Decision Support Systems 199 (2025) 114544

Contents lists available at ScienceDirect

Decision Support Systems

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

Beyond assumptions: A reference architecture to enable unsupervised process discovery from video data[☆]

Niklas Wördehoff [a,c,*], Andreas Egger [a], Wolfgang Kratsch [a,b], Fabian König [a], Maximilian Röglinger [a,c]

[a] FIM Research Center, Branch Business & Information Systems Engineering of the Fraunhofer FIT, Alter Postweg 101, Augsburg, 86159, Germany
[b] Technical University of Applied Sciences Augsburg, Alter Postweg 101, Augsburg, 86159, Germany
[c] University of Bayreuth, Wittelsbacherring 10, Bayreuth, 95444, Germany

ARTICLE INFO

Keywords:
Process mining
Video data
Unstructured data
Computer vision
Unsupervised learning
Reference architecture

ABSTRACT

Process mining has developed into one of the most important research streams in business process management. Despite its successful application to improve process performance in industry, there is still substantial potential to be realized in the coming years. One of them is the use of unstructured video data to enable the analysis of previously unobservable parts of processes. Existing approaches derive event logs from video data by extracting a predefined set of potentially relevant activities. As this set is typically determined using a process model or input from process experts, rather than the available video data, current solutions are unable to identify activities that extend beyond the presumed process behavior, limiting transparency in process analysis. Therefore, this study aims to develop a solution that enables the extraction of actual process behavior from video data, as opposed to assumed process activities. Following a design science research methodology, we developed and evaluated the Reference Architecture for Video Event Extraction (RAVEE), which enables the identification of individual process steps in an unsupervised manner. We performed several evaluation activities to ensure the completeness and applicability of the RAVEE. A prototypical instantiation of the RAVEE further demonstrates its ability to extract process-relevant events from video data on two real-world datasets.

1. Introduction

Process mining aims to discover, monitor, and improve processes by extracting knowledge from event logs readily available in information systems to point out possible courses of action [1]. Owing to its ability to provide evidence-based decision support, process mining, recognized as a big data technology, has gained high relevance in both science and practice. As a result, it has evolved into one of the most active research streams in the Business Process Management (BPM) community [2,3]. At the same time, the commercial process mining market — projected to grow at double-digit rates in the coming years and reach \$2.3 billion by 2025 [4] — has attracted major software vendors (e.g., Celonis) who have entered with great success [5].

A major challenge in generating high-quality event logs for process mining is the reliance on readily available data that must be suitable for analysis [6]. Since companies usually do not collect data specifically for process mining purposes, the data must be preprocessed into formats that support process mining. This can be challenging, especially when data is not available in a structured form. About 80% of all data is unstructured [7], which increases the barriers to the application of process mining in enterprises. Without appropriate solutions, unstructured data is often left unprocessed or not stored at all. We follow Kratsch et al. [8] by referring to the resulting parts of processes that cannot be fully captured in information systems as 'blind spots' in this paper. Uncovering these blind spots can be very beneficial, as they typically contain additional contextual information about processes and can thus help improve their examination in an end-to-end fashion [9].

Accordingly, current research has taken initial steps to exploit unstructured data, including text [10–12], sensor [13,14], and image or

☆ This work was partly supported by the Bavarian Research Foundation (Bayerische Forschungsstiftung) [grant number AZ-1550-20].
* Corresponding author at: FIM Research Center, Branch Business & Information Systems Engineering of the Fraunhofer FIT, Alter Postweg 101, Augsburg, 86159, Germany.
E-mail address: niklas.woerdehoff@fim-rc.de (N. Wördehoff).

https://doi.org/10.1016/j.dss.2025.114544
Received 27 June 2024; Received in revised form 22 September 2025; Accepted 22 September 2025
Available online 3 October 2025

video data [8,15]. Provided that data protection regulations are observed [16], video data are particularly valuable for uncovering blind spots, especially when other data sources are unavailable. This is most relevant for manually executed process steps, where installing sensors would often be impractical or costly. Despite recent advances in computer vision, video-based approaches to process mining remain limited. Our Systematic Literature Review (SLR) (see supplementary material in our repository[1]) shows that existing approaches are typically tailored to specific use cases [15,17–19], and only a few provide generic means of extracting process activities from video data [8,20,21]. A key challenge is the need to predefine which activities are relevant before they can be extracted into event logs (e.g., [8,15]). This reliance on predefined sets of activities — typically derived from process models or expert assumptions — introduces an inherent human bias by focusing analysis on expected process behavior rather than actual observed behavior. In this work, we use the term 'bias' to denote this human bias arising from assumptions about expected behavior. Consequently, deviations, unexpected process steps, or emerging patterns — crucial for process improvement — may go undetected.

To the best of our knowledge, no existing video-based approach derives BPM-relevant process steps without relying on predefined activity sets. This hinders objective decision support, as insights are drawn from assumed process flows rather than data-driven evidence, the very foundation of process mining [1]. Recent literature on Decision Support Systems (DSS) stresses the importance of overcoming this limitation. For example, Andrews et al. [22] call for extending the toolchain to cover the entire event log lifecycle, among others, by advancing log extraction methods. Similarly, Kratsch et al. [8] advocate moving beyond supervised learning in video-based log extraction, arguing that unsupervised methods can capture a broader range of process activities without being restricted to predefined activity classes. Together, these perspectives highlight the potential to strengthen decision support by enabling data-driven, unbiased insights. Therefore, this study aims to contribute to transparent process discovery and objective decision support by investigating the following research question: *How can actual process activities be derived from video data as part of process discovery?*

We employed the Design Science Research (DSR) methodology [23] to address this research question and developed the RAVEE, an Reference Architecture (RA) designed to extract actual process behavior from unstructured video data. Unlike existing approaches that rely on predefined activity sets [8,15], the RAVEE enables the unsupervised identification of individual process steps and their transformation into event logs suitable for process mining. By instantiating the RAVEE as a software prototype, we provide guidance for its practical implementation and evidence of its applicability in a process mining context. We evaluated our prototype through established evaluation activities [24] demonstrating its potential to support transparent, data-driven process discovery. These findings extend existing knowledge by introducing a foundation for unsupervised video-based process discovery, eliminating the bias associated with predefined activity sets. The RAVEE thereby opens new avenues for research and enables organizations to analyze processes based on actual operational dynamics rather than assumed process flows, supporting evidence-based decision-making.

The remainder of this paper is structured as follows: Section 2 provides an overview of process mining and introduces relevant information on computer vision. Section 3 presents the research design, which is based on the DSR process [23]. Sections 4 and 5 are structured according to the phases of the DSR process. Section 4.1 defines the requirements for a potential solution to our research question. Section 4.2 describes the architectural design of the proposed RAVEE, which is evaluated in Section 5.1. Section 5.2 contains the prototypical instantiation of the RAVEE and its evaluation on two real-world datasets. Section 5.3 examines the practical use of the instantiated RAVEE in a case study. Finally, Section 6 concludes our work by outlining its limitations and suggesting future research opportunities.

[1] https://github.com/RAVEE-PM/RAVEE/blob/master/Supplementary_Material_SLR.pdf.

2. Theoretical background

2.1. Process mining

Upon successful application, process mining techniques can provide holistic insights into underlying processes and their complex structures [25]. By incorporating techniques and methodologies from diverse adjacent domains and situating them within the context of process mining, it becomes feasible to discover, monitor, and improve actual processes [1].

Initially, van der Aalst et al. [1] defined three basic process mining use cases. These encompass discovery (i.e., producing a process model from an event log without prior knowledge), conformance checking (i.e., comparison of an existing process model with an event log of the same process for conformance between the two), and enhancement (i.e., improvement or extension of existing process models using actual process information recorded in event logs). In addition to these established use cases, several process mining perspectives have been delineated in the literature to concentrate on acquiring specific insights [1,26,27]. The most frequent of these are the control-flow perspective (emphasizing activity flow), the organizational perspective (focusing on the resources involved in the different activities), and the temporal perspective (focusing on execution times and activity frequency). Van der Aalst [28] refined the process mining framework to include a distinction between pre-mortem data (pertaining to ongoing cases) and post-mortem data (pertaining to already completed cases). Moreover, the framework differentiates between de jure models (reference process models) and de facto models (descriptive model of actual process activities). These key use cases are then extended by adding seven others. Together, this results in ten techniques that can be applied depending on the data available, the desired model, and the perspective used.

