Juho Vihonen

Sequential Detection Applied to Line-Scan Gray Level Defect Imaging
Juho Vihonen

Sequential Detection Applied to Line-Scan Gray Level Defect Imaging

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB111, at Tampere University of Technology, on the 31st of August 2009, at 12 noon.
Preliminary assessors

Prof. Dr. Pekka Toivanen
Department of Computer Science
Faculty of Business and Information Technology
University of Kuopio
Finland

Prof. Dr. Juha Röning
Department of Electrical and Information Engineering
Technical Faculty
University of Oulu
Finland

Opponents

Prof. Dr. Björn Kruse
Department of Science and Technology
Linköping University
Sweden

Prof. Dr. Juha Röning
Department of Electrical and Information Engineering
Technical Faculty
University of Oulu
Finland

Custos

Prof. Dr. Ari Visa
Department of Signal Processing
Faculty of Computing and Electrical Engineering
Tampere University of Technology
Finland

ISSN 1459-2045
Abstract

The detection of abnormal changes in the nominal characteristics of industrial applications is a problem typically related to the safety of the technological process and the quality of production. Given a record of measurements, the best solution is often the quickest detection possible of the occurrence of disorder, with as few false alarms as possible. Particularly in signal processing the ability to detect efficiently small changes in possibly long samples of data is of crucial importance. At the present, sequential algorithms are not generally used in image processing, while some of them operate as standards, e.g. in sampling-based inspection of goods. Perhaps one of the reasons is that no proofs of optimality, particularly for the so-called cumulative sum procedure, were available until quite recently. By taking data sequentially, one can deal with problems which have no fixed-size solutions or which at least are computationally prohibitive. Dynamic programming is also often important with intrinsically sequential problems.

Web inspection systems are utilized for electro-optical quality control in paper production lines. To be able to image the entire paper web accurately, such an inspection system may contain tens of cameras having built-in parallel processing for real-time detection and classification of defects and other surface flaws. The web inspection systems have no major difficulties in detecting large defects with high contrast. However, detection of physically small-sized defects having low contrast and unknown dynamics is far from easy under the strict constraints on the available processing time. At the expense of making sample size a random variable, the developed detector is shown to provide computationally inexpensive, yet effective performance in less simple cases that pose problems for current fixed sample size detectors. To meet the real-time requirements, we introduce extreme values as a valuable tool for reducing the data at hand. Even though both the sequential detection and dynamic programming are in some sense areas which are not very actively pursued today, the novel approach presented in this thesis will hopefully reawaken interest in these important fields.

Keywords: sequential change detection, dynamic programming, cumulative sum, web inspection, imaging, extrema
Preface

The work presented in this thesis was carried out during the years 2005-2009 at the Department of Signal Processing, Tampere University of Technology, Finland. The work formed part of two projects called MIMER and ODEN. ODEN can be seen as a successor to MIMER that is now an on-going project funded by industry and the Finnish Funding Agency for Technology and Innovation (Tekes).

Sequential analysis has not been a vibrant statistical field within our University and so it is reasonable to say that the subject found me. Some problems are intrinsically sequential, which should have certain implications for the design of a technological solution. As far as I believe, many of the discussed issues are therefore relatively timeless and will continue to be relevant in the never-ending quest for better performance. I have also attempted to make this thesis more widely accessible than most of the literature which it is based on. Without sacrificing rigor in the mathematics, the style is occasionally tutorial. However, the contributions are not solely mine and I would like to thank a number of people who have directly or indirectly contributed to the MIMER and ODEN projects as well as to this thesis. First, for all the funding, I am particularly indebted to Pekka Yrjölä at Tekes. I also thank my supervisor, Prof. Ari Visa, who has been a constant source of intellectual guidance and patience. In addition I want to thank my colleagues Juha Jylhä, Timo Ala-Kleemola, Marja Ruotsalainen, Riitta Kerminen, and Jarmo Kauppila for brilliant inspiration, countless discussions, and technical assistance during the past several years. Additionally, the proofreading and helpful suggestions on technical issues of Juhani Rauhamaa, ABB Oy, have been absolutely superb. I would also like to thank Tommi Huotilainen, ABB Oy, for digital material and cheerful advice. All of these people who worked with me while writing this thesis, have indeed been a rich source of ideas and a stimulus to me to improve mine. The constructive comments of the pre-examiners, Prof. Pekka Toivanen and Prof. Juha Röning, are most gratefully appreciated.

Finally, I thank my family and my beloved Tuuli for their everlasting support. The support of Nokia Foundation is also gratefully acknowledged.

Tampere, Juho Vihonen
July 2009

Juho Vihonen

Contents

Abstract i
Preface ii
Contents iii
List of Publications v
Contribution vi
List of Abbreviations x
Basic Notation xi

1 Introduction 1
  1.1 Background ...................................................... 5
  1.2 Objectives ....................................................... 10
  1.3 Outline .......................................................... 12

2 Change Detection Using the CUSUM Algorithm 14
  2.1 Survey of Optimal Properties ................................. 15
  2.2 Relation to Off-Line Estimation of the Change .............. 23
  2.3 Assessing Run-Length ......................................... 26
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 Line-Scan Defect Monitoring</td>
<td>33</td>
</tr>
<tr>
<td>3.1 Selecting Most Likely Pixels</td>
<td>34</td>
</tr>
<tr>
<td>3.2 Detection of Change with Unknown Parameters</td>
<td>37</td>
</tr>
<tr>
<td>3.3 Examples of Defect Segmentation</td>
<td>41</td>
</tr>
<tr>
<td>4 Discussion</td>
<td>48</td>
</tr>
<tr>
<td>5 Conclusion</td>
<td>51</td>
</tr>
<tr>
<td>Bibliography</td>
<td>54</td>
</tr>
<tr>
<td>Errata</td>
<td>63</td>
</tr>
<tr>
<td>Publications</td>
<td>64</td>
</tr>
</tbody>
</table>
List of Publications

This thesis is based on the following publications. Most of the ideas have already been published in the two articles below, and the listed first conference paper describes the early steps of the conducted research. Yet, the author wishes to present a more complete treatment and, to some extent, readdress and critically re-examine the proposed techniques. Thus, the thesis contains some unpublished material, too.


Publication IV is referred to in the text most frequently. All the Publications are also briefly described next. The cited literature is listed at the end of this thesis.
Contribution

To help meet the detection challenges ahead, we show how the classical formulations in the theory of sequential analysis and change detection can enrich and broaden the subject of on-line surface defect imaging. The related algorithms are often inherently recursive and satisfy stringent requirements with respect to accuracy and efficiency. We claim that it is appealing to apply the theory of sequential change detection if the imaged abnormal change of interest is small and persistent. Particularly, the evidence in favor of an abnormal change can be conveniently strengthened by capturing new frames in the case of a low-contrast defect. Contributions of Publications I-V are listed and summarized below.

I This paper presents an early application of the CUSUM algorithm for line-scan imaging. The imaging problem is new in sequential analysis and the algorithm is clearly immature. The idea of adopting sequential analysis for ABB’s high definition line-scan cameras originates from the author’s earlier experience in remote sensing, where detection procedures have revolved around fixed sample size tests (see e.g. Vihonen [97]). Since web imaging systems produce frames in a temporal order, sequentially, it was the author’s original idea to apply the CUSUM algorithm to the web inspection problem. The residual analysis discussed in this paper was used to influence the CUSUM statistic. But since the assumption of periodicity of the surface profile fluctuation cannot be satisfied at a sufficient level, parametric models otherwise useful in spectral estimation do not appear to be that easily applicable for detecting subtle defects reliably in our case. In this regard, many authors have modeled pixel intensities as an autoregressive process, which is of course not a novel approach. However, these observations changed the course of the whole line-scan research. Particularly the fruitful discussions with T. Ala-Kleemola have contributed to this paper, as well as the author’s perspective and his knowledge as to how the theory of sequential analysis can be best applied to the web imaging problem later.

II The paper co-authored with J. Jylhä relates to aircraft fatigue management. The reported problems of parametric models and conventional Fourier analysis led to the necessity to develop a new signal model for structural high frequency
vibrations, mostly because vibration intensity and frequency band may vary considerably over a very short period of time. A lesson learned in this work is that one may easily cancel highly useful pieces of information if the typically harmless filtering operations for spectral manipulation are applied without carefully knowing the nature of the problem at hand; the same applies with spatial frequencies that cause the observed intensity fluctuation of the surface profile in the web imaging problem too. On the basis of collective observations made by J. Jylhä, T. Ala-Kleemola, and the author, J. Jylhä developed a novel signal decomposition for the F-18 Hornet structure strain modeling research. To reduce intervals of physical inspections, the decomposition extracts high-frequency vibration for fatigue modeling of certain problematic structural details of aircraft. In determining the temporarily most significant strain components that practically dictate the induced damage, extreme values of the strain signal form the core of many analyses in the field of fatigue management. However, the discoveries made turned our attention to the extreme values practically unexploited in imaging. In Publications III and IV, the author shows these values to be of considerable importance also in imaging-related hypothesis testing. For further reference, the subsequent work to extend the lifetime connects the caused fatigue to flight maneuvers in order to appraise the practice of the whole Finnish Air Force F-18 fleet; see Ruotsalainen et al. [85], Jylhä et al. [39].

The next development stage of the proposed sequential detector is described in this article, but all the benefits of extreme values have not yet been found. Consequently, the importance of extreme values is emphasized from the viewpoint of web imaging and a simple correlation-based update of the CUSUM statistics is also proposed in order to cope with the optics-related issues. Since they quantify variations in the mass distribution of pulp and other raw materials in a simple manner, it was the author’s original idea to simplify the contents of line frames using the extreme values. The nature of data in web imaging is also conceptually similar to that of Publication II in the sense that, for example, neither the variance nor the resolution are constant over the imaged web surface, partly due to the effect of illumination. As the extrema describe nonstationarities of a signal, the author suggested that they contain the preferable information for efficient sequential hypothesis testing. Inspired by the connections between remote sensing and capabilities of the new generation CCD cameras of the web imaging system, highly usable “Viterbi-like” generalization and further methodological advances were yet to come. The cooperation of T. Ala-Kleemola and J. Rauhamaa is gratefully acknowledged.
IV In this article, the novel sequential detector, now called the DP-CUSUM, is developed to a mature stage: compared with a state-of-the-art fixed sample size detector, superior performance is observed with small persistent changes. With too small a sample size, an evolutionary change often goes unnoticed and, as the benchmarking in the article shows, the simple criterion of minimizing the expected number of necessary observations is found to be an important step in the development of enhanced detection for imaging. Hence, integrating the DP formalism and the CUSUM algorithm provides a sound foundation for detection and further development. The contributions are two-fold: the author shows also that the efficiency of detection of a state-of-the-art technique can be increased dramatically via extreme values. Consequently, it was the author’s original idea to introduce dynamic programming and sequential change detection jointly with the aspects related to web imaging and extreme values. The mathematical assistance of T. Ala-Kleemola and technical assistance of J. Rauhamaa greatly improved this article.

V The paper co-authored with M. Ruotsalainen applies the new ideas to chemical agent detection and identification with so-called swept-field ion mobility spectrometry. The CUSUM-based detector is run continuously as a preprocessor, and the neural net-based identifier is used only when an alarm is triggered. The key idea here is that we use a very sensitive detector for change detection and an identifier that double-checks a number of sweeps before the final alarm is given. Using separate detection and identification solutions as proposed is common practice in recognition-oriented signal processing, as is also the case with the web inspection application. To properly select the sweeps under hazardous exposure for categorization, the author introduced a new generic change point estimation rule that had its origin in the segmentation experiments discussed in Publication III. The experiments in this paper verified that the sequential estimation of the change points, where sweeps before and after the change are utilized collectively according to the author’s original idea, improved the overall identification considerably when compared with an existing identification system. Due to the relatively broad range of ion mobilities observed under exposure, the DP formalism was not utilized in this paper as presented elsewhere in this thesis.
ing valuable comments that resulted in significant improvements in the manuscripts. The author, on the other hand, provided comments that resulted in improvements in Publications II and V. The sequential experiments in Publication V were also based on the author’s derivations and implementation. The co-authors have seen these descriptions and agree with the author.
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARL</td>
<td>Average run-length</td>
</tr>
<tr>
<td>CCD</td>
<td>Charged-coupled device</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative density function</td>
</tr>
<tr>
<td>CMF</td>
<td>Cumulative mass function</td>
</tr>
<tr>
<td>CUSUM</td>
<td>CUmulative SUM (algorithm)</td>
</tr>
<tr>
<td>DP</td>
<td>Dynamic programming</td>
</tr>
<tr>
<td>i.i.d.</td>
<td>Independent and identically distributed</td>
</tr>
<tr>
<td>MLE</td>
<td>Maximum likelihood estimate</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability density function</td>
</tr>
<tr>
<td>PMF</td>
<td>Probability mass function</td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver operating characteristics</td>
</tr>
<tr>
<td>SDRL</td>
<td>Standard deviation of run-length</td>
</tr>
<tr>
<td>SPRT</td>
<td>Sequential probability ratio test</td>
</tr>
</tbody>
</table>
## Basic Notation

Indicated usually by sub- or superscripts, some symbols may have locally more specific other meanings.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$</td>
<td>Parameter(s).</td>
</tr>
<tr>
<td>$\theta_0$</td>
<td>Parameter(s) before change.</td>
</tr>
<tr>
<td>$\theta_1$</td>
<td>Parameter(s) after change.</td>
</tr>
<tr>
<td>$t_0$</td>
<td>Change time.</td>
</tr>
<tr>
<td>$\hat{t}_0$</td>
<td>Estimate of change time.</td>
</tr>
<tr>
<td>$t_a$</td>
<td>Alarm time.</td>
</tr>
<tr>
<td>$\tau_{fa}$</td>
<td>Mean time between false alarms.</td>
</tr>
<tr>
<td>$\tau_d$</td>
<td>Mean delay to detection.</td>
</tr>
<tr>
<td>$\tau^*_{d}$</td>
<td>Worst mean delay to detection.</td>
</tr>
<tr>
<td>$h, -\epsilon, \gamma$</td>
<td>Thresholds.</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Magnitude of jump (change).</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Mean.</td>
</tr>
<tr>
<td>$y, x$</td>
<td>Argument of the distribution functions.</td>
</tr>
<tr>
<td>$f(x), p(x)$</td>
<td>Probability density function, PDF.</td>
</tr>
<tr>
<td>$p_\theta(y)$</td>
<td>Parameterized probability density function, PDF.</td>
</tr>
<tr>
<td>$s_k$</td>
<td>Logarithm of the likelihood ratio at time $k$.</td>
</tr>
<tr>
<td>$F(x), P(x)$</td>
<td>Cumulative distribution function, CDF.</td>
</tr>
<tr>
<td>$N$</td>
<td>Number of observations.</td>
</tr>
<tr>
<td>$N(0, 1)$</td>
<td>Normal distribution with zero mean and unit variance.</td>
</tr>
<tr>
<td>$P(x)$</td>
<td>Probability of the event $x$.</td>
</tr>
<tr>
<td>$E$</td>
<td>Expectation.</td>
</tr>
<tr>
<td>$H_0$</td>
<td>Null-hypothesis.</td>
</tr>
<tr>
<td>$H_1$</td>
<td>Alternative hypothesis.</td>
</tr>
<tr>
<td>$k$</td>
<td>Discrete time instant.</td>
</tr>
<tr>
<td>$y_k$</td>
<td>Observation (random variable) at $k^{th}$ sampling.</td>
</tr>
<tr>
<td>$g_k$</td>
<td>Decision function (statistic) of the CUSUM algorithm.</td>
</tr>
<tr>
<td>$m_k$</td>
<td>Current maximum value of $g_k$ after its last renewal.</td>
</tr>
</tbody>
</table>
$S_k^j$  Log-likelihood ratio for observations from $y_j$ to $y_k$.

$K$  Number of frames used by the DP search procedure.

$I_n$  Scoring sum of a pixel $x_n$ at the $n^{th}$ frame.

$j$  Number of candidate pixels.

$\ell$  Number of CUSUM-based tests in the image plane incorporating the measurement history.

$A$  Amplitude of the signal embedded in additive noise.

$P_d$  Probability of detection.

$P_{fa}$  Probability of false alarm.
Chapter 1

Introduction

Many monitoring problems can be formulated as the problem of detecting a change in the parameters of a stochastic system. Any change in the parameters of the system that occurs either instantaneously or at least very fast with respect to the sampling period of the measurements provides typically a starting point for the design and performance analysis of a change detection algorithm. In most applications, however, the main problem is to detect small changes such as deviations from the normal operating conditions of an industrial process. Thus, the discussed changes by no means refer to abnormal events of large magnitude only. The primary concrete application that interests us performs electro-optical, recognition-oriented quality monitoring, which is called web inspection when the system is used at a paper mill. Fig. 1.1 illustrates a schematic of the web imaging system with electronic cabinets and peripheral devices. Relying on high definition line-scan cameras, see Fig. 1.2, the web inspection system, also called a web imaging system, is capable of producing a continuous flow of high-resolution line images by sampling the scene constantly. The resulting image consisting of several frames is comparable to a 256 gray-level photograph. Since a line image, termed also a line frame, is produced at each sampling, the paper web’s motion past the line camera produces a two-dimensional image ready for digital processing. In imaging, barely distinguishable entities can be expressed by using the word “low-contrast”. At a paper mill, holes, stains such as oil and dirt spots, wrinkles, and other surface defects, even squeezed bugs, represent the essential examples of anomalies that degrade the final product; see Fig. 1.3. Since these sorts of defects can be modeled by an additive change in the mean intensity taking place somewhere in the image plane, their detection can be approached by monitoring for an unusually systematic change of intensity in the normal operating conditions. Partitioning the given image into its constituent parts is often called segmentation. Broadly, the segmentation of a signal is typically the first step of a recognition procedure in which features of data are extracted more accurately in
CHAPTER 1. INTRODUCTION

Figure 1.1: A web imaging system with a scalable architecture according to requirements of the papermachine on which it is installed. The main parts of a modern paper web imaging system are a camera beam in the cross-direction of the web, a light source beam, defect information processing electronics, and an operator station. For the camera beam, high quality papers may require multiple line-scan cameras per meter; low camera density is often connected to sensible cost. Other additional devices include a pulse encoder for speed measurement, alarm devices, and colormarkers to facilitate the manual removal of defects at a winder or re-reeler; see [77].

order to produce interpretations regarding the signal’s content; see, e.g., Loncaric [54], Zhang and Lu [109]. However, we shall not label or interpret the entities found.

The principal subject of this thesis is applying sequential hypothesis testing for imaging. Detection is viewed as a special two-state segmentation case: the frames are first reduced to interesting and non-interesting pixels for which we develop sequential change detection. For the problems studied in detail, there exist fixed sample solutions too, but the reason for introducing a novel sequential detector is to provide greater efficiency, partly in terms of its far-reaching generalization capability. The so-called cumulative sum procedure, known as the CUSUM algorithm, proposed by Page [65] during the 1950s is introduced as a valuable tool for change detection in imaging. Because of its optimal properties, first discussed by Lorden [55] in the
Figure 1.2: A high definition line-scan camera. The light-sensitive elements (pixels) are organized in the cross-direction of the web in the sensor chip of this line-camera. To cope with the fast movement of the paper web, built-in special signal processing circuits and parallel processing capability make it possible to realize advanced real-time defect detection algorithms, which could not be run fast enough in general purpose PCs. In addition, digitized images can be further processed by computers at the operator station; see Rauhamaa and Reinius [77], Rauhamaa [76].

1970s, this algorithm seems to deliver the best results in many cases, though often its use has meant not using what appears to be the “best” method for a particular problem. In many cases, we shall therefore ignore the possibility of finding optimal solutions and concentrate instead on sequential change detection issues that one might encounter when working with machines that can see. In this thesis, we solve the problem of detecting low-contrast abnormalities using a statistical approach, though we speak about changes in the parameters of the underlying probability distributions. From this point of view, the pixels describing details on the web surface are a realization of a fluctuating random process, which can easily give rise to false alarms. In the web imaging system, false-alarm-limited operation is important for two reasons: in addition to the fact that it is computationally expensive to process
false alarms at the recognition stage, the operator will very quickly stop using the monitoring tools if his or her present reasonings are inconsistent with the system’s automatic reports. For robust operation, utilizing the recognized “best practices” of sequential analysis correctly is therefore of crucial importance and we apply the formalism of sequential change detection to continuous electro-optical imaging to the extent that there is a unifying theme to many already published works. We also adopt some properties of dynamic programming (DP), which is a central procedure for dealing with intrinsically sequential problems having some unknown dynamics.

Figure 1.3: Some examples of real anomalies on the surface of the paper web. The first on the left shows an oil spot, the second is a wet spot of dirt, the next is a fragment of badly wrinkled paper sheet, and the last one on the right is a piece of paper with reduced thickness. From the point of view of this thesis, not all the cases are challenging, but they represent nicely many cases where 100% reliability is expected, hopefully, from a single detection procedure.

In this thesis, sequential hypothesis testing is developed for imaging in the conventional non-Bayesian, non-decision-theoretic context. The expert reader might criticize the approach since, within the process industry, the goal of planning is often to minimize some expected loss by selecting the course of action that minimizes the total expected cost. For example, adjusting some actuator may have several different outcomes, but ordinarily the loss incurred for making an error is greater than the loss incurred for being correct. There are many texts available on this; see e.g. Duda et al. [17], Bishop [11]. With the exception of some cases, the optimal
Bayesian solution is in practice frequently merely conceptual in the sense that it cannot be defined analytically. At a paper mill, for example, planning a corrective action often involves many imponderables which make it practically difficult, even if theoretically possible, to formulate a reasonable loss function in a probabilistic context. Although the web imaging system forms cause-and-effect relationships and interprets the observed defects at the recognition stage, its ultimate goal is to find as many defects as possible, practically without false alarms. Executing adjustments, e.g. for the coating blade, is left in practice for the operator to decide since the questions to be answered do not usually have mathematically neat boundaries. To do this automatically in the sequential framework, the developing theory of so-called fault isolation and change diagnosis could be utilized. The objective is to determine, upon detection of change in a process, which one of the possible changes has actually occurred; see e.g. Nikiforov [62], Lai [50] for some more or less complex examples. In this sense, perhaps the greatest problem is that the pieces of information found appear to fit many different raw-material and process-related scenarios, and very likely many appropriate actions are required, e.g. to repair equipment. Obviously, fault isolation and diagnosis-related inferences are far beyond the scope of this work.

1.1 Background

The modern theory of sequential analysis came into existence in the 1940s in response to demands for more efficient inspection procedures and weapons testing. Its principal architect, A. Wald, admirably summarizes these early steps, which can be regarded as extraordinary accomplishments at that time, in his book [104]. During that period, the word “inspection” typically referred to testing for the proportion of defectives on the basis of a random sample which was taken e.g. from a large batch of industrial products or war-time supplies that could be judged good or defective. Over the past few decades there has been a significant increase in the number of real problems concerned with continuous inspection, which nowadays is often considered a synonym for sequential change detection. Here, the terms “continuous” and “sequential” refer to some sort of monitoring or process control through which we obtain and process observations constantly. Since much of the interesting information lies in the observed changes, condition-based digitalized processing, both on-line and off-line, has become widely used. The reason for this is simple: many real problems are concerned with questions such as safety and with ecological and economic issues. All this, as well as the exponential growth of the available information, have led to a number of modifications being proposed for Wald’s sequential probability ratio test, SPRT. During this decade, it has finally also become possible to give a somewhat complete theoretical analysis of many of the proposals and hence
to understand them better. In this context, the efforts of Basseville and Nikiforov [9] are commendable and their book is repeatedly cited in this work. They also describe other competing possibilities that could be applicable for detecting a change. Fine treatment is also provided by Siegmund [89] on clinical trials and by Lai [51], who describes also many important unsolved problems.

A topic closely related to the sequential analysis and intensively investigated is time-series analysis. Typically, the assumptions underlying these investigations are that the properties or parameters describing the data are either constant or vary slowly with time. Though the problem of estimation exists to some extent with all change-detection algorithms, the parameters are often also partly known, e.g. because of earlier experience. Now let us imagine a process which produces a potentially infinite sequence of observations $y_1, y_2, \ldots$. Initially, the process is “in-control” in the sense that we are satisfied with the record of $y$’s, without taking any action. At some unknown time $t_0$ the process experiences a change and becomes “out-of-control”, meaning that our task is to infer from the $y$’s that this change has actually taken place. Appropriate action should be taken “as soon as possible” since, after the change at time $t_0$, we use abnormal measurements which may have costly practical consequences. This means that minimization of the amount of data necessary to detect the change is of crucial importance in order to signal a change in the process parameters from their target values. According to Basseville and Nikiforov [9], the sequential detection problem can be formally described as follows. Before the unknown change time $t_0$, the so-called conditional density parameter $\theta$ is assumed constant and said to be equal to $\theta_0$; i.e., the illustrative technological process is “in-control”. After the change, the parameter is assumed to be equal to $\theta_1$, signifying that the process is “out-of-control”. The on-line change detection problem is to detect the occurrence of the change as soon as possible, with a low fixed rate of false alarms before $t_0$. The alarm time $t_a$ is the time at which the change which has occurred is detected. If $t_a = n$, it is sufficient to continue sampling up to time $n$, which explains the on-line point of view and the sequential nature of the sampling. In the case of fixed sample size, or an off-line change detection problem, the estimation of the change time $t_0$ is not usually required and hypotheses testing of the observations $y_1, \ldots, y_N$ is done, where $N$ is fixed. The off-line approach is often useful while designing a decision or estimation function that is finally implemented on-line.

