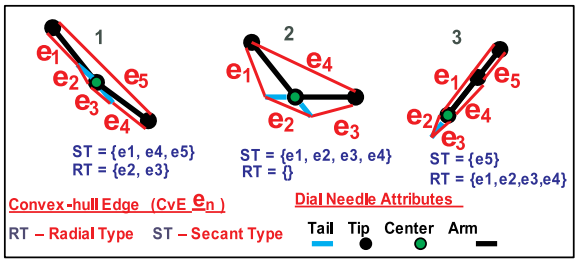


Fig. 4. Radial and Secant Type CvE, under 3 typical orientations: 1 - dials maximally open, 2 - semi open dial tips & tails, 3 - minimally open (dial overlap). Reprinted from [6].

obtain a convex-hull of the dial needles. (2) Using *Hu*-invariant image moments for contour template matching instead of using CHT (Circular Hough Transform) to detect dial face profile.

3.5.1. Design assumptions

Combined functional accuracy of each processing step: feature extraction (Section 3.3), image segmentation (Section 3.3.2) and measurement extraction Algorithm 1 is subject to the following assumptions: (1) Gauge dial needles vary in length, therefore length based image segmentation is used. It is assumed a typical multi-dial gauge each dial arm can be identified by its length. For example, typically on a clock face the seconds, minutes and hours have needle arm lengths in the longest to shortest order respectively. Alternate image segmentation scheme may be possible that may use other shape features such as colour to distinguish the dial arms. (2) Rotary Dial profile - currently it is assumed the dial gauge or clock face to be detected, as any object having a circular outer boundary profile. This assumption initially drove the adoption of the Circular Hough Transform in the feature extraction step (Section 3.3). (3) Constant illumination - computer vision based data acquisition performance is susceptible to various practical limitations, such as lighting conditions and image stability, which must be evaluated independently. Current version of ViDAQ assumes acquired image is under static illumination containing no shadows and is acquired from a stable platform free of any mechanically induced vibrations.

3.5.2. Improvement 1- perimeter approximate polygon CvE

Naive utilization of convex-hull based on RDP algorithm [35], as initially adopted in the image segmentation (Section 3.3.2) step, to enumerate the list of convex-hull edges, leads to a unique class of ViDAQ [6] acquisition errors. This is owing to malformed contour errors which usually occur due to poor resolution of input image or when dial needle orientation brings them in close proximity of each other or partially overlapping (E.g. Fig. 9). CvEs resulting from malformed contours cause misidentification of dial arms (based on length) leading to incorrect values being interpreted by ViDAQ measurement extraction step (E.g. minutes needle may be interpreted as hour or vice-versa). Whilst, malformed contours occur in real-time image processing, existing standard image quality compensation techniques using morphological operators (dilation and erosion) may be explored.

In addition as previously indicated, RDP based convex-hull yields a minimal vertex cover set. This results in an envelope (polygon) that bounds each isolated shape in the input image by only retaining extreme peaks and troughs (convexity defects) outlining the shape. Therefore, it may fail to detect shallower vertices that may lie between two extreme peaks, as shown in Fig. 5-(i). This issue often causes ViDAQ to miss the shorter dial arm (E.g. hour needle) that may appear between two longer dial arms in 3 dial configuration.

An alternative to using a RDP algorithm [35] based CvE enumeration, is to rather seek a polygon that approximates the perimeter of the original contour shape with an adjustable upper bound for perimeter approximation error **(ε)** (Fig. 5-(ii)). A controllable parameter for

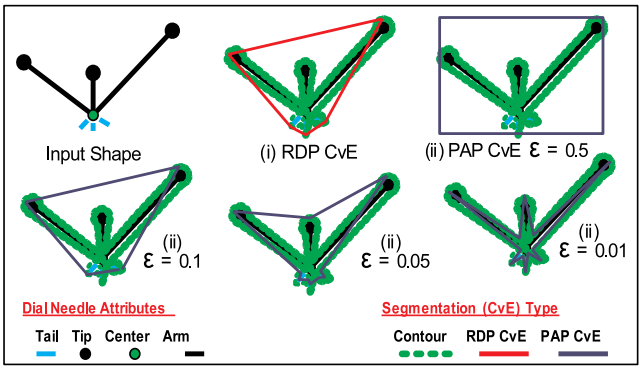


Fig. 5. Image Segmentation using Two Convex-Hull Edge (CvE) based enumeration techniques :(i) RamerVDouglasVPeucker (RDP) (ii) Perimeter Approximated Polygon (PAP); (i) RDP CvE Often misses smaller dial tips. (ii) PAP CvE output at various approximation error values $\varepsilon = \{0.5, 0.1, 0.05, 0.01\}$ result in progressively better dial tip resolution and detection.

approximation error allows the ViDAQ algorithm to optimize the memory usage, speed and detection accuracy appropriately while adapting to various dial orientations.

Depicted in Fig. 5 is the initial contour shape (green boundary) consisting of over 500 points for the CvE vertex list, which imposes higher time and memory cost for sort and search operations for every input image in real-time to detect all dial tips; In Fig. 5-(ii), PAP based CvE list for **(**$\varepsilon = 0.5$**)** yields only 4 CvE vertices for the bounding box (blue boundary) enveloping the entire needle contour shape, which is a very coarse means of dial tip detection; **(**$\varepsilon = 0.1$**)** case is similar to RDP CvE (Fig. 5-(i)) in that minor peaks are not captured in CvE vertex list; For **(**$\varepsilon = 0.05$**)**, yields in a slightly tighter bounding envelope consisting of approximately 10 CvEs vertices however, the dial tips may not accurately coincide with CvE vertices; **(**$\varepsilon = 0.01$**)** is the optimal case that yields a list consisting of approximately 15 CvE vertices, that accurately coincide with each needle tip.