2.2. Event log generation for process mining

The starting point of every process mining analysis is an event log generated from actual process data. Therefore, the quality of the underlying data is of utmost importance [1,29]. Accordingly, Reinkemeyer [29] defines the existence and reliability of process data as one of the three most important success factors for process mining projects. The first step in generating an event log from actual process data is to define and delineate the process under investigation with all its key activities [29,30]. In a further step, all data sources are identified, whereby the data can originate from a wide variety of source systems such as databases, enterprise resource planning systems, document management systems, and other structured, semi-structured, and unstructured data sources [31,32]. After collection and extraction, the data must be preprocessed to generate an event log. The required effort for this step varies considerably depending on the structure of the source data [29]. Typically, various data quality problems must be solved, such as detecting and removing duplicates, identifying missing attributes and event data, and determining relevant events [31,32]. To assess the log data quality for usability in process mining, van der Aalst et al. [1] defined five different maturity levels. In this context, the higher the maturity level, the higher the data quality. According to van der Aalst et al. [1], data from the third maturity level and above can be used to generate event logs. This third maturity level comprises automatically generated data but without a systematic approach. There is certainty that the logged events correspond to reality, but not that they are complete [1]. Event abstraction concepts may also need to be considered at this stage. Process mining algorithms expect abstract, high-level events that describe their respective activity states [33]. If the current level of abstraction is still too low, it needs to be adapted using event abstraction concepts. Moreover, it is also essential to consider that different levels of granularity of activities may be required depending on the evaluation objective (e.g., low granularity for overall process analysis vs. high granularity for detailed analysis) [1]. Once the relevant data has been collected from the available sources and identified, extracted, and transformed as needed, it can be transferred to a target system and an event log can be generated [31].

Event logs are the starting point for process mining, capturing traces of executed processes as collections of events, each associated with a case, activity, and timestamp [1]. One of the most widely used formats for storing event logs is the eXtensible Event Stream (XES) standard [34]. XES is designed to represent event-driven data through event logs and event streams, ensuring compatibility with various process mining techniques. Each recorded event must include at least a case identifier (e.g., a case number) to link it to a specific process instance, an activity name, and a timestamp to maintain the correct sequence of events and therefore meet the requirements of an event log [25]. Additionally, XES allows for the enrichment of event logs with optional attributes, such as resource allocation, cost information, and other contextual details, providing deeper insights into process execution [31].

Data for event log creation may also originate from unstructured data sources [31,32], which can take various forms such as text, sensor, or video data. Among these, video data offer a valuable supplementary source for obtaining event logs that provide a more holistic representation of reality. However, video-based event log creation faces data quality challenges that must be addressed to ensure a maturity level suitable for further analysis. The approaches found in our SLR (see Section 1) [8,15,17–21] tackle these challenges in different ways (a brief overview is available in our repository[2]).

2.3. Computer vision

Computer vision applies mathematical techniques to visual input such as images or videos to extract meaningful information [35,36]. In contrast to image processing — which modifies or enhances images to create new results [37] and lies outside the scope of this study — computer vision focuses on interpreting visual data. Closely associated with Artificial Intelligence (AI), it underpins numerous technologies across domains [38–40] and is enabled by a wide range of algorithms. The techniques most relevant to this paper are introduced below.

Traditional computer vision approaches can be distinguished from deep learning approaches [41]. Starting with the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) of 2012, approaches based on deep learning gained strong popularity, mainly due to their high accuracy [42]. In recent years, these modern computer vision algorithms have predominantly been based on Convolutional Neural Networks (CNNs). These neural networks gradually extract relevant features from input data through multiple layers. The final feature set is processed in a fully connected layer to generate predictions [43].

In contrast, traditional computer vision pipelines involve a preprocessing step followed by feature extraction to reduce input data to essential features. Subsequently, a learning algorithm is applied for classification to obtain an output class or label [44]. This feature extraction step is circumvented for CNNs as deep learning algorithms allow direct processing of unstructured data like images without manual feature extraction [45].

Although deep learning approaches avoid the need for hand-crafted features and often achieve superior performance, certain aspects (e.g., avoidance of deep learning black boxes) still favor traditional computer vision methods [41]. For a more detailed discussion, see O'Mahony et al. [41] and Nixon [44]. In general, traditional approaches remain useful when their complexity is manageable or when the generated feature vectors offer advantages. Since the clustering algorithm by Sarfraz et al. [46], which we employ for automatically detecting process steps as part of the RAVEE, relies on such feature vectors, the following section elaborates on their extraction in traditional computer vision.

Before feature extraction, a preprocessing step is performed. This step can involve several aspects, among others, adjusting or normalizing contrast and brightness effects or image size [47]. In case the visual input is a video, as in this paper, video-specific elements have to be considered: The input, usually available as a video stream, has to be split into individual frames to enable framewise analysis. Limiting frames per second (FPS) might also be advantageous in reducing data and computing time [48].

After preprocessing, relevant features of individual input frames are extracted to create a video frame representation, aiming to remove unnecessary information for tasks like classification. For instance, object differentiation often requires comparing shapes rather than color values or backgrounds. Therefore, local features are typically adept at object tracking. These local features are distinctive features reproduced similarly in corresponding images [44].

Local features are valuable representations for both images and videos. In the video domain, feature descriptions can be created by tracking interest points across a sequence of images to collect motion information in trajectories [49]. One approach to this is the use of dense trajectories, which are formed by tracking densely sampled points with the help of optical flow fields, as described by Wang et al. [49]. To better account for camera motion, Wang and Schmid [50] later proposed improved dense trajectory features.

To obtain an image representation for classification, feature encoding is applied [51]. There are several approaches to encoding features, including Fisher Vector and bag of features. A bag of features, or a bag of visual words, is constructed by extracting local descriptors of an image and then assigning them to the closest entry of a visual vocabulary [52]. This codebook is generated by clustering all descriptors using k-means. Hence, k clusters are obtained, each with one centroid. These centroids represent the main features of the training dataset. After that, a feature vector, a k-dimensional representation of the input data, can be calculated. Another approach to feature encoding is to compute a Fisher Vector [51]. The difference from the bag of features approach is that the Fisher Vector considers both the mean and the variance between the video descriptors and a Gaussian Mixture Model. In this case, the Gaussian Mixture Model serves as a visual vocabulary. An essential characteristic of the k-means approach discussed earlier is that each descriptor is assigned to exactly one cluster. In doing so, no attention is paid to how that item is associated with a cluster. In a Gaussian Mixture Model, this is ensured by comprising each cluster of a Gaussian, which represents the distribution of the associated descriptors. The Fisher Vector uses the Gaussian Mixture Model to describe how a feature set deviates from a learned distribution. This leads to a k(2D+1)-dimensional vector, where k corresponds to the number of clusters of the Gaussian Mixture Model and D to the dimensionality of the input descriptors [51,53].

After creating a video frame representation, it can be further evaluated in the next step. There are various approaches to this (e.g., Support Vector Machine (SVM) [54]), whereby a large part is based on supervised approaches. Here, the algorithm has to be trained on specific output categories for the classification task. This implies some limitations for practical application, such as that the possible output categories must already be known before an analysis can be carried out. However, there are also unsupervised approaches, such as the TW-FINCH algorithm [46]. As discussed earlier, an unsupervised learning problem such as this one aims to find meaningful relationships between given input data. The TW-FINCH algorithm performs action segmentation using an unsupervised clustering method. Through action segmentation, an action is assigned to each frame of the input video, whereby these actions are primarily high-level activity instances performed by humans [55]. Since the approach is unsupervised, there is no need for detailed annotations, as would be the case in other supervised action segmentation approaches. Instead, the clustering algorithm

[2] https://github.com/RAVEE-PM/RAVEE/blob/master/Supplementary_Material_SLR.pdf.