With the web imaging application, on-line extraction of the useful information becomes essential, primarily due to the temporal nature of the imaged scenes. As the paper web moves, more and more frames describing the web’s surface profile are obtained with very short intervals between them. Detection and segmentation are typically the early steps in on-line extraction of the useful information, but unfortunately doing this autonomously is one of the most difficult tasks in digital image processing since eventual failure is almost surely guaranteed with weak solutions. Sev-
eral general-purpose algorithms and techniques, such as those discussed by Therrien et al. [93] and more recently by Radke et al. [73], have been developed for detection and segmentation. These articles summarize several basic image change detection approaches starting from the use of simple threshold, image differing, masks, and estimation techniques without emphasizing a particular application area; see also e.g. Gonzalez and Woods [25], Pratt [70], Shapiro and Stockman [87]. The actual algorithms for the surface defect detection and segmentation used in commercially available systems have not been reported due to intellectual property constraints. However, systematic reviews of recent advances which treat defect detection and surface inspection as texture analysis problems are given by e.g. Kumar [48], Xie [107]. According to Kumar [48], statistical, spectral, and stochastic model-based methods form the three main categories of the fabric defect inspection taxonomy. In statistical approaches, the defect-free regions are typically assumed to extend over a significant portion of the image, making for example texture measures of regularity useful on materials that exhibit a global structure. A suitable gray level thresholding limit can also often be determined straightforwardly from the defect-free region, though many different images may share the same histogram, which increases the risk of error. Similarly, the high degree of periodicity of basic texture in the fabric permits the use of spectral methods and for example the Fourier and wavelet transforms are widely applied tools for characterizing different scales and discontinuities of textures effectively; see also e.g. Rioul and Vetterli [82]. Generally, the spectral approaches work well if the frequency components associated with a defect-free region do not share the band of defective regions. Stochastic model-based approaches are typically more suitable in cases where the surface profile shows more or less random fluctuation. Because of the fluctuations, the size of segments is often adapted to the local properties of the pixels with respect to the same characteristics, such as intensity and boundary line. If the task requires more than one pass through all the pixels, the computational needs may end up being far from what is acceptable in an on-line inspection system; see e.g. Moon [58], Xu and Wunsch [108], Höppner et al. [33] about clustering. As this short review shows, there are several types of defect detection and segmentation schemes, and depending on the application, one may be preferred to the others. But in spite of the fact that the literature on fabric defect detection using digital imaging is vast, apparently none of the over 300 papers mentioned in the surveys [48, 107] proposes anything at all sequential. In the on-line change segmentation problem—as we define it for the web imaging application—, the problems of on-line detection and segmentation coexist and decisions should be reached sequentially, as when frames are captured. This on-line nature continues to complicate the applicability of some earlier work (see e.g. Iivarinen et al. [36]), which in other words indicates that the algorithms currently available for the web inspection are not completely satisfactory.
The typical way to assess the performance of a detector is to derive properties of the optimum detector after the signal of interest has been corrupted by i.i.d. Gaussian noise. Particularly for small entities embedded in a near-uniform background, a common way to maximize the probability of detection in real-time is to perform convolution between the corrupted signal and its “replica” for correlation; see e.g. Proakis [71], Gonzalez et al. [26]. The issue of optimality was apparently first discussed under the term “matched filter” by van Vleck and Middleton [96] during the 1940s. With appropriate preprocessing, such as passing the original image through a high-pass filter or other spectral manipulations, see e.g. Oppenheim and Schafer [64], Stoica and Moses [92], Lim [53], the image statistics have been shown to be highly Gaussian; see Hunt and Cannon [34]. Thus, closely related to the case with matched filtering, the basic method for searching a set of pixels that describe some interesting event is to loop through all the pixels in the image and compare them with a known model, called a template. In detection, the central assumption of such an approach is that the events of interest have a repetitive form of geometry which yields high correlation if a well-matching event, like a discrete spot of dirt, appears again on the fast-moving paper web. The goodness of match is assessed by calculating the sum of products between the coefficients in the template and the corresponding pixels in the area spanned by the template. Because the output of the template depends on the intensity of the input signal but not on the signal’s detailed characteristics, the method is simple to implement as well as to understand. However, typically more than one template are needed for different scales and rotations, and so the accuracy of binary decisions depends primarily upon the accuracy and completeness of the available templates; see e.g. Uenohara and Kanade [95] and the recent advances described there. When the error margins are small, there exists only a handful of methods that are useful across many domains of application. One such method is called dynamic programming, DP for short. The term was originally used in the 1940s by R. Bellman (see e.g. Dreyfus [15]) to describe the procedure of solving problems where the best decisions are found one after another. The Viterbi algorithm, named after its developer A. Viterbi, is a DP procedure used for finding the most likely sequence of so-called hidden states, or causes, that result in a sequence of observed events; see e.g. Forney [22] and Viterbi [103, 102].

From the point of view of this thesis, the chart in Fig. 1.4 summarizes the setup: a sensor acquires measurements, a digital image, describing an event so that preprocessing improves this data in ways that increase the chances of success of subsequent processing, such as detection and segmentation. At present, solid state CCD cameras having low noise and good linearity are the most common sensing devices used in industrial machine vision; see e.g. Gonzalez and Woods [25], Haley and Kondepudy [27]. Since a CCD is based on fixed sensing elements of equal size so that their readout frequency is basically limited by analog-to-digital conversion, these devices
Figure 1.4: The early processing of many systems can be broken down into components such as those shown here. A sensor converts physical events to signal data. After that, a preprocessor modifies the input signal according to some directives. At the detection stage, if an interesting change takes place, indication of its existence is given. Since information is usually incomplete and noisy, some uncertainty is always involved in the decisions.

provide precise spatio-temporal quantization that enables accurate representation of imaged events in a computer. However, a simple lens obeys an optical law called the \( \cos^{4\text{th}} \) law; see e.g. Holst [32], Klein and Furtak [46]. This law basically says that the natural light intensity fall-off is approximately proportional to the fourth power of the cosine of the angle at which the light impinges on a sensor array; a fragment of raw data is shown in Fig. 1.5. Built-in image preprocessing compensates the intensity fall-off effect when converting raw sensor data to standard image, but the lens inflicts also a geometry-related error through the picture elements located further away from the center of the lens. If the angle is oblique, the change in the shape and size of an imaged defect may become considerable, but for optimized detection a template-based detector may be matched only at the center of the field of view. Although not objectionable to the average viewer at low levels, this mismatch can significantly impair computer vision detection algorithms that rely on precise geometrical appearance in analyzing a scene. The subsequent DP formalism, which in optimum circumstances may replace a bank of templates, is capable of
tackling this typical problem of varying level of detail, but because of change detection issues we will intentionally pay less attention to this type of detail. Although only little time can be spent if compared with off-line analysis (see e.g. Wu et al. [106]), tailored and defect-specific preprocessing can also lead to a detection and segmentation performance far exceeding that which can be obtained using simple normalization techniques; see e.g. Astola and Kuosmanen [3], Friedman [23], Haykin [29] for further reference and different application domains.

![Graph](image)

Figure 1.5: The intensity fall-off effect illustrated using a sheet of standard drawing paper. This phenomenon of brightness attenuation away from the frame center is prevalent with vision systems and is compensated for example in the snapshots shown in Fig. 1.3. It should be noted that the area covered by each pixel changes when viewed from increasing angles and the nominal resolution is typically reached only for pixels near the center of the frame.

### 1.2 Objectives

Current change detection algorithms have been developed mostly for elementary industrial applications, including e.g. fault detection in individual sensors, analysis of actuator signals, monitoring of fluid and gas flows, or simply inspection of goods. In our attempt to capture features of the CUSUM algorithm for use with arrays of data, the first motivation lies in the definition of criteria used for deriving the algorithm. As already mentioned, the typical on-line problem is to detect the occurrence of a change as soon as possible, with a fixed rate of false alarms before the change time \( t_0 \). In the web imaging application, the optimal solution, according to the above-mentioned criterion, is basically a trade-off between quick detection and few false alarms. Though many of the detected minor defects do not necessitate instant
1.2. OBJECTIVES

planning of corrective action, quick reporting of the monitoring system is ultimately economical. The second motivation comes from practical experience with different types of detection algorithms for image processing which, as far as we believe, do not employ the fundamental properties of sequential analysis. While at least such approaches are virtually nonexistent, detectors based on fixed template matching have become popular in spite of their rather problem-specific and dedicated nature, though the amount of data needed for a definite decision depends often on the data itself. In that context, a third motivation stems from computational efficiency. In Fig. 1.1, cross-direction resolution of the web is determined by the camera density and the number of light-sensitive elements in the CCD in cross-direction. Machine direction resolution depends roughly on pixel size and exposure time. Assuming that the paper width is 10 meters and machine speed 40 meters per second, a maximum of 400 square meters of paper may have to be inspected at each second. If a resolution of one square millimeter is the target, this means that approximately 400 Mbytes of information have to be processed each second, as explained by Rauhamaa and Rein ius [77]. Though a high camera density allows distributed computing, the volume of acquired data is huge, which emphasizes the need for resource-constrained, yet effective, algorithmic solutions to process the data at hand. Since typically less than 0.00001 % of the data, or pixels as here, may be considered interesting, a computationally inexpensive ability to detect and segment small changes using built-in electronics of the smart CCD cameras without the need to store or iterate large volumes of data at a time is of key interest. An on-line change detection and segmentation algorithm exhibiting the above-mentioned properties is potentially a powerful tool for the web imaging system.

Low-contrast defects occupying a few pixels per line frame and choice of detector are interdependent and linked to the question of efficiency. To summarize the proposed detector-concept briefly, the DP is adopted to enumerate a search space of candidate pixels under a simple scoring scheme in order to update CUSUM-based decision statistics. The objective in using these is to incorporate the measurement history. Because the line-scan cameras have an integrated short-term frame buffer, a limited number of the most recent frames is available for rapid access with strict constraints on the available computing time. The objective of the DP is to reduce a number of these frames to a suitable form so that optimal pixels of a low-contrast defect in the frame buffer, and those yet to come, can be found using only a fraction of all the pixels available compared with off-line analysis, e.g. at the operator station. In spite of the fact that the line-scan cameras are fully packed with state-of-the-art specialized signal processing circuits, the DP breaks down the overall detection problem in order to accommodate the constraints on processing time. Because each defect is unique, detectability of low-contrast defects depends on the statistical tool that is used for detection. No universal detector exists. As already mentioned,
defect detection of web surfaces has been accomplished historically by thresholding and using matched filters. If a sequential detector declares detection after a random number of observations but never later than a fixed-size detector, it has a reasonable claim to be regarded as more efficient, assuming that the probabilities of error are equal. Hence, the detection problem is approached from the point of view of “quickest detection”, which is in our case another way of saying that e.g. a template-based detector may reject a plausible but unproven hypothesis too early as only a fixed number of observations are taken at a time though more might be available, as when frames are captured constantly. From many comparative studies too, such as Roberts [84], Basseville [8], it can be deduced that even less complicated sequential tests can outperform conventional fixed-size detectors at least in efficiency. Since sequential techniques, like the CUSUM algorithm, can incorporate the measurement history, this thesis seeks to provide practitioners with an understanding of the application of sequential analysis to on-line imaging. The subsequent formalism is equally applicable to other imaging-related applications, including infra-red, radar, and sonar sensors, and more conventional video surveillance; see e.g. Holst [32], Klemm [47], Foresti et al. [21].

1.3 Outline

In the following chapters, the emphasis is on the CUSUM algorithm and DP to the extent considered necessary for most of the engineering community. While focusing on the most recent advancements, the theoretical treatment is believed to be balanced in terms of the simple models suitable for on-line detection. The basic assumptions and reasoning are made as explicit as possible to facilitate comprehension. Readers mainly interested in methodology who are willing to accept that a correctly parametrized CUSUM algorithm provides quickest detection, may completely skip the introductions of Chapter 2. Although a glimpse at the DP formalism at the beginning of Chapter 3 might be useful, readers familiar with standards of sequential change detection may in fact go straight on to the experiments in the middle of Chapter 3.

There are roughly only two parts in this thesis. The first part, containing Chapter 2, addresses sequential change detection. The second part contains the three chapters which follow below and presents our new concepts regarding defect detection. As a preliminary to our study of sequential detection in on-line imaging, we discuss the SPRT and its relation to the CUSUM algorithm in Chapter 2. Although it is unlikely that the SPRT would be used in practice, it provides the fundamentals for studying the CUSUM algorithm. Readers who want complete mathematical results and proofs must consult other books or articles, indicated in references. The main reason for
this is that although the meaning is usually clear, there are technical mathematical
problems that often make the proofs very complex. Because of this, the well-known
case describing a change in the mean of a noise-obscured signal is considered as our
primary example. We also consider aspects of change-time estimation together with
some intuitive performance indexes of the CUSUM algorithm.

The algorithm to be developed should allow us to detect several types of abnormal
events with minimum computational effort. To achieve this, the main conceptual
ideas are introduced in Chapter 3 in a context built upon the basics discussed in
Chapter 2. Since it is application-driven, designing a sequential detector also in-
cludes the problem of choosing a decision rule. In this sense, Chapter 3 forms the
core of this thesis, where the main ideas introduced in Chapter 2 are applied to the
web imaging problem. The goal is to detect abnormal changes in the mean intensity.
A change that does not modify the mean value is considered undetectable. Thus,
it should be noted that the thesis is primarily concerned with very simple models—
especially those assuming normal distribution in addition to simplifications neces-
sary for real-time computation. For pathological defects, such as X- or O-shaped
changes, for example, clustering could be utilized for putting the problematic statist-
tics together, but such cases are omitted in this work. Indeed, the examples selected
for treatment have been kept rather simple, such that the impression may arise
that methodological guidance is not really required. As a result, some nonessential
coordinate indexing, for example, has been omitted.

Chapters 4 and 5 provide the concluding remarks. This thesis is intended to be
a bridge between sequential analysis and applied imaging problems. Despite the
relative simplicity of our examples, the task is not that straightforward as there is
clearly an implicit lack of relevant theory to which the examples could be related (see
e.g. [94, 38]). With modern technological solutions, many of the cited mathematical
investigations appear also to fall short, although the adopted DP, for example, has
roots in the well-known principle of maximum likelihood. The same relation exists
for the CUSUM algorithm. Consequently, the mathematical methods, the CUSUM
algorithm and the DP formalism, have been chosen because they provide a unified
treatment for a range of real problems.
Chapter 2
Change Detection Using the CUSUM Algorithm

Elementary quality control is concerned with many application areas and it plays an important role in modern industries. Considering the numerous factors that influence production, e.g. at a paper mill, the measured values of consistency, temperature, pressure, etc. can be considered random variables, a fact which makes statistical change detection algorithms relevant for the purpose of quality control and continuous monitoring. In the following, different faults and anomalies in technological processes are assumed to lead to nonrandom and deterministic changes in model parameters. In practice, a closed-form solution to a real problem may not exist, but as a convenient level of abstraction the probability distributions of the measurements may be considered to depend critically upon, for example, the quality of the raw materials. Even without complete understanding of the underlying physics, most of the time quick detection of the disorder is highly desirable simply to see if the technological process has to be stopped, checked, and repaired. However, if the web imaging system detects a minor defect, this is naturally too early to form any conclusions regarding the origin of the defect or even consider drastic corrective measures.

In very general terms, there are two reasons for introducing sequential analysis into change detection. According to Siegmund [89], one reason is to solve more efficiently a problem which has a fixed sample solution; i.e., on-line change detection problems are often close to conventional change detection problems. The other is to deal with problems for which no fixed sample solutions exist. For instance, if observations are collected in on-line mode with high frequency, batch processing and conventional fixed-size algorithms may be too complex or at least too time-consuming. Some problems are also intrinsically sequential and cannot be approached without taking their sequential aspects into account, but the necessity to apply an on-line algorithm
2.1 Survey of Optimal Properties

Let us assume that we obtain an independent random variable \( y_k \) at time \( k \) with a probability density function, abbreviated by PDF, \( p_\theta(y_k) \) depending upon one or more scalar parameters described by \( \theta \). For coherent notation in terms of the related literature, we assume that the parameter \( \theta \) is equal to \( \theta_0 \) or \( \theta_1 \). It is convenient to have these parameters known in the theoretical analysis and an illustration of \( N(0, 1) \) and \( N(2, 1) \) Gaussian densities is given in Fig. 2.1, which plays the role of our main example through this chapter. Our primary problem is to detect if \( y_k \) is at the state \( \theta_1 \), which we begin solving by introducing first the SPRT and then the CUSUM algorithm within a general statistical framework. Particularly, our subsequent descriptions of the two algorithms for hypothesis testing are based on a concept that is very important in mathematical statistics, namely the logarithm of the likelihood ratio, defined by

\[
s_k = \ln \frac{p_{\theta_1}(y_k)}{p_{\theta_0}(y_k)},
\]

which we refer to as the log-likelihood ratio. Computing a log-likelihood ratio is typically used for deciding between two hypotheses based on the value of the ratio. Further generalizations of hypothesis testing to more than two states are often more difficult to derive but are nevertheless easy to apply, since they basically involve deciding between the other states using pairs of likelihood ratios just as presented. In academic literature, the function (2.1) is often referred to as the sufficient statistic with respect to the parameter \( \theta \), if no other statistic which can be calculated from \( y_k \) provides any additional information like here; see the early work of Fisher [20]. Negative values of the log-likelihood ratio mean that \( y_k \) was likely to occur at the
CHAPTER 2. CHANGE DETECTION USING THE CUSUM ALGORITHM

Figure 2.1: Based on a study of many examples of some phenomenon, a PDF represents a simple way to characterize the probability distribution of measurements by using a mathematical function. Here, \( p_{\theta_0}(y_k) \) and \( p_{\theta_1}(y_k) \) describe the different states of the process that is being monitored. Observations drawn from these Gaussian distributions are used in many of the following figures.

state \( \theta_0 \). Positive values mean that \( y_k \) was more likely to occur at the state \( \theta_1 \). Assuming that \( E_{\theta_0} \) and \( E_{\theta_1} \) denote the expectations of \( s_k \) under the two distributions \( p_{\theta_0} \) and \( p_{\theta_1} \), we summarize the key statistical property of the ratio (2.1) as follows:

\[
E_{\theta_0}(s_k) < 0 \quad \text{and} \quad E_{\theta_1}(s_k) > 0,
\]

which means that a change in the parameter \( \theta \) changes the sign of the mean value of the log-likelihood ratio.\(^1\) Next, we introduce a notation which is used throughout the chapter. Let

\[
S^k_j = \sum_{i=j}^{k} s_i
\]

be the sum of log-likelihood ratios for the observations from \( y_j \) to \( y_k \), \( k \geq j \). The typical behavior, obviously due to (2.2), of this log-likelihood ratio corresponding to a change in the mean of a Gaussian sequence with constant variance is shown in Fig. 2.2. Generally, the summation \( S^k_j \) is used with different algorithms, but being a function of the data \( y_i \), it is fundamentally a statistic. In the sequential framework,

\(^1\)In practice, the log-likelihood ratio (2.1) itself is often not actually used in the test, though various criteria can lead to it at the derivation stage; see e.g. [43]. For the Gaussian distribution with mean value \( \mu \) and constant variance \( \sigma \), the changing parameter \( \theta \) is \( \mu \) and the PDF is \( p_\theta(y_k) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_k-\mu)^2}{2\sigma^2}} \). The sufficient statistic (2.1) of this special case can be rewritten as \( s_k = \frac{\mu_1 - \mu_0}{\sqrt{\frac{\mu_1^2 + \mu_0^2}{2}}} \) \( (y_k - \frac{\mu_1 + \mu_0}{2}) \), where the subtraction \( y_k - \frac{\mu_1 + \mu_0}{2} \) reflects the important property (2.2).
the decision is taken with the aid of what is called a stopping or decision rule, which refers to the fact that the sampling and testing continue ideally as long as the state $\theta_0$ prevails. Instead of the standard Neyman-Pearson detection criterion [61], by which we maximize the probability of detection subject to the constraint of a fixed false alarm probability, see also e.g. Kay [43], different types of mean delays are typically used to characterize the performance of sequentially operating detection algorithms, as we shall next see.

Consider a sequence of independent random variables $y_1, y_2, \ldots$ with density $p_\theta$. Until the unknown time $t_0$, the parameter $\theta$ is $\theta = \theta_0$ and from $t_0$ becomes $\theta = \theta_1$ so that $\theta_0 \neq \theta_1$. Our problem is to detect the change as soon as possible. In general terms, testing of the two hypotheses can be written:

$$
\begin{align*}
\text{H}_0 & : \theta = \theta_0 \quad \text{and} \\
\text{H}_1 & : \theta = \theta_1.
\end{align*}
$$

Figure 2.2: Piecewise constant mean (dotted line) and corresponding Gaussian signal (solid line) before and after the change (first row). The behavior of the log-likelihood ratio $S_k$, $k = 1, 2, \ldots, 50$ shows a negative drift before change and a positive drift after change (second row). The x-axis corresponds to $k$ and is also referred to as time.
If the parameters $\theta_0$ and $\theta_1$ are known, they specify the population distribution completely. Then, the hypotheses are often called simple in the related literature. Let $t_a$ be the alarm time at which a detection occurs; $t_a \geq t_0$. Given that an on-line algorithm declares detection at this instant, $t_a$ may also be referred to as stopping time. For estimating the efficiency of the detection, both $t_0$ and $t_a$ are generally very useful as they provide convenient expressions for the mean time between false alarms and mean delay for detection. We define the mean time between false alarms as the following quantity:

$$\tau_{fa} = E_{\theta_0}(t_a).$$  \hspace{1cm} (2.5)

The delay for detection is related to the ability of the on-line algorithm to give an alarm when a change actually occurs. This delay depends upon the behavior of the process both before and after the change time, which makes the issue of a convenient definition for this delay more difficult than the delay above. Let us assume that $y_1, \ldots, y_{t_0-1}$ constitutes all the past observations from time 1 to $t_0 - 1$. As the change time $t_0$ and the past observations $y_1, \ldots, y_{t_0-1}$ have an effect on the alarm time $t_a$, the conditional mean delay for detection can be written as the following expectation:

$$E_{\theta_1}(t_a - t_0 + 1|t_a \geq t_0, y_1, \ldots, y_{t_0-1}).$$  \hspace{1cm} (2.6)

The conditional mean delay can be used in several different ways, but the main benefit of this expression is that it makes it possible to define other delays to detection; Basseville and Nikiforov [9], for example, list a few of them. These different delays can be defined with respect to the distributions of the change time $t_0$ and the past observations or using a supremum. Particularly, the conditioning is useful as it can reveal the worst mean delay to detection; i.e., the case in which the decision statistic at the change time is the least favorable with respect to speed of detection. The worst mean delay can be defined as the following quantity:

$$\tau^* d = \sup_{1 \leq t_0 < \infty} \text{ess sup} \ E_{\theta_1}(t_a - t_0 + 1|t_a \geq t_0, y_1, \ldots, y_{t_0-1}).$$ \hspace{1cm} (2.7)

For a set of real numbers, “sup” is defined to be the smallest real number that is greater than or equal to every number in the set. Though we prefer using the max-operator more common with programmers, the essential supremum, denoted by “ess sup”, is the proper generalization for measurable functions of the maximum; see e.g. Stark and Woods [91]. In (2.7), the two suprema are computed for the change time and for the behavior of the process before the change. This criterion, representing the most pessimistic performance situation, was introduced by Lorden [55]. By comparing our examples in Figs. 2.3 and 2.4, the situation related to an algorithm performing better than another should become clear.

In order to examine the previously introduced basic problem, we begin by considering the simple example where the SPRT provides the stopping rule. Wald’s SPRT [104]
employs the decision statistic (2.3) starting with $j = 1$. The boundaries of the SPRT are $-\epsilon$ and $h$, which represent here some conveniently chosen thresholds; i.e., $\epsilon \geq 0$ and $h > 0$. If $S^k_1 \leq -\epsilon$, the decision taken suggests accepting the hypothesis $H_0$ and we deduce that the process is at the state $\theta_0$. If $S^k_1 \geq h$, the hypothesis $H_1$ is accepted and detection is declared. However, accepting $H_1$ means also that the final decision is taken which yields the stopping time. Thus, at time $k$, we continue sampling when $-\epsilon < S^k_1 < h$ or exit at time

$$t_a = \min\{k | (S^k_1 \geq h) \cup (S^k_1 \leq -\epsilon)\}, \quad k = 1, 2, \ldots$$  \hspace{1cm} (2.8)$$

The typical behavior with repeated use of the SPRT is depicted in Fig. 2.3, where 3, 13, 17, 20, 22, and 24 are the times at which the hypothesis $H_0$ is accepted in each successive cycle. The first time at which the hypothesis $H_1$ is accepted, the SPRT stops observation and does not restart a new cycle; i.e., the change is detected at time 37 in Fig. 2.3. In testing a simple hypothesis against a simple alternative with i.i.d. observations, a SPRT is optimal in the sense of minimizing the expected sample size both in the case of $H_0$ and $H_1$ among all tests having no greater error probabilities; see e.g. Siegmund [89] for more details with criticism. The error probabilities refer to the probability of erroneously rejecting $H_0$ at $\theta_0$ or accepting $H_0$ while the true state is $\theta_1$. If at least one of the decision boundaries of the SPRT is large, the condition where $t_a \gg 1$ corresponds to the situation in which the decision function of the SPRT (2.3) has drifted close to the one of its two decision boundaries, while the true state is the opposite. Hence, while the SPRT has minimum expected sample size at $\theta_0$ and at $\theta_1$ according to the so-called Wald-Wolfowitz theorem [105], the maximum sample size over $\theta_0$ and $\theta_1$ can be considerably greater than the optimal fixed sample size with the same error specification. In short, the worst-case mean delay to detection (2.7) may become large. To improve the efficiency, Wald’s theory can be modified in order to achieve other optimal properties, as we shall next show. For a Bayesian approach, see also Shiryaev [88].

The concomitant possibility of the alarm time $t_a$ becoming excessively large seems intolerable in practice, since in change point detection the delay for detection should be the minimum as it represents a period of time during which the underlying process is out-of-control without action of the monitoring system. Nevertheless, the CUSUM algorithm’s [65] significance in change detection problems stems from the properties of SPRT. Particularly, the CUSUM algorithm can be viewed as a renewal process wherein each renewal takes place at the termination of an SPRT that was started at the previous renewal. The boundaries of the SPRT are 0 and $h$, and the word “renewal” refers to the decision statistic reaching 0, which always restarts the SPRT algorithm. Based on the repeated use of SPRT, the CUSUM signals a change at the alarm time

$$t_a = \min\{k | g_k \geq h\},$$  \hspace{1cm} (2.9)$$
where the CUSUM algorithm’s decision rule can be compacted in a recursive manner as

$$g_k = \max\{g_{k-1} + s_k, 0\}, \quad k = 1, 2, \ldots$$  \hspace{1cm} (2.10)

versus \(k\), given \(g_0 = 0\) as the initial decision statistic. The typical behavior is depicted in Fig. 2.4. Note the shorter alarm time than with the SPRT in Fig. 2.3. Replacing the SPRT’s lower threshold \(-\epsilon\) by zero was first intuitively suggested by Page [65]. This intuitive suggestion was later formally proved to be the optimal value, which we discuss in the following.