The perimeter approximated polygon (PAP) is more adept at accurately resolving major and minor peaks and troughs of the contour shape as shown in Fig. 5-(ii). Therefore, PAP based convex-hull is also less susceptible to yield CvEs that may result due to malformed contours. Moreover, resulting shorter list of convex-hull vertex coordinates shape extremity vertices also helps using repeated sort and pop operation on the CvE list, to successively extract first, second, third longest dial arm candidates on a clock or gauge in real-time (repeated sort and pop operation is required to actively filter out erroneous breaks in convex hull edges that may otherwise be incorrectly detected as a dial tip).

3.5.3. Improvement 2- contour template matching

Contour template matching using *Hu*-invariant image moments [32,36] is an approximate but faster shape detection technique compared to Circular Hough Transform (CHT) [29], that was initially utilized in the feature extraction (Section 3.3) step to detect rotary dial profiles in the input image. Image moments are commonly utilized in image processing applications to derive various unique shape specific properties based on Eq. (3) (where $I(x,y)$ is pixel intensity in a binary image at location (x, y) and M_{ij} is the raw moment of order $i + j$. For example, the centroid of a shape can be located at: $\bar{x} = \frac{M_{10}}{M_{00}}$ and $\bar{y} = \frac{M_{01}}{M_{00}}$ using raw moments. Other higher order moments such as central moments μ_{pq} (Eq. (4)) are further specified using a binomial combination of raw moments M_{ij} (Eq. (3)).

$$M_{ij} = \sum_{x} \sum_{y} x^i y^j I(x,y) \tag{3}$$

$$\mu_{pq} = \sum_{p}^{m}\sum_{q}^{n}\binom{p}{m}\binom{q}{n}(-\overline{x})^{(p-m)}(-\overline{y})^{(q-n)}M_{mn} \tag{4}$$

Similar to central moments which, are shown to be translation invariant, other rotation and scale invariant moments can also be derived using Eq. (3). As per *Hu* [36] seven moments can be specified that are translation, scale and rotation invariant that can uniquely summarize image shapes and thus, can be used to match shapes using templates.

Contour matching is utilized to detect similar shapes in images that resemble a template shape, such as a circular or rotary dial face. A match template is generated by first generating the contour of a given shape template, which is then used to calculate the seven *Hu*-invariant image moment values. Finally, only *Hu* moment tuple values of the template shape need to be retained for comparison (within a specified tolerance) against the *Hu* moment values of the target shape in the input image. Hence, using *Hu*-invariant moments, effectively reduces shape matching to comparison of two row or column vectors containing ordered list of *Hu* moment tuples - an operation that is optimized for both high speed approximate shape detection with lower false positive rate.

Contour template matching using *Hu* moments can also be generalized to extend shape matching more than just basic geometrical shapes (E.g. circle, oval, box, etc.) to other various industry standard dial shape profiles (E.g. half dials, arc dials, etc). This is an advantage over the restriction posed by CHT [29] (Section 3.3.1) that can only be used to detect exact rotary dials and that is quite sensitive to otherwise even a slight oblongness that can cause the rotary dial to be not detected at all.

Following results compare the performance differences and relative merits of previous two improvements.

4. Experiments

Experiments were conducted to evaluate the functionality and performance of the original ViDAQ [6] application code against the new code based on two design improvements described earlier (Sections 3.5.2, 3.5.3). Result data is the time value reading obtained by processing images of rotary dial clock. These, experiments included profiling the runtime performance and data acquisition error while processing clock face images sourced from two static image datasets. Secondly, ViDAQ accuracy and precision performance was measured while using a live camera feed to acquire time value reading from a live rotary dial clock positioned at various distances from the camera. Finally, in order to test the feasibility of ViDAQ to run on an embedded system, a Raspberry PI (ver. 2 model B) was used to run the ViDAQ source code to acquire time values from a dial watch from a fixed (80*cm*) distance under static lighting conditions over an extended period of time. The acquired time value readings were communicated via Ethernet for remote monitoring and trending.

4.1. Static image test

Static image dataset of dial clock faces were generated using a script. Specifically, two distinct datasets: $2-Dial$ and $3-Dial$ clock image datasets consisting of 720 and 43,200 images respectively showing all possible (seconds, minute, hour) dial positions for a 12 h span. The ViDAQ algorithm was implemented using *python 2.7 OpenCV 2.4.8* library and tested on a Windows 7 machine {Intel i5-480M (2.66 Mhz, 3MB *L*3 cache), 4 GB DDR3 RAM}.

A test script was written to sequentially stream each image from either two image datasets to the ViDAQ algorithm. ViDAQ output consisting of time reading and dial angle value were then compared to the expected time and dial angle values for each input image. The resulting absolute error values were then captured in a delta-log file for further analysis.

4.2. Streaming video test

In this test a USB camera {Logoitech C720 HD Webcam} was used to capture clock dial face at 1280 × 720 resolution. The video stream was fed to the ViDAQ algorithm at 1/10 the frame rate to avoid operating system resources from being overwhelmed.

Live streaming test entailed manually recording both the time reading values output by ViDAQ algorithm and the actual time indicated by the clock. Few variations of clock dial face designs were also experimented to ascertain if shape of needles and design had any impact on visual data acquisition.

Finally, a distance test was conducted, to ascertain the accuracy and precision performance of ViDAQ with respect to the distance between the target watch and the camera. Remote measurement distance stops were chosen based on the distance a human operator would typically encounter while reading industrial panel mounted dial gauges.