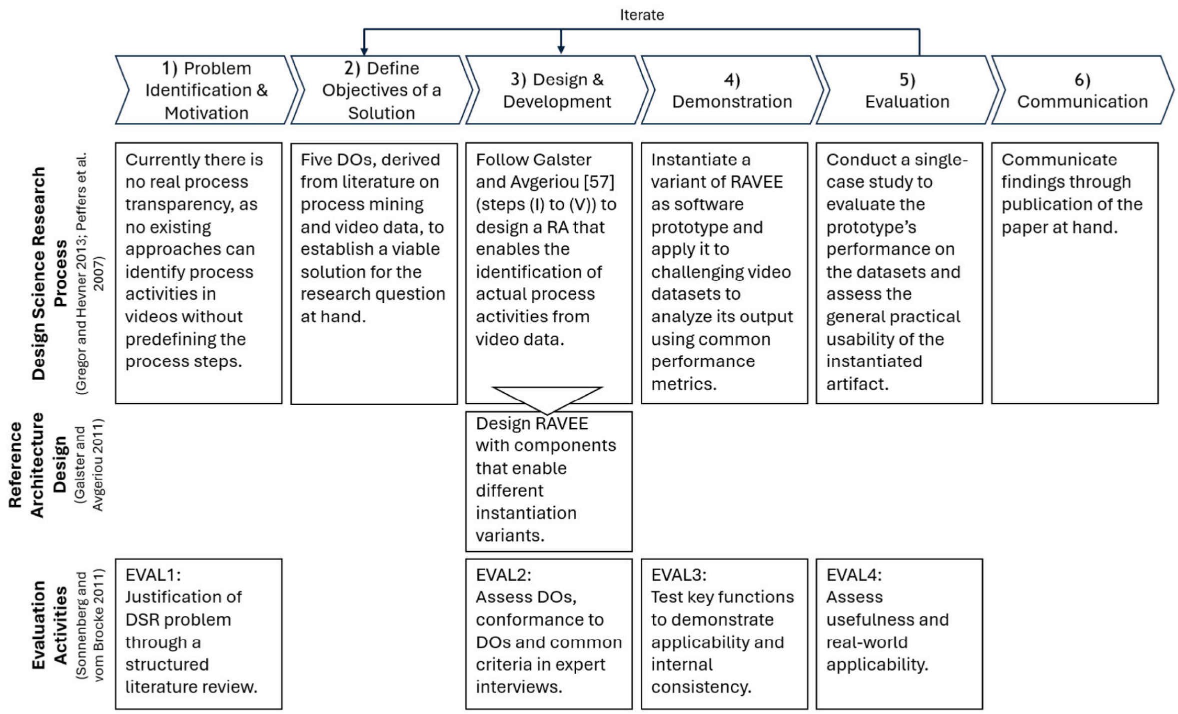


Fig. 1. Overall research design based on Peffers et al. DSRM [23], Sonnenberg and vom Brocke [24], and Galster and Avgeriou [57].

groups the frames of a video in a way that makes them semantically consistent. This is done by grouping the individual frames in a hierarchical clustering based on their spatial and temporal similarities. To be precise, the algorithm connects frames based on their feature space proximity and their respective position in time [46]. Thus, the input video can be divided into segments assigned to specific clusters. These clusters, in turn, correspond to individual actions that occur in the video. This enables videos to be divided into meaningful segments without the need for training on the target activities.

3. Research design

As an answer to our research question, we developed the RAVEE to systematically derive actual process activities from video data. To build an artifact within DSR, we developed an RA and instantiated it as a software prototype to demonstrate its utility and feasibility [56]. We followed the DSR reference process proposed by Peffers et al. [23], commonly referred to as the DSRM, which consists of six phases. Since Peffers et al. do not provide guidance on designing RAs — one of the main goals of this paper — we complemented the DSRM with a specific method for RA development presented by Galster and Avgeriou [57], which has been successfully applied in related contexts (e.g., [8]). Galster and Avgeriou formulate six steps to construct an empirically grounded RA. Since the proposed steps of Galster and Avgeriou partially overlap with those of the DSRM, we only conducted steps not covered by the overall DSRM process.

To ensure the rigor of the conducted research through continuous evaluation of the artifact, we followed the evaluation activities according to Sonnenberg and vom Brocke [24], which specify when, how, and what to evaluate. Of these, two evaluation activities are conducted ex-ante (EVAL1 and EVAL2) and two ex-post (EVAL3 and EVAL4). As required by Sonnenberg and vom Brocke [24], relevant evaluation criteria (i.e., understandability, completeness, real-world fidelity, internal consistency) were considered throughout the evaluation process. Fig. 1 depicts the resulting overall research design, which is described below.

In **Phase 1** (problem identification and motivation) of the DSRM [23], we derived related literature at the intersection of video data and process mining in an SLR [58] (an in-depth description of the SLR can be found in our repository[3]). The comparison with the related literature, as presented in Section 1, ensures the novelty and relevance of the problem (EVAL1). In **Phase 2** (define objectives of a solution), we established Design Objectives (DOs) to describe how the new artifact is expected to support solutions [23], drawing on relevant literature at the intersection of process mining and computer vision (Section 4.1). In **Phase 3** (the design and development phase), we created the RAVEE (Section 4.2). Here, we followed the previously mentioned steps by Galster and Avgeriou [57].

These steps are (I) decision on the type of RA, (II) selection of design strategy, (III) empirical acquisition of data, (IV) construction of RA, (V) enabling RA with variability, and (VI) evaluation of RA. We incorporated these steps as follows. For the type of RA (I), we created a preliminary facilitation RA designed to be implemented in multiple organizations. We adopted this type of RA as the RAVEE should be valid in as many application areas as possible and be of practical relevance. Since the RA was created by independent researchers, domain independence is ensured. As for selecting the design strategy (II), we chose a strategy supporting the preliminary nature of our RA. For this reason, we created our artifact by taking advantage of the research-driven nature of this prescriptive strategy and using related research as a basis. We conducted an SLR on video data in process mining to ensure empirical data acquisition (III) and to gain insights into existing approaches. Furthermore, we collected relevant literature on process mining and computer vision to provide a sound foundation for our RA (Section 2). Subsequently, we constructed our RA (IV) based on the empirical data collected in the previous step. To properly document the design of the RA, we chose a UML-based documentation level. Thereby, we ensured a comprehensible design for the preliminary RA that both provides operational support for a real-world implementation and leaves room for extensions and adjustments. To enable variability

[3] https://github.com/RAVEE-PM/RAVEE/blob/master/Supplementary_Material_SLR.pdf.

Table 1
Design objectives for the RAVEE.

DO	Description	Derived from
DO1	**Explorative and data-based activity identification and detection:** To allow for transparent process analysis of video data, the artifact must enable the identification and subsequent detection of observable rather than presumed activities (i.e., be data-driven or curiosity-driven) in an exploratory manner.	[1,59]
DO2	**Cross-domain applicability:** To be applicable in a wide variety of application areas and business domains (e.g., manufacturing, construction), the artifact must not be restricted to predefined use cases.	[25,60]
DO3	**Extraction of activities at different abstraction levels:** As process mining analyses require events at varying levels of granularity, the artifact must be capable of generating activities of different levels of granularity based on the given video input.	[1,33]
DO4	**Consideration of contextual process knowledge:** To allow for the consideration of process/domain-specific knowledge, the artifact must support contextualizing discovered activities to distinguish between relevant and irrelevant activities.	[1,30,61]
DO5	**Event log generation:** In order to utilize the extracted events for subsequent process mining analyses, the artifact must export a standardized process event log format.	[31,60]

(V) in the RA, we used an annotation that allows for optional components and different case-specific design variants, and therefore RA instances.

We only performed steps (I) to (V) of Galster and Avgeriou [57] in this work, as we followed Sonnenberg and vom Brocke [24] for our evaluation activities. Accordingly, in EVAL2 (Section 5.1) (performed instead of step (VI)), we compared the RAVEE against related work in a competing-artifacts analysis. Additionally, we conducted expert interviews on our DOs and RA to demonstrate that the RAVEE solves the previously stated problems.

We instantiated the artifact as a software prototype to demonstrate the use of RAVEE in **Phase 4**, the demonstration phase (Section 5.2). We then applied it to two real-world video datasets to assess its performance and feasibility (EVAL3). We applied our prototype to parts of each dataset to obtain and analyze event logs using several suitable performance metrics. **Phase 5**, the evaluation phase (Section 5.3), concludes with EVAL4, focusing on the assessment of the artifact's practical use. In this phase, we conducted a case study focused on the application of the prototype to one of the datasets, drawing on the insights of a domain expert involved in the dataset creation to evaluate the prototype's performance and practical applicability. Finally, **Phase 6** (communication) is fulfilled by publishing this paper.

4. Design and development

4.1. Definition of design objectives

This phase of our DSR process, as outlined in Fig. 1, deals with the definition of DOs, i.e., "description[s] of how a new artifact is expected to support solutions" [56]. Our RA should adhere to these DOs to provide a viable solution for obtaining relevant process activities from video data as a starting point for process discovery. Table 1 summarizes the DOs derived from relevant research in process mining (Sections 1 and 2). As part of a later evaluation step (Section 5.1), the DOs were also validated for relevance, completeness, and level of implementation in a joint assessment with all co-authors, as well as with experts.