In sequential analysis, expected sample sizes are the primary measures of performance. More precisely, the efficiency of the CUSUM algorithm is measured using the worst mean detection delay (2.7). The standard minimax criterion for defining an optimal detection scheme can be stated as follows: Find an alarm time \(t_a\) that minimizes the worst mean delay to detection (2.7) subject to a given fixed lower bound on the mean time between false alarms (2.5). That is, one tries to maximize the period between false alarms and minimize the delay to detection, which explains the term “minimax”. For independent observations, Lorden [55] established the optimality of the CUSUM algorithm from an asymptotic point of view, which is often
2.1. SURVEY OF OPTIMAL PROPERTIES

Figure 2.4: Typical behavior of the CUSUM decision statistic $g_k$. Note the shorter alarm time 35 if compared with the SPRT in Fig. 2.3.

called first-order optimality; see Pollak [69]. His result, deduced from the properties of SPRT, states that when the mean time between false alarms (2.5) approaches infinity, the CUSUM algorithm minimizes the worst mean delay to detection. The asymptotic optimality is the CUSUM algorithm’s most important property in practice because a low rate of false alarms is always desirable. Based upon the same criterion of worst mean delay, another optimality result for the CUSUM algorithm is proven by Moustakides [60] in a nonasymptotic framework. His result basically says that the stopping time of the CUSUM algorithm has the smallest worst conditional mean detection delay even under the constraint that false alarms occur with a small mean period. An alternative proof for this result is given by Ritov [83] and the optimality applies, under Lorden’s criterion [55], for most discrete time cases of practical interest. Other major developments are skillfully summarized by Lai [51].

Although optimality theory has not been very influential in the development of sequential analysis in the last four to five decades, the above results state that the CUSUM-based schemes exhibit the best worst-case behavior in detecting changes. In most cases of practical interest, the worst case mean delay to detection (2.7) occurs when the CUSUM statistic (2.10) has been reset to zero immediately before
the change time $t_0$. Because of the renewal process, the relevant value of the decision function at the change time is then $g_{t_0 - 1} = 0$. In the conditioning (2.6), the behavior of the process before the change time is practically assumed nonrandom and fixed so that the change time $t_0$ is set to equal to one. In this sense, the change point itself at time $t_0$ initializes the CUSUM algorithm after a renewal of the SPRT, signifying that the past measurement history has been least favorable for quick detection. Hence, we can define the mean delay to detection as the following quantity:

$$\tau_d = \mathbb{E}_{\theta_1}(t_a), \quad (2.11)$$

which represents the most relevant situation as it says that the two delays (2.7) and (2.11) are equivalent. Typically, the conditioning (2.6) promotes, however, more complicated mathematical expressions if other change detection algorithms are to be analyzed; see e.g. Basseville and Nikiforov [9] for more detailed treatment and proofs. To sum up, the quantities (2.5) and (2.11) operate as standard design tools for CUSUM-type algorithms and they describe the expected behavior before and after the change time $t_0$, ideally, for known distributions. Below, a more complete analysis is given in Section 2.3.

In this work, a template-based detector is assumed to operate on the basis of fixed sample size, as discussed in Chapter 1. In practice, the maximum sample size is often limited due to memory constraints as well as computational power. On the other hand, the CUSUM algorithm incorporates the measurement history in a simplistic manner through (2.10), but because it possesses the discussed optimality properties in our setup, it is not an ad hoc procedure. Hence, assuming that both detectors handle the data according to its natural balance using (2.1) with equal false alarm specifications, the CUSUM-based detector outperforms the fixed-size detector when the change magnitude is close to background noise level. The result was reported in the early experimental study by Roberts [84] during the 1960s, which also shows that the history of application of sequential analysis does not go hand-in-hand with its theoretical evolution described previously. Since the CUSUM incorporates the measurement history, it is also possible that $g_{t_0 - 1} > 0$ after taking an arbitrarily large number of observations. From the point of view of quickest detection, the number of observations at $\theta_1$ required by the CUSUM algorithm to set an alarm may end up being smaller than that required by a fixed sample size detector with equal false alarm specification. Keeping in mind our main problem, low-contrast defect detection, the issues discussed provide good motivations to introduce the CUSUM algorithm for web imaging.
2.2 Relation to Off-Line Estimation of the Change

Up to now we have discussed only the sequential detection issue. For estimating the change time $t_0$ we shall consider an important relation of the sequential testing to the equivalent off-line case using the log-likelihood ratio. The relation stems from the so-called MLE algorithm inferred by Hinkley [30], where the abbreviation MLE stands for maximum likelihood estimate. In Page’s CUSUM algorithm the unknown change point $t_0$ can be estimated by maximum likelihood, and the key idea in another work by Hinkley [31] was that there is much redundant extra work involved in getting the MLE of the change time using conventional off-line means. Although the analog is simpler to use than the MLE algorithm itself, some amount of information is lost by not doing the MLE with all the available data. The subject is treated well by Basseville and Nikiforov [9] and we shall again follow their treatment partly.

Given a sample of $N$ independent observations, we assume that there exists a change point at an unknown time $t_0$ such that before $t_0$ the parameter $\theta$ is equal to $\theta_0$ and after the change it is equal to $\theta_1 \neq \theta_0$. The parameters $\theta_0$ and $\theta_1$ are assumed to be known. To obtain the maximum likelihood estimate $\hat{t}_0$ from a sample $y_1, \ldots, y_N$ of $N$ observations, we have to maximize the so-called log-likelihood function

$$\hat{t}_0 = \arg \max_{2 \leq k \leq N} \left[ \sum_{i=1}^{k-1} \ln p_{\theta_0}(y_i) + \sum_{i=k}^{N} \ln p_{\theta_1}(y_i) \right], \quad (2.12)$$

where we assume that at least one observation comes from each distribution; see Hinkley [30]. The previous equation can be rewritten as

$$\hat{t}_0 = \arg \max_{2 \leq k \leq N} \left[ \ln \prod_{i=k}^{N} \frac{p_{\theta_1}(y_i)}{p_{\theta_0}(y_i)} + \ln \prod_{i=1}^{N} p_{\theta_0}(y_i) \right], \quad (2.13)$$

where the rightmost term of the equation is a constant for the given sample. As the constant can be neglected, the estimate of the change time is therefore

$$\hat{t}_0 = \arg \max_{2 \leq k \leq N} \sum_{i=k}^{N} \ln \frac{p_{\theta_1}(y_i)}{p_{\theta_0}(y_i)} \quad (2.14)$$

with a close connection to the cumulative statistic (2.3). Fig. 2.5 shows the summation $\sum_{i=k}^{N} \ln \frac{p_{\theta_1}(y_i)}{p_{\theta_0}(y_i)}$ for $k = 2, \ldots, N$, where the MLE of $t_0$ is the x-axis value of the maximum value of the sum. Since the assumption that a change point exists is typically the result of testing of the hypotheses (2.4), the cumulative statistic (2.3) provides the basis of both change point estimation (2.14) and quickest detection (2.9). Next, we shall first clarify the connection and then unite these properties in order to introduce a highly usable sequential segmentation rule.
Figure 2.5: Typical behavior of the MLE algorithm’s cumulative sum $S_k^N$; $N = 50$. The MLE of the change time is the value of x-axis corresponding to the maximum value of the cumulative sum (solid line). The sketched threshold $h$ reveals the CUSUM algorithm’s connection with change point estimation.

In this work, we are only interested in estimating $t_0$ if the CUSUM algorithm implies its existence by triggering an alarm at an unknown time $t_a$. If we change the signs of the MLE statistic values in Fig. 2.5, simple visual inspection reveals that the obtained result has a bias, but it matches exactly the curve in the Fig. 2.4 after time $t_0$. Hence, $S_{t_0}^{t_a} = S_{t_0}^N - S_{t_{a+1}}^N \geq h$ corresponds to a decision rule where an alarm is given when the deviation of the maximum value of the sum is greater than the threshold $h$, which is illustrated in Fig. 2.5. Note that a renewal process of the CUSUM algorithm takes place at time $t_0 - 1$. Since this allows writing $g_{t_a} = S_{t_0}^{t_a}$, the estimate of the change time $t_0$ is simply equal to the last renewal time plus one, and it comes with a delay equivalent to the mean delay to detection (2.11). Because of this delay, we also prefer to use the term “on-line” rather than “real-time” for relatively fast detection. The interested reader is also advised to see the treatments of Hinkley [30, 31] for precision issues and the distribution of the estimation error, which we omit here.

As we recall from Fig. 2.2, the typical behavior of the log-likelihood ratio shows a
negative drift before a change, and a positive drift after a change of equal rate. This fact is important when a change has been detected and its duration is of interest. In practice, the monitored process may or may not return to the nominal state \( \theta_0 \), but if it does so, the drift of the CUSUM algorithm’s decision function turns to negative, and monitoring the difference of \( g_k \) with respect to its current maximum provides an estimate of the end of the change, exactly as shown in Fig. 2.5. For concreteness, let us assume that there exist two change points. The first change happens at an unknown time \( t_0 \) such that, before \( t_0 \), the parameter \( \theta \) is equal to \( \theta_0 \) and after the change it is equal to \( \theta_1 \neq \theta_0 \). However, after the second change point, at an unknown time \( t_1 \), our monitored statistical process returns to its nominal state; i.e., the parameter \( \theta \) after the change point \( t_1 \) is equal to \( \theta_0 \) so that \( t_1 \geq t_0 \). In segmentation, the goal is to partition data into its constituent parts. The segmentation rule is then, at each time instant, to monitor if our threshold \( h \) has been exceeded and, if so, to compare the difference with a threshold \( \gamma \) at the same time. For \( 0 \leq \gamma \leq h \), the rule can be formulated as follows:

\[
(t_{a_0}, t_{a_1}) = \min\{k\left| (g_k \geq h) \cap (m_k - g_k \geq \gamma) \right\},
\]

where

\[
m_k = \max_{i \leq k} \{g_i\}
\]

is the current maximum value after the last renewal of the CUSUM algorithm. The estimate of the change time \( t_0 \) is equal to last renewal time plus one, and it is obtained by monitoring \( g_k \geq h \) as usual. The estimate of the second change point \( t_1 \) is the index \( i \) corresponding to the maximum value of \( g_k \), which is obtained by monitoring the difference \( m_k - g_k \geq \gamma \). Now it becomes clear that this detection rule is nothing but a comparison between the CUSUM algorithm’s decision statistic \( g_k \) with an added adaptive threshold \( \gamma + g_k \). Because of the statistic \( g_k \), the second threshold \( \gamma \) is not only modified on-line, but it also keeps a complete memory of the entire information contained in the observations ideally between \( t_0 \) and \( t_1 \). The rule is illustrated in Fig. 2.6 for \( h = \gamma = 15 \). There, \( t_{a_0} = 23 \) and \( t_{a_1} = 43 \). To minimize the risk of undersegmentation, the threshold \( \gamma \) has to be chosen so that the mean time between alarms should not be too much less than the mean time between possible other successive jumps if such are to be expected. This applies to the selection of the threshold \( h \) too, but we shall address the issue of missed detections later. The estimated change times provide us with the boundaries of the segments, which plays a major role in the latter part of this work. The rule (2.15) is also successfully utilized in the case study presented in Publication V and for example by Kauppila et al. [41].
2.3 Assessing Run-Length

Typically, the frequency of alarms, which depends on the time interval between samples, plays a greater role than the probability of alarms per sample or observation. For instance, an algorithm with a mean time between false alarms less than the delay to detection is obviously of no interest. To introduce a particular function that contains the information in (2.5) and (2.11) related to the underlying statistical process, we define the following function of the parameter $\theta$:

$$\text{ARL}(\theta) = E_{\theta}(t_a).$$ (2.17)

Ever since this so-called ARL function was introduced by Aroian and Levene [2], it has become the main performance criterion for several change detection algorithms.
2.3. ASSESSING RUN-LENGTH

The ARL function (2.17) defines, at $\theta_0$, the mean time between false alarms, and at $\theta_1$, the mean delay to detection, which is also clear from the two equations (2.5) and (2.11). In the sequential framework, the usual method for comparing the performance of detectors on the basis of ARL function is to design the procedures so that their ARL values are the same at $\theta_0$ and then compare the ARL values over the range of parameters $\theta_1$ considered to be important.

The log-likelihood ratio (2.1) summarizes all of the relevant information supplied by the random variable $y_k$. However, Page [65] did not explicitly say that the increment $s_k$ of the CUSUM algorithm (2.10) should always be the log-likelihood ratio. This situation is connected with the availability of information about $\theta_1$, which must be accommodated in the decision rule. Often $\theta_0$ is known, but $\theta_1$ is not. Unfortunately, if the CUSUM algorithm is used in situations where the actual parameter values are unknown or different from the preassigned $\theta_0$ and $\theta_1$, its previously described optimal properties are lost. In effect, the efficiency of our detector decreases, which typically denotes both a higher rate of false alarms and longer delays to detection, or simply an imbalance between them. Starting with $g_0 = 0$ and using a reference value $\nu$, a practical equivalent to the equation (2.10) is obtained by writing

$$g_k = \max\{g_{k-1} + y_k - \nu, 0\}, \quad k = 1, 2, \ldots$$

(2.18)

where $y_k$ is the observation or, as often in quality monitoring, some useful sample statistic. As soon as $g_k$ reaches $h$, an alarm is triggered because the mean of $y_k$ might be larger than $\nu$; i.e., the reference value $\nu$ is designed to identify positive deviations of the mean of $y_k$ at $\theta_0$ so that $\nu \geq \mu_0$. In fact, there are three possible a priori choices that can be made for selecting $\nu$. Applicable in most cases of practical interest, the first consists of choosing the parameter $\nu$ as a minimum interesting level or magnitude of jump. In the second, we choose it in accordance with the most likely magnitude of jump based on earlier experience of the process behavior. The third choice for $\nu$ is a kind of worst-case value from the point of view of the cost of a non-detected change versus the false alarm rate. From these three standpoints, proper selection of $\nu$ can lead to an almost optimal algorithm, if compared with the performance achievable using the log-likelihood ratio. For example, from the point of view of minimum interesting magnitude of change, the limit case is $\nu = \mu_0$. This situation occurs when all possible positive jumps are to be detected, whatever their magnitude. Switching the sign of data is sufficient too, if we intend to detect negative deviations. Note also that an obvious way to modify the CUSUM algorithm in the case of an unknown post-change parameter $\theta_1$ is simply to estimate it. This

\(^2\)Previously, the sufficient statistic (2.1) for the Gaussian distributions with a constant variance $\sigma$ was shown to be $s_k = \frac{\mu_1 - \mu_0}{\sigma^2} (y_k - \frac{\mu_1 + \mu_0}{2})$; $\mu_1 \geq \mu_0$. In (2.18), the multiplicative term $\frac{\mu_1 - \mu_0}{\sigma^2}$ is canceled since it can be incorporated in the threshold $h$. As a user-defined parameter, $\nu$ substitutes the term $\frac{\mu_1 + \mu_0}{2}$ so that $\nu \geq \mu_0$ and the property (2.2) is not lost.
leads to the so-called generalized likelihood ratio test, which is “self-tuning”, but however does not have such a convenient recursive form or low memory and computation requirements as the CUSUM algorithm; see e.g. Basseville and Nikiforov [9]. Although we basically discuss the computation of the ARL function next, we consider fixed values of the CUSUM parameters $h$ and $\nu$ from now on. We also omit the parameters $\theta_0$ and $\theta_1$ occasionally for simplicity since $h$ and $\nu$ can be easily standardized, as we shall next discuss. Imaging-driven strategies with nonlinear aspects for choosing them are discussed at the end of Chapter 3.

In reality, the simple binary hypothesis testing problem (2.4) is a so-called composite hypothesis testing problem as at least the $\theta_1$ distribution is difficult to specify completely. If the increment of the CUSUM algorithm cannot be computed using the log-likelihood ratio, it is of key interest to compute the mean delays (2.5) and (2.11) for other parameter values in terms of robustness, which we do later in this section for (2.17). For analyzing the CUSUM algorithm, however, there is quite an extensive literature on different techniques available. Also Page [66], who introduced the CUSUM algorithm, discussed the computation of the ARL. But to describe some of the variety of ideas in the earlier works, Johnson [37], for example, assessed the suitability of the CUSUM parameters based on error probabilities and Barnard [5] proposed the so-called V-mask that received wide popularity and is even today used in some software packages (see e.g. JMP®). Furthermore, Goal and Wu [24], for example, proposed solving a system of linear algebraic equations in order to compute the ARL function (2.17), for which also Siegmund [89] provided a sophisticated approximation based on the properties of SPRT. Certain “rules of thumb” are also discussed by Montgomery [57]. The list could be continued, but nowadays many of these techniques can be considered carry-overs of the pre-computer era. These often had limited accuracy or, as in the case Johnson’s paper [37], even lacked meaningful interpretation of actual performance. We shall now review some recent insights by discussing the so-called run-length probability mass distribution for which we then provide an accurate approximation according to Luceño and Puig-Pey [56].

Let us assume that the only underlying hypothesis is that the values used to build the CUSUM statistic $g_k$ are i.i.d. random variables. Let $0 \leq z < h$ and, for clarity, let us assume that $g_0 = z$ is the current value of the CUSUM statistic in (2.9). The probability that the threshold $h$ is reached after exactly $n$ steps for the first time can be stated as

$$P(n|z) = P(t_a = n).$$

(2.19)

If $F$ is the CDF of $y_k$, the probability that the threshold $h$ is reached after the first observation is

$$P(1|z) = 1 - F(h + \nu - z),$$

(2.20)

since it is clear that $P(g_1 \leq h) = P(z + y_1 - \nu \leq h) = F(h + \nu - z)$. Following from
the law of total probability (see e.g. [9, p. 75]), the probability that the threshold \( h \) is reached for the first time in exactly \( n \) steps is

\[
P(n|z) = \int_{0}^{h} P(n-1|x) f(x+\nu-z) \, dx + P(n-1|0) F(\nu-z), \quad n = 2, 3, \ldots ,
\]

where \( f \) is the PDF of \( y_k \). Now, provided that \( z = 0 \), the performance of the CUSUM algorithm can be assessed in terms of the whole run-length probability mass distribution. For discrete distributions, the PDF is often called a probability mass function. We shall use the abbreviation “PMF” for run-length probability mass distribution and similarly “CMF” for the cumulated values of PMF. Later, using the same notation, we mostly consider the ARL simply a quantity for the given distributions \( f \) and \( F \), which we assume known a priori. Subsequently, the ARL is therefore easily evaluated along with the standard deviation.

Direct numerical solution of the integral (2.21) is prone to numerical instability. However, whenever we want to evaluate numerically an integral with recursion, the Gaussian quadrature can be utilized for an approximate solution; see e.g. Kincaid and Cheney [44]. Although very small errors can occur, the accuracy of the approximation is much dependent on the accuracy of the Gaussian quadrature, which is governed by the number of so-called Gauss-Legendre points. With the same notation, after approximating the whole PMF, the expectation of discrete random variable \( P(t_a = n) \) defines the ARL:

\[
\text{ARL} = \sum_{n=1}^{\infty} nP(t_a = n).
\]

The standard deviation of the run-length, denoted by SDRL, can be given as

\[
\text{SDRL} = \sqrt{\sum_{n=1}^{\infty} (n - \text{ARL})^2 P(t_a = n)},
\]

which equates the second-order moment of the centered run-length. Concerning approximation (2.21), the article of Luceño and Puig-Pey [56] provides a comprehensive reference. The proposed algorithmic solution is particularly convenient because the computation of the ARL (2.22) proceeds from \( n = 1 \) onwards and may be stopped after a sufficient amount, like 99.9 %, of the probability mass has been cumulated. With minor manipulations, the method of Goal and Wu [24] can also be used to save computation time at the expense of introducing small errors when the ARL is large. This is due to the fact that the run-length distribution at the nominal “noise-only” state is often close to a geometric distribution like that reported, for example, in the early studies by Ewan and Kemp [18] and Brook and Evans [12]. However, the most
accurate values are provided by using recursion, which is why we never switch to the geometric approximation of Goal and Wu [24]. A few run-length distributions are illustrated in Fig. 2.7.

![Figure 2.7: Some examples of run-length distributions computed using the method proposed by Luceño and Puig-Pey [56]. More details are given in Table 2.1.](image)

For a Gaussian distribution with unit variance Table 2.1 gives the ARL and SDRL along with the median and some quantiles of the run-length distribution for a few selected values of the standardized parameters $\mu - \nu$ and $h$. The word “quantile” refers to the percentage of probability mass that we have cumulated onwards from $n = 1$ up to the indicated run-length and the CMF incorporates this information. The x-axis of Fig. 2.7 and the numbers in Table 2.1 should be interpreted as the factors of the sampling interval. If the sampling interval were equal to one hour, all the run-lengths would be in hours. Compared with the time when run-length analysis was first developed, the presented values are practically error-free and describe the performance in detail, denoting that analysis of the CUSUM algorithm has grown to maturity. It is worth noting that the SDRL is often almost as large as the ARL. This means that even when the ARL is large, a relatively large percent of alarms occur shortly after the CUSUM algorithm has been reset to zero. The same result is also observed if one looks at the lower quantiles in Table 2.1. For example, for $h = 1.5$, 1% of the alarms would occur not later than 15 samplings, though the ARL is well over one thousand. Consequently, at the state $\theta_0$, this would mean that the probability of early false alarm is high. The close relation of ARL and SDRL reveals also the inherent difficulty in sequential experimental design: a substantial number of experiments may be required for the sake of accuracy in the performance indexes.
2.3. ASSESSING RUN-LENGTH

Table 2.1: ARL, SDRL, Median, and 1 %, 5 %, and 25 % lower and upper quantiles of the run-length distribution for a CUSUM algorithm with standardized parameters $\mu - \nu = -1.75$ and $h$, assuming the underlying distribution to be Gaussian with unit variance.

<table>
<thead>
<tr>
<th>$h$</th>
<th>ARL</th>
<th>SDRL</th>
<th>1%</th>
<th>5%</th>
<th>25%</th>
<th>Median</th>
<th>75%</th>
<th>95%</th>
<th>99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.50</td>
<td>1418.5</td>
<td>1417.8</td>
<td>15</td>
<td>73</td>
<td>408</td>
<td>983</td>
<td>1966</td>
<td>4248</td>
<td>6530</td>
</tr>
<tr>
<td>1.25</td>
<td>647.6</td>
<td>647.0</td>
<td>7</td>
<td>34</td>
<td>187</td>
<td>449</td>
<td>898</td>
<td>1939</td>
<td>2980</td>
</tr>
<tr>
<td>1.00</td>
<td>308.3</td>
<td>307.8</td>
<td>4</td>
<td>16</td>
<td>89</td>
<td>214</td>
<td>427</td>
<td>923</td>
<td>1418</td>
</tr>
</tbody>
</table>

In order to approach the issue of missed detections in the case of a change of finite duration, some ARL, SDRL, upper, and lower quantiles for a few standardized parameter values are given in Table 2.2. As we will recall from the discussion above, the results in Table 2.1 could be interpreted as showing that the CUSUM algorithm is prone to early false alarms. Now, if we look at the run-lengths corresponding to different quantiles in Table 2.2, the run-lengths become shorter the closer the mean $\mu$ gets to our minimum interesting jump $\nu$. The notation $\mu - \nu$ simply denotes that there is drift with the indicated mean in the increment of the CUSUM algorithm. For instance, for $\mu - \nu = 0.5$, the run-length that corresponds to the 99 % quantile is equal to 12. Obviously, this is an indication of quick detection, since 99 % of detection would occur not later than 12 samplings. Though the CUSUM algorithm is fully sequential, it is relatively easy to determine its parameters so that they almost surely allow detection of events at $\theta_1$ with finite duration. The probability of early detection is high too; i.e., for $\mu - \nu = 0.5$, 50 % of detections would occur not later than three samplings. The same behavior can be observed, but at a coarser level, if one looks at the ARL’s.

Table 2.2: ARL, SDRL, Median, and 1 %, 5 %, and 25 % lower and upper quantiles of the run-length distribution for a CUSUM algorithm with standardized parameters $\mu - \nu$ and $h = 1.5$, assuming the underlying distribution to be Gaussian with unit variance.