5. Results and evaluation

First experiment (Section 5.1) uses static dataset images to evaluate: ViDAQ functionality (Section 5.1.1) and Runtime performance profile (Section 5.1.2).

Under functionality analysis, various sources of data acquisition errors: Quantization(Section 5.1.1), Malformed contours(Section 5.1.1) and Systematic errors(Section 5.1.1) are evaluated. For each source of error relative comparisons are drawn between results obtained for ViDAQ application code with new improvements (using *Hu*-invariant moment for feature extraction + PAP based CvE enumeration for image segmentation) vs. original ViDAQ design (using CHT + RDP based CvE enumeration technique).

The second test, is for the live camera feed to test ViDAQ error rate under controlled conditions, while only the distance between camera and target watch is changed (Section 5.2).

5.1. Static image dataset test

ViDAQ data acquisition error is tallied as each clock face image is processed by ViDAQ application utilizing either the RDP based (Fig. 7a and b) or PAP based (Fig. 8a and b) CvE algorithms for $2-Dial$ and $3-Dial$ static image datasets respectively.

5.1.1. Data acquisition errors

ViDAQ [6] absolute data acquisition errors typically can be attributed to 3 main reasons: Quantization, Malformed contours and Systematic errors.

Quantization Errors. This error is related to the quantization step which, in turn dictates the accuracy at which a dial needle must be resolved to acquire accurate reading.

Quantization or round-off errors are seen more clearly for $2-Dial$ data set (Fig. 7a as shown enclosed by green and grey rectangular box), normally leads to $\pm$ 1 value error in the acquired values of minutes reading than in hours reading.

Above error can be attributed due to higher resolution (6°/min deflection - smaller quantization step) required in the measurement of minutes dial angle compared to a lower resolution (30°/h deflection - larger quantization step) required for measurement of hours dial angle value. Higher resolution measurement for minutes value reading results in relatively higher error probability.

Extreme $\pm$ 59 error overshoot noticeable in Fig. 7a and more significant in Fig. 7b (enclosed by grey rectangular box), are caused when either the acquired value is 0 while the actual value is 59 min or vice-versa, results in a large absolute error (due to wrap-around) which is actually also attributed to quantization error.

Quantization error can be compensated within the ViDAQ value extraction logic by taking into account previously known states of

minutes and hours dials to smartly correct any minor quantization errors. Examples of two typical instances of quantization errors as discussed above are shown in Fig. 9: (b) - where 0 versus 59 min error occurs owing to conversion wrap-around from 59 to 0 min; (c) - is the case where ± 1 value difference occurs due to round-off error, which is more common in minutes reading than in hours reading due to relative difference in quantization steps (as discussed previously).

For 2 – *Dial* image data set, RDP [35] based CvE (Fig. 7a), as compared to perimeter approximated polygon (PAP) based CvE (Fig. 8a), shows significantly higher rates for both quantization and wrap-around errors in minutes value reading (blue spikes). Evidently, PAP CvE performs better for 2 – *Dial* image data set.

For 3 – *Dial* image dataset, RDP CvE (Fig. 7b) performs similar to PAP CvE (Fig. 8b) as the latter has approximately similar rate of both quantization and round-off errors for minutes readings (blue spikes).

In Summary, above results reveal ViDAQ Acquisition error using PAP based CvE technique performs moderately better if not similar to RDP [35] based CvE for dial tip detection. Quantization error rate in visual data acquisition is a function of dial scale graduation which directly affects the resolution or quantization step at which the dial needle tips must be resolved (detected) by ViDAQ for accurate measurement. Therefore, a CvE enumeration technique (E.g. RDP vs. PAP) can affect the degree of accuracy achieved for dial tip detection at a given image resolution, camera specification, dial orientation and illumination.

Malformed Contours (CvE) Error. The second source of error is caused when the contour that is generated (E.g. output in step 4 of Fig. 6) in the Feature Extraction step of ViDAQ processing (Fig. 3a) is malformed (i.e. has discontinuities and incompletely formed). Such type of error results in ViDAQ algorithm to misidentify or completely not detect dials in a multi-dial gauge.

Malformed contours emerge as the artefacts of (step 3: Dilation mask in Fig. 6) applying morphological operators in Fig. 3a, whose parameters do not adapt to dial shape, orientation and illumination condition. Since, each dial orientation represents a unique contour shape, this error type is both dial position and shape dependent.

Malformed contour error can be seen in Fig. 9(a), where both dial tips have been incorrectly detected as having equal lengths due to the merging effect noticed between the dials (CvE e1, e2) when they are close proximity with each other (i.e. when dial needle make acute angles).

Malformed contour acquisition errors are more pronounced specially when acquiring dial gauge readings via a camera under uncontrolled lighting conditions. Currently this is compensated by manually adjusting the dilation parameters tuned to work with a specific style of dial and under a given lighting condition.

For a viable ViDAQ system, malformed contour errors rate must be kept less than the quantization error rate. It may be compensated either

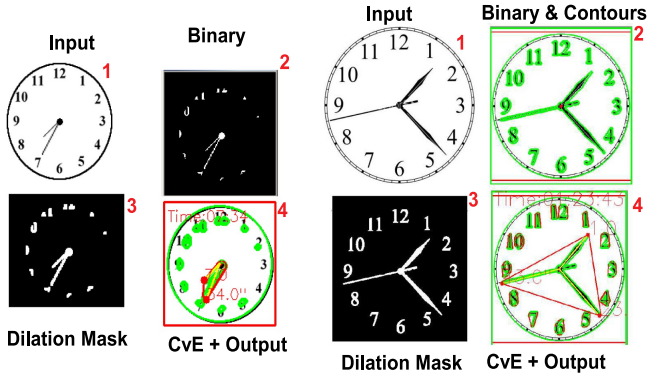


Fig. 6. ViDAQ output - (left) 2 – *Dial*: Input (600 × 600) Output:07:34, (right) 3 – *Dial*: Input (380 × 400) Output:01:23:43. Reprinted from [6].

by using a look-up table or fuzzy logic to dynamically select appropriate morphological and Otsu's thresholding [27] parameters based on dial angle.