4.2. Design specification

4.2.1. Architectural design of RAVEE

Fig. 2 illustrates our main contribution, the RAVEE, which enables the identification of actual process behavior from video data. In doing so, RAVEE first detects individual process steps in an entirely unsupervised manner and subsequently enriches them to construct an event log composed of valid events.

RAVEE builds on Kratsch et al.'s ViProMiRA [8], which provides a structured approach to supervised video-based process mining. However, ViProMiRA relies on predefined user input to identify events, limiting its ability to uncover deviations and emergent process steps. In contrast, RAVEE autonomously extracts process steps from video data, enabling a more explorative, data-driven analysis.

To achieve this, we retained ViProMiRA's three-layer structure — Data Preprocessor, Information Extractor, and Event Processor — but introduced a fourth layer, the Feature Extractor, to enable unsupervised detection. The Data Preprocessor remains largely unchanged as it prepares video data for analysis through standard preprocessing techniques. However, the downstream layers underwent significant modifications: the Information Extractor fundamentally shifted from a supervised classification approach to an unsupervised clustering-based method, requiring replacing core components and new processing logic. The Event Processor was adapted to focus specifically on extracting relevant events rather than supporting a holistic process mining system.

These modifications represent a substantial departure from ViProMiRA, making RAVEE the first RA capable of unbiased event extraction. It enhances process transparency and decision support by autonomously identifying process steps without relying on predefined assumptions or user input. This eliminates bias in event extraction and strengthens process mining capabilities. While a human-in-the-loop is required for subsequent activity labeling, relevant process steps are initially discovered without user input, ensuring decisions are based on actual process behavior rather than predefined assumptions. As required by Galster and Avgeriou [57], RAVEE provides a flexible framework with various instantiation options, enabling explorative, data-driven process mining using computer vision across different application areas. It introduces optional, configurable components — indicated by dotted frames — that can be tailored to specific use cases. The following sections introduce the RAVEE and its components in detail.

4.2.2. Data Preprocessor layer

The **Data Preprocessor** layer creates an initial interface between raw video data and the following subsystems. Several preprocessing steps are consolidated in the Video2Frame Converter component (e.g., converting a video into individual frames and adjusting the frame rate and resolution).

4.2.3. Feature Extractor layer

The **Feature Extractor** receives the preprocessed frame batch and extracts the required video features to produce a representation of the video reduced to the relevant information. This corresponds to creating a video frame representation as described in Section 2.3. To prepare the frame batch for several specific feature extraction methods, the additional Video2Grey Converter component can convert an RGB color scheme to a grayscale scheme for the frame batch. This optional component enables feature extraction methods that have proven to perform better on grayscale representations of a video stream, as well as images where gray scaling eliminates unnecessary background noise.

The **Feature Generator** generates a video frame representation provided to the subsequent layers for further processing steps. The Feature Generator comprises up to five subcomponents, which can be combined for specific use cases to generate the best possible video frame representation for the selected approach. Different video frame representations can be needed because various algorithms with different input needs can support the action segmentation step performed in the Information Extraction subsystem.

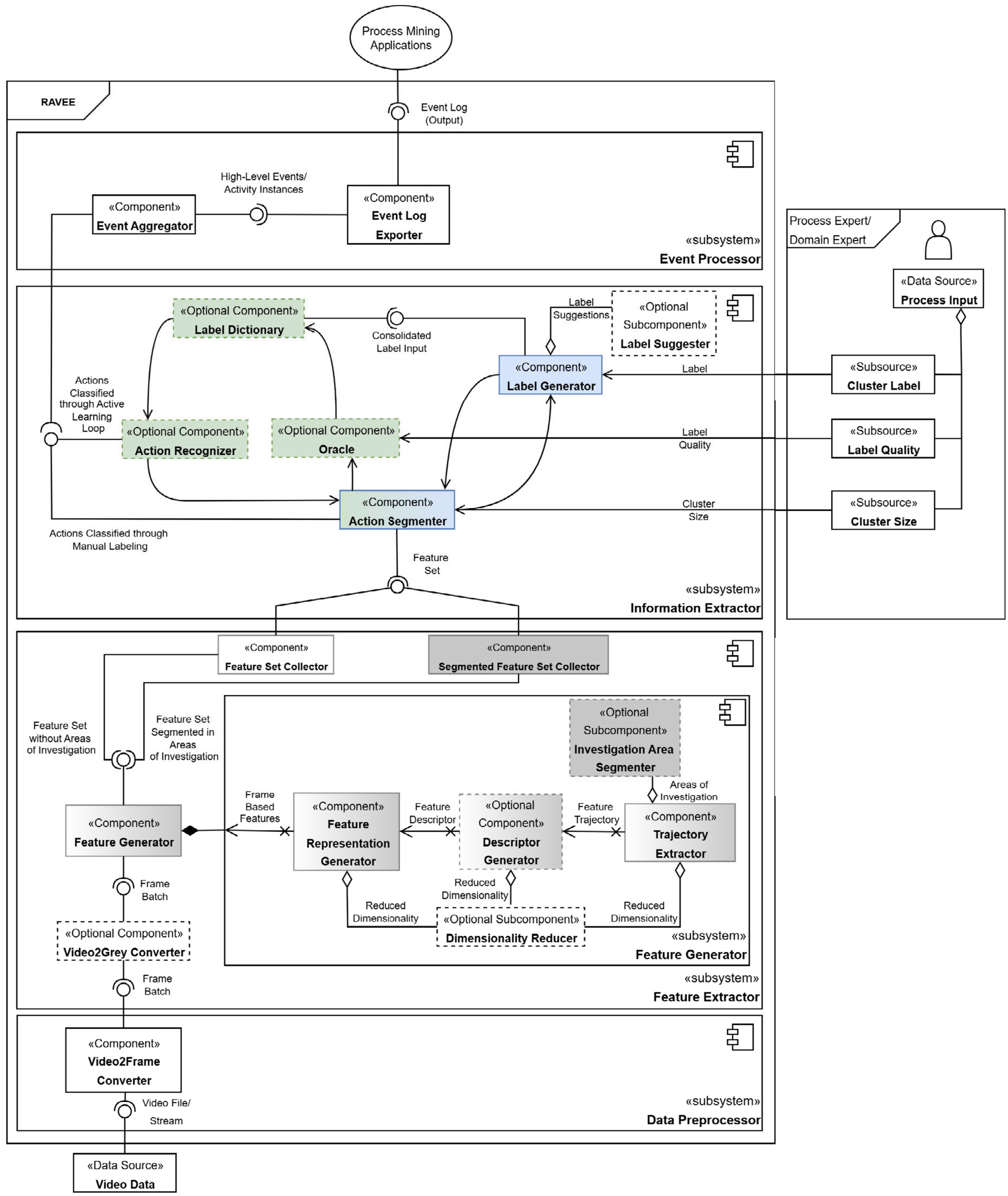


Fig. 2. The RAVEE as a UML Diagram.

As an initial step, the Trajectory Extractor can extract local spatio-temporal features by tracking interest points through video sequences. The optional Investigation Area Segmenter subcomponent can be applied for additional information about which features were tracked in which area of a frame. Employing this optional subcomponent, several Areas of Investigation can be defined in which the feature trajectories are tracked separately. This enables independent process steps that take place simultaneously in different areas of the image to be evaluated independently in the subsequent steps.

Using the optional Descriptor Generator, feature descriptors can be computed on the previously generated trajectories to encode relevant information and differentiate one feature from another. The feature descriptors normalize the trajectory vectors using different approaches depending on the descriptor type. Accordingly, depending on the application scenario, different descriptors vary in suitability for further steps (e.g., HOG descriptors are well-suited for static images, whereas HOF performs better with excessive camera movement).

Furthermore, the Dimensionality Reducer can reduce the dimensionality of the raw feature trajectories, their respective descriptors, or the frame-based features. Using the Dimensionality Reducer is often advantageous, as computations in subsequent steps can be accelerated due to lower dimensionality. Likewise, a decorrelation of the data can lead to better results depending on the following step [53]. Both the Descriptor Generator and the Dimensionality Reducer represent optional components that can be included depending on the needs and structure of the required feature representations. However, as already discussed, their deployment must be tailored to each scenario and can, therefore, not be predefined for every use case within the scope of the RA.