<table>
<thead>
<tr>
<th>$\mu - \nu$</th>
<th>ARL</th>
<th>SDRL</th>
<th>1%</th>
<th>5%</th>
<th>25%</th>
<th>Median</th>
<th>75%</th>
<th>95%</th>
<th>99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.5</td>
<td>549.7</td>
<td>548.9</td>
<td>6</td>
<td>29</td>
<td>159</td>
<td>381</td>
<td>762</td>
<td>1645</td>
<td>2529</td>
</tr>
<tr>
<td>-0.5</td>
<td>21.1</td>
<td>19.9</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td>15</td>
<td>29</td>
<td>61</td>
<td>93</td>
</tr>
<tr>
<td>0.5</td>
<td>3.5</td>
<td>2.4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>8</td>
<td>12</td>
</tr>
</tbody>
</table>

The presented results are valid for i.i.d. samples, but sequential change detection involving dependent observations is also an important topic, particularly in the engineering literature on fault detection and diagnosis; see e.g. Basseville and Niki-
forov [9]. Although it is relatively easy to extend the CUSUM algorithm to non-independent observations, for example with a recursive Kalman filter, see also Lai [51], Hanlon and Maybeck [28], the assumption that observations or samples are independent is typically satisfied provided that our sampling interval is large enough or, otherwise, some appropriate prewhitening is applied. Certainly, the Gaussian distribution in our examples can also be replaced by a distribution corresponding to the distribution of the increment being used, but as a word of caution the ARL may change substantially depending on the shape and parameters of the underlying distribution; see Luceño and Puig-Pey [56]. In practice, even relatively short-term drifts around the mean of a signal may trigger alarm earlier than anticipated. As already mentioned, a basic engineering approach is to remove such unwanted variations of a signal with time- or frequency-domain filtering, for which there are many texts available as for example mentioned in Chapter 1. Next, intrinsic changes and abnormalities are selected so that they are not directly observed, which represents the key difficulty and justification for favoring DP and continuous statistical control using the CUSUM algorithm. However, the outcome of the analysis here and in the following Chapter 3 has depended at least as much upon the conceptual framework employed to analyze the CCD line frames as it has upon the image data itself.
Chapter 3

Line-Scan Defect Monitoring

Electro-optical inspection systems with digital image processing tools are widely used in the process industry for inspecting goods and materials. At a paper mill, these systems can help the papermaker to enhance mill productivity. The real image material we consider in this chapter has been collected from a controlled laboratory setup by experts using the discussed real-time web inspection system. As described in Chapter 1, the inspection system is based on high definition CCD line-scan cameras. The custom-made camera technology and advanced digital analysis reveal the fine details of defects, which makes it possible to reduce production disturbances. The system detects holes and other surface defects which, as discussed by Landry [52], have been recognized to cause significant quality and productivity problems in the paper industry. Since the whole area of the fast-moving paper web needs to be inspected in real-time, web inspection systems replace human inspectors in order to provide constant feed-back on paper properties.

Detection of surface anomalies is critical to the runnability of the paper machine and the condition of production equipment. There are many types of paper surface flaws and, in general, large defects are more critical than small ones. Arising, for example, in the coating of a magazine paper or in the wet end of a paper machine, many defect types are common to various paper grades, whereas some are more specific to certain grades only. For those having a repetitive form of geometry, a well-matching template yields easily detectable, high correlation each time such a defect appears. It is obviously relatively easy to detect also a defect with a large contrast range, for example, by thresholding. Paper users are also concerned about defects whose diameter is only a fraction of a millimeter. If a low-contrast defect extends over several frames in the machine direction but occupies only a few pixels per line frame, the line frames accessible in the short-term frame buffer within the limits of available processing time may not provide sufficient evidence for forming a definite decision. As we shall next discuss, sequential analysis can be expected to provide
most benefit for web imaging if the evidence can be appropriately strengthened by incorporating the measurement history. The current quality requirements have resulted in a situation where distinguishing the fine details more precisely than before is of interest in delivering products of correct quality to customers.

It is self-evident that paper mills want to produce paper that meets specifications set by their customers. To extend the arsenal of detection techniques for web inspection, see e.g. Rauhamaa and Reinius [77], Rauhamaa [74, 75, 76], the “theoretical” method in Publication IV provides some technical computing and simulations intended to give a reasonable approximation and understanding regarding the proposed detector. In the latter part of this chapter, those results are also shown to accurately portray the outcome of using the DP and the CUSUM algorithm in real-life. The reader interested of noise and calibration-related aspects may also find the treatment of Haley and Kondepudy [27] useful and in line with this work. The various noise sources corrupting digital pixel values of the CCD lead in the normal distribution.

3.1 Selecting Most Likely Pixels

In classic speech-to-text recognition, the acoustic signal is treated as the observed sequence of events, and a string of text is considered to be the “hidden cause” describing the content of the acoustic signal; see e.g. Rabiner [72], Silverman and Morgan [90], Scharf and Elliott [86]. Whereas in this case the Viterbi algorithm, see e.g. [22, 103, 102], finds the most likely string of text based on maximum likelihood if it is given the acoustic signal, we can also use it to find the most likely pixels of interest if it is given a sequence of images. The basic operation is sketched in Fig. 3.1. These techniques are used also, for example, in infra-red imaging to hypothesize the kinematics of an airplane with a simple model and return a sequence of identified flight patterns simultaneously with slightly postponed detection; see e.g. Barniv [6], Barniv and Kella [7], and Arnold et al. [1]. As the cited literature shows, the DP formalism allows differing techniques to be utilized with domain knowledge arising in other change detection applications. In our change detection problem, the connection with the aforementioned DP literature can be stated as follows. To cope with the fast movement of the paper web, line frames describing the surface profile of the paper web are captured constantly with high frequency. Thus, the corresponding pixels describing an observed defect arrive in a sequence. Hence, in the spirit of ideal DP, computing the most likely sequence of pixels up to a certain time \( k \) must only depend on the observed pixels at time \( k \), and the most likely sequence at time \( k - 1 \). If a procedure scores every pixel, our problem is to find the highest “scoring” alignment of pixels.

In spite of the fact that low-contrast defects have more or less noisy and distorted
3.1. SELECTING MOST LIKELY PIXELS

Figure 3.1: Evolution of the Viterbi algorithm over seven time instants. The dots represent the known states in the graph, called a trellis. At each time, the best score for each state at the next time instant is determined. Based on our measurements, the arrows form sequences of states so that sequences with a low score do not survive to the next time instant and are deleted. By the time $k+6$, the best sequence of states from $k$ to time $k+3$ has been determined unambiguously. For further reference, see e.g. Moon and Stirling [59].

Visual features, the property that the score of a pixel can be expressed as a sum of incremental scores between pixels adjacent in time provides a way to find a sequence of pixels that are likely to be at $\theta_1$. Whereas post-processing and analysis with a large number of parameters generally requires a large amount of memory and computation power, models useful for on-line change detection are usually much smaller than physical models and models for recognition-oriented tasks. For this work, particularly the DP formalism of Tonissen and Evans [94], presented in the context of remote sensing and so-called track-before-detect techniques, is useful: each pixel at each frame is assumed to have a region of so-called candidate pixels at the previous frame. The minimum search region size should be greater than one in order to calculate a score of at least one pixel, and in order to center the window, it should be of odd size. For two-dimensional image and one-dimensional line frame
data, the minimum region size is basically set at $3 \times 3$ and $1 \times 3$, respectively. This is illustrated below in Fig. 3.2, but the heart of DP, as used in [94], is that the pixel-wise scoring sums at each frame depend only on the maximum of the scoring sums of these candidate pixels. A scoring sum $I_n$ of a pixel $x_n$ at the $n^{th}$ frame is then

$$I_n = \max\{I_{n-1}\} + x_n,$$

(3.1)

where $I_0 = 0$. The scoring, at time $k$, over a group of $K$ frames is simply a summation (3.1) of intensity values up to the $k + K - 1^{th}$ frame with $I_{k-1} = I_0 = 0$. Commonly, the estimate of a target state is additionally related to the observations through a known model, which we omit here to limit computational complexity. It is worth noting that equation (3.1) is close to a standard integration and, based on the largest score sum in the case of a positive change in the intensity, it allows efficient search for pixels likely to be at $\theta_1$ without the need to know the actual change parameters or dynamics.¹ Statistical properties of the DP scores are further investigated by Johnston and Krishnamurthy [38].

In the context of remote sensing, apparently the most accepted approach to track localization is based on a three-dimensional matched filter proposed by Reed et al. [78, 79, 80]; see also Bar-Shalom and Li [4]. Obviously, a template-based approach would be valid here too, but it is often even difficult to describe visual features of low-contrast defects using words. Fundamentally, the scoring (3.1) is of a recursive type, and we also stress that the essential difference between a recursion and a proper DP algorithm is that the DP saves its intermediate scores, whereas the recursion does not. The arguments in favor of and against the approaches are as follows. On the one hand, for a large $K$, making a full search of all possible alignments for scoring is time-consuming, in particular with the recursion. On the other hand, as the DP algorithms are often expressed via recurrence relations between tables holding intermediate results, their proper implementation may be complicated. However, a potential speed gain arises from re-use of the intermediate scores.

¹In Chapter 2, the sufficient statistic (2.1) for the Gaussian distributions with means $\mu_0$ and $\mu_1$ and a constant variance $\sigma$ was shown to be $s_k = \frac{\mu_1 - \mu_0}{\sigma^2}(y_k - \frac{\mu_0 + \mu_1}{2})$; $\mu_1 \geq \mu_0$. As the absolute values of $s_k$ are unimportant, the multiplicative term $\frac{\mu_1 - \mu_0}{\sigma^2}$ and subtraction of the constant $\frac{\mu_0 + \mu_1}{2}$ can both be canceled. These terms do not modify alignments of best pixels after the maximization of the DP recurrences (3.1). Hence, the DP incorporates a built-in MLE relation and, if a threshold is introduced, also a built-in detector of fixed sample size; see Publication IV.
3.2 Detection of Change with Unknown Parameters

A line-scan CCD array produces a two-dimensional image by exploiting relative motion between the scene and the sensor. In our following examples, vertical extent of the images is basically obtained by sequentially gathering rows of horizontally oriented line frames. Though very small scan times allow practically freezing the motion, the specialized frame buffer can provide only short-term storage requiring rapid processing with only a limited number of frames available at a time. In practice, it will usually happen that \( \theta_0 \) is known rather than \( \theta_1 \). Therefore, we assume that this is regularly so. Our goal is to effectively separate the pixels that are similar or brighter with respect to intensity compared with the minimum interesting magnitude of jump \( \nu \). We assume that the observed pixels at \( \theta_1 \) are in a sequence and that the sequence corresponds to their intensity values, time, and place in the cross-direction. The equation (3.1) is naturally based on the assumption that the place of a pixel at \( \theta_1 \) at time \( k - 1 \) is unknown but within some given bounds at time \( k \); i.e., the candidate region of the DP procedure should be selected accordingly. Therefore, computation of the most likely sequence up to a certain point \( k \) depends ideally only on the most likely sequence at point \( k - 1 \), i.e., the sequence of pixels with the highest score (3.1). Now consider that each pixel at the \( k - 1 \)th frame has a CUSUM statistic \( g_{k-1} \). As a kind of worst-case scenario, assume also that the change occupies a single pixel per frame. For quickest detection, the definition (2.9) denotes choosing a pixel of the \( k \)th frame with the largest score sum (3.1) within the candidate region for each \( g_{k-1} \). With the aid of dynamic scoring illustrated in Fig. 3.2 (a), a CUSUM statistic \( g_k \) of the pixel \( y_k \) at the \( k \)th frame is then

\[
g_k = \max\{0, \max\{g_{k-1}\} + y_k - \nu\},
\]

where \( \max\{g_{k-1}\} \) denotes the maximum of CUSUM candidates; see Fig. 3.2 (b). Since the change point is unknown and may be masked by noise, other pixels that have smaller DP score sums restart new CUSUM statistics (3.2) so that \( \max\{g_{k-1}\} = g_0 = 0 \). Whenever a new frame is received, the scores (3.1) are simply revised for (3.2). For obvious reasons, the algorithm is called the DP-CUSUM; see Publication IV.

Before introducing more practical examples, the leftmost image in Fig. 3.3 (a) shows our simulated test pattern embedded in i.i.d. additive \( N(0, 1) \) noise. Details are not easily visible to the naked eye, and for clarity the occurrence of the embedded pattern

\[\text{2}\]The DP scores (3.1) remain invariant if they are computed in reversed order for the same frames. In this sense, the candidate region is here flipped “backwards” to get the highest of CUSUM statistics at time \( k - 1 \); see Fig. 3.2 (b).
is illustrated on the right in Fig. 3.3 (a). The amplitude $A$ of the signal is equal to 1.85. To mimic line-scan memory-constrained processing, the image is processed row-wise from top to down and the result is presented in Fig. 3.3 (b). The cumulated pixels before a detection are in gray, and the last pixel causing the detection is in black. After a detection, the whole algorithm, comprising the decision statistic (3.2) and the decision rule (2.9), is restarted, i.e., each of the candidate statistics $g_{k-1}$ is set to zero. The given parameters yield a mean false alarm interval of over one thousand frames.

Note the short distances between black dots in the upper part of Fig. 3.3 (b). As discussed in Chapter 1, the probability of early detection is high, but not all the true pixels at $\theta_1$ are recovered, however. The reason for this is that, though the DP recovers reasonably good alignments of pixels, the pattern has loosely defined, partly unknown dynamics. In badly noise-obscured cases like the one presented, high noise masks many of the pixels actually at $\theta_1$. If the temporal evolution of the embedded pattern were modeled as a sequence of more strictly defined states at a succession of discrete instances, the possible pixels that the pattern is able and likely to occupy—while being a stochastic process—could be limited at each line frame. From this point of view, performance of the DP procedure is discussed comprehensively by Tonissen and Evans [94]; see also Johnston and Krishnamurthy [38]. In our case, improperly recovered pixels at the detection stage can often be recognized and corrected at the recognition stage. Once it is established that a change at $\theta_1$ is present, one could here deduce the instantaneous frequency of the
3.2. DETECTION OF CHANGE WITH UNKNOWN PARAMETERS

Figure 3.3: (a) A noisy and the actual pattern, $A = 1.85$. (b) Detected pixels using the DP-CUSUM algorithm for $1 \times 9$ candidate region, $K = 10, \nu = 1.75, h = 7$; see Publication IV.

shown frequency modulated sinusoid by utilizing the pixels picked up by the detector for the task; see e.g. the minimum variance estimation in [42].

First utilized in Publication III and then in Publication IV, perhaps the most important result in terms of computational complexity is addressed next. The following approximations are based on a study by Arnold et al. [1]. If we consider the parameters $K$ and $j$, the penalty associated with a Viterbi-type DP algorithm is in approximately $j^2(K-1)$ increasing computation; i.e., the maximization procedure for $I_{k+K-1}$ is performed $K-1$ times with no more than $j$ allowable candidates for each of the $j$ pixels. This means that the optimization over $K$ frames is decomposed into smaller problems at each frame and the intermediate scores are stored in memory. With an exhaustive search, for which a Viterbi-type DP algorithm substitutes efficiently, the complexity of some $j^{K-1}$ maximizations is significantly higher. While trying to reduce complexity, one of our key innovations was to utilize “data about data” referring to local maxima of each line frame; see Fig. 3.4. While in generic imaging an interesting change can be a combination of step edges, ridges, ramps, etc., it is worth emphasizing that the essential information for hypothesis testing is often contained in extrema, i.e., in the largest or smallest measurements. The subject of extreme value modeling can be traced back to the 1940s. The work of Rice [81] is still relevant and a more recent treatment is presented by, for example, by Coles [13]; see also Oakley [63]. In effect, local maxima simplify the content of a line frame efficiently and, as a result, these pixels maximize the DP scores (3.1). For example, since only 33% of data were local maxima in the Fig. 3.3, the complexity
of the exhaustive search is reduced to about \((0.33j)^{K-1}\), and by analogy the DP is reduced to \((0.33j)^2(K-1)\) without practically changing the results. With the real data in our later example (Fig. 3.8 (a)), only 18% of data were local maxima, which emphasizes the redundancy of data in the web imaging. For reproducible research, the DP-CUSUM detector utilized here an exhaustive search as such a search is feasible via local maxima: frame-by-frame, the scores (3.1) were maximized over the \(K\) frames prior to the update of CUSUM statistics (3.2). The conventional nearest neighbor –logic was adopted if no extrema were within the DP candidate region. By holding the intermediate alignments of pixels and revising the DP scores, say, every fifth frame for \(K = 10\) makes also a huge difference to complexity in this type of scoring without practically affecting the detection if compared with an exhaustive search that is called for repeatedly each time after a new frame is received. A deeper insight into efficient scoring is given in Section 3.3.

![Figure 3.4: Local maxima “×” and minima “O” of a vector.](image)

The usual method for comparing the performance of detectors is to design the procedures so that their false alarm rates are equal and then compare the detection performance over the range of parameters \(\theta\) considered to be important. While not using fixed templates, the relevant comparison is to use the DP scores for fixed sample size detection as was originally proposed by Tonissen and Evans [94]. The results in Publication IV show that explicit degradation of detection is often encountered with too small sample sizes, which is bound to happen in high noise. In the electro-optical quality management of paper production lines, see the web inspection system in Fig. 3.5, large fluctuations occur in the recorded intensity levels when the papermaking process is in-control. More or less randomly occurring low-contrast defects are, additionally, basically always assimilated into the background, which further hinders their detection if weak solutions are utilized. In many cases, allowing the DP-CUSUM statistics to cumulate over a longer period of time provides a discrimination of high degree. Next, we shall illustrate a generic approach for online segmentation. Then a solution involving more specific planning is presented and analyzed in more detail. The treatments are, however, somewhat cursory in order to introduce the concepts clearly without case-specific optimization.
3.3. **EXAMPLES OF DEFECT SEGMENTATION**

The main desired properties of an on-line segmentation algorithm are few false alarms and missed detections, and low detection delay. Using our earlier illustration in Fig. 3.2 (b), Fig. 3.6 adds a simplistic red-shaded area which is a hypothetical defect of interest that the picture elements of the CCD quantize. The red area spans two pixels at the third frame, which should both contribute to quickest detection, preferably using an approach exhibiting low complexity. The problems of detection and segmentation coexist and, in our first case, estimating pairs of change points brings the key information. Extending the previous formulations, the estimation is also conducted frame-by-frame describing the width of each defect in the cross-direction of the web. Since they deviate substantially from the majority of easily detectable flaws, the detection of scratches and wrinkles becomes difficult if the
flaw’s width is less than a tenth of a millimeter, contrast is very low, and its length is only a few decimeters; see Rauhamaa and Reinius [77]. Hence, the cross-direction segmentation of the web, which was discussed first, is omitted in the second case and alternative discriminatory measures are presented. It is in the nature of things that benefits do not always come without penalties. For the smallest of defects, utilizing a segmentation closer to the generic image processing becomes more or less useless as it virtually only increases the risk of oversegmentation that tends to lead to false alarms. For this reason the second case shows an example of tailored DP design and more contextual parametrization of the CUSUM.

\[
g_k = \max\{0, \max\{g_{k-1}\} + y_k - \nu\} + g_k^{\text{cross}} - y_k + \nu, \quad (3.3)
\]

where \(\max\{g_{k-1}\}\) denotes the maximum of CUSUM candidates at time \(k - 1\), as earlier. The cross-direction sum, \(g_k^{\text{cross}} - y_k + \nu\), includes subtraction and addition in
Figure 3.7: Monitoring renewals and maxima of the statistic (2.10) allows cost-efficient estimation of the width of a defect frame-wise, i.e., in the cross-direction of the web; see discussion in Chapter 2. Some plausible points that are not detected when the frame is processed forwards are more likely detected when the pixels are processed in reversed order; see also Basseville and Nikiforov [9].

order not to incorporate the intensity of pixel $y_k$ and $\nu$ twice; for $g^{\text{cross}}_k = 0$ the rule (3.2) applies. A fragment of high resolution line-scan image is shown in Fig. 3.8 (a). The snapshot includes five defects on the surface of a standard copying paper. As we can see from the color-coded results in Fig. 3.8 (b), using the segmentation rule (2.15) and (3.3) gives relevant results in all of these less simple situations where transitions between the background and defects tend to be slow rather than abrupt. Fig. 3.8 (c) shows the cumulated decision statistics; the colors of the upper solid curves match the segmented defects in Fig. 3.8 (b), whereas the dotted curves at the bottom illustrate the background variation. To facilitate the detection of short-term changes yet have a relatively high or, as in this case, low intensity, the dash-dotted piecewise linear red curve operates as our decision boundary, similar to those boundaries used to truncate clinical trials; see Siegmund [89]. Consequently, if a smaller low-contrast defect is suddenly immersed in a strongly fluctuating background, more sophisticated techniques become necessary.
Figure 3.8: (a) A fragment of high resolution line-scan image with a variety of defects. (b) shows the detected defects using the DP-CUSUM algorithm with segmentation in the cross-direction of the web; the DP parameters were the same as in Fig. 3.3. (c) shows the cumulated statistics (solid lines) corresponding to the color-coded segments in (b). Note the piecewise linear decision boundary: the threshold $h$ takes values in function of the integrated intensities in (c). The other CUSUM parameters, $\nu$ and $\gamma$, were selected suitably to help visualization without any optimization.
3.3. EXAMPLES OF DEFECT SEGMENTATION

Case 2: The selected candidate region and $K$ frames determine the number of combinations of pixels used for building the DP scores (3.1). Because the score recurrence incorporates this combinatory aspect, combining DP and sequential detection more tightly than discussed above depends essentially on limiting the greatest number of DP scores (3.1) whenever a new frame is used in the scoring. Note that this may also be thought of as a problem-specific subject. In the case of score recurrence, the approach we prefer to use is simply to hold the sequences for each end-frame pixel $x_{k+K-1}$ (see for example Fig. 3.2 (b)) that have the maximum DP scores (3.1) from among those scores that have originated from the same pixel $K$ frames earlier. Assuming that the maxima over multiple frames contain all the relevant detail about the spatial relations at $\theta_1$, the overall number of DP scores (3.1) for each end-frame pixel, or maximum to be precise, $x_{k+K-1}$ is therefore merely connected to the number of maxima within the limits of viable dynamics at the $k^{th}$ frame. Correspondingly, this makes it easy to build CUSUM statistics for each one of the DP scores using just (2.18). Since all of the CUSUM statistics (2.18) for each end-frame pixel $x_{k+K-1}$ comprise at least a partly different history of pixels, it is worth noting that they also include the statistic equivalent to the maximum CUSUM statistic (3.2) (as would for example obviously result in Fig. 3.2 (b)) and therefore the difference between the equations is only conceptual. The holding of highly correlating histories of pixels is certainly not necessary for speed gain. A pragmatic illustration is given in Fig. 3.9. Here, the risk of recovering suboptimal pixels is small because the number of DP scores inherently connected with CUSUM statistics has been kept large in the manner indicated. As far as we believe, this type of tailored DP has allowed us to come close to the limits of what seems to be possible to detect reliably even by the naked eye, including also low-contrast cases where quantized intensity values both at $\theta_0$ and $\theta_1$ occasionally lead to equal DP scores; this is also the situation in the illustration. All the pixel values are equalized in the range $[0,255]$ and centered around 128 and, for easy demonstration, the reference value $\nu = 120$ used was simply doubled each time the least-squares fit of a first-degree polynomial to the coordinates of the DP scores resulted in a mean absolute error greater than three. With inappropriately smaller values of $\nu$ in terms of the prevailing background gray-level mean $\mu$, the same strategy applies well to safe elimination of CUSUM statistics that might trigger alarms while occurring frame after frame nearly collinearly along the maximum viable dynamics. In this setup, such statistics may often be regarded as oblique-angled artifacts with the coordinates constantly lying in the same marginal part of the candidate region. Owing therefore to its privileged role within our parametrization framework, the weighting of the CUSUM parameter $\nu$ as suggested here is by no means restricted solely to the coordinate properties. On the contrary, the choice of a particular weighting should be driven by the defect type, its shape, and area as in geometry-based template
CHAPTER 3. LINE-SCAN DEFECT MONITORING

matching. Assuming for example that the state of a pixel is conditioned according to the states of the nearby pixels, the iterative segmentation with Bayesian basis of Besag [10] is a computationally efficient method for utilizing the intensity level of a pixel and information from the neighborhood simultaneously. For stripped-down methodology, see also Vihonen et al. [98, 99, 100, 101].

Figure 3.9: (a) A zoomed fragment of high resolution line-scan image with a low-contrast wrinkle. For $K = 20$, $1 \times 17$ candidate region and $\gamma = 40$, (b) shows the recovered pixels most likely at $\theta_1$ when using the DP-CUSUM with our Viterbi-type evolution. Just as in Fig. 3.1, note how the purple-coded pixels describe a dominating set of pixels from which all the others with different color codes originate. They, however, have not “survived” due to the segmentation rule (2.15), which takes quick action in consequence of the discussed coordinate-based doubling of $\nu$ otherwise set to 120. Although not shown, the maximum of the CUSUM statistics is in this case easily distinguishable as it would naturally coincide with the purple-coded pixels. For this, the acceptance zone for $H_1$ can be selected similarly as shown in Fig. 3.8 (c).

The discussed methods have been designed for small defects, but on-line segmentation, in its computational simplicity, is intended to give only sufficient initial insight into change characteristics; see e.g. Basseville and Nikiforov [9]. The characteristics should be examined thoroughly at subsequent processing stages, such as recognition. For example in Fig. 3.8 (b), a more global identification to discriminate between characteristics inside the purple-coded segment is evidently necessary. Some tools for the web imaging system are described, for example, by Kunttu [49], where the essential idea is to break down the detected event into its various components and assess each component separately in favor of shape description and segmentation. Other useful techniques are described by, for example, Kassam and Poor [40], Parizeau and Plamondon [67], Kittler et al. [45], Del Bimbo [16], Costa and
3.3. EXAMPLES OF DEFECT SEGMENTATION

Cesar [14]. Subject to more or less natural or process-related factors, fluctuation of the imaged paper surface forms interference which is basically unknown a priori and difficult to quantify accurately. The latter case shows that it is, however, relatively easy to formulate problems with flexible sequential solutions so that good results are obtained with simple computer calculations. On-chip programmers have always regarded simple calculations advantageous in real-time applications. The necessary DP and CUSUM computations can also be performed independently on different segments of the line-frame, which is essential for parallel computation implementation. As a short summary, it is certainly up to the designer to add and refine the weighting of \( \nu \) and the adjustment of \( h \) up to a level sufficient for his or her requirements. This is essential since the observed intensity values may not provide sufficient discrimination like also demonstrated in Fig. 3.10. For example, one can conjure up an image of a decision tree that adopts a hierarchy of techniques so that the performance depends flexibly on a combination of metrics. For techniques developed earlier for different types of defects without stringent theoretical and distributional assumptions, see for example Peura and Iivarinen [68], Iivarinen and Visa [35]. Such techniques have the potential to increase overall detection performance at least in the form of reduced false alarms.