For 2 – *Dial* image data set, RDP [35] based CvE (Fig. 7a), as compared to perimeter approximated polygon (PAP) based CvE (Fig. 8a), shows significantly higher error rate in minutes value reading (blue spikes with red circle tops). However, while PAP CvE (Fig. 8a) has no instances of error in minutes reading, it shows (green spikes with red circle tops) a few instances of malformed contours causing error in hours reading (E.g. due to issue shown in Fig. 9(a)). Evidently, PAP CvE performs better for 2 – *Dial* image data set.

For 3 – *Dial* image dataset, performance of PAP CvE (Fig. 8b) out performs RDP CvE (Fig. 7b), as former case has lower error rate shown by blue spikes with red circle tops in Fig. 8b as compared to higher error rate shown by red box in Fig. 7b for both minutes and hours readings.

In summary, PAP based CvE performs better than RDP based CvE technique. This is owing to the fact that CvE enumeration technique (E.g. RDP vs. PAP) can affect the degree of accuracy achieved for dial tip detection at a given image resolution, camera specification, dial orientation and illumination. Nevertheless, PAP generally offers faster dial tip detection.

Systematic Error. Lastly, in the multi-dial (3-dial) case, systematic errors may commonly arise due to needle overlap condition. As shown in Fig. 10, the shortest (hours) needle may be hidden behind either of the longer dial needles, which in current implementation is resolved by assuming the hours hand is always hidden behind the minutes hand.

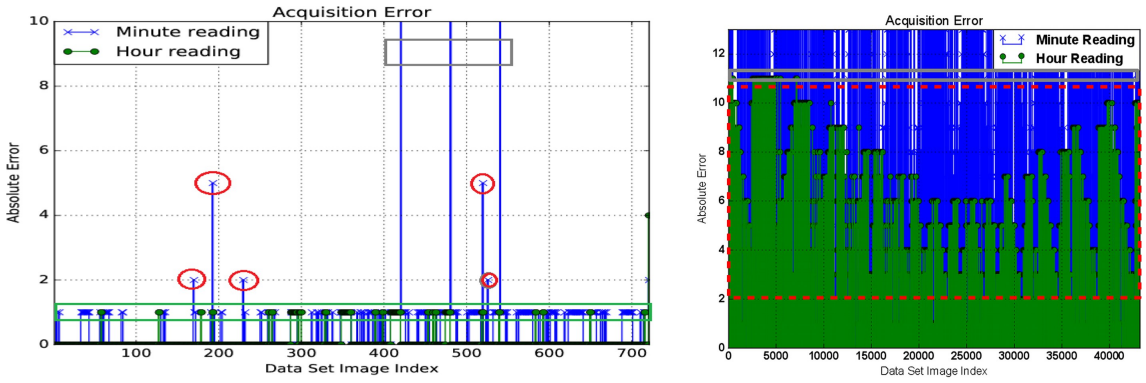
For instance in Fig. 10(a) the hours needle, actually hidden behind seconds needle, is by default assumed by ViDAQ code to be hidden behind minutes hand therefore, hours reading output momentarily reverts to 04 from actual value of 12. However, in Fig. 10(b) the overlap of hours arm is barely avoided while, in Fig. 10(c) the seconds arm moves one second ahead and covers the hours arm, which results in hours reading to change to 06. As noted in previous examples the systematic error persists repeatedly for specific dial orientations. Therefore, above scenario is just one instance of a systematic error among other possible causes that may occur in visual data acquisition which may need to be resolved carefully per use case basis.

For 3 – *Dial* image data set, RDP CvE (Fig. 7b) performs wors than PAP CvE (Fig. 8b). As seen in former case, PAP CvE has higher error rate (shown by green spikes inside the red box in Fig. 7b) for hours reading originating due to both malformed contour and systematic error. While for PAP CvE (Fig. 8b) only systematic error exists, which is clearly visible as a periodic pattern (green spikes inside magenta box) of error rate for hours reading. The periodic sawtooth pattern emerges owing to the recurring overlapping orientation between hours and seconds needle every hours (E.g. Fig. 10(b) and (c)).

In summary, systematic error in visual data acquisition generally may be present owing to the inherent design of the dial or gauge that may obfuscate the reading of values visually in certain dial orientations. For instance, this is similar to the parallax error a human experiences while reading a dial. This category of error is independent of any ViDAQ design and must be resolved carefully by other cross validation techniques to detect such an error and then correct it.