Finally, the Feature Representation Generator creates a dimensionally adjusted representation of the previously computed features for every single video frame. Thus, each frame is then represented by a feature representation.

The extraction of the feature representations can be adapted to different input scenarios through the various components. Cross-domain applicability is ensured since the feature extraction step can also be performed for videos of all types, regardless of the application area in which the videos originated (DO 2).

4.2.4. Information Extractor layer

The **Information Extractor** layer obtains the previously computed video frame representations and generates a frame-based clustering of the individual process steps. Subsequently, activity labels are assigned to the individual process steps. This is realized by the two color-coded processing loops in the subsystem, both of which receive relevant process information from a human-in-the-loop instance. The human-in-the-loop should be a process or domain expert to be able to give contextual process insights.

In the blue loop, a dictionary with activity labels is created based on the individual process steps and contextual process information. The optional green loop, in turn, uses this dictionary to train a supervised approach for automated recognition of the process steps. The two loops consist of different components, which enable them to achieve their functionalities.

The Action Segmenter component is present in the blue loop on the right-hand side of the Information Extractor subsystem layer and the optional green loop on the left-hand side. This component divides the individual video frames into clusters corresponding to individual process steps. Depending on whether the Investigation Area Segmenter subcomponent has been used in generating the feature set, the feature set contains either a single representation per frame or multiple ones corresponding to different Investigation Areas. This input is utilized in the Action Segmenter component, which uses the feature representations to solve a clustering problem. Given the fully unsupervised approach, it allows for exploratory, data-based process step identification from video data (DO 1).

The blue loop also contains a Label Generator component. Here, a human-verified dictionary of action labels is created from a previously selected representative sample of clusters. This is achieved by labeling the representative cluster sample obtained from the Action Segmenter through the human-in-the-loop. Thus, the contextualization of discovered activities is enabled (i.e., the consideration of contextual process knowledge), which is consistent with DO 4. The label input for the representative cluster sample is then used to assign a label to each of the obtained clusters, thereby generating a label dictionary. The optional Label Suggester subcomponent can assist the human-in-the-loop during labeling by leveraging pre-trained algorithms (e.g., activity recognition models) to identify typical process steps from video data. When the Label Generator receives a video segment — previously clustered by the Action Segmenter as a single process step — the Label Suggester generates a suggested label in the background, which the human-in-the-loop can then review and accept.

Depending on the aggregation level at which the human-in-the-loop wants to examine the video, a desired number of clusters can be chosen, which affects cluster size and, hence, the granularity of detected activities. Hence DO 3, the extraction of activities at different abstraction levels, is facilitated. The blue loop can be executed multiple times at different aggregation levels, resulting in different cluster numbers, hence creating a multi-labeling for individual frames since they can belong to different clusters of various aggregation levels. In this way, all process-relevant information can be included in the label dictionary.

The optional green loop contains an Oracle component, a Label Dictionary, an Action Recognizer, and the Action Segmenter previously discussed. Although not the primary focus of our contribution, the green loop can enhance the efficiency and scalability of the labeling process. This is particularly relevant in real-world applications, where labeling every process step from the blue loop is impractical. The label dictionary previously created in the blue loop serves as the initial training set for an active learning approach in the green loop. This input is stored in the Label Dictionary component. The information here included is used to train a supervised model in the Action Recognizer component. This component makes predictions on data from a candidate pool (i.e., the output of the Action Segmenter component). The human-in-the-loop within the Oracle component then assesses these predictions for their quality. After sufficient training, the active learning loop can recognize and label known process steps without further human input, thus enabling automated process activity detection (DO 1). The amount of training data and the number of active learning feedback loops required to achieve sufficient model performance depend on several factors. These include the confidence threshold for automatic detection (e.g., usually higher in medical applications than in logistics), the complexity of the activities to be identified, the number of distinct activities, and their frequency in the training data. More broadly, the required training effort is influenced by both the desired output quality and the structure of the available training data.

4.2.5. Event Processor layer

The output of the blue loop — or the optional green loop if enabled — can be fed into the Event Aggregator component of the **Event Processor** subsystem layer. The Event Processor subsystem takes the information from the Information Extractor subsystem, processes it and converts the information into a format suitable for event log creation, generating logs for subsequent process mining activities. The Event Aggregator component receives previously computed activity instances and converts them into data suited for creating an event log. Using event abstraction concepts (for a detailed description, see Koschmider et al. [62]), activity instances are generated from low-level events. In our work, the individual activity labels, each assigned to a frame, are available as input. This input corresponds to the low-level events following [62]. Since these low-level events are already available in an ordered, clustered form, the first occurrence of a low-level event corresponds to the starting point of a high-level event and the last to its end. Because the high-level events follow one another sequentially, they aggregate to form the activity instances, which can be assigned to a process instance. Finally, the Event Log Exporter component forms an interface for further process mining applications. The hereby created output in the form of an event log can be used in ex-post process mining applications (e.g., for process discovery, conformance checking, and enhancement) since the Event Log Exporter transforms all generated input to a process-mining-compatible format. Through the generation of an event log and the resulting ability to further use the information obtained, DO 5 is also satisfied.

5. Evaluation

5.1. Evaluation of the design specification (EVAL2)

5.1.1. Competing artifacts analysis

To ensure that our artifact addresses the problem outlined in Section 1 and meets our DOs (Section 4.1), we conducted a competing-artifacts analysis as part of EVAL2. In addition, we performed 10 semi-structured interviews following the methodology of Myers and Newman [63]. For the competing artifacts analysis, we compared the design of our RA with existing solutions (i.e., competing artifacts)

Table 2
Analysis of competing artifacts.

	DO 1	DO 2	DO 3	DO 4	DO 5
RAVEE	● Unsupervised clustering mechanisms enable exploratory activity discovery	● Cross-domain applicability through generic processing mechanisms and variability points	● Enables activity extraction at different levels of granularity	● Process/domain expert contextualizes discovered activities	● Dedicated subsystem for event log generation
[17]	○ Gaze points towards areas of interest are assigned to process segments	◑ Not considered, but approach could be transferred to other application areas	○ Only certain, predefined activities, linked to specific areas of interest	◑ No dynamic integration of contextual process knowledge in the information extraction process	● Transform their data into an XES event log for further analysis
[20]	○ Extract process details from unstructured data and enrich existing process models	● Sufficient training data enable transfer to various application areas	○ Enriches predefined process models at their level of granularity	● Contextualizes existing process activities with data from various sources	● Enriches XML log of existing process model
[21]	○ Predefined activities with the help of computer vision capabilities detected	● Sufficient training data enable transfer to various application areas	◑ No explicit abstraction levels, but training at different levels allows varying granularity	◑ No dynamic integration of contextual process knowledge in the information extraction process	● Generation of event log suitable for process mining
[15]	○ Trajectories mapped to predefined work activities by unsupervised clustering	○ Approach is tailored to a specific scenario in the manual assembly	○ Only certain, predefined activities on a given abstraction level	◑ No dynamic integration of contextual process knowledge in the information extraction process	● Extract data in the form of an event log and convert it to XES format
[18]	○ Predefined process phases are detected by material detection and state classification	○ Approach is tailored to a specific scenario in the manual assembly	○ Only predefined process phases on a given abstraction level	◑ No dynamic integration of contextual process knowledge in the information extraction process	○ Do not create an event log, rather utilize data only in real-time analysis
[8]	○ Predefined activities with the help of computer vision capabilities detected	● Variability points and capabilities allow adaptations for different domains	◑ No explicit abstraction levels, but training at different levels allows varying granularity	◑ No dynamic integration of contextual process knowledge in the information extraction process	● Dedicated subsystem for event log generation
[19]	○ Predefined group interactions derived among others from gaze behavior	○ Approach is tailored to a specific scenario in the education environment	○ Only certain, predefined activities, linked to specific behavior	◑ No dynamic integration of contextual process knowledge in the information extraction process	◑ Generate Excel document that can be analyzed with Disco

Table 3
Further information on interview participants (RE = Research Expert; IE = Industry Expert).