![Figure 3.10](image_url)

Figure 3.10: A typical histogram of background gray levels mixed with the intensity values of a low-contrast wrinkle. If compared with Fig. 2.1, the states \( \theta_0 \) and \( \theta_1 \) are not separable by intensity only.
Chapter 4

Discussion

With the long history of investigations and applications within industry, the CUSUM algorithm, proposed by Page [65], is generally considered to be a key change point detection tool. Compared with fixed-size tests of equal false alarm probability, several studies have reported that the CUSUM algorithm is preferable in terms of efficiency when the minimum interesting change magnitude is small; see e.g. Roberts [84], Basseville [8], Basseville and Nikiforov [9]. This practical property stems from the properties of Wald’s SPRT [104] and studies of the time it requires to reach a decision. The CUSUM algorithm basically operates as long as the decision taken, see (2.4), does not admit $H_1$. If the previous decision is $H_0$, the decision statistic is set to zero and a new cycle restarts the algorithm. The ARL function defines at $\theta_0$ the mean time between false alarms and at $\theta_1$ the mean delay for detection. Together these define the CUSUM algorithm’s performance exactly as the Neyman-Pearson rule [61] is defined with the aid of false alarm and detection probabilities, using some numerical decision boundaries. The decision regarding the state of the process at any time depends on the whole prior measurement history and not just on the most recent observations. Hence, all the information is embedded solely in the decision statistic $g_k$, which can often be updated with operations close to standard integration. An other significant property is the availability of change time estimation with close relation to the optimum MLE formalism. Though in many cases not taken into account, much of the MLE’s computational effort can be avoided by proper parametrization of the CUSUM algorithm, which is also the key issue from the point of quickest detection. The developed DP-CUSUM detector with a sequential algorithmic basis was here used to detect and segment changes considered interesting from an observed sequence of line frames. As the adopted DP-based search procedure finds ideally the most likely pixels at the interesting state $\theta_1$, the simple formalism of the CUSUM algorithm is easily adapted for other applications of imaging, too.
The so-called receiver operating characteristics, ROC for short, are used to quantify the ability of a detector to discern between signal and noise; see e.g. Fawcett [19]. At this point, the reader is referred to the ROC analysis in Publication IV as it is considered to give a reasonable approximation and more thorough understanding. In imaging, the questions of recovering the noise-obscured pixels at \( \theta_0 \) precisely and overall detection performance are, however, interrelated. Referring to the progress of Tonissen and Evans [94], Johnston and Krishnamurthy [38], and the references therein, no generic tools or explanatory uniting concept appear to exist for the DP procedure. Though the DP is precise in many practical cases, developing an accurate closed-form measure of error seems unrealistic due to the dependencies of the score recurrences. Thus, as a word of caution regarding the ROC, the effect of the DP-CUSUM integrating a mix of pixels at \( \theta_0 \) and \( \theta_1 \) cannot be—and has not been—rigorously accounted for, for example, in our Publication IV. As also indicated in the article, the given expressions for ROC analysis, \( f_\ell(x)^1 \) and \( p_\ell(x)^2 \), see the proofs below, map the run-length analysis of the CUSUM to the imaging case in a plain manner. In this sense, the approach is basically no exception to many useful approximations in sequential analysis: simplicity enables one to see the central issues clearly; see e.g. Basseville and Nikiforov [9]. Just like Tables 2.1 and 2.2, the analysis allows one to explore the run-lengths and quantiles of the DP-CUSUM in a rather realistic setting without being sidetracked into complicated simulations which, as explained in Section 2.3, tend to become entangled in massive probability calculations. A more pragmatic aspect in terms of the ARL and detection is discussed next.

As the noise-obscured signal and detection results shown in Fig. 3.3 illustrate, the DP does not find all the pixels at \( \theta_1 \) correctly. In spite of the errors, this does not necessarily imply bad overall detection performance; see also the discussion of Tonissen and Evans [94]. For humans, visual features are dominant characteristics for identification. Hence, for some readers, it may appear obvious that the DP-CUSUM detector recovers the pattern to an easily identifiable level, though a closer look reveals that quite coarse inaccuracies are shown in Fig. 3.3 (b). The reason for this is that humans tend to form a tentative hypothesis about what they see, based on what they have seen earlier, which is difficult to factor out as the embedded pattern is also exposed to the reader on the same page. Hence, this may interfere

---

1Let us assume that we have \( j \) random variables so that \( x_1 < x_2 < \cdots < x_j \). Let \( F_i(x) \) be the CDF of \( x_i \). Since \( x_i \) will be less than any given \( y \) if and only if all \( x_i, i \in [1, j] \), are less than \( y \), then the CDF is \( F_y(x) = P_y(x < y) = \prod_{i=1}^j P_i(x < y) = \prod_{i=1}^j F_i(x) = [F(x)]^j \) if all \( x_i \) are i.i.d. random variables and \( F_i(x) = F(x) \). The PDF is \( f_y(x) = D(F_y(x)) = j[F(x)]^{j-1} f(x) \), where \( f(x) = D(F(x)) \) and \( D(\cdot) \) is the derivate operation.

2Let us assume that \( x \) denotes a stopping time. For i.i.d. stopping time, finding the probability that \( x_1, \ldots, x_\ell \) are all greater than \( y \) proceeds similarly as above while it must be remembered that now we have \( P_i(x > y) = 1 - P_i(x \leq y) = 1 - F_i(x) \).
with perception, particularly, if the pattern is further compared with the modest number of recovered pixels by the DP detector in Publication IV. More scientifically, however, one should also recall that the ARL at $\theta_0$ was well over one thousand frames, denoting that it is highly unlikely that we see any false alarms in their conventional meaning. Because alarms are raised constantly, they are correlated in time too, whatever the actual error in the coordinates. Hence, these alarms exhibit an underlying systematic aspect and, because of this, it remains possible to infer a model and explain the behavior at the recognition stage, which the mind does almost automatically.\footnote{The ROC curve for the case with the signal $A = 1$ embedded in noise, Fig. 3 (b) in Publication IV, may give the impression of a completely undetectable change. In reality, however, the DP-CUSUM would raise alarms with a shorter mean interval than the precomputed ARL at $\theta_0$ would indicate, leaving the underlying dynamics of a change unclear.} The ultimate accuracy is, of course, limited by the intrinsic noise of the pixels.

Although the established objectives are often not that complex and can be easily formulated as a simple sequential problem, many practitioners find testing of the obtained measurements one-at-a-time impractical or at least unattractive. For the field of imaging, perhaps the main reason for this is that surprisingly little, virtually no, methodology has been available to guide the design of sequential algorithms. Using sequential analysis in practice requires some level of discipline, but pragmatic solutions have been sought for decades for many difficult technical problems which can also be utilized to expand the conceptual horizon of this work. As the examples presented particularly in Chapter 3 show, it is relatively easy to formulate problems with flexible sequential solutions that address the multiple needs of users for coping with uncertainty in both technological and natural worlds. For instance, no accurate knowledge of the noise or signal statistics is required with the presented formulations. Having understood the problem, the gains in using sequential methods, with or without elegant Bayesian problem formulation, may turn out to be substantial as it is ultimately economical to try to reach decisions in the shortest possible time.
Chapter 5

Conclusion

Improvements in hypothesis testing over the present state of the art can be effected only by departing from the typical, simplifying assumptions step-by-step and introducing a more discriminatory way of reasoning. Because abrupt or gradual changes by no means imply only changes of large magnitude, even small improvements can have large benefits. A correctly parametrized sequential detector is rarely initialized by the change point itself. Thus, at the expense of the delay to detection being a random variable, detection can often be carried out efficiently with fewer samples than with the conventional fixed sample size hypothesis testing of equal false alarm probabilities. Though this might introduce a small error in change point estimation, the core of sequentiality is basically that the hypotheses are continuously tested by collecting data, and the effective sample size depends on the quality of the data itself. As no detector can prove causal connections in its decisions beyond all possible doubt, it may occasionally take a very long time for a sequential test to finish. However, though this uncertainty is in many cases frustrating, it is very likely an accurate reflection of the true situation. From this point of view, the drawbacks of fixed-size tests are usually related to compactness and computational efficiency regarding on-line detection. Obviously, the adopted DP formalism is by no means necessary if more elaborate procedures are available, but for the web imaging problem flexibility of the DP and sequentiality of the CUSUM algorithm complement each other well.

For standard time-series analysis, the CUSUM algorithm is often a near-optimal change point detection procedure and to achieve quickest detection, it can incorporate the whole measurement history. With the aid of the DP formalism, we applied the CUSUM algorithm to sequential change detection in imaging, which is basically the main contribution of this thesis. The CUSUM algorithm and the DP procedure are, of course, not novel, but as far as we believe, change detection in imaging has not been approached from the point of quickest detection with the DP reducing the
frames to the most likely sequences of pixels of interest. Being therefore rich enough to embrace many detection problems in signal and image processing, and since the tools and equations for the task are provided here, the whole chain of analysis, including the ROC curves, is applicable to other cases too, whatever the underlying PDF or the frame dimension. We also note that the recent work by Luceño and Puig-Pey [56] has had a significant impact on our understanding of the change detection problem. Since the PMF is approximated numerically, the advent of 64-bit processor architecture, owing to its massively increased floating point accuracy, is also expected to provide increased accuracy with the necessary computations. The problem of detecting small abnormalities and changes in images is of considerable interest in areas such as aerial photographs, radar, sonar, and infra-red image analysis that all have issues in common with this work. At the best, only a different portion of the electro-magnetic spectrum is used and so application of the developed DP-CUSUM formalism is by no means limited.

Choosing the right elements and flexible features maximize the impact and usefulness of sequential analysis. Because it also becomes necessary, in terms of optimum performance, to identify the main properties of the events of interest, greater attention should be paid to more intensive exploitation of information already on hand. In our case, one of the key innovations was to utilize local extrema, which are often very eye-catching in images, too. Local minima and maxima describe only changes in the intensity and, like in many practical applications, a large part of the information contained in the measurements lies in their nonstationarities. Obviously, if a positive jump in the signal level takes place, the probability of local maxima exceeding some positive threshold is higher than that of local minima. Based on this simple fact, reducing the complexity of the DP-based search, even by several orders of magnitude, is a straightforward task. Here, in addition that to the fact that the overall number of pixels is reduced, the average number of candidates within the candidate region is also reduced, which further reduces the number of operations required to find a sequence of pixels most likely at $\theta_1$. As computational power is not wasted, utilization of a few extra frames might well become possible within the available processing time, which usually limits the degree of uncertainty in badly noise-obscured scenarios.

There are undoubtedly occasions when fixed sample size tests are feasible for the designer, but sequential tests are not. Fixed-size tests should be favored if parameters of the event of interest are known and, particularly, if the event has some finite known duration; i.e., if a compact and efficient test with properties close to a matched filter exists, there is basically no need for a sequential detector. However, if the parameters describing a change in the data are either constant or slowly time-varying over a longer period of time and the magnitude of the change itself is small, sequential algorithms for change detection may turn out to be very efficient.
and superior performance-wise. Otherwise, and in the opinion of experts in the field, there seems to be no general or persuasive scientific reason for preferring a sequential test to a fixed sample size test if the facts primarily concerning efficiency and small changes are excluded. In many practical applications, short- and long-term changes may have to be monitored together and, though the proposed DP-CUSUM detector is basically equivalent of a multitude of fixed sample size detectors in a single package, a variety of problems can be solved efficiently by combining the fixed-size and sequential frameworks. Thus, when designing and experimenting change detection algorithms, it is typically beneficial to implement a few fixed-size detectors along with a sequential detector in order to strike a balance between computational requirements, special cases, and performance.

To sum up, we treated the on-line web inspection problem as a sequential hypothesis-testing and segmentation problem with the null hypothesis representing the absence of defects. Allowing for an algorithmic speed gain, extreme values were found to describe regularities of the imaged surface structure efficiently. The flowchart in Fig. 5.1 links our Publications with the presented central web imaging issues.

Figure 5.1: Summary of the work and its evolution in our Publications.
Bibliography


Errata

In Publication I, equation (19): $M_{n|n-1}$ should read $M_{k|k-1}$. 
Publications
Publication I

ON SEQUENTIAL ON-LINE OUTLIER DETECTION AND A LINESCAN APPLICATION

J. Vihonen, T. Ala-Kleemola, R. Kerminen, J. Jylhä, A. Visa

Tampere University of Technology, Institute of Signal Processing
Korkeakoulunkatu 1, P.O. Box 553, FI-33101 Tampere, Finland

ABSTRACT

Industrial quality monitoring is increasing rapidly, and challenging signal environments with requirement of steady performance pose conflicting demands to on-line tests. The sequential probability ratio test (SPRT) and the Kalman filter (KF) are proposed as two tools for detection and recognition-oriented signal processing. A modified sequential test is suggested and applied to a linescan problem.

1. INTRODUCTION

Systems ranging from mechanical vibration monitoring to elegant sensor life-cycle management and fault detection have become increasingly popular in industry. Many of these applications adopt and benefit from on-line detection of a change. Linescan camera applications have also established a firm foothold in several civilian and military applications.

Excellent techniques exist for the detection of changes under the assumption of statistical independence. Assuming that the measured signal does not admit the independence assumption, the appearance of increased-variance observations should be distinguished as transient state. For example in papermaking, variations in mass distribution of pulp and other raw materials result in thick or thin spots or even holes, giving dependencies some structural form. As it might be expected, accurate segmentation and decomposition of such local signal characteristics is of interest. Two tools, optimum in their own categories, are proposed for this task. A wide-ranging image change detection survey can be found in [1].

The Page’s detector [2] is based on the idea of Wald’s SPRT [3]. These methods do not need to determine the number of measurements for the test in advance. In addition to simplicity, a notable property of SPRT is that with given fixed values of error probabilities it is optimal for testing between two independent distributions in the sense that it minimizes average run length [4]. Appealingly, Page’s test and SPRT work well in practice for dependent sequences.

Stochastic local process transitions can be captured through KF residual analysis. For hypothesis testing, this is an attractive property since complete process-characterizing transient state models are not necessarily required. Assuming that the true process state model is known, the filter is optimal in the sense that it minimizes the Bayesian mean squared error for each new estimation [5]. The filter is also the optimal linear minimum mean squared error estimator, if the Gaussian statistics are invalid. A recent KF residual treatment can be found in [6].

With simple deduction, it becomes obvious that stepwise process tracking capitalizes unexpected, “hard” transitions. Slow or “soft” changes remain easily undetected. Therefore, both the KF and SPRT have differing advantages. Adopting the expertise of these methods to a single detector is proposed. Here, the residual check is embedded in SPRT, which is otherwise treated as Page’s test (a.k.a. CUSUM). The joint use renders quicker detection and more overall sensitivity.

2. PROBLEM STATEMENT

Manufacturing abnormalities appear, for instance, in papermaking and cast metal surfaces. There exist imaging solutions to track down such process-related defects. The system grazed here uses a linear array of charge-coupled device (CCD) detectors in which the detectors scan their field-of-views in a direction orthogonal to the moving surface. For every performed scan, a linescan spatial surface profile is captured in each detector. The spatial extent of a defect can range arbitrarily, which restricts the use of off-line approaches in terms of undefined memory usage and processing time. In other words, a complete two-dimensional image could be provided, if restrictions concerning memory and computing capacity were ignored. Hence, the change detection methodology provides preferential tools for linescan outlier detection. Specifically, the arriving data needs to be processed line-by-line, i.e., sequentially. The recognition problem is not addressed here.

Desired inspection properties include few false alarms and missed detections, and low detection delay. Next, a new test is proposed and the discussed methods are outlined. The reader is encouraged to refer to [5, 7]. In further discussion, its application is suggested and some CCD detector responses for real measured linescan images are examined. Finally, the conclusions are drawn.

Thanks to the National Technology Agency of Finland (TEKES) and Nokia Foundation for funding and support. Special thanks to Mr. J. Rauhamaa, ABB Oy, for co-operation and the provided test images.

Author’s email: juho.vihonen@tut.fi
3. 1-D PAGE-LIKE RECURSION FOR ON-LINE HYPOTHESIS TESTING

Page suggested the use of repeated test for the hypotheses:

\[ H_0 : \ \theta = \theta_0 \]  \hspace{1cm} (1)  
\[ H_1 : \ \theta = \theta_1. \]  \hspace{1cm} (2)

The algorithm is based on the concept of the logarithm of the likelihood defined by

\[ g(x) = \ln \frac{p_{\theta_0}(x)}{p_{\theta_1}(x)}, \]  \hspace{1cm} (3)

where \( p_{\theta_0} \) and \( p_{\theta_1} \) represent the probability density functions of the random variables under two distributions. In the particular case where the distribution is Gaussian with mean value \( \mu \) and constant variance \( \sigma^2 \) the statistic can be written as

\[ g(x) = \frac{1}{\sigma} \left( x - \frac{\mu_0 + \mu_1}{2} \right). \]  \hspace{1cm} (4)

Let \( k \) denote the time index. The proposed recursion augments the standard test and is given by

\[ \arg \min_k \{ S_k \geq h \}, \]  \hspace{1cm} (5)

where

\[ S_k = \max \{ 0, S_{k-1} + g(y_k + \epsilon_k \gamma) \}. \]  \hspace{1cm} (6)

The point here is that generally faults range in duration and magnitude. Especially, short duration raises the risk of misdetection for a long-term test, which is why the term \( \epsilon_k \gamma \) is introduced in (6). Assume that \( \epsilon_k \) is given by some auxiliary process observer:

\[ \epsilon_k = \begin{cases} 1, & \text{likely transient state} \\ 0, & \text{normal state} \end{cases}. \]  \hspace{1cm} (7)

By further setting

\[ \gamma = \frac{\mu_1 - \mu_0}{2}, \]  \hspace{1cm} (8)

the functional property of (4) reduces the recursion to minimum possible jump detector, whatever their magnitude [7]. The step (8) is simply for allowing unconstrained statistic development under Gaussian statistics. Partly steered by some auxiliary supervision, detection is declared when the statistic \( S_k \) exceeds \( h \).

4. 2-D SEQUENTIAL LINESCAN PROCESSING

Let the \( j \)th observation of \( k \)th scan be denoted by \( y_{k,j} \). The recursion for line-by-line sequential processing is the same as in Sect. 3, except that the expression (6) is replaced with

\[ S_k^j = \max \{ 0, S_{k-1}^j + g(y_{k,j} + \epsilon_k^j \gamma) + \sum_{i \neq j} \epsilon_i^j g(y_{k,i} + \epsilon_i^j \gamma) \}, \]  \hspace{1cm} (9)

which continues with the score function accumulation according to the spatial extent of a transient. Suppose that two proposed algorithms are run in parallel; the first for detecting an increase in mean, and the second for detecting a decrease in mean. For example for positive tail of the distribution, accumulation is easily accomplished by

\[ S_k^j = \begin{cases} \max \{ \epsilon_k^j S_k^{j+1} \}, & \text{if } \epsilon_k^j = 1 \\ S_k^{j-1}, & \text{otherwise} \end{cases} \]  \hspace{1cm} (10)

for all \( n \in \{ i | y_{k,i} > \mu_0 \} \). So, the maximization of statistic value is acceptable only if the observation is in the right tail of the underlying distribution and has acceptable risk level.

5. MODEL AND RESIDUAL UTILIZATION

The value of \( \epsilon_k^j \) may be based upon thresholding the current prediction error after an inverse filtering procedure, as it will be discussed next. The KF and the state-space model have been used in a variety of signal processing and control applications. Whereas the KF provides an elegant solution for residual generation, the surface modeling issue is highly complex. The general formulations of the chosen concepts are briefly summarized next.

5.1. Used surface model

Under a stable process state, the observed noise and surface responses are assumed to be realizations of \( p \)th-order Gauss-Markov process, i.e.,

\[ s_k = \sum_{i=1}^{p} a_i s_{k-i} + u_k, \]  \hspace{1cm} (11)

where \( u_k \sim N(0, \sigma_u^2) \). The statement may be regarded as an autoregressive, AR(\( p \)), process excited by white Gaussian noise [5] having state-space representation:

\[ s_k = As_{k-1} + Bu_k, \quad k \geq 0, \]  \hspace{1cm} (12)

and \( u_k \sim N(0, Q_k) \). An extensive treatment of AR-identification can be found in [8].

5.2. Residual utilization

Using the measurement model

\[ y_k = h^T s_k + w_k, \]  \hspace{1cm} (13)
\[ w_k \sim N(0, \sigma_w^2), \] and without specifying further details, the KF recursion can be written

\[ \hat{s}_{k|k-1} = A\hat{s}_{k-1|k-1} \]

\[ M_{k|k-1} = AM_{k-1|k-1}A^T + BB^T \]

\[ k_k = \frac{M_{k|k-1}h}{\sigma_w^2 + h^TM_{k|k-1}h} \]

\[ r_k = y_k - h^T\hat{s}_{k|k-1} \]

\[ \hat{s}_{k|k} = \hat{s}_{k|k-1} + k_k r_k \]

\[ M_{k|k} = (I - k_kh^T)M_{n|n-1}. \]

The optimality of the above recursive algorithm is based on knowing the quantities \( Q \) and \( \sigma_w^2 \). Note that the steady-state values can be precomputed by iterating. The KF residual monitoring is used for model validating. Based on the assumption that no failures have occurred, the residual sequence is theoretically well described as a zero mean Gaussian sequence with variance \( \sigma^2 = \sigma_w^2 + h^TMh \).

The confidence bounds for \( c_k^2 \) are easily constructed. If \(|r_k^2|\) is greater than \( 3\sigma \), the current sample is declared as unacceptable, and the KF is initialized with the expected normal-state signal mean value. Each scan is filtered from both ends and minimum jump processing is used only if both thresholded residual sequences indicate fault. See [7] for more sophisticated analysis methods.

### 6. EXAMPLES

AR(4)-model is used with the recursive-in-order Burg identification for readily transient-free data. The stochastic noise input to the AR-model should represent the true noise of the process, which determines \( \sigma_w^2, \sigma_a^2 \), is set here to 0.0025\( \sigma_a^2 \), which emphasizes the uncertainty of the model, but allows process tracking. The threshold \( h \) was chosen to provide false-alarm free performance, and \( \mu_1 = \mu_0 + 4 \).

Comparison is made between the proposed recursive scheme and the iterated conditional modes (ICM) [9], which is an iterative, well-known, and well-performing segmentation algorithm. The ICM was evaluated in the context of hypothesis testing. In Fig. 1 (a), a heavy pulp spot is captured. Badly wrinkled surface is observed in Fig. 2 (a). The original images are scaled for better view. Additional figures represent the image processed using different methods, and gray and white colors represent the different tails of the background distribution. The double-sided ICM iterated the images several times using 3x3 window with threshold and neighborhood influence parameters set to be the best found compromise for the both cases. The used window in (9) was chosen correspondingly, i.e., \( i \in [j - 1, j + 1] \).

In the general case, the extension in (9) should treat a transient starting from \( y_{k-m}^j \) to its other edge \( y_{k+n}^j \) as one. It is not trivial to determine the true \( m \) and \( n \) of a transient, although Page’s test could be used. Here, these parameters are chosen to be \( m = n = 1 \), which is obviously ad hoc. The used choice amounts to maximize the decomposition resolution. However, the implemented ICM was expected to give the ground truth in Figs. 1, 2. Even so, the proposed recursion seems to exhibit much alike results. In fact, the suggested method may be interpreted as a recursive version of the ICM. Note that the local unif statistic is forced down and vice versa, which stems from (7). Hence, the phase (7) may be compared to the ICM initialization and the extension in (9) similarly to the conditional probability maximization.

The KF is designed to filter any deviations in the measurements and predictions using updates. Thus, the softer the change is, the harder the task becomes to tackle with residuals. The KF’s error covariance becomes also easily large after identification. Hence, even a relatively large residual falls well within its bounds (not to even mention bounds with few step prediction using forward projected covariance). The used simple model is most suitable for cases where the process change is abrupt. This, on the other hand, is related to numerous factors such as pixel size, sweep time, and focusing. Hence, the possibility of using multiple models can’t be ruled out.
Fig. 2. An example of a very wrinkled paper surface and decomposition results. The ICM and the proposed recursion show both decent performances.

Keeping in mind the ICM’s fixed window size, the recursive statistic incorporates the measurement time history. This gives the Page-like recursion an obvious advantage. The simple hypothesis testing justifies also the minimum jump processing, since it is unnecessary to test a sample twice. Unfortunately, the problem statement, to be precise, line-by-line processing results in lack of symmetry of the sequential testing. Note that a horizontally-oriented transient (Fig. 2) represents the pathological case for the standard Page’s test, but the proposed recursion handles it pretty well.

8. CONCLUSIONS

We proposed new Page-like, minimum memory recursion with supervision to enhance and accelerate short-duration signal detection. It involves Kalman filter residual analysis with affordable computation. By recognizing that the Page’s test is optimal for only some jump magnitude, the residual treatment can be used to liberate the score statistic development in clear outlier cases. Despite the ad hoc nature of the neighborhood use, which led to a recursive algorithm with features similar to the ICM, the same principle was adopted to linescan data processing with satisfactory outcome. The approach may be further improved by using change time estimation. Now the spatial alarm time from the Page’s test is used for segmentation.

9. REFERENCES


Publication II

MODELING STRUCTURAL VIBRATIONS FOR AUTOMATED AIRCRAFT FATIGUE MONITORING

Juha Jylhä¹, Juho Vihonen¹, Timo Ala-Kleemola¹, Riitta Kerminen¹, Jarkko Tikka², and Ari Visa¹

¹ Institute of Signal Processing
Tampere University of Technology
Korkeakoulunkatu 1, FIN-33720 Tampere, Finland
Email: first_name.second_name@tut.fi
² Patria Aviation Oy
Lentokonetehdaantie 3, FIN-35600 Halli, Finland
Email: jarkko.tikka@patria.fi

Abstract: Knowledge about high frequency vibrations in aircraft structures is insufficient. This paper proposes a strain signal model which allows the high cycle fatigue to be monitored. In a Finnish Air Force (FiAF) research program, an extensive operational loads measurement system is instrumented in one F-18 fighter. The system includes high quality strain gauges and enables our perusal of stress manifestation in critical structural details. Essential strain signal properties consist of vibration intensity and frequency which vary over time. This variation causes inconvenient non-stationarity for the analysis. In our model, vibration is analyzed and represented based on rainflow cycle count enduring the harmful variation. Presented experiments concern two F-18 structural details: the leading edge flap hinge and the vertical stabilizer attachment. The results demonstrate the meaning of the high cycle fatigue and the appropriateness of the signal model.