5.1.2. Run-Time performance

Run time performance of the ViDAQ code was determined by capturing wall clock execution time of various code segments (*P*0 – *P*8) while processing images from each static image dataset (2 – *Dial* and 3 – *Dial*). The execution time is probed at 4 critical processing stages of ViDAQ application code, as shown in Fig. 6: (1) Input image scaling (*P*0) and feature extraction (*P*1) to detect the rotary dial (meter) face (using CHT or *Hu*-moment contour matching); (2) Binarization (*P*0) which includes conversion to gray scale followed by thresholding; (3) Application of morphological operators (dilation/erosion) to accentuate



(a) (b)

Fig. 7. ViDAQ Acquisition Absolute Error using RDP based CvE for Dial Tip detection (a) For 2 – *Dial* Image Data Set. Error Types: Malformed CvE (red circle); Round-off error (grey box); Quantization errors (green box) (b) For 3 – *Dial* Image Data Set. Error Types: Malformed CvE (dashed red box); Round-off error (grey box). Sub-figure (a) reprinted from [6]. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

dial arm edges for contour detection followed by selection of dial face ROI (*P*2); Finally (4) Convex-hull edge list (CvE) generation (*P*2) (using RDP or PAP based CvE) is done to localize dial arm tip positions (*P*6), which is required to determine respective dial angles (*P*8). (*P*3, *P*5, *P*7 are reserved for debug code stubs and consume negligible execution time.)

The Figs. 11a, b, 12 a and b for (2 – *Dial* and 3 – *Dial* datasets) show for each code segment (wall clock) execution run time, which has been normalized with the complete execution time spent processing per image.

CHT and RDP CvE - Run Time Profile. Running average of normalized run time values are plotted against data set image index, for the ViDAQ application code using CHT (circular Hough-Transform) for dial face detection and RDP [35] based CvE generation in Fig. 11a and b.

Clearly, code segment *P*1 for CHT (Circular Hough transform) - feature extraction for dial face detection consumes approximately 75% for both Fig. 11a and b (2 – *Dial* and 3 – *Dial* datasets respectively) and currently using standard *OpenCV* library routines. While, *P*2 contour feature extraction for CvE generation using RDP consumes approximately 5%.

In summary, execution run time results shows CHT as a computationally intensive step to detect the rotary dial face. Other code segments specifically feature extraction using RDP based technique to generate CvE list that only retains extremities (Section 3.5.2) thus, relatively uses less computational effort discovering other minor peaks

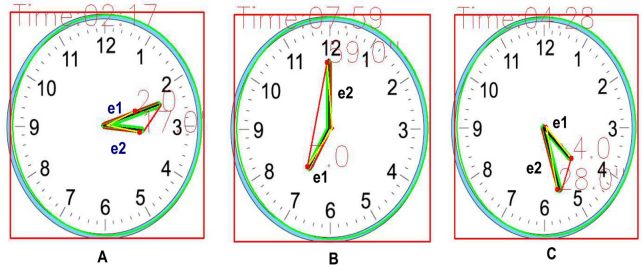
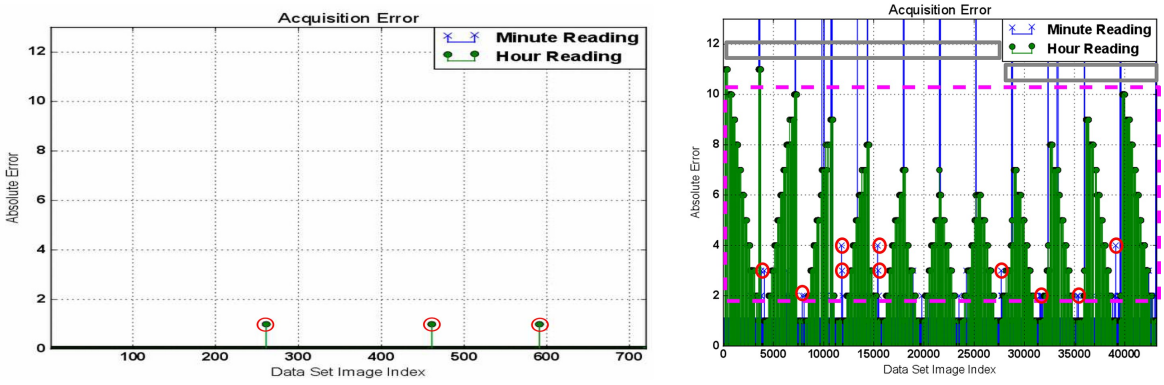


Fig. 9. Common Errors: (a) Malformed Contour causes dials edges (CvE) to be incorrectly detected - Output:02:17 vs. 03:12; Quantization Errors (b) Wrap-Around Error - Output:07:59 vs. 07:00 and (c) Round-off Error - output 04:28 vs. 04:27. Reprinted from [6].

thereby associated with yielding higher ViDAQ error due to malformed contours.

Hu-Moment and PAP CvE. Run time profile for ViDAQ application code that utilizes *Hu*-moment based contour matching (Section 3.5.3) for detecting circular dial face profiles and PAP based CvE enumeration technique (Section 3.5.2) is shown in Fig. 12a and b for 2 – *Dial* and 3 – *Dial* dataset respectively. Run time profile shows a similar pattern for 2 – *Dial* and 3 – *Dial* datasets with the only difference being a slight scale up in run time while processing 3 – *Dial* dataset.

Code segment (*P*1) for *Hu*-moment contour matching (feature



(a) (b)

Fig. 8. ViDAQ Acquisition Absolute Error using PAP based CvE for Dial Tip detection (a) For 2 – *Dial* Image Data Set. Fewer instances of quantization errors in Hours Reading than seen in Fig. 7. (b) For 3 – *Dial* Image Data Set. Error Types: Malformed CvE (red circle); Systematic error (dashed magenta box); Round-off error (grey box) (Seconds reading error was negligible therefore not included for clarity). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

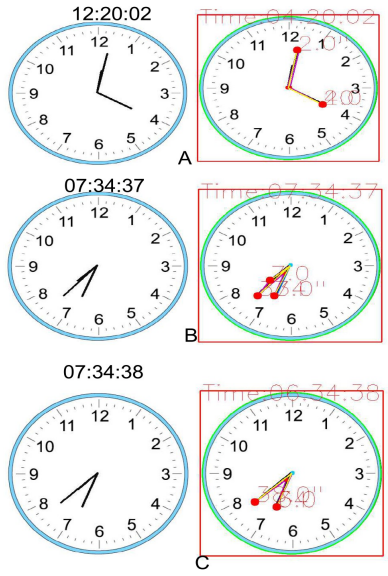


Fig. 10. Systematic error in 3-Dial test due to hours needle overlap detection (a) Output: 04:20:02 vs. 12:20:02 (fail); (B) Output: 07:34:37 vs. 07:34:37 (pass); (c) Output 06:34:38 vs. 07:34:38 (fail).

extraction) for dial face detection consumes approximately 40% to 50% of run time which is lower than that consumed by CHT (Fig. 11b).