Expert abbreviation	Job title	Years of experience in process mining	Experience at the interface of process mining and unstructured data
IE1	Director Hyperautomation Transformation and Strategy	Over 2 years	Yes
IE2	Cluster Lead Data Analytics	Over 4 years	No
IE3	Data Scientist Research and Development	Over 4 years	Yes
IE4	Business Process Excellence Senior Manager	Over 4 years	No
IE5	Business Process Management Expert	Over 6 years	Yes
IE6	Solution Engineer	Over 5 years	Yes
IE7	Principal Consultant	Over 5 years	Yes
RE1	Doctoral Researcher	Over 2 years	Yes
RE2	Doctoral Researcher	Over 3 years	Yes
RE3	Doctoral Researcher and In-house Consultant	Over 2 years	Yes

and assessed their compliance with the DOs. The comparison with the competing artifacts derived in the SLR (as previously discussed, see our repository[4] for details) can be found in Table 2. To enable a quick overview, we assigned circles to all cells that symbolize whether a DO is either met fully (●), partially (◑), or not at all (○). We justified these joint assessments of all co-authors by a qualitative, condensed statement.

5.1.2. Expert interviews

The semi-structured interviews [63] involved discussions with industry experts and researchers specializing in process mining and computer vision. Table 3 provides an overview of participant backgrounds.

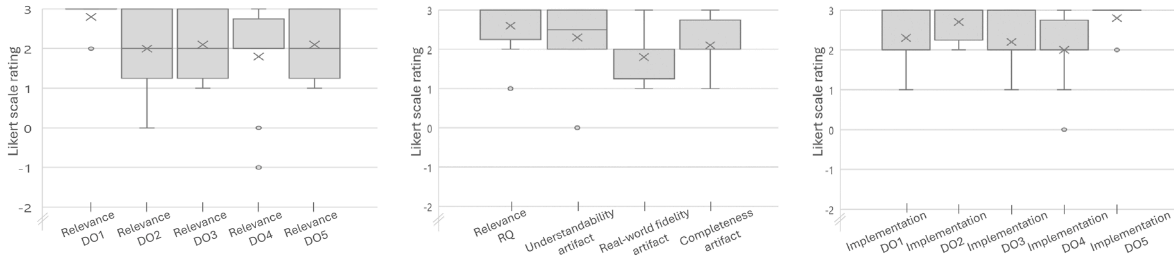
We structured the interviews into two segments, using a Likert scale [64] (−3 to +3) and brief discussions for further assessment. The first segment introduced process mining concepts and evaluated the relevance of our research gap and the resulting DOs, as shown in Fig. 3(a). The second segment introduced relevant computer vision capabilities and the RA, assessing common EVAL2 criteria (understandability, completeness, and real-world fidelity) (see Fig. 3(b) for the experts' ratings) alongside the implementation of the DOs (see Fig. 3(c)). A detailed summary of quantitative responses is provided in our repository.[5]

All experts confirmed the relevance of our research question, noting that although video data is not available in every process, it holds considerable potential where applicable. They emphasized that computational costs must be balanced against the benefits, but acknowledged that our approach establishes a foundation for leveraging video data in process mining while mitigating human bias.

Regarding our DOs, DO 1 was unanimously supported. For DO 2, experts highlighted the trade-off between generalizability and domain-specific effectiveness, agreeing that a flexible framework that can be adjusted to specific use cases is most valuable. DO 3 was well received,

[4] https://github.com/RAVEE-PM/RAVEE/blob/master/Supplementary_Material_SLR.pdf.

[5] https://github.com/RAVEE-PM/RAVEE/blob/master/Supplementary_Material_EVAL2_and_4.pdf.



(a) Responses Regarding the Relevance of Our DOs, Ranging From DO 1 (Far Left) to DO 5 (Far Right).

(b) Responses (From Left to Right) Regarding the Relevance of Our Research Question, the Understandability, the Real-World Fidelity, and the Completeness of Our Artifact.

(c) Responses Depicting How Well the DOs Were Implemented, Ranging From DO 1 (Far Left) to DO 5 (Far Right).

Fig. 3. Boxplot illustrations with the consolidated responses from the interviewed experts.

with two experts (one industry expert (IE) and one research expert (RE)) suggesting that fine-grained results, which can be aggregated across abstraction levels, could suffice, while others highlighted that even an initial overview at moderate granularity can provide essential insights. Most experts endorsed DO 4, noting that contextual information elevates insights, though two experts (one IE, one RE) cautioned that the manual integration could be resource-intensive. Generating standardized event logs (DO 5) was widely seen as important, though some experts noted that event log creation is generally manageable in later stages.

Experts rated the understandability of our RA well, acknowledging that while not self-explanatory, it is comprehensible based on the supplementing explanations. Regarding the real-world fidelity of our RA, experts agreed that our RA enables the development of prototypes for various deployment scenarios. However, they emphasized that specific implementation challenges, such as managing noisy data and addressing privacy concerns, require further consideration. The completeness of the RA was unanimously supported, with suggestions such as incorporating real-time analysis and automated activity labeling.

Regarding the implementation of our DOs, one IE suggested that the human-in-the-loop labeling approach could reintroduce assumptions, suggesting transfer learning to mitigate potential bias (DO 1). Two experts also noted that our focus on video data means process steps outside the frame may be missed. DO 2 was validated without substantial comments. For DO 3, two IEs suggested a live preview of process activities at different granularities. Regarding DO 4, potential enhancements include incorporating process environment data and highlighting deviations from standard processes. DO 5 was rated as '(very) efficiently' implemented by all experts.

Some experts suggested integrating automatic label generation using pre-trained models, which we addressed by introducing the 'Label Suggester' subcomponent. Overall, the feedback strongly supports our approach, confirming that our DOs and RA effectively address the research problem of deriving actual process activities from video data. Some expert recommendations were noted for incorporation in future work, while keeping the current study's scope focused.

5.2. Demonstration: Software prototype (EVAL3)

5.2.1. Instantiation of the artifact as software prototype

To demonstrate its applicability and internal consistency, we instantiated the RAVEE (Fig. 2) as a software prototype. The prototype is designed to extract individual process steps from previously unprocessed video data and convert them into an XES-compliant event log. To evaluate its performance and feasibility, we applied the prototype to two datasets [65,66], following the evaluation approach of Sonnenberg and vom Brocke [24]. To demonstrate the applicability of our RA, not all aspects depicted in Fig. 2 had to be implemented in the instantiation for this paper. Following Tuunanen et al. [67], DSR can be decomposed into smaller, self-contained "decision units" that contribute to distinct elements of design knowledge. This aligns with the principle that only components directly tied to the novel scientific contribution require evaluation. Accordingly, the optional green loop — focused on active learning — was not considered in the instantiation as part of EVAL3. While this loop is intended to increase efficiency and thus applicability to larger datasets in real-world applications, it is not essential to answering our research question. Furthermore, active learning techniques have already been extensively studied and applied across various (scientific) domains and thus do not require further assessment here (see [68] for a general overview). Instead, our instantiation relies on the blue loop in Fig. 2, where the entire dataset — rather than a representative sample — is labeled and then transformed into an event log. This procedure has the disadvantage of increased labeling effort for the process/domain expert. Additionally, separate event logs are created for each selected granularity level, rather than using multi-labeling in the green loop. However, we argue that the present instantiation is sufficient to test the key functions and demonstrate the performance and feasibility of the RA.

The prototype was written in Python, using a prebuilt artifact in C++ for feature trajectory extraction. Docker [69] was employed to create a suitable interface to integrate this artifact into the main program. Our software repository's supplementary material section provides an in-depth description of the instantiation of all the prototype components.[6] Further information on the installation of the necessary applications and on conducting your own test runs using your own dataset can be found in our software repository.[7]

5.2.2. Application of the software prototype to real-world datasets

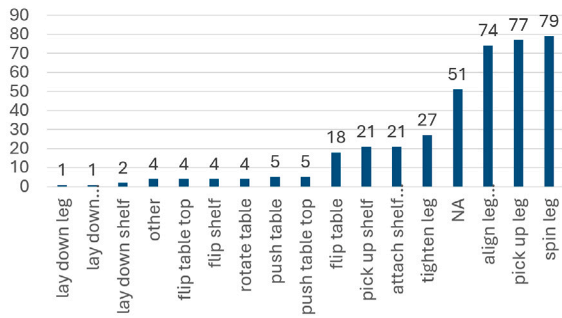
To evaluate the prototype — and by extension, the RA — in terms of performance and feasibility in accordance with EVAL3, we applied it to the IKEA Assembly Dataset [65] and the Solve4X Dataset [66].

The IKEA Assembly Dataset [65] was selected because it meets the prototype's requirements: the recorded assembly steps occur sequentially, which aligns with our event aggregator's assumption that high-level events do not run in parallel. As the current instantiation does not include the Investigation Area Segmenter, parallel activity instances cannot be analyzed (see our repository's supplementary material for details).