INTRODUCTION

With a continuous electronic supervision, condition of aircraft structures can be observed with a reduced interval of physical inspections. This means a significant reduction in maintenance expenses, and the risk to experience a structural failure decreases. The subject is widely discussed in [1].

An overview of recent Finnish aeronautical fatigue investigations is given in [2]. FiAF launched a research program to improve automated fatigue tracking of F-18 Hornet [3], [4]. The baseline is the existing monitoring system provided by the original equipment manufacturer. The purpose of the program, led by Patria Aviation, is stress loads prediction. Neural networks are employed to return the loads via low-rate flight parameters stored in the
aircraft data system. Figure 1 illustrates this “predictive” strain and fatigue modeling. To provide necessary information for the fatigue analysis, FiAF instrumented one F-18 with so-called Hornet Operational Loads Measurement (HOLM) system. The instrumentation comprises about 60 high quality strain gauges and an enhanced flight data recorder. Within the program, it turned out that the strain vibration has a significant role in the life cycle of certain structural details such as the vertical stabilizer attachment. Unfortunately, the vibration is not straightforward to characterize. The issue is found to be crucial for the automated fatigue prediction.

This paper considers an “explanatory” strain modeling which supplements the overall predictive model illustrated in Figure 1. Our model refines the HOLM strain gauge data for neural network training. The main contribution centers on incorporating the vibration into the model. Basically, the issue is to express a high frequency strain occurrence at a time window in a compact way. This can be regarded as a parametric compression. The parameters include information relevant for fatigue. Then, the neural networks predict these parameters revealing the realistic condition of structures.

Figure 1. Simplified modeling blocks of a flight parameter based fatigue analysis; see a more detailed description in [4]. Our contribution focuses on the explanatory modeling and its conversion back to strain. That is to say, a strain compression and decompression.
Vibration characterization issue is typically approached by frequency analysis. Conventional schemes rest on applying the Fourier transform. Recent research examines wavelets [5] and an empirical mode decomposition [6]. In this paper, the stress fluctuation is characterized based on the rainflow cycle counting. It is compared to a Fourier-based method. The model suitability for fatigue description is appraised in results where also the demand of the model is underscored.

The outline of the paper is as follows. In the next section, the strain signal is considered in general level. Then the explanatory modeling and the model application in fatigue prediction are discussed. Also, the results regarding the two F-18 structural details are given. Finally, there are discussion and some critical remarks about the paper.

**STRAIN SIGNAL**

The HOLM instrumentation provides the appropriate information including strain gauge measurements. Here, the gauge data is considered as follows. The strain signal

\[ s = \varepsilon_s + \varepsilon_v + s_d \]

where \( \varepsilon_s \) denotes the static component, \( \varepsilon_v \) vibration component, and \( s_d \) disturbances. The disturbances comprise noise along with distortions and interferences originating in F-18 systems and in the measurement system. In HOLM, strain gauge implementation is designed keeping an eye on the formerly unknown stress behavior. Measurement dynamics enables to study the whole range of varying loads in structures.

Here we consider merely strain treatment and assume that \( s_d \) is removed by preprocessing beforehand. In the notations, we favor relative strain \( \epsilon \) instead of stress \( \sigma \). Their relation is straightforward. Since the actual stress level in aircraft structures is practically always in the linear elastic region, Hooke's law is valid. Further, we mostly refer to relative strain as strain.

In our model, the strain consists of the static component and the vibration component, i.e., low and high frequency bands in practice. The reason for the division derives from the required compression. The strain has to be represented in a time window. For one window, the static part constitutes a strain offset, and its representation is an explicit signal decimation. A cut-off frequency \( f_{\text{cutoff}} \), separating \( \varepsilon_s \) and \( \varepsilon_v \), is chosen obeying the requirements for the window length. Obviously, the decimation is excluded with \( \varepsilon_v \).

We present two approaches to describing \( \varepsilon_v \). When exploring the strain signal, we noted that the vibration takes place intermittently with varying frequency. The phenomenon has even some stochastic, noise-like properties, but fundamentally, the problem reduces to variable stationarity. Typically, fatigue is related to about fifth power of strain cycle amplitude. Thus the core of the characterization is including the accurate amplitude information into the signal model. Both our approaches rest on extracting vibration frequency and amplitude of dominant components in short time windows. The number of those concurrent components is open in advance but invariant within our analyses. In the first approach, we use short-time Fourier transform (STFT) and, in the second one, rainflow cycle counting.
Fourier transform is a conventional method for a frequency analysis. Assuming certain level of stationarity, the transform converts the signal from the time domain to the frequency domain. Here the non-stationarity and the fatigue-amplitude sensitivity complicate the application. Windowing is required. The article [7] is a classic reference on the employment of discrete Fourier transform to time-frequency analysis.

Next we discuss our STFT application for vibration characterization. We perform the STFT for $\varepsilon_v$. Window length $T_{\text{win}}$ and window step $T_{\text{step}}$ determine windowing. Those windowing parameters govern model sampling time $t_m$ and window length $N$ (in samples): $t_{m+1} = t_m + T_{\text{step}}$, and $N = f_s T_{\text{win}}$ where $f_s$ is the sampling rate of the strain signal. So the signal window to be transformed comprises $N$ samples of $\varepsilon_v$ at time instant $t_m$. The STFT result, complex valued $E_k^\varepsilon$ denotes the strain intensity (and phase) within the window. In this time-frequency plane representation, $t_m$ refers to time and $k$ to frequency dimension. Related radial frequency $\omega = 2\pi f_s k / N$. See more detailed STFT description in [7].

The Fourier-based characterization includes one or more of the strongest samples from the Fourier spectrum. From every spectrum, we select $D$ highest peaks (absolute value) having indices

$$h_1 \ldots h_D \in \left\{ 1 \ldots \left\lfloor N/2 \right\rfloor \right\}. \quad (2)$$

The frequency content tend to spread, and we aim to find $D$ separate elements. Fourier model parameter vector

$$\theta_v^\varepsilon = \left[ \varepsilon_{h_1}^\varepsilon \mid E_{h_1}^\varepsilon \right] \ldots \left[ \varepsilon_{h_D}^\varepsilon \mid E_{h_D}^\varepsilon \right] \omega_{h_1}^\varepsilon \ldots \omega_{h_D}^\varepsilon \right]^T \quad (3)$$

consisting of $1 + 2D$ elements. The static component $\varepsilon_{h_i}^\varepsilon$ derives simply from low-pass strain signal sampled at $t_m$.

To be precise, the representation by $E_k^\varepsilon$ demands stationarity—is reliable only when there is coherent oscillation through the window. In reality, that is not the case even with a short window from strain signal in question. The non-stationarity leads to inaccuracies in fatigue prediction.

**RAINFLOW-BASED VIBRATION MODELING**

A demand of precision in the amplitude characterization guided us to study rainflow cycle counting. Literature recognizes several rainflow methods. The one discussed here obeys the ASTM standard [8]. Every strain hysteresis loop is tracked and read into an output matrix $R$ in order of its beginning time $t$. This rainflow count
\[ \mathbf{R} = \begin{pmatrix}
\varepsilon_1^1 & \ldots & \varepsilon_L^1 \\
\varepsilon_1^m & \ldots & \varepsilon_L^m \\
p_1 & \ldots & p_L \\
t_1 & \ldots & t_L \\
T_1 & \ldots & T_L
\end{pmatrix} \] (4)

where one column represents single vibration cycle. Indices 1…\(L\) imply cycle's order number. All essential data concerning fatigue tracking is included rigorously: cycle amplitude \(\varepsilon_a\), cycle mean value \(\varepsilon_m\), the type of cycle \(p\) (full or half cycle: 1 or 0.5), beginning time \(t\), and cycle period \(T\). Note that \(\varepsilon_a\) exactly equals the cycle amplitude in the original signal.

We examine \(\mathbf{R}\), window by window, and determine the temporarily dominating components. To eliminate the static component, we dismiss cycles whose \(T > 1/f_{\text{cutoff}}\) before further processing of \(\mathbf{R}\) (analogous to high-pass filtering). Then the short-term maximum amplitude

\[ \varepsilon_a^* = \max_i \varepsilon_a^i \] (5)

where index \(i\) indicates cycle order number in the window of \(\mathbf{R}\): \(t_m - T_{\text{win}}/2 \leq t_i < t_m + T_{\text{win}}/2\). So \(\varepsilon_a^*\) denotes the maximum vibration amplitude within \(T_{\text{win}}\) at \(t_m\), and it basically dictates the damage induced. In addition, it is essential to know the cycle count within the window. Hence, we calculate a short-term vibration density on amplitude ranges tied to the short-term maximum amplitude. But not all the frequency content contributes. After a preliminary investigation, adjusting the ranges becomes a simple task in terms of the fatigue. We define the amplitude ranges by a vector of weighting coefficients: \(\delta = [\delta_1 \ldots \delta_{D+1}]^T\). \(D\) is the number of ranges; \(0 \leq \delta_d < 1\); and \(\delta_{d+1} < \delta_d\). Bounds \(\varepsilon_a^* \delta_d\) limit the ranges. For range index \(d\), the short-term vibration density

\[ f_d^* = \frac{1}{T_{\text{win}}} \sum_{\varepsilon_a^* \delta_{d+1} < \varepsilon_a^* \delta_d \leq \varepsilon_a^* \delta_d} p_i, \quad d = 1 \ldots D. \] (6)

Also here, \(i\) is limited to the window. For every component, cycles outside the range are dismissed. Figure 2 demonstrates this rainflow processing.

It is reasonable to specify that the first bound equals \(\varepsilon_a^*\), i.e., \(\delta_1 = 1\). The first range \([\varepsilon_a^* \delta_2, \varepsilon_a^* \delta_1]\) covers the biggest cycles and typically governs the result in respect of fatigue. Even alone, it may be sufficient for the analysis.
Rainflow model parameter vector

\[ \hat{\theta}_R^\alpha = \begin{bmatrix} \varepsilon_\alpha^{i_1} & \varepsilon_\alpha^{i_2} & f_1^{i_1} & \cdots & f_D^{i_2} \end{bmatrix}^T \]  

(7)

consisting of \(2 + D\) elements. Compare to Fourier model (3); the static component is identical here. Figure 3 shows an example of the rainflow model output.

The fundamental concept of our rainflow approach is quite analogous to the Fourier approach. In the STFT-based analysis, we extract the vibration parameters (intensity and frequency) using Fourier transform. Similarly, rainflow contains cycle amplitude and occurrence time, but it provides rather amplitude spectrum instead of frequency spectrum. We survey frequency bands with STFT whereas the amplitude ranges with rainflow. While scanning the time-frequency plane involves issue with non-stationarity in the STFT approach, there are no limiting assumptions about signal shape in the novel rainflow approach.

One can see that the short-term maximum is the only parameter describing the amplitude behavior. \(\varepsilon_\alpha^{i_1}\) follows a curve on the crest of the oscillation, i.e., the envelope curve. Most important, attaching the cycle densities to the envelope by \(\delta\) provides an insight into the challenging vibration.
FATIGUE CHARACTERIZATION BASED ON PRESENTED MODELING

This section considers the fatigue life expenditure (FLE) analysis. In the FiAF program, the neural networks are trained to produce the strain parameters. The purpose is to predict the FLE. Our model ability to describe the strain and fatigue information is discussed next. Thereafter presented experimental results contain FLE analysis for two structural details of F-18 and clearly suggest the rainflow model to be applied.

Conversion back to strain

Common, strain or stress-based, fatigue evaluation tools cannot handle the presented strain as input data; see Figure 3. Typically, the favored input types are either raw stress signal or stress turning points. A natural choice to estimate the model suitability is to use the original strain signal as a point of comparison. We produce the model, the signal parametric form (3) or (7), from the original signal and then reconstruct the signal vice versa. Finally, we set the original FLE against the reconstructed signal FLE.

Both our parametric models (3) and (7) have an analogous interpretation. The static component serves as strain offset. $D$ dominant higher frequency components are assumed to represent the vibration adequately. Each, with index $d$, comprises the envelope ($|E_m|^{d}$ or $\varepsilon_m^{d}\delta_d$) and the frequency ($\omega_m^{d}$ or $f_d^{\omega}$).
The signal reconstruction is based on orthogonal sine element addition. First, we form static strain by resampling $\varepsilon_{st}^s$ to a desired rate $f_{rs}$ that does not necessarily equal the original $f_s$. Since the vibration involves typically much higher frequencies than model sampling rate $1/T_{\text{step}}$, $f_{rs}$ has to be higher than $1/T_{\text{step}}$ in which case the reconstruction demands oversampling. We generate strain vibration consisting of the sine elements with separate frequency. As in the inverse Fourier transform, the frequency resolution $\Delta f$ of the sine elements is inverse of $T_{\text{win}}$ which ensures the orthogonality. We incorporate the temporary envelope $\delta_s$, the frequency $f_{rs}$ from the $D$ concurrent components into the sine elements. This processing involves oversampling the parameters: resultant is amplitude $\varepsilon_k(n)$ for every element ($n$ is time index). Last, we add the static element and vibration elements together. The reconstructed strain

$$\varepsilon_{\text{reco}}(n) = \sum_{k=0}^{K} \varepsilon_k(n) \cos \left( \frac{2\pi f_k}{f_{rs}} n \right)$$  \hfill (8)

where $f_k = k\Delta f$, and $K$ yields the highest possible output frequency. The cosine function simulates the strain oscillation. As a special case, $\varepsilon_0$ is the static component. This procedure is applicable for both models (3) and (7). Figure 4 expresses the reconstruction quality for the strain and fatigue representation.

![Figure 4](image)

Figure 4. These graphs clarify the strain reconstruction and address the FLE progress. Although the actual FLE is not presented here, the model ability to track the essential cycles is evident.
We do not construct the signal window by window although the both presented vibration models rest on windowing. Overlapping windows having the same frequency may cancel each other due to phase difference between them rather than merge properly. A fundamental idea is to construct every single sine element (with frequency $f_k$) coherently over time. Consequently, the result is realistic and lacks discontinuities in transient states as it is visible in the uppermost graph in Figure 4.

**Experimental results**

The model quality is assessed by comparing original strain gauge signals and model-based reconstructions. The FLE of both is estimated based on standardized design values and related material information available. The stress representation with only the low-pass component is considered as well since it should compare well with the conventional FLE monitoring. Hence the meaning of the vibration becomes evident for the structures concerned. Figure 5 and Figure 6 present FLE ratios depicting the model ability to describe the fatigue information. Value one is the goal. Table I gives the model adjustment parameters. Nine test flights that represent FiAF typical F-18 use are considered. The most salient message is revealed by the rightmost bars of both Figures: the overall quality all the flights included.

![Fatigue life expenditure ratio](image)

**Figure 5.** Figure illustrates the FLE ratio between the modeled and the measured signals. Results from nine test flights are presented. The strain gauge in question is located in the leading edge flap hinge area of an F-18. The rightmost bars depict the FLE ratio of all the nine flights together. RF refers to the rainflow approach, BF to the Fourier approach (Fourier base function), and LP to signal model with merely the static low-pass component.
Results from nine test flights are presented. The strain gauge in question is located in the vertical stabilizer attachment of an F-18. The rightmost bars depict the FLE ratio of all the nine flights together. RF refers to the rainflow approach, BF to the Fourier approach (Fourier base function), and LP to signal model with merely the static low-pass component.

Table I. The rainflow model parametrizing in the experiments.

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{step}$</td>
<td>0.1 s</td>
</tr>
<tr>
<td>$T_{win}$</td>
<td>0.4 s</td>
</tr>
<tr>
<td>$f_{cutoff}$</td>
<td>5 Hz</td>
</tr>
<tr>
<td>$\delta$</td>
<td>$[1, 0.4]^T$</td>
</tr>
</tbody>
</table>

**DISCUSSION**

The results prove that the rainflow model is able to represent the strain vibration acceptably. See Figure 5 and Figure 6. As can be expected, the rainflow model is throughout more competent compared to the Fourier-based model. In the FLE results for the leading edge flap hinge, the overall difference between the rainflow modeled and the measured signal is 16%. Without the vibration (only the static strain), the difference is 29%. For the vertical stabilizer attachment, corresponding readings are 1.9% and 88%. In this case, the need of the vibration model is obvious.
The rainflow model is applied to the neural network-based predictive fatigue modeling, and preliminary results are promising [4]. The similar modeling error as above is 3% for the vertical stabilizer.

The requirements of the model depend on the structural detail in question. The rainflow model adjustment parameters have an effect on the capability to characterize the signal. Windowing affects the correlation between consecutive windows. Thus the critical behavior in transient states is adjustable through $T_{\text{win}}$ and $T_{\text{step}}$. Of course, if model sampling rate modification is restricted, $T_{\text{step}}$ can not be decided freely. But $T_{\text{win}}$ provides some scope for adjustment. The more the successive windows overlap, the higher is the correlation between them. Thus the variation in the model output may be affected which is necessary, for example, with the above-mentioned neural networks. In addition, the choice of the amplitude ranges (bounds) is important. The more there is deviation in the cycle amplitudes within one window, the larger is the number of ranges needed. For applications where only fatigue information is relevant, an assumption of only one dominant component (one range) might be acceptable. Nevertheless in that case, the frequency content is assumed to be limited.

In non-stationary signal analysis, the empirical mode decomposition [6] is a promising method. We tested it with the signal from the vertical stabilizer. Preliminary results were practically identical with the rainflow method though the empirical mode decomposition appeared to be computationally intensive. It seems that these approaches having no strict presumptions about signal shape are effective.

The work continues with the automated fatigue tracking. The connection between a flight maneuver profile and the caused fatigue is one of our future challenges.

CONCLUSION

This work originated from an observation in the FiAF research program: the vibration intensity in certain F-18 structures seemed to be surprisingly high. We analyzed a mass of strain gauge data recorded in a number of F-18 flights focusing mainly on upper part of the spectrum. Our contribution is a novel strain model including the rainflow-based vibration representation. Accordingly, the high cycle fatigue can be considered in the automated fatigue tracking. In general, the model can be used whenever there is a need to describe the complex strain occurrences in a compact way.

ACKNOWLEDGMENT

The authors would like to thank the Finnish Funding Agency for Technology and Innovation, Patria Aviation, the Finnish Air Force Headquarters, and the Nokia Foundation for funding and support.
REFERENCES


Publication III


This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the Tampere University of Technology’s products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this material, you agree to all provisions of the copyright laws protecting it.
Metadata in Sequential Real-Time 2-D Detection

Juho Vihonen, Timo Ala-Kleemola, Juha Jylhä, Riitta Kerminen, Juhani Rauhamaa, Member, IEEE, and Ari Visa, Senior Member, IEEE

Abstract—Extraction of patterns is an important low-level operation in several vision applications. In particular, this work is motivated by the problem of detecting unknown vague structures in images with a minimum sample number. Tracing of an unknown structure is allowed by the spatial 2-D placement of extremum. Often, this permits near-optimum testing of hypothesis, which also accentuates the descriptive value of extrema. The proposed sequential approach guarantees predictable detection delay, which is a significant advantage in real-time use. The performance is demonstrated with simulations and real experiments, where transients have unknown starting time.

Index Terms—Extrema, image processing, sequential detection, shortest-path.

I. INTRODUCTION

S

EQUENTIAL detection methods have been well known for decades. The early work by Marcus et al. [1], among many others from a wide field, essentially credits to the test procedure suggested by Wald [2]. Linescan signal processing compares with 2-D, spatio-temporal image processing, and very general detection approaches are usually adopted. The application of sequential detection methods to linescan has not attracted attention. We show that viable processing of the measurements can be accomplished also with specific line-type design. Often, sequential tests outperform fixed sample-size tests.

The problem of predicting the statistical properties of local extrema arises in some domains. Taking the points where the first derivative of a function vanishes and solving the signs of the second derivative is often a simple way to locate local minima or maxima. Unfortunately, the theory defining the mean density of local extrema (the expected number of local extrema per unit area in 2-D) and their distribution appears to be incomplete in the literature. No widely accepted approach exists. The problem has been addressed, e.g., in [3]–[5].

Despite the lack of any general method for evaluating the extrema distribution, this fundamental signal property is of significant interest. In particular, local extrema are not randomly chosen points but reflect the signal’s geometric structure. Here, the term “metadata” is selected to emphasize this “data about data” relationship. Their sequences can often be considered to consist of a deterministic and a stochastic component. In an elementary case, the observed uncertainty is simply due to system noise. If a magnitude change takes place, the probability of local maxima exceeding some (positive) threshold is higher than that of local minima. Respectively, the local maxima and minima form preferential sample sets for the testing of increase and decrease in signal level.

The starting time of a transient signal is often unknown. In such a case, sequential tests can be favored. Wald’s sequential probability ratio test (SPRT) is optimal in terms of the quickest decision, given some fixed values of error probabilities and simple hypotheses [2], [6]. It is, however, treated here as Page’s test [7], known as CUSUM. Using the lower threshold of zero, Page’s key idea was to restart the SPRT algorithm as long as the alternative hypothesis is rejected. In this way, the algorithm minimizes the worst case delay to detection under a constraint on the average delay between false alarms [8], [9]. Basseville et al. [10] wrap also other results up well.

A variety of problems can be expressed in terms of finding a path through a directed graph. In sequence estimation and optimum path selection, the Viterbi algorithm [11] (VA) has been widely used. In general, a dynamic programming algorithm requires a probabilistic state which can be propagated by a recursion relation from stage to stage. Fundamentally, our problem is to detect defects, but the VA itself seems ill-fitting. As it follows, integrating CUSUM and the local extrema as higher information reduces the suggested method to a Viterbi-type, shortest-path algorithm.

To yield plausible 2-D tracing, Section II extends Page’s method with a spatial, extrema-based linking algorithm. Relevant signal properties are discussed in Section III. Typical behavior of our algorithm is enlightened by showing a connection with conventional CUSUM parameter selection. Case studies are presented in Section IV with two simulated examples and one real-world image taken with a charge-coupled device (CCD) linescan camera. The CCD image has been captured from a papermaking line, revealing the diverseness of defects. As a secondary object, the given examples demonstrate low-level real-time segmentation and defect isolation. Further details are briefly discussed in Section V, and concluding remarks are given in Section VI.

II. SEQUENTIAL 2-D HYPOTHESIS TESTING WITH EXTREMA LINKING

Suppose we have collected a discrete array of data \( D \), where

\[
s_k^n = \begin{cases} D(k, m), & k \geq 0, 1 \leq m \leq N \\ 0, & \text{otherwise} \end{cases}
\]

Finally, the review of this manuscript and approving it for publication was Dr. Malcolm Macleod.

J. Vihonen, T. Ala-Kleemola, J. Jylhä, R. Kerminen, and A. Visa are with the Institute of Signal Processing, Tampere University of Technology, FI-33720 Tampere, Finland (e-mail: juho.vihonen@tut.fi; timoala-kleemola@tut.fi; juha.jylaha@tut.fi; riitta.kerminen@tut.fi; ari.visa@tut.fi).

J. Rauhamaa is with the ABB Oy, Process Industry, Web Imaging Systems, FI-00381 Helsinki, Finland (e-mail: juhanni.rauhamaa@fi.abb.com).

Digital Object Identifier 10.1109/LSP.2006.888427
and spatial coordinates \( k, m \in \mathbb{Z} \). Let \( s_k^m \in \mathbb{R} \) and let \( k \) denote the time index. Then, we have a measurement vector

\[
s_k = [s_k^1, \ldots, s_k^N]^T.
\] (2)

The two hypotheses are defined as

\[
H_0 : \theta = \theta_0 \\
H_1 : \theta = \theta_1
\] (3)

that denote some appropriately chosen quantities, which parameterize the different process states. Let the logarithm of the likelihood be defined by

\[
g(x) = \ln \frac{p_{\theta_1}(x)}{p_{\theta_0}(x)}
\] (4)

where \( p_{\theta_0} \) and \( p_{\theta_1} \) represent the probability density functions of the random variables under two distributions. They describe the previous hypotheses. Suppose that \( \{S_k^m; k \geq 0\} \) is a process with independent increments related to the \( m^{th} \) component of \( s_k \). Let \( t_a \) denote the alarm time so that

\[
t_a = \arg \min_k \{S_k^m \geq h\}.
\] (5)

A detection is declared when \( S_k^m \) exceeds the threshold \( h \). Let \( s_k^1, s_k^m, \) and \( s_k^n \) be successive local extrema of \( s_k \) in order such that every local maximum is followed by a local minimum and vice versa. This compresses \( s_k \) and yields the discussed metadata-concept. The edges of \( m^{th} \) extremum are the preceding \( l^{th} \) and ensuing \( n^{th} \) extremum indices, i.e., \( m \in (l, n) \). The first and the last values of \( s_k \) may also represent edges. We formulate the linking for the local maxima as follows. For each measurement vector, new tests are initialized and existing ones are updated. The tests merge or diverge for new paths

\[
S_k^m = \max \left( 0, \max_i \left\{ S_{k-1}^i + g(s_k^m) \right\} \right), \quad \forall i \in [l, n]
\] (6)

where \( s_k^m \) is local maximum. The statistic \( g(s_k^m) > 0 \) assigns the initial value for \( S_k^m \), if \( \{S_k^m\} \equiv \emptyset \). The progression is of random-walk type, and a generalization for the local minima is obvious, e.g., by switching the sign of data.

The proposed algorithm’s one-dimensional (1-D) interpretation as CUSUM with varying window-size, which is chosen according to the behavior of the observations of the entire past, applies exactly after local extrema linking. When \( H_1 \) is true, the best 2-D route minimizes the delay of detection. No paths need to be preserved or searched backward, which is typical in the VA. Instead, the number of branches may be indefinitely large, since the VA-like probabilistic state is propagated through the CUSUM algorithm. This requires only single output per the traced extremum at a time instant, namely \( S_k^m \).

III. STATISTICS AND DELAY ANALYSIS FOR CCD

The following treatment is given in the context of our linescan application. We start with some light intensity issues and move on to the selection of decision statistic and parameters.