Code segment (*P*0) consumes approximately 35% to 45% of run time since, PAP based CvE generation step requires an additional pre-processing of using Gaussian blurring (Section 3.3.1) to remove any sharp vertices following Binarization to reduce false positive instances of dial tips. Gaussian blurring was not utilized for RDP based CvE (Fig. 11a and b), and hence *P*0 consumes only 5% run time.

Code segment (*P*2) PAP based CvE contour feature extraction consumes approximately 15% to 30% of overall run time, which is 15% more in comparison to what RDP based CvE consumes (Fig. 11a and b). This is expected since PAP based CvE enumeration technique is a non-optimized custom developed operation that uses several sort and pop operation (Section 3.5.2) to discover candidates for dial tips. Nevertheless, PAP based CvE yields lower ViDAQ error performance.

Next noticeable time consuming segments *P*8 (Angle measurement and time value extraction routine) utilizing approximately 5% and *P*6 (Dial arm and tip selection) utilizing approximately 3% of overall processing time. Both *P*6 and *P*8 are custom code segments implemented based on Algorithm 1.

5.2. Live streaming video test

Table 1 tabulates manually captured data while ViDAQ is tested using a live video stream. The data is used to ascertain the dependence of ViDAQ data acquisition accuracy and precision, when the subject rotary dial face (clock) is located at distances varying between 40 cm (centimetre) to 3 m (meter) from the camera (Section 4.2) Accuracy here represents mean deviation in acquired time value versus actual time reading at a given acquisition distance (shown column wise). Precision here represents the mean spread between the acquired time values captured at various acquisition distances when the clock time remains unchanged (shown row wise).

The data (Table 1) shows an average accuracy of ± 1 min to ± 3 min (in minutes values) when a reading is taken at a distance of 1.5 m, where the source of error is predominantly caused due to quantization effects (Section 5.1.1). Moreover, Table 1 data also reveals ViDAQ acquisition precision has a weaker dependence, compared to accuracy, on the ViDAQ acquisition distance when other factors such as target stability and lighting conditions remain unchanged. At distances greater than 2 m the accuracy suffers drastically due to increased instances of malformed contours (Section 5.1.1) resulting in misinterpretation of dial tips by the ViDAQ algorithm.

Though, data acquisition accuracy and precision of ViDAQ's prototype algorithm is satisfactory for distances lower than 2 m, the current design has further room for improvement to increase the acquisition distance using higher resolution images for practical applications. Furthermore, future work should also extend ViDAQ performance to address external factors such as target illumination and image stability.

6. Conclusion and future work

This work further extends the earlier work on Visual Data Acquisition - ViDAQ [6] by evaluating two improvements in the image processing stages that utilize *Hu*-invariant image moments [32,36] for faster feature extraction and perimeter approximated polygon (PAP) for faster image segmentation. The first improvement, focuses on shape template matching based on *Hu*-moments, which is a more efficient and approximate technique for dial shape detection compared to Circular Hough Transform (CHT) [29], that is both computationally intensive and sensitive to detecting rotary dial shapes with perfect circular profiles only.

The second improvement, using PAP instead of the RDP [35] algorithm, generates convex-hull edge (CvE) list required for detecting dial tips, is more efficient, i.e. runs at higher speed and uses lower memory. Above improvements resulted in approximately 40% to 30% speed-up in run-time execution for 2 – *Dial* and 3 – *Dial* data sets

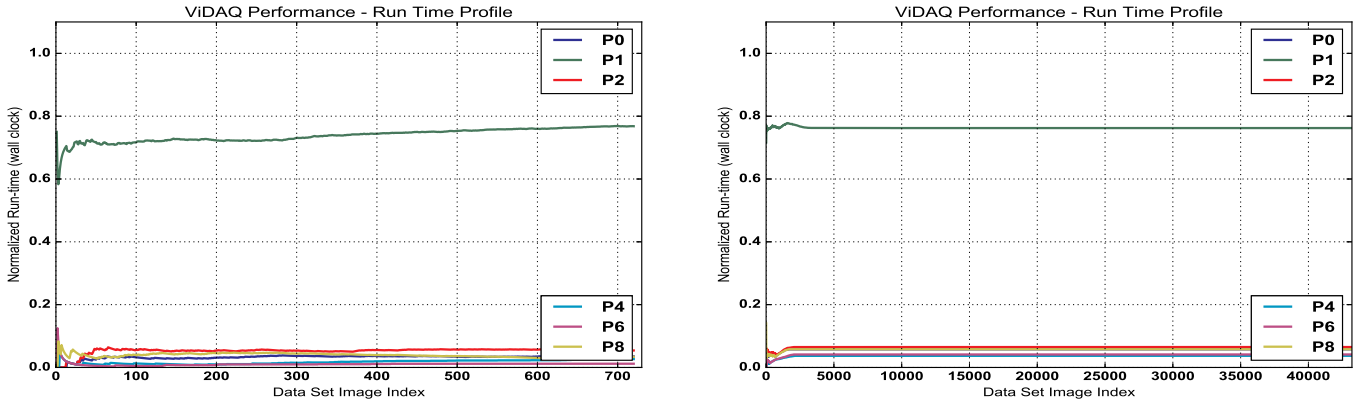


Fig. 11. ViDAQ Normalized (running average) Runtime profile (P0 - P8 are code segments); Using CHT for Dial Face detection and RDP based CvE for Dial Tip detection (a) 2 – *Dial* Image Data Set. (b) 3 – *Dial* Image Data Set.