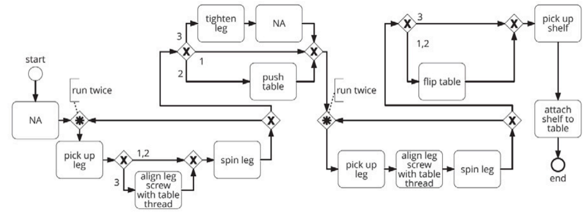
The dataset includes multimodal video recordings of actors assembling furniture. For our evaluation, we focused on the TV bench assembly, which features 43 actors performing the process in varying

[6] https://github.com/RAVEE-PM/RAVEE/blob/master/Supplementary_Material.md.

[7] https://github.com/RAVEE-PM/RAVEE.



(a) Number of Activity Instances in the Ground Truth (GT) of Our Subsample.



(b) Exemplary Assembly Process of Three Instances From the Ground Truth of Our Subsample.

Fig. 4. Information on the Ground Truth (GT) of our IKEA dataset subsample.

sequences and settings. We randomly selected a subsample of 20 videos for analysis. As illustrated in Fig. 4(a), some activities (e.g., picking up or spinning a leg) occur frequently, while others (e.g., laying down the tabletop) appear rarely. Additionally, Fig. 4(b) highlights three process variations, showing that each assembly sequence follows a highly individualized pattern.

The multimodal Solve4X Dataset [66] depicts different IT asset management processes, carried out by two IT administrators and 10 different customers. In addition to the sensor and IT system recordings, video recordings are also available from two different camera perspectives for the scenes, with provided Ground Truth (GT) annotations.

We analyzed Scene 01 and Scene 02, containing three instances of asset disbursement to clients and two instances of the inventorying of new assets. Only the video clips from camera 2 were used for the evaluation, as the instantiation of RAVEE cannot combine different camera angles. Camera 2 was chosen because it covers the entire image area except for the door, i.e., the process steps 'enter/leave room' are the only ones not captured. The two scenes contain 14 different process steps across the two distinct processes. Time intervals without process steps are labeled as 'Not specified'.

As a preparatory step for evaluation, we downsampled the GT annotations for both datasets to one-second intervals and generated corresponding one-second labels per video using our prototype. This involved several steps: First, the prototype generated frame-based clusterings for each video, with the number of clusters matching the number of activity classes in the GT to ensure comparability. Then, through a human-in-the-loop process, individual cluster segments were labeled using only the activity labels found in the GT, again to ensure comparability. The resulting frame-based annotations were then aggregated to one-second intervals. This yielded two comparable logs per video: a True_Log with GT labels and an Extracted_Log with labels generated by the prototype.

5.2.3. Event log analysis

The Extracted_Log produced by our prototype provides a suitable means to evaluate its performance. For this purpose, we assessed how well the Extracted_Log complies with the actual observable behavior according to the GT (True_Log).

First, we assessed the quality of the clustering by comparing it to the GT. We used the Rand Index (RI), which measures the proportion of data point pairs assigned to the same or different clusters in both clusterings. We also applied the Adjusted Rand Index (ARI), a variation of the RI that adjusts for the chance grouping of elements. Across the entire IKEA dataset subsample, the RI is 0.8777, and the ARI is 0.7193. For the Solve4X subsample, the RI is 0.8921, and the ARI is 0.7067. These values are relatively high for both datasets, indicating a strong similarity between GT and our clustering, thus suggesting that the clustering algorithm performed well in capturing the underlying data structure. This is reinforced by the fact that the analysis was conducted on a per-second basis rather than on the level of activity instances. An increased degree of noise and variability is usually expected at such a detailed level.

Next, we evaluated how well the process flow in the Extracted_Log aligns with the True_Log, focusing on three metrics: accuracy (the proportion of observed activities (Extracted_Log) correctly identified relative to the total number of observed activities), precision (the proportion of detected activities that were correctly identified), and recall (the proportion of observable activities correctly detected).

The comparison was conducted on a per-second basis, enabling a direct match between prototype labeled and GT activities. A correct match for an activity pair requires identical labels (e.g., 'spin leg' in both logs). This approach also mitigates issues caused by not implementing the green loop (Fig. 2), as our prototype tends to split temporally extended activities into multiple clusters. Due to the absence of the green loop, this behavior cannot be corrected through its learning process. Although this results in fewer activity labels in the Extracted_Log, expert labeling ensures alignment with the GT.

Given the multi-class classification problems on imbalanced datasets, we used weighted averaging for precision and recall, adjusting for the distribution of detected events. Table 4c summarizes the resulting performance metrics.

The accuracy of 77.22% for the IKEA dataset and 71.13% for the Solve4X dataset can be considered a good result, especially when accounting for the analysis on a per-second basis and the fact that no predefined activities were specified. For the IKEA dataset, it can be seen (Table 4a) that three out of the 17 activities were not detected in any of the videos in our sample. These three activities ('lay down leg', 'lay down table top', and 'push table') are all characterized by their short duration (usually less than two seconds) and by the fact that they are often performed in conjunction with other short activities (e.g., 'push table' right before 'flip table'). A manual examination of the data indicates that the prototype grouped these short activities into larger clusters. Furthermore, the prototype identified an activity instance not listed in the GT but labeled as 'NA' according to the GT. Manual review confirmed it as 'inspect leg'. However, since this label was absent in the GT, it was counted as incorrect for performance evaluation. For the Solve4X dataset (Table 4b), only 'Enter room' and 'Leave room' were not detected, as these activities are not visible in camera 2's field of view.

The average weighted precision of 75.37% for the IKEA dataset and 73.66% for the Solve4X dataset suggests that the prototype correctly identified a large proportion of the detected activities. Table 4a presents the weighted average precision and recall for each of our 18 activities of the IKEA dataset (i.e., the 17 activities from the GT and the manually identified activity 'inspect leg'). Table 4b presents the weighted average precision and recall for the 14 different activities of the Solve4X dataset. It is evident from the precision values of shorter activity instances that these were either not identified at all ('lay down

Table 4
Performance metrics for our sample.

(a) Precision and recall for each label of our Extracted_Log from the IKEA dataset.

	NA	Align leg screw [...]	Attach shelf [...]	Flip shelf	Flip table	Flip table top
Precision	0.8148	0.5000	0.7452	0.5455	0.4948	0.4333
Recall	0.4018	0.2308	0.8997	0.4286	0.6667	0.8125
	Inspect leg	Lay down leg	Lay down shelf	Lay down table top	Other	Pick up leg
Precision	0.0000	0.0000	1.0000	0.0000	0.5455	0.4352
Recall	0.0000	0.0000	0.2000	0.0000	0.3158	0.8165
	Pick up shelf	Push table	Push table top	Rotate table	Spin leg	Tighten leg
Precision	0.4717	0.0000	0.7500	0.5625	0.9359	0.7573
Recall	0.2976	0.0000	0.5000	0.6923	0.9235	0.6094

(b) Precision and recall for each label of our Extracted_Log from the Solve4X dataset.

	Chat	Check asset quality	Checkout asset [...]	Create system entry [...]	Enter room
Precision	0.7500	0.9600	0.7607	1.0000	0.0000
Recall	0.5455	0.8727	0.7807	0.9559	0.0000
	Handover asset [...]	Install and configure [...]	Leave room	Move [...] storage	Not specified
Precision	0.3333	1.0000	0.0000	0.5185	0.6591
Recall	0.8095	0.6829	0.0000	1.0000	0.5088
	Pick [...] self service	Pick [...] warehouse	Test asset quality [...]	Unpack asset	
Precision	0.2000	0.6296	1.0000	1.0000	–
Recall	1.0000	0.9341	0.6316	0.2000	–

(c) Performance measures comparing the results of our prototype (Extracted_Log) with the GT (True_Log).

	RI	ARI	Accuracy	Precision	Recall
IKEA dataset	0.8777	0.7193	0.7722	0.7537	0.7722
Solve4X dataset	0.8921	0.7067	0.7113	0.7366	0.7113

leg', 'lay down table top', and 'push table') or only partially correctly identified ('flip shelf', 'flip table top', 'handover asset to user', 'pick asset from self service desk'). A manual examination revealed that the activity boundaries set by the prototype (start and end time of an activity) slightly deviated in some cases from those of the GT. As a result, some very short activities were not identified at all, or other identified activities were detected for a slightly longer duration than would be correct according to the GT, hence strongly influencing their precision due to their short overall activity duration.