A. Sufficient Statistics

A CCD cell receives radiation from the object surface and converts it to a scene. Inherently, photon arrival and whether it produces a charge or not is random. The photodetector conversion of optical photons to signal charges may follow a Poisson process, which introduces a measurement variance proportional to the signal magnitude. On the other hand, amplifier and circuit noise determines the noise floor of the device at low signal values. By Healey et al. [12], the intensity of a pixel \((k, m)\) is a random variable

\[
D(k, m) = \mu(k, m) + N_F(k, m) + N_C(k, m)
\] (7)

where, as discussed above, \( N_F(k, m) \) relates to the signal \( \mu(k, m) \) but \( N_C(k, m) \) does not. When the light intensity is proper for high-resolution scene capturing, the Poisson noise \( N_F(k, m) \) dominates \( N_C(k, m) \) and it approaches a Gaussian distribution. Rice [13] has derived the distribution of maxima in a 1-D Gaussian process. He showed that the spectral properties contribute and the distribution shape is similar to Rayleigh or approximates a Gaussian. The latter prevails here, since \( N_F(k, m) \) plays such a significant role. Small, hardly eye-catching transients are very background-like.

The statistic \( g(x) \) can be written as

\[
g(x) = \frac{1}{\sigma} \left( x - \mu_0 - \frac{\nu}{2} \right)
\] (8)

when the distributions under both hypotheses can be approximated Gaussian with mean values \( \mu_0, \mu_1 \), and constant variance \( \sigma^2 \). Setting \( \sigma^2 \) constant is possible under invariable intensity, and

\[
\nu = \mu_1 - \mu_0 \quad (\mu_1 > \mu_0)
\] (9)

implies the change magnitude. Useful details can be found in [10] and [14] as well, where a fine treatment of Poisson processes is given close to this context. Appealingly, \( \nu \) can now be chosen according to minimum magnitude of interest, which seems convenient in contrast to, e.g., photon hit-rates.

B. Approximation Effects and Parameter Selection

In what follows, the solid curve in Fig. 1 represents the “true” CUSUM behavior of the average run length (ARL). It is computed with Siegmund’s method having Gaussian statistics [10, 15, pp. 186–189]. The almost exactly overlapping dash-dot line represents simulated ARL values using the local maxima drawn in random order from the standard Gaussian 1-D signal. The x-axis symbol \( \mu \) denotes the mean value of the increment of the decision function, i.e., \( \mu = E[g(x)] \).

As one may note, the general deviation from the Siegmund’s values is here clearly negligible. By its nature, the Gaussian approximation decreases the ARL for a negative drift. This may be worth remembering while assessing the mean time between false alarms, especially with low \( h \) values. Note that the overall 2-D false alarm interval is dependent on the number of local
Because of some supplementary manipulation, we do not address it for now.

IV. SMALL-SIGNAL JUMP DETECTION AND EXPERIMENTS

The following figures present the original image and the paths that maximize $S_k^{m}$ and develop detection. The originals on the left are processed recursively line-wise from up to down, which mimics linescan. On the right, a string of extrema after restart to alarm time $t_a$, is sketched with the gray color. The black represents the path after $t_a$ to the estimated transient end. The first time a detection is declared, we do not stop the observation nor restart a new cycle. The execution is continued until the transient ends. If the statistic $S_k^{m}$ exceeds $h$ after a cycle, the simple stopping rule is $g(s_k^{m}) < 0$.

These results are obtained for $h = 25$ and $\nu = 4\sigma$. In the real CCD case, the standard deviation is estimated using the first few lines. Figs. 2 and 3 are simulated test images. They present two different signals in homogeneous backgrounds, but the latter is more pragmatic. For both cases, $\mu_0$ and $\sigma$ are fixed as in Section III. Note the spatial non-smooth paths: local minima or maxima are linked line-by-line, free of spatial scale. No false alarms occur and, in fact, solving $x$ in (8) with the known jump magnitude reveals that the chosen threshold is conservative at least in off-line sense. Fig. 4 presents a practical case where the maxima of $S_k$. Because of some supplementary manipulation, we do not address it for now.

Fig. 2. Simulated ring and x-pattern in additive white $N(0,1)$ Gaussian noise. Their widths are one and three pixels, respectively, with magnitude of $\pm 3$. The upper and lower ring parts represent the worst case defect orientation for the algorithm. Owing to up-to-down asymmetry, they remain undetected.

Fig. 3. Previous Fig. 2 after 2-D low-pass filtering and addition of $N(0,1)$ white noise. The filter does simple $3 \times 3$ spatial neighborhood averaging, leaving the background and the patterns similarly floculent. Paths with inferior statistic are discarded. So the “x” appears incomplete, but there is only minor difference in $t_a$ compared to Fig. 2.

Fig. 4. High-resolution linescan CCD image visualizes a badly wrinkled paper surface. Though adaptation is simple and uses a single scan-to-scan updated $\mu_0$, smooth defect tracing with fast detection characterize the performance.

CCD camera is placed above a fast moving paper web. Interestingly, despite the nominal patchiness of the paper surface data, the extrema align, scan-by-scan, randomly with uniform density. Sketched defect paths reveal heavily structured extrema-patterns. The large defect on the middle-left is traced also well, although the shape of the object under view is diversiform and unknown. This observed positioning is crucial in terms of simplified physical surface description. In consequence, off-line revision of the hypotheses and possible reforming of $g(x)$ could spring to mind.

V. DISCUSSION

The above test images illustrate circumstances where satisfactory performance of several optional techniques can not be guaranteed. For instance, the blurred “x” in Fig. 3 became undetectable to ICOV-algorithm [16], which utilizes Laplacian and gradient (Sobel) operators. Matched filters produce optimal results with known signals. However, as the scene sharpness degrades gradually from the centre towards the edges (especially due to the “cosine-to-the-fourth” law), the optics-related mismatch may become substantial. In addition, the real-time requirement aggravates the task even further.

Simplified description of subtle transients can enhance practical 2-D detection and tracing performance. In the presented approach, the spatially displaced local maxima or minima are linked in time. Note that the best path is traversed, e.g., in ridges and ravines. In this sense, the proposed linking resembles...
template matching in its accuracy. Line-wise merging may also become inefficient in comparison with defect's size, which perhaps delays alarming. Even though the detection is inevitable, spatially very large transients (Fig. 4) do not represent the small signals for which the method was designed for. Obviously, off-line processing is not limited to the vertical direction. With horizontal processing the points become orthogonally selected (Figs. 2 and 3). In real time, the statistic $S^R_k$ is easily coupled with a horizontally developed statistic having the same coordinates.

The line-wise positioning of the local extrema in the CCD scene is in constant change. An interested reader may verify this and the fluctuating intensity of the surface by placing a bright source of light behind a blank sheet of copying paper. Capturing this type of surface profile requires high resolution, wide system dynamics, and ample number of bits. The local extrema line-up along darker and lighter spots, where also the algorithm tends to traverse. In our case, embedding nominal surface variation to the parameter $\nu$ is pure simplification of a complex optical process. Nevertheless, the smallest changes are stochastic, and taking action requires permanent observation.

It is easy to modify the proposed algorithm to be more suitable for spatially large defects by allowing free merging and accordingly summing the decision function. However, the algorithm’s sensitivity increases dramatically which requires techniques to limit overwhelming false alarms. The possibility of open cycle continuation to infinity seems tolerable as well, since ample data throughput typically characterizes linescan processing. Using neighboring extrema as spatial edges is very likely beneficial in further segmentation. This work continued the earlier work [17], which only scratched the topic. Here, we generalized the CUSUM algorithm from 1-D to 2-D in Viterbi-like fashion.

VI. CONCLUSION

Sequential detection with shape description through local extrema linking was found reliable and competitive, which is our main contribution. If compared to block-based approaches, a sequential change detection scheme, if available, is more attractive. This is due to its easy implementation, performance gain,1 and robustness to the unknown starting time of the transient signal. Theoretically sound, deliberately deteriorated, and real test cases were studied, and indication of optimal path selection for small curvilinear defects was observed. A fractional part of the real data were also local extrema for which a handful of tests were running at a time and required further attention. Like CUSUM, the proposed algorithm is extremely simple. They also share statistical properties. However, it is certainly up to the designer to refine, e.g., the decision function or stopping criterion up to a level which is sufficient for his/her requirements. It is likely that fields that require precise registration (e.g., retinal vessel tracing in medical image analysis) find the approach useful.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful comments and suggestions.

REFERENCES

Publication IV

Directional Filtering for Sequential Image Analysis

Juho Vihonen, Juha Jylhä, Timo Ala-Kleemola, Marja Ruotsalainen, Jarmo Kauppila, Tommi Huotilainen, Juhani Rauhamaa, Member, IEEE, and Ari Visa, Senior Member, IEEE

Abstract—Image sequences produced by, e.g., electro-optical sensors are being used increasingly for continuous quality monitoring and surveillance. This letter considers the important problem of multiframe change point detection of signals with unknown dynamics. We propose a new CUSUM-based (Cumulative SUM) detector for detecting noise-obsured signals in a sequence of frames. Our sequential detector adopts directional filtering, a search philosophy of interesting pixels, of state-of-the-art detector [1] that employs dynamic programming (DP) for so-called "track-before-detect" (TBD) processing. In case of a persistent change, a detailed comparison reveals our CUSUM-based detector superior to the DP-based TBD detector.

Index Terms—Dynamic programming, image analysis, sequential detection.

I. INTRODUCTION

SUCCESSES in the sequential change detection applied to image processing are not numerous. However, the results of earlier works for detection of weak signals, so-called "track-before-detect" (TBD) techniques [1]–[8] restricted to surveillance, allow development of very efficient sequential detectors to other fields of imaging, too. In this letter, dynamic programming (DP) combined to sequential detection is shown to be of considerably more value in multiframe change detection than has been recognized. We have restricted our studies to grayscale line frames.

A single frame is often insufficient for detection of a noise-obsured signal. Provided that an interesting signal occupies the same pixel in each frame, detection can be enhanced by summing up a group of frames pixel-by-pixel. If an interesting signal occupies a different pixel in each frame, the same result is obtained by "aligning" frames appropriately prior the summation; see, e.g., [8, Fig. 4, p. 240]. To search these pixels for enhanced detection, a DP-based TBD [1], termed here the DP-TBD detector, uses a simplified version of the Viterbi's DP algorithm [9] applied over a fixed number of frames.

The CUSUM (Cumulative SUM) algorithm [10] in its optimum form detects a change point (in this work, a jump in signal magnitude) with a minimum average delay under constraint on the frequency of false alarms; see [11]–[13]. These so-called average run-lengths (ARLs) define the CUSUM performance, as the Neyman–Pearson rule is defined with the aid of false alarm and detection probabilities using numerical decision boundaries; see, e.g., [14] and [15]. To sum up, the time between triggered detections is stochastic, but the CUSUM is optimal if tuned properly.

Imaging sensors consist of sensing elements (pixels) organized in either a 1-D vector or a 2-D matrix grid, which defines the maximum dimensions of a frame. However, no matter the frame dimensions, the Viterbi's DP algorithm [9], as used in [1], finds recursively the most likely sequence of interesting pixels. For time series—to which the DP algorithm reduces the pixels of a sequence of frames—the CUSUM is optimal change point detection procedure. The purpose of this letter is in this way to apply the CUSUM to multiframe sequential change detection and to compare its pros and cons to the near-optimum DP-TBD detector of fixed sample size.

Next, operation of the CUSUM and DP-TBD detector are shortly reviewed. Then, our new CUSUM-based detector is introduced. In Section III, detection performance of the new CUSUM-based detector is compared with the DP-TBD detector using ARLs, receiver operating characteristic (ROC) analysis [16], and an example consisting of 1-D vector frames to mimic our imaging application [17], which also makes the graphical presentation easier than achievable by 2-D frames. Concluding remarks are made in Sections IV and V. This letter continues our earlier work [18], but we postpone reviewing the results up to the point where they become relevant.

II. MULTIFRAME DP FOR CUSUM

In change point detection, we wish to decide between the following two hypotheses as quickly as possible:

\[ \begin{align*}
H_0 : \theta &= \theta_0, \\
H_1 : \theta &= \theta_1
\end{align*} \]

(1)

where \( \theta_0 \) and \( \theta_1 \) parameterize the different states of a stochastic process. If both probability density functions (PDF) \( p_{\theta_0}(x) \) and \( p_{\theta_1}(x) \) are known and observations \( x \) are i.i.d.,¹ the Neyman–Pearson decision function, \( p_{\theta_0}(x)/p_{\theta_1}(x) \), yields the optimum CUSUM statistic; see, e.g., [14]. However, when operating with intermediate amounts of information, one can write \( g(x) = x - \nu \) for the decision function, where \( \nu \) is a reference value used to identify positive deviations of \( \nu (\nu \geq \mu) \), the mean of \( x \). We prefer using this form as an accurate knowledge of the noise \( \theta_0 \) or signal statistics \( \theta_1 \) is required. The CUSUM statistic is defined as \( S_k = \max\{0, S_{k-1} + g(x)\} \), where \( S_0 = 0 \). We refer to \( k \) as time. If a change from \( \theta_0 \) to \( \theta_1 \) occurs at some point, quickest detection is described by the alarm time \( t_a \) and stopping rule with the threshold \( h \)

\[ t_a = \min\{k | S_k \geq h\} \]

(2)

¹The abbreviation i.i.d. stands for collection of independent and identically distributed random variables.
so that \( \{ S_k; k \geq 0 \} \) is a stochastic process with independent increments.

Suppose each pixel at each frame has a region (e.g., \( 3 \times 3 \); see [1, Fig. 2, p. 1442]) of so-called candidate pixels at the previous frame. The heart of DP as used in [1], is that the pixel-wise “scoring” sums at each frame depend only on the maximum of the scoring sums of these candidate pixels. A scoring sum \( I_n \) of a pixel \( x_n \) at the \( n \)th frame is then

\[
I_n = \max \{ I_{n-1} \} + x_n
\]

(3)

where \( I_0 = 0 \). The DP scoring [1], at time \( k \), over a group of \( K \) frames is simply a summation (3) up to the \( k + K - 1 \)th frame with \( I_{k-1} = I_0 = 0 \). If \( I_{k+K-1} \) exceeds a threshold \( h_{DP} \), the DP-TBD detector declares decision \( H_1 \).

To find the pixels most likely occupied by the signal, one can certainly backtrack and so “revert” the end-frame sums to pixels of the first frame of the group. In Fig. 1(a), a large score sum (3) is backtracked to a pixel \( y_k \).

Consider that each pixel at the \( k \)th frame has a CUSUM statistic \( S_{k-1} \). For quickest detection, the definition (2) denotes choosing a pixel of the \( k \)th frame with the largest score sum (3) within the candidate region for each \( S_{k-1} \). With the aid of DP in Fig. 1(a), a CUSUM statistic \( S_k \) of the pixel \( y_k \) at the \( k \)th frame is then

\[
S_k = \max \{ 0, \max \{ S_{k-1} \} + g(y_k) \}
\]

(4)

where \( \max \{ S_{k-1} \} \) denotes the maximum of CUSUM candidates; see Fig. 1(b). Since the change point is unknown, other pixels (that have small score sums) restart new tests (4) so that \( \max \{ S_{k-1} \} = S_0 = 0 \) whenever a new frame is received, the DP scores are simply revised for (4). For obvious reasons, the algorithm is called the DP-CUSUM.

Because of the small \( 1 \times 4 \) “frames” compared to the \( 1 \times 3 \) candidate region, a single test (4) occupied Fig. 1(b). If the candidate region was, say, \( \ell \)-times smaller than frame dimension, it follows that the \( k \)th frame is always occupied by about \( \ell \) tests (4) cumulated ever since the first frame. These \( \ell \) tests can incorporate the whole measurement history.

III. ASSESSING PERFORMANCE

In the studies to follow, a detailed performance comparison is given. For concrete benchmarking, a change that occupies only a single pixel per frame is assumed. To begin with, assume that we receive a sequence of 1-D line frames (vectors) each consisting of 50 elements. As it is in our imaging application (see [17]), this sequence forms a “normal” 2-D image. The image is assumed to contain thermal i.i.d. Gaussian \( N(0,1) \) noise at state \( \theta_0 \) as basically any image can be transformed into such by appropriate prewhitening or local mean removal. As is the natural way with the DP procedure, we assume fixed candidate region of \( 1 \times 9 \) pixels (\( j = 9 \)), and ten frames (\( K = 10 \)) is also sufficient in a variety of cases; see [1]. Next, we first acquire the PDF of \( y_k \) for a test (4) at the noise-only state \( \theta_0 \). The result is then adopted to \( \ell \) simultaneous tests to yield so-called run-length probability mass function (PMF) along with the cumulative mass function (CMF) necessary for ROC analysis of the DP-CUSUM.

A. Obtaining Run-Length Distribution at \( \theta_0 \)

For \( K = 1 \) and \( j \) i.i.d. random variables having the PDF \( f(x) \) and the cumulative distribution function (CDF) \( F(x) \). PDF of the pixel \( y_k \) in (4) is easily shown to be \( f_{\ell,j}(x) = \int [F(x)]^{j-1} f(x). \)

(4)

For any \( K > 1 \), the PDF renders the PDF of \( y_k \) difficult to determine precisely. However, to account for \( K > 1 \), introducing a small positive term \( \phi \) so that \( y_k \sim f_{\ell,j}(\phi) \) appears sufficient: through trial and error, selecting \( \phi = 2 \) is quickly verified accurate for \( K = 10 \) as we shall next show. Once the PDF of \( y_k \) is presumed known, the CUSUM algorithm’s run-length PMF \( p(x) \) and CMF \( P(x) \) can be obtained using a trouble-free technique presented in [19]. The ARLs are obtained by computing the mathematical expectation of \( p(x) \). Consequently, the run-length PMF of the quickest detection (2) for \( \ell \) tests (4) can be given through relation \( p_{\ell,\phi}(x) = \ell [1 - P(x)]^{\ell-1} p(x) \).

The proof is simple and thus omitted. Here, each test (4) occupies a space of nine pixels and since each vector frame contains 50 samples, the number of tests (4) is \( \ell = 50 \times 9 = 550 \). To describe a “minimum” interesting intensity of a pixel, let \( \nu \) be equal to 1.75.

For \( K = 10, j = 9, \phi = 2, \ell = 550, \) and \( \nu = 1.75 \), the ARLs at \( \theta_0 \) are shown in Fig. 2 as a function of the threshold \( h \). After validating the ARLs with simulation as presented, the approximated PMF can be concluded to be accurate enough.

B. Comparison in ROC Space

A ROC graph depicts the balance between the probability of detection \( P_D \) and the probability of false alarm \( P_{FA} \) among all samples available [16]. In ROC space, the (0,1) point is called the perfect classification; i.e., no uncertainty is involved in deciding \( H_1 \). Such situation is generally unreachable and, in practice, \( P_D \gg P_{FA} \) for a detector to be useful.

Analysis in [20] was adopted to produce the ROC graph for the DP-TBD detector in Fig. 3(a). Being too low for reliable detection with a single frame, \( A \) is the amplitude of an interesting signal. Note that \( A = 0 \) at the noise-only state \( \theta_0 \). Since the DP-TBD does not incorporate measurement history, the both \( P_D \) and \( P_{FA} \) are frame-wise; see [20]. To go beyond the shown trends for the case \( A = 1.85 \) plus \( N(0,1) \) noise, the DP-TBD attains \( P_D \approx 0.2 \) for \( P_{FA} \approx 0.001 \) while \( h_{DP} = 22.5 \) denoting that approximately 20% of the pixels would be recovered with 0.1% risk level. However, since the DP-CUSUM does not a remark, the cited literature contains plenty of applicable methodology for detection of multiple signals, signal disappearance, and segmentation.

3 \times 3 \) region is equivalent for a sequence of 2-D frames (matrices); see [1].

4 For \( K = 1 \), the DP is redundant and the search becomes one of choosing a large pixel from noise.
incorporate the measurement history, a ROC space in Fig. 3(b) shows the cumulative balance between the true and false $H_1$ decisions as new frames are processed: $P_{\text{fa}}$ and $P_{\text{d}}$ are expressed as function of run-length (rl) and thus denoted by $P_{\text{ful}}$ and $P_{\text{dul}}$. With the same $P_{\text{ful}} \approx 0.001$ (that corresponds to $1/1313$, the inverse of the ARL at $\theta_0$ in Fig. 2 for $h = 7$) as above, e.g., $P_{\text{ful}} = 0.1$ denotes that 10% of the false alarms would occur not later than 163 frames. For $A = 1.85$ plus $N(0, 1)$ noise, this corresponds to $P_{\text{ful}} \approx 0.99$ denoting that 99% of the detections would occur within the same 163 frames. In fact, to go further beyond the shown trends for $A = 1.85$, 50% of decisions $H_1$ would occur not later than 32 frames with lower than 1% risk being a false alarm. In other words—by incorporating the whole measurement history—the DP-CUSUM can achieve $P_{\text{d}}$ near unity with the same $P_{\text{ful}}$ as the DP-TBD, but at worst, this comes at the cost of a much larger number of frames integrated. Compare also the ARLs in Fig. 3(b) for an easier insight.

Indeed, achieving $P_{\text{d}} \approx 0.2$—using the words of Tonissen and Evans [1, p. 1444]—“highlight the rapid performance degradation for $A < 2$” of the DP-TBD detector.

The results in Fig. 3(b) do not incorporate effects of the DP algorithm finding and a test (4) integrating a mix of pixels at $\theta_0$ and $\theta_1$, though such is bound to happen in high noise. In spite of the DP enabling near-optimum estimation, particularly for $A \to 0$, the analysis loses its comparative simplicity and one should be somewhat skeptical of the curves.

C. Comparison With a Pattern Having Unknown Dynamics

The leftmost image in Fig. 4(a) shows our simulated test pattern with $A = 1.85$ embedded in i.i.d. $N(0, 1)$ noise. Details are not easily visible to the naked eye, and for clarity, the occurrence of the embedded pattern is illustrated on the right in Fig. 4(a). The image is processed row-wise from up to down. If a score sum (3) exceeds $I_{\text{DP}} = 22.5$ after processing ten rows, the pixel $y_k$ (see Section II) is shown in black in Fig. 4(b). Though this processing is repeated at each row, the DP-TBD recovers roughly 20% of the pixels. In Fig. 4(c), the DP-CUSUM alarms with much better recovery of $\theta_1$ pixels; the cumulated pixels before a detection are in gray, and the last pixel causing the detection is in black. After a detection, the whole algorithm is restarted, i.e., each $S_{k-1}$ is set to zero as in (2) to meet the behavior presented in Section III-B.

Note the short distances between black dots in the upper part of Fig. 4(c). Exactly as predicted in Section III-B, the probability of early detection is high (remember that 50% of correct $H_1$ decisions occur not later than 32 frames). Hence, assessing the quality of $H_1$ decisions in function of the frames processed can prevent effectively bias in any subsequent higher-level analysis such as, e.g., ignoring false alarms. Except the very general $P_{\text{ful}}$ and $P_{\text{dul}}$, the DP-TBD incorporates no such cumulative knowledge.

IV. DISCUSSION

Our CUSUM-based algorithm integrates pixels according to a DP-based search that recovers the most likely $\theta_1$ pixels sequentially using some $K$ frames at a time. For a change having a finite change time, a matched filter [21]—to which the DP-TBD in its adaptivity can be compared with—is optimum. When the change is persistent but small, a test based on a fixed sample size is less effective [22] and, if tuned with the true pre- and post-change distributions, the CUSUM minimizes the average delay to detection. These optimum characteristics are well complementary, however. By sharing the DP-procedure, simultaneous dual-detection with both the DP-TBD and DP-CUSUM detector is very cost-efficient. As discussed next, one surely seeks something simple, too, in TBD.

The penalty associated with the DP algorithm is in approximately $\rho^2(K - 1)$ increasing computation; i.e., the maximization procedure for a $I_{k+K-1}$ is performed $K - 1$ times with no
The algorithms are by no means exclusionary: in dual-operation, their optimum characteristics can complete each other very well. Hopefully, this work both renews attention to DP as well as highlights the benefits of sequential change detection in image processing.

REFERENCES


VIHONEN et al.: DIRECTIONAL FILTERING FOR SEQUENTIAL IMAGE ANALYSIS

Fig. 4. (a) Noisy and the actual pattern, $A = 1.85$. (b) Detected pixels using the DP-TBD, $K = 10$, $h_{DP} = 22.5$. (c) Detected pixels using the DP-CUSUM, $K = 10$, $\nu = 1.75$, $h = 7$. More than $g$ allowable candidates for each of the $g$ pixels. With a naive exhaustive search, which a DP algorithm substitutes efficiently, the complexity $f^{K-1}$ is significantly higher. However, as the exhaustive search is conducted whenever a new frame is received, the chance of recovering the true pixels at $\theta_1$ is also higher. The key idea of our early work [18], like outlined in the title, was to utilize “data about data” referring to local maxima. By virtue, these pixels maximize the DP scores and, since only 33% of data were local maxima, the complexity of the exhaustive search is reduced to about $0.33f^{K-1}$ and, by analogy, the DP to $(0.33)^2(f - 1)$ without adverse effect on $P_{T1}$ or $P_{FN}$. Making the result very attractive, all TBD techniques have high computational demands as their $P_{T1}$ depends crucially on the quality of the search. To leave no room for speculation on the best achievable search performance for $K = 10$, however, the both detectors utilized here exhaustive search as such a search becomes feasible via local maxima. To close up, adapting $K$ to “as small as necessary” and, as suggested by a reviewer, to remove outliers of the DP-based search, e.g., with particle filtering [23] might allow more efficient and robust implementation.