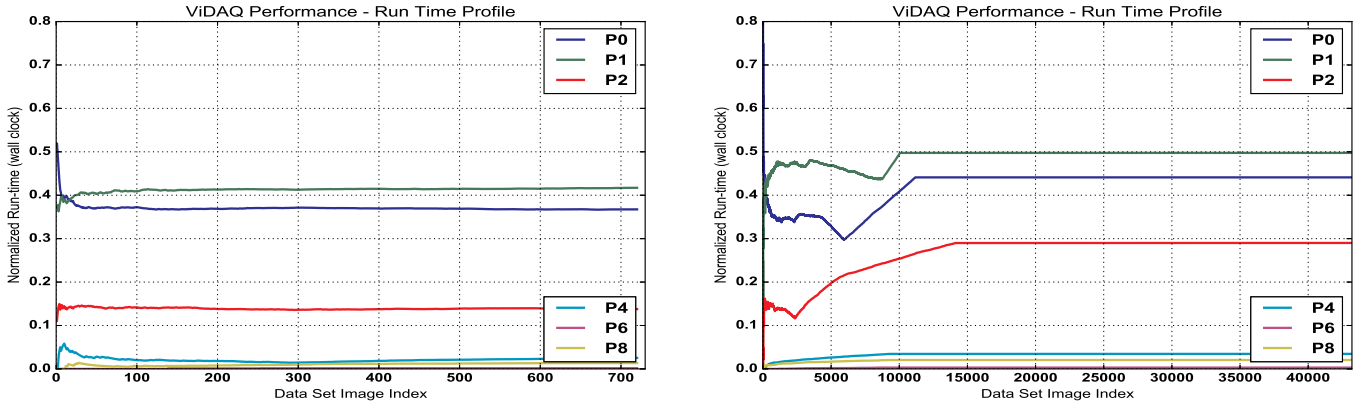


Fig. 12. ViDAQ Normalized (running average) runtime profile (P0 - P8 are code segments); Using *Hu*-moment for Dial Face detection and PAP based CvE for Dial Tip detection (a) 2 – *Dial* Image Data Set. (b) 3 – *Dial* Image Data Set.

Table 1
ViDAQ live video streaming test (Reprinted from [6]).

Actual time	40 cm	80 cm	1 m	1.5 m	2 m	3 m
6:21	6:21	6:20	6:21	6:21	6:25	1:09
6:09	6:08	6:09	6:09	6:10	6:11	3:23
3:29	3:30	3:29	3:32	3:29	3:26	9:20
2:38	2:38	2:37	2:38	**3:36**	**3:36**	n/a
12:20	12:19	12:18	12:20	12:19	12:19	n/a
1:55	1:54	1:55	1:54	**11:09**	**1:56**	n/a
2:37	2:39	2:40	2:40	**3:28**	**3:29**	n/a
3:00	3:02	3:02	3:03	3:03	**12:03**	n/a
2:21	2:22	2:22	2:22	2:20	2:20	n/a
7:15	7:15	7:16	7:17	7:17	7:17	4:15

Accuracy: ± 1 min ± 1 min ± 2 min ± 3 min ± 1 h ± > 1 h.

respectively.

ViDAQ is being developed with target application in industrial control rooms that use legacy HMI instruments such as rotary dial meters, gauges, indication lights, etc. Furthermore, extensibility of EYE-on-HMI [9] framework is also addressed in adapting ViDAQ for monitoring various HMI designs in order to detect human-in-the-loop errors. Likewise, future work will focus on improving ViDAQ accuracy using real-time camera video streams to improve its practical feasibility.

Conflict of interest

The authors declare no conflicts of interest.

References

[1] IAEA, International nuclear event scale (ines).
[2] IAEA, Insag-7 safety report the chernobyl accident, 1992.
[3] Nuclear Energy Institute, Lessons from the 1979 accident at three mile island, 2014.
[4] Atomic Energy of Canada Limited, Chalk River Laboratories, The chalk river accident in 1952, 1992.
[5] D. Manzey, et al., Human performance consequences of automated decision aids: the impact of degree of automation and system experience, J. Cogn. Eng. Decis. Mak. (2012). 1555343411433844.
[6] H.V.P. Singh, Q.H. Mahmoud, ViDAQ: a framework for monitoring human machine interfaces, Real-Time Computing (ISORC), 2017 IEEE 19th International Symposium, IEEE, 2017.
[7] S. Song, et al., The nuclear power plant information system for remote users, Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium on, 1 IEEE, 2001, pp. 381–385.
[8] Z. Drias, et al., Analysis of cyber security for industrial control systems, Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC), 2015 International Conference on, IEEE, 2015, pp. 1–8.
[9] H.V.P. Singh, Q.H. Mahmoud, EYE-on-HMI: a framework for monitoring human machine interfaces in control rooms, Electrical and Computer Engineering (CCECE), 2017 IEEE 30th Canadian Conference on, IEEE, 2017, pp. 1–5.
[10] Y. Chen, Industrial information integration:a literature review 20062015, J.Industr. Inf. Integr. 2 (2016) 30–64, https://doi.org/10.1016/j.jii.2016.04.004.
[11] A. Kazemian, X. Yuan, et al., Computer vision for real-time extrusion quality monitoring and control in robotic construction, Autom. Constr. 101 (2019) 92–98, https://doi.org/10.1016/j.autcon.2019.01.022.
[12] M.H. Ali, A. K., et al., Vision-based robot manipulator for industrial applications, Procedia Comput. Sci. 133 (2018) 205–212, https://doi.org/10.1016/j.procs.2018.07.025. International Conference on Robotics and Smart Manufacturing (RoSMa2018)
[13] S. Guo, Q. Diao, et al., Vision based navigation for omni-directional mobile industrial robot, Procedia Comput. Sci. 105 (2017) 20–26, https://doi.org/10.1016/j.procs.2017.01.182. 2016 IEEE International Symposium on Robotics and Intelligent Sensors, IRIS 2016, 17–20 December 2016, Tokyo, Japan
[14] R.A. Kharjul, V.K. Tungar, et al., Real-time pedestrian detection using svm and adaboost, Energy Systems and Applications, 2015 International Conference on, IEEE, 2015, pp. 740–743.
[15] S. Sownthar, C. Manikandababu, Visual monitoring and alerting system for arc type meters, Int. J.Res. Comput. Appl. Rob. ISSN:2320-7345 (2014) 29–41.
[16] D.S. Bhushan, A review paper on automatic meter reading and instant billing, Int. J.Adv. Res. Comput. Commun. Eng. 4 (1) (2015).
[17] Y. Fujita, Y. Hamamoto, Automatic reading of an analogue meter using image processing techniques, IEEJ Trans. Electron. Inf. Syst. 129 (2009) 901–908.
[18] L. Zhao, Y. Zhang, Q. Bai, et al., Design and research of digital meter identifier based on image and wireless communication, Industrial Mechatronics and Automation, 2009. ICIMA 2009. International Conference on, IEEE, 2009, pp. 101–104.
[19] R. Ocampo-Vega, G. Sanchez-Ante, et al., Image processing for automatic reading of electro-mechanical utility meters, Artificial Intelligence (MICAI), 2013 12th Mexican International Conference on, IEEE, 2013, pp. 164–170.
[20] M. Gellaboina, G. Swaminathan, V. Venkoparao, Image based dial gauge reading, 2013, US Patent App. 13/237,147.
[21] J. Kennedy, T. Baird, Analog utility meter reading, 2014, US Patent App. 14/295, 669.
[22] L. Zhao, Y. Zhang, Q. Bai, Z. Qi, X. Zhang, Research of digital meter identifier based on dsp and neural network, Imaging Systems and Techniques, 2009. IST'09. IEEE International Workshop on, IEEE, 2009, pp. 402–406.
[23] D. Shu, S. Ma, C. Jing, Study of the automatic reading of watt meter based on image processing technology, Industrial Electronics and Applications, 2007. ICIEA 2007. 2nd IEEE Conference on, IEEE, 2007, pp. 2214–2217.
[24] Y. Tang, C.-W. Ten, et al., Extraction of energy information from analog meters using image processing, IEEE Trans. Smart Grid 6 (4) (2015) 2032–2040.
[25] Z. Yang, W. Niu, et al., An image-based intelligent system for pointer instrument reading, Information Science and Technology (ICIST), 2014 4th IEEE International Conference on, IEEE, 2014, pp. 780–783.
[26] L. Xu, et al., An automatic recognition method of pointer instrument based on improved hough transform, Applied Optics and Photonics China (AOPC2015), 9675 International Society for Optics and Photonics, 2015, p. 96752T, https://doi.org/10.1117/12.2202805.
[27] D.M. Eler, R.E. Garcia, Using otsu's threshold selection method for eliminating terms in vector space model computation, Information Visualisation (IV), 2013 17th International Conference, IEEE, 2013, pp. 220–226.
[28] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, Orb: an efficient alternative to sift or surf, Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011, pp. 2564–2571.
[29] D.H. Widyantoro, K.I. Saputra, Traffic lights detection and recognition based on color segmentation and circle hough transform, Data and Software Engineering (ICoDSE), 2015 International Conference on, IEEE, 2015, pp. 237–240.
[30] O. Vincent, O. Folorunso, A descriptive algorithm for sobel image edge detection, Proceedings of Informing Science & IT Education Conference (InSITE), 40 (2009), pp. 97–107.
[31] A. Amanatiadis, I. Andreadis, Performance evaluation techniques for image scaling

algorithms, Imaging Systems and Techniques, 2008. IST 2008. IEEE International Workshop on, IEEE, 2008, pp. 114–118.
[32] J. Flusser, T. Suk, Pattern recognition by affine moment invariants, Pattern Recognit. 26 (1) (1993) 167–174.
[33] B. Catanzaro, B.-Y. Su, et al., Efficient, high-quality image contour detection, Computer vision, 2009 IEEE 12th international conference, IEEE, 2009, pp. 2381–2388.
[34] X. Sun, Q. Chen, Defects detecting of gloves based on machine vision, Real-time Computing and Robotics (RCAR), IEEE International Conference on, IEEE, 2016, pp. 169–173.
[35] J.E. Hershberger, J. Snoeyink, Speeding up the Douglas–Peucker Line-Simplification Algorithm, University of British Columbia, Department of Computer Science, 1992.
[36] M.-K. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inf. Theory 8 (2) (1962) 179–187.