V. CONCLUSION

This work combined the conceptually advanced Viterbi’s DP to the simple CUSUM algorithm for sequential multiframe detection of badly obscured signals otherwise being in risk of remaining undetected. Representing state-of-the-art benchmark, the proposed fully sequential DP-CUSUM was compared with the DP-TBD detector [1], which verified statistically convincing potential and high performance of our novel solution.
Publication V

Cumulative sum and neural network approach to the detection and identification of hazardous chemical agents from ion mobility spectra

Marja Ruotsalainen\textsuperscript{a}, Juho Vihonen\textsuperscript{a}, Juha Jylh\text{"a}\textsuperscript{a}, Jarmo Kauppila\textsuperscript{a}, Osmo Anttalainen\textsuperscript{b}, and Ari Visa\textsuperscript{a}

\textsuperscript{a}Tampere University of Technology, Department of Signal Processing, P.O. Box 553, FIN-33101 Tampere, Finland; \textsuperscript{b}Environics Oy, P.O. Box 349, FIN-50101 Mikkeli, Finland

ABSTRACT

The detection and identification of hazardous chemical agents are important problems in the fields of security and defense. Although the diverse environmental conditions and varying concentrations of the chemical agents make the problem challenging, the identification system should be able to give early warnings, identify the gas reliably, and operate with low false alarm rate. We have researched detection and identification of chemical agents with a swept-field aspiration condenser type ion mobility spectrometry prototype. This paper introduces an identification system, which consists of a cumulative sum algorithm (CUSUM) -based change detector and a neural network classifier. As a novelty, the use of CUSUM algorithm allows the gas identification task to be accomplished using carefully selected measurements. For the identification of hazardous agents we, as a further novelty, utilize the principal component analysis to transform the swept-field ion mobility spectra into a more compact and appropriate form. Neural networks have been found to be a reliable method for spectra categorization in the context of swept-field technology. However, the proposed spectra reduction raises the accuracy of the neural network classifier and decreases the number of neurons. Finally, we present comparison to the earlier neural network solution and demonstrate that the percentage of correctly classified sweeps can be considerably raised by using the CUSUM-based change detector.

Keywords: Gas detection, gas identification, cumulative sum, neural network, ion mobility spectrometry

1. INTRODUCTION

The desire for protecting people from hazardous chemical agents has created a need for gas identification systems that continuously monitor the environment. Typical gases to identify are chemical warfare agents (CWAs) and toxic or explosive industrial compounds. Because in critical situations, human lives are in danger, an identification system should give early warnings and identify gases reliably. In addition, false alarm rate should be low. These requirements should be fulfilled despite diverse environmental conditions and gas concentrations. Reliable gas identification is a challenging task, and it requires well designed gas detection sensors as well as proper processing of measurement data. For assisting the identification, many gas detector devices provide measurements on environmental quantities, such as humidity and temperature.

Typical technologies used for detection of hazardous gases are various semiconductor microsensors,\textsuperscript{1} micro-electro-mechanical-system (MEMS)\textsuperscript{2} or surface acoustic wave (SAW)\textsuperscript{3} sensors, electro-chemical (EC)\textsuperscript{3} sensors, and ion mobility spectrometry (IMS).\textsuperscript{4} Semiconductor sensors react to the exposure of chemical agents by

\begin{itemize}
    \item Marja Ruotsalainen: E-mail: marja.ruotsalainen@tut.fi
    \item Juho Vihonen: E-mail: juho.vihonen@tut.fi
    \item Juha Jylh\text{"a}: E-mail: juha.jylha@tut.fi
    \item Jarmo Kauppila: E-mail: jarmo.kauppila@tut.fi
    \item Osmo Anttalainen: E-mail: osmo.antalainen@environics.fi
    \item Ari Visa: E-mail: ari.visa@tut.fi
\end{itemize}
resistance change, MEMS- or SAW-based sensors are sensitive to adsorbed gas mass, and EC-sensor response is based on chemical reactions in sensor’s electrolyte. As an identification technique, IMS is founded on the fact that ion mobility of various gases in the electric field is different. Diverse ion mobilities yield to compound-specific ion mobility spectra, which can be used as a kind of “signature” for chemical agents. A conventional IMS technique is a drift tube. In the drift tube IMS, gas ions flow through the drift region, which has a constant uniform electric field, and a detector plate at the end of the region collects the charges of ions. Then, an ion mobility spectrum is generated using detector current and the drift times of the ions. An alternative IMS technique is an aspiration condenser IMS, where the gas flows through the so called aspiration condenser cell. That cell contains condensers, each of which create an electric field, and detector plates, which collect the ions and generate output signals. This paper concentrates on a swept-field aspiration condenser type IMS prototype. The term “swept-field” refers to a technique, where an ion mobility spectrum is generated by sweeping bias voltage. Then separate collector plates for every mobility range are not needed. As an output, the device produces an ion mobility spectrum in every sweep. In short, aspirating IMS technology makes possible to manufacture gas detector devices that are sensitive, reliable, portable, and inexpensive.

There is some earlier work concerning the identification of gases using aspiration condenser type IMS. Many of them are written from the sensor technology point of view, and they introduce ways to improve identification by further developing sensors and measurement methods. Pattern recognition is considered, for example, in the works of Vuorimaa et al. and Ilonen et al. Vuorimaa et al. have applied their fuzzy self-organizing map for the detection of chemical agents, and they have compared their approach to the nearest neighbor and learning vector quantization methods. Ilonen et al. on the contrary, have tested the performance of a neural network, a Bayesian classifier, and a decision tree for the identification of CWAs. These two works use an aspiration condenser type IMS device that has static bias voltage so it is not similar to our device. A swept-field IMS has been used by Rosenblatt et al. They have applied a neural network to identify anesthetic agents from ion mobility spectra. According to their preliminary study, a neural network seems to be a very potential categorizing method for that purpose. Earlier pattern recognition related works are mainly concentrated on the categorization of gases. Research on the change detection of ion mobility spectra has been minor even though it is generally important part of identification systems, e.g., in industrial monitoring applications.

Our gas identification system consists of two main blocks: detection and identification (Figure 1). The detection block continuously monitors the ion mobility spectra produced by the IMS device and reports changes from the normal state, i.e. the possible exposure of a hazardous chemical. Using a CUSUM-based change detector, we get the estimate of exposures duration and we are able to select the changed ion mobility spectra for the further analysis. In the identification block, the changed spectra are categorized into predefined classes which represent the potential agents. Data from the environmental sensors is exploited to make identification of agents precise in varying conditions. We have further developed the neural network approach presented by Rosenblatt by using principal component analysis (PCA) to reduce ion mobility spectra before the categorization. This preprocessing raises the accuracy of the neural network.

Section 2 considers the aspiration condenser type IMS prototype, that has been used in our measurements, and describes the measurement data. Section 3 introduces the use of the CUSUM algorithm in our detection problem. Section 4 presents our neural network classifier and explains how we reduce ion mobility spectra with PCA. Section 5 documents the comparison of our neural network approach to the one without PCA and illustrates the positive impact that change detection has on the classification results. Sections 6 and 7 give discussion and the conclusions, respectively.

2. SWEEPT-FIELD ASPIRATION CONDENSER TYPE IMS PROTOTYPE AND MEASUREMENT DATA

This section introduces the measurements discussed in this paper. As a measurement device, we have used a swept-field IMS prototype by Environics Oy. Figure 2 illustrates its principal structure. The main components of the device are an ionizer, a reference electrode pair, bias plate, and two detector plates. The first detector plate is called a front electrode, and the second a collector. The bias plate is located on the other wall of the detection cell and the front electrode as well as the collector on the other. An electric field between the plates is produced by a voltage difference, and its magnitude can be modified by changing bias voltage $V_b$. At the
Figure 1. Our gas identification system. A gas sample is measured using ion mobility spectrometry and various environmental sensors. The detection block continuously monitors the measured ion mobility spectra and reports changes that indicate a possible exposure of hazardous chemicals. The changed ion mobility spectra are then reduced and categorized. In categorization, data from the environmental sensors is exploited to make identification of agents precise in varying environmental conditions.

Figure 2. The principal structure of the swept-field aspiration condenser type IMS prototype (taken from Anttalainen et al.\textsuperscript{13} with permission). At the beginning, sample gas is ionized. Then, ions formed flow through the electric field. They end up on either of the detector plates or flow straight through the cell, depending on their mobility $k$. Only the latter detector plate is used for making measurements while the front plate works as a filter. A sweep of the bias voltage $V_b$ provides the assessment of the whole ion mobility spectrum.
beginning of a measurement, sample gas is ionized by the ionizing source. Formed ions flow through the electric field. They end up on either of the detector plates or flow straight through the cell depending on their mobility $k$. Only the collector is used for making measurements while the front electrode works as a kind of filter that captures ions of high mobility and thus limits the range of ions that can land on the collector. Measurements of different parts of the ion mobility spectrum can be made by adjusting the electric field. The reference electrode pair is used to create a signal from unionized sample. Finally, this signal is subtracted from the actual signal measured by the collector, and the difference is given as an output.

In the presented IMS prototype, the bias voltage is swept over a predefined set of voltages resulting in a set of measurements from the collector at each sweep. Figure 3 shows an example of four sweeps representing measurements of AIR and DIMP in two relative humidity conditions, 10% and 86%. In X-axis, there is bias voltage and in Y-axis collector plate current. The left side of the figure refers to ions of negative polarity and the right side to ions of positive polarity. When an exposure of an agent begins, the peak of the positive polarity side of the spectrum shifts to the right, as can be seen in the figure. High humidity causes the same kind of shift but in both positive and negative ions. Diverse environmental conditions form the biggest challenge for classification. They contribute to the form of ion mobility spectra via complicated chemical reactions, and there are so far no good ways to normalize their influence. As a result, some ion mobility spectra measured in diverse environmental conditions can resemble each other even though the sample gas is different.

The IMS measurements, discussed in this paper, consist of various tests which illustrate the situation encountered by a continuous monitoring system. Each test has roughly 260 sweeps. All tests have been conducted in a laboratory using the same protocol. At the beginning of the test, air flows into the IMS device. After awhile, sample gas is released. At the end of the test, the exposure is over and air flows again into the device. The measurements are preliminary, and they include only nerve gas simulants DIMP and DMMP and two nerve gas agents referred to as agents A and B. The concentration and relative humidity of the sample gases has been altered. For example, in the case of DIMP three distinct concentrations and five humidity conditions have been used. Other environmental variables such as the temperature and air pressure have remained constant.

In order to train classifiers and estimate their performance we had to label the measured sweeps. We went through all the test recordings and determined, based on the data, the sweeps when the sample gas began to flow and when it ceased to flow. Then we labeled all the sweeps before and after the exposure as AIR and the sweeps in the exposure as the sample gas known to be present.
3. CUMULATIVE SUM -BASED CHANGE DETECTOR

Time series analysis is an intensively investigated topic. In these investigations, it is usually assumed that the properties or parameters describing the data are either constant or slowly time-varying. To characterize obtained measurements using a mathematical function, consider a sequence of independent random variables \( y_1, y_2, \ldots \) with density \( p_0 \). Until the unknown time \( t_0 \), the parameter \( \theta \) is \( \theta = \theta_0 \) and from \( t_0 \) it becomes \( \theta = \theta_1 \) so that \( \theta_0 \neq \theta_1 \). In general terms, testing of two hypotheses can be written as follows:

\[
H_0 : \ \theta = \theta_0, \ \text{and} \ \ H_1 : \ \theta = \theta_1.
\]  

The typical on-line problem is to detect the occurrence of a change from \( \theta_0 \) to \( \theta_1 \) using a record of measurements as soon as possible and with a fixed rate of false alarms before the unknown change time \( t_0 \). The optimal solution, according to the above-mentioned criterion, is basically a trade-off between quick detection and few false alarms, implying here of a comparison between the losses caused by the costs of protective actions and search for the origin of the possibly hazardous compound in the air. Because these issues tend to lead in complicated cause-and-effect relationships, sequential hypothesis testing is developed here for IMS measurements in the conventional non-Bayesian, non-decision-theoretic context.

Let \( t_a \) be the alarm time at which detection occurs; \( t_a \geq t_0 \). Given that an on-line algorithm declares detection at this instant, \( t_a \) may also be referred to as stopping time. If the parameters \( \theta_0 \) and \( \theta_1 \) are known, they completely specify the population distribution. Then, the so-called log-likelihood ratio would summarize the whole of the relevant information supplied by the measurement \( y_k \) but, in practice, this is usually not the case. We refer to \( k \) as time. The CUSUM algorithm announces a change at the alarm time

\[ t_a = \min\{k|g_k \geq h\}, \]  

where, assuming that the parameter \( \theta_1 \) is not known, the CUSUM algorithm’s decision rule can be compacted in a recursive manner as

\[ g_k = \max\{g_{k-1} + y_k - \nu, 0\}, \quad k = 1, 2, \ldots \]  

versus \( k \), given \( g_0 = 0 \) as the initial decision statistic. The reference value \( \nu \) is designed to identify positive deviations of \( \mu \), the mean of \( y_k \). As soon as \( g_k \) reaches \( h \), an alarm is triggered because the mean \( \mu \) might be larger than \( \nu \); i.e., \( \nu \geq \mu \) at \( \theta_0 \). If compared with a clean reference sample, this form of \( g_k \) is justified because the compounds of our interest produce positive shifts in collector plate currents as Figure 3 shows.

In this work, we are interested in estimating \( t_0 \) if the CUSUM algorithm implies its existence by triggering an alarm. Now let us assume that there are two change points. The first change happens at an unknown time \( t_0 \) such that before \( t_0 \) the parameter \( \theta \) is equal to \( \theta_0 \) and after the change it is equal to \( \theta_1 \); \( \theta_1 \neq \theta_0 \). However, after the second change point, at an unknown time \( t_1 \), the monitored process returns to its normal state; i.e., the parameter \( \theta \) after the change point \( t_1 \) is equal to \( \theta_0 \) so that \( t_1 \geq t_0 \). In segmentation, the goal is to partition data into its constituent parts. As we shall show next, the segmentation rule is then, at each time instant, to monitor if our threshold \( h \) has been exceeded and, if so, to compare the difference \( m_k - g_k \) with a threshold \( \gamma \) at the same time. For \( 0 \leq \gamma \leq h \), the rule can be formulated as follows:

\[ (t_{a_0}, t_{a_1}) = \min\{k|(g_k \geq h) \cap (m_k - g_k \geq \gamma)\}; \]  

where

\[ m_k = \max_{i \leq k}\{g_i\} \]

is the maximum value after the last renewal of the CUSUM algorithm. The estimate for the change time \( t_0 \) is equal to last renewal time plus one and it is obtained by monitoring for \( g_k \geq h \). Here, the word “renewal” refers to the decision statistic reaching 0, which always restarts the CUSUM algorithm. The change point estimation issue is not further clarified here, but the reader is advised to see the fine book by Basseville and Nikiforov for further reference. The estimate of the second change point \( t_1 \) is the index \( i \) corresponding to the maximum value of \( g_k \), which is obtained by monitoring the difference \( m_k - g_k \geq \gamma \). As an on-line equivalent, both of these

...
estimation strategies stem from the so-called maximum likelihood estimation algorithm.\textsuperscript{15} The segmentation rule is illustrated in Figure 4 for $h = \gamma = 15$; here, $t_{a_0} = 24$ and $t_{a_1} = 45$.

Because the first order aspiration IMS measurement is an integral measurement over a relatively broad range of mobilities, aspiration IMS devices typically have a relatively poor ion mobility resolution. That is, the ions will predominantly produce currents over a number of swept bias voltages that govern the electric field of the collector, at which the measurement is made. Theoretically, however, any desired resolution in the ion mobility domain can be achieved by increasing the number of used bias voltages that eventually leads to a characteristic ion mobility spectrum for a given gaseous compound. Because of broad range of mobilities, we suggest using the segmentation rule (4) for each bias voltage, termed also channels, separately. In Figure 5, real sweeps are processed; the white line indicates the maximum of the CUSUM statistics, yielding an estimate of the change’s peak value, duration and channel. To avoid a substantial increase in the length of this work, we shall omit the so-called run-length analysis used in assessing false alarm and detection delays. For this, however, there are many texts available; e.g., the technique of Luceño and Puig-Pey\textsuperscript{16} can basically be adopted without complicated manipulations.

4. NEURAL NETWORK CLASSIFIER

This section introduces our neural network solution to the gas identification problem. We decided to use a neural network classifier because it can solve complex problems due to its parallel distributed structure and ability to learn and generalize.\textsuperscript{17} Besides, Rosenblatt et al.\textsuperscript{12} have shown a neural network to be suitable for identification of anesthetic gases. In their experiment, the neural network was trained using 22 collector plate voltage values within the bias voltage range of -6 to 6V. The compounds, we wish to identify, are nerve gas simulants and agents. Now, the useful information is concentrated on the positive polarity side of the ion mobility spectrum while the negative polarity side of the spectrum is dominated by the impurities of the air. Therefore, we decided
Figure 5. Output of the aspiration condenser type IMS in gray and estimated change in white. The first sweeps were adopted for reference to estimate the mean values of the normal states channel-wise.

Figure 6. Blocks of our neural network solution. At first, collector plate currents and relative humidity are normalized. Then the principal components of the normalized currents are computed. Finally, 16 principal components and normalized relative humidity are fed to the neural network, which has five outputs. Each output represents one of the gases we wish to discriminate.

to exploit 30 points from the positive polarity side of the spectrum in categorization. In contrast to Rosenblatt’s work, our measurements are swept from bias voltage 20V to -20V and they are performed in diverse humidity conditions. Because changes in humidity have a considerable influence on ion mobility spectra, we use humidity as a neural network input too. Thus, our input data consists of 31 attributes, i.e., 30 collector plate current values and relative humidity.

Figure 6 illustrates the blocks of our neural network solution. At first, preprocessing is performed. The attributes are normalized to mean 0 and standard deviation 1. Then PCA is used to compress the information from the channels. PCA is a linear technique, which projects multidimensional data onto a lower dimensional subspace in a way that is optimal in a sum squared error sense.\textsuperscript{18} That is, it retains those characteristics of the data that contribute most to its variance. This kind of approach is reasonable, because nerve gas agents shift the peak of the ion mobility spectra causing high variance on the channels near the peak. These channels are the most useful in the discrimination of agents. By discarding principal components whose contribution to total variation is less than 0.001, we are able to reduce the number of channel attributes from 30 to 16. These 16 attributes and normalized relative humidity are used as an input to a multilayer feed-forward neural network. The neural network has five outputs, one for each gas we wish to discriminate. When training a neural network,
we set the target outputs so that the gas in question has value one in its output, and the other outputs are zeros. This, of course, requires that the sweeps are labeled before training.

In our identification system, a neural network works together with a CUSUM-based change detector. Figure 7 gives an example on how the CUSUM-based detector selects sweeps for categorization. At first, the CUSUM algorithm detects a change, which begins from sweep 70 and ends at sweep 134. Then the neural network categorizes these sweeps. We can see that some sweeps at the beginning and at the end of the exposure are categorized as AIR while the other sweeps are correctly categorized as DIMP.

5. CATEGORIZATION RESULTS

We compared the performance of our neural network approach with the one where data was not preprocessed using PCA. We tested two four-layer neural network topologies, which corresponded to the one used by Rosenblatt.\(^\text{12}\) The larger of the topologies had 31 neurons in the input layer while the smaller had 17 neurons. In addition, both topologies had 15 neurons in hidden layers, and 5 neurons in the output layer. The larger topology was intended for the non-reduced data and the smaller for the data preprocessed using PCA. We trained the larger network for the reduced data, too, to confirm our results. All neural networks had hyperbolic tangent sigmoid transfer function in all layers, and they were trained using resilient backpropagation. We randomly chose about a half of the laboratory tests for training and validation while we left the other half for testing. Ten different training and test sets were generated in order to get reliable results. In one test arrangement, we trained 100 networks for each of the ten training sets. Altogether, we trained 3000 neural networks. PCA was implemented as described in Section 4. In the case where PCA was not used, data was only normalized to mean 0 and standard deviation 1.

Table 1 shows the average sum squared errors (SSE) of sweeps in three test arrangements, i.e., how close the outputs of the neural networks are to their target outputs. In the first test, PCA is not used. The two other test arrangements, on the contrary, demonstrate our neural network solution where PCA is utilized. The
Table 1. Average sum squared errors of sweeps when data is preprocessed using PCA and when it is not. Two neural network topologies are tested. For clarity, values are multiplied by 1000.

<table>
<thead>
<tr>
<th></th>
<th>Test 1</th>
<th>Test 2</th>
<th>Test 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preprocessing</td>
<td>PCA not used</td>
<td>PCA used</td>
<td>PCA used</td>
</tr>
<tr>
<td>AIR</td>
<td>7</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>DIMP</td>
<td>104</td>
<td>87</td>
<td>82</td>
</tr>
<tr>
<td>DMMP</td>
<td>85</td>
<td>77</td>
<td>76</td>
</tr>
<tr>
<td>Agent A</td>
<td>398</td>
<td>329</td>
<td>319</td>
</tr>
<tr>
<td>Agent B</td>
<td>22</td>
<td>13</td>
<td>12</td>
</tr>
<tr>
<td>Average</td>
<td>123</td>
<td>102</td>
<td>99</td>
</tr>
</tbody>
</table>

Table 2. Percentages of correctly categorized sweeps when CUSUM-based change detector is used and when it is not used. Sweeps are preprocessed using PCA before categorization. Results for the agent A are weaker than for all the other considered gases most likely due to the smaller training data set.

<table>
<thead>
<tr>
<th></th>
<th>Not used</th>
<th>CUSUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preprocessing</td>
<td>PCA used</td>
<td>PCA used</td>
</tr>
<tr>
<td>AIR</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>DIMP</td>
<td>96%</td>
<td>100%</td>
</tr>
<tr>
<td>DMMP</td>
<td>96%</td>
<td>100%</td>
</tr>
<tr>
<td>Agent A</td>
<td>83%</td>
<td>97%</td>
</tr>
<tr>
<td>Agent B</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

results of each test arrangement are based on 1000 trained neural networks. These results are shown as average SSEs because errors represent the accuracy of the networks better than percentages of correctly classified sweeps. Errors for the nerve gas agent A are larger than errors for all the other gases considered because we had fewer measurements from that agent and we therefore used a notably smaller training set for it than for the other gases. The last row of the Table 1 shows the average errors for the three test arrangements. The smallest error is obtained by using the smaller of the tested neural network topologies for the data preprocessed using PCA. Average error in this case is even 20% smaller than error in the worst case where PCA is not used. Overall, we can say that when preprocessing data using PCA, the accuracy of a neural network can be improved and the number of neurons can be reduced.

In the preceding results, the CUSUM-based change detector was not employed, but we showed how the use of PCA raises the accuracy of a neural network. Table 2 introduces results where we compare percentages of correctly classified sweeps in cases where our CUSUM detector is used and where it is not used. We utilize here the same 1000 neural networks as in test arrangement 3 in the preceding results. In the results of Table 2, the category of a sweep is decided directly according to the maximum output of a neural network regardless its value. For the maximum output, one could also set a threshold that has to be exceeded before declaring the classification. That way, the false decisions could be reduced, even dramatically, in pursuance of leaving obscure events unidentified. However, we want to describe the overall behaviour of the proposed approach. Besides, the unidentified events can be considered erroneous results in many applications.

In the CUSUM approach, all sweeps, before and after the estimated change, are categorized as AIR whereas sweeps of the change are categorized based on majority voting. Percentages of correctly categorized sweeps are
notably higher in the case where the CUSUM-based change detector is used than in the case where it is not used. The CUSUM-based change detection makes categorization results more reliable.

6. DISCUSSION

Requirements for gas identification systems are commonly strict. The system should be sensitive and alarm when the concentration of the hazardous chemical agent is very small, but at the same time, the false alarm rate should be kept low. For this reason, it is very beneficial to separate detection and identification from each other. In that case, a very sensitive detector can be used because the identification block double-checks the sweeps before the alarm is given. In addition, the detector carefully selects the sweeps for the further analysis so that noise in the measurements does not cause false alarms so easily than in the case where a separate detector is not used. Yet another benefit of using separate detection and identification blocks is that only the detector must be run continuously and the identifier only when needed. This reduces the amount of computations because detectors are usually less computing intensive than identification algorithms. The lesser the amount of needed computations the lesser is the power consumption of a chemical detector device. Small power consumption is an important feature especially in portable detector devices.

When considering the results, the reader must be aware of two things. Firstly, all measurements were done in a laboratory with a prototype device, and the only environmental variable changed was relative humidity. The real world, of course, provides more diverse environmental conditions as well as impurities that can disturb the identification. Nonetheless, the data was real, and the used measurement protocol well illustrated the situation to be encountered by a continuous monitoring system. The second thing to remind the reader of is about labeling. All measured sweeps needed to be manually labeled before the classifier could be trained. The exact beginning of the exposure was hard to determine in some measurements because, due to the used measurement protocol, it could take a couple of sweeps before the intended concentration of the sample gas was attained. These sweeps in the transition were labeled as the sample gas. That decreased the number of correctly classified sweeps in test arrangements where a change detector was not used because sample gas had so small concentration in the first few sweeps of the transition that the sweeps resembled air. Figure 7 illustrates the situation. Inexact labeling affected the results of the CUSUM-based detector as well. In some measurement, the CUSUM detected a change even before the manual labeling indicated it. In these cases, some AIR-labeled sweeps were categorized as a sample gas, which decreased the percentage of correctly classified AIR sweeps.

It is not the best way to assess the results of the CUSUM-based detector on a sweep-by-sweep basis, but however, it demonstrates the positive effect that a separate change detector has on the categorization results. In future, it would be interesting to examine long lasting real-life measurements using our identification system and to develop an enhanced logic that decides when alarms are given to users. Then we could calculate the probabilities of correct and incorrect detections and estimate how soon an exposure is detected. Even though the CUSUM detects the changes reliably, the gas identification is the task of the classifier so the performance of the classifier is equally important. Detectors and classifiers do not work perfectly, but the double-check mechanism of our identification system decreases the amount of false alarms.

7. CONCLUSIONS

This paper introduced a novel approach to the detection and identification of hazardous chemical agents from ion mobility spectra. We considered an identification system, which is run continuously. The key idea was to separate the detection and identification blocks from each other to improve the performance of the system. We used a CUSUM-based change detector and a neural network classifier. Our approach provided an estimate of exposure’s duration and allowed the gas identification task to be accomplished using carefully selected measurements. As a further novelty, we proposed the preprocessing of the ion mobility spectra using PCA. The use of PCA improved the accuracy of a neural network and reduced the number of neurons. When using the proposed CUSUM change detector and the neural network classifier together, classification results were considerably better than without a separate detector.
ACKNOWLEDGMENTS

The authors would like to thank the Finnish Funding Agency for Technology and Innovation and Environics Oy for funding (Grant No. 2955/31/06).

REFERENCES