In contrast, the average weighted recall of 77.22% for the IKEA dataset and 71.13% for the Solve4X dataset can mainly be attributed to the fact that some activities did not visually stand out significantly from others. Generally, the prototype performed very well in recognizing visually distinct activities (e.g., 'attach shelf to table', 'spin leg', 'create system entry for asset'). However, activities that do not visually differ much from their preceding or following activities (e.g., 'align leg screw with table thread' and 'pick up leg', or 'pick asset from self service desk' and 'unpack asset') were sometimes assigned to the same activity cluster by the prototype. In the second example, the video clearly shows that unpacking begins as soon as the asset is lifted from the table, blurring the visual boundaries of the activities. In addition, as with the precision, the problem is that shorter activities were not assigned to distinct activity clusters and were therefore not recognized (e.g., 'lay down leg').

To further evaluate the prototype's performance, the activity identification's robustness was evaluated in the next step. For this purpose, video '0007_black_table_02_01_2019_08_16_13_21' from the IKEA dataset was analyzed ten times by our prototype in a dedicated experiment. As before, the clustering was based on the number of clusters according to the GT, and the results were compared on a per-second basis. To assess the consistency of the clustering results across different runs, we computed the pairwise similarity between all 10 clustering outputs using two established metrics: the ARI and the Jaccard coefficient. Both metrics have their own characteristics, such that the ARI is better suited to assessing overall structure, as it allows robust comparisons and eliminates random effects. On the other hand, the Jaccard coefficient is good for looking at pure pairwise similarities, but it can distort cluster evaluations with many small groups. For a comprehensive analysis, we calculated the ARI and Jaccard coefficient for all pairwise comparisons between the 10 clustering results, yielding 45 comparisons. We then computed the mean and standard deviation for each metric to assess the overall stability of the clustering algorithm.

The mean ARI across all pairwise comparisons was 0.8196, with a standard deviation of 0.0617. These results indicate a high level of clustering consistency, with relatively low variability between runs. The high ARI suggests that the clustering structure is well preserved across multiple runs, demonstrating the algorithm's robustness. The mean Jaccard coefficient was 0.6231, with a standard deviation of 0.1562. Compared to the ARI, the Jaccard coefficient yielded a lower mean and a higher standard deviation, indicating that some variations exist in the exact grouping of frames. This suggests that while the overall clustering structure remains stable, there are fluctuations in the precise assignment of some frames. This observation is consistent with our previous findings, which suggest that some very short activities were not identified at all or that identified activities were detected for a slightly longer or shorter duration than would be correct according to the GT, hence strongly influencing the pairwise comparison of frames for the Jaccard coefficient.

Overall, the prototype proved highly effective in detecting and identifying visually distinct activities lasting more than a few seconds (i.e., >= three seconds). It is worth emphasizing that the prototype was able to accurately identify a majority of the activity segments relevant to the process flow on a per-second basis, solely from the video data without any prior input. These promising results highlight the substantial potential for various process mining applications. Thus, EVAL3 has confirmed the performance and feasibility of our prototype and, consequently, of the RAVEE.

5.3. Evaluation of practical use (EVAL4)

5.3.1. Case context and study design

To evaluate the practical use of our instantiated artifact (EVAL4), we conducted a case study on the application of the prototype to the Solve4X dataset [66]. We focused on assessing the RAVEE framework and the results obtained in EVAL3. To gain expert feedback grounded in the dataset's creation and structure, we carried out a semi-structured interview [63] with one of the dataset's authors. His unique domain-specific knowledge made him particularly well-positioned to assess the artifact's effectiveness from the perspective of someone deeply familiar with the dataset's creation, structure, and underlying logic: as a creator of the dataset, the expert was highly familiar with its structure, challenges, and intended use, enabling a well-informed assessment of the artifact's output quality. Moreover, having personally performed the labeling process during dataset creation, the expert was uniquely positioned to reflect on how the prototype could have supported or improved this workflow — providing insights not only on the artifact's technical results, but also on its potential impact in practice.

The interview followed the guidelines for a semi-structured interview by Myers and Newman [63], and was structured in two main segments, combining Likert-scale [64] items (ranging from −3 to 3) with open-ended questions to enable both quantitative assessment and qualitative exploration.

5.3.2. Practical use and usability

The first segment assessed the practical use and usability of RAVEE and the software prototype, including potential areas for improvement. Initially, the expert reflected on the challenges associated with the Solve4X dataset, including the need to integrate two different camera angles to obtain a complete representation of the process, which is not always optimal for observing specific activities. Additionally, the complexity of workflows and process patterns makes it difficult to interpret process activities when relying solely on visual data. Another significant challenge for creating the GT was the subjectivity in defining event boundaries, as start and end points often overlap and require expert judgment.

After being introduced to RAVEE and our software prototype, the expert stated that the system would have been 'very helpful' (Likert rating: 3) in increasing the efficiency and consistency of the dataset labeling process. The prototype's ability to provide consistent event segmentation was considered particularly beneficial for reducing subjective inconsistencies among dataset annotators. The expert pointed out that the dataset authors had differing interpretations of activity boundaries during the initial dataset labeling process, which required manual alignment. The prototype's automated segmentation would have eliminated this inconsistency, ensuring a standardized definition of activity start and endpoints. The expert also rated the prototype's usability and efficiency positively (both rated as 2), suggesting that further improvements would be necessary for large-scale real-world applications, mainly by integrating additional AI-assisted support, such as the 'Label Suggester' subcomponent or the active learning loop of RAVEE. Although these features were not included in our prototype, as they are not part of our contribution and have already been evaluated (see Section 5.2.1), their integration could further enhance the system's efficiency in practical deployment.

Upon reviewing the EVAL3 results, the expert found the prototype's performance highly promising and noted that the system's effectiveness would improve further by integrating the active learning loop. He also reflected on a limitation of the approach, namely that the prototype relies on visually distinct activities to ensure clear differentiation. However, he viewed this as a potential advantage rather than a shortcoming, as it could help identify instances where activities were not performed properly. For example, rushed task execution might make an activity appear less visually distinct, highlighting areas where process deviations occur. Overall, the expert found the generated process instances 'meaningful' (2) and recognized the prototype's ability to identify new, previously unconsidered process steps as 'high' (2), noting that the results 'very well' (3) matched his expectations. He concluded that integrating the prototype into the Solve4X dataset's GT labeling workflow would have been highly beneficial. He recommended exploring additional ways to enhance human-in-the-loop support to improve efficiency further.

5.3.3. Insights on activity granularity

The second segment focused on the activity extraction on different levels of granularity. The expert emphasized that granularity is a crucial factor when analyzing video-based datasets, as process activities can be captured at various levels of detail. He stressed that the level of granularity required depends on the specific use case, as different stakeholders may require different levels of process detail. For instance, a high-level overview of key process steps may be sufficient for managerial decision-making, whereas fine-grained insights into individual actions may be essential for operators or analysts conducting detailed workflow evaluations. The expert expressed to be 'confident' (2) in the clustering results across different levels of granularity, finding them informative and valuable depending on the intended application. He also highlighted the importance of a recommended granularity setting, as selecting an appropriate level can be challenging when first exploring a dataset. Without clear guidance, those with limited prior knowledge of the dataset may find it difficult to determine the most meaningful level of granularity for their analysis. The expert suggested integrating hierarchical clustering in future work to enhance the utility of granularity analysis further. This approach would allow users to begin with a coarse-grained process view and progressively refine their analysis by 'drilling down' into finer details as needed. Such a feature could also prove valuable in analyzing overlapping process steps, addressing one of the key challenges identified in the dataset. Hierarchical clustering could significantly improve process understanding and exploration by enabling analysts to first explore broader process structures before zooming into specific details.

To sum up, this case study provided valuable insights into the application of the RAVEE framework in a real-world process mining scenario. While the findings are not statistically generalizable, they reflect the practical implications of deploying the artifact in a real-world setting. The expert's experience as a dataset author enabled a grounded evaluation of the prototype's functionality, limitations, and potential for impact. The findings confirm the practical applicability and usefulness of the instantiated artifact and, by extension, the RAVEE. The expert's feedback highlights the potential of the RAVEE to improve process transparency — and, consequently, decision support — by enabling the extraction of actual process behavior rather than assumed behavior through unsupervised process event generation.

6. Conclusion

6.1. Summary and contribution

This paper presents the RAVEE, an RA designed to extract actual process behavior from unstructured video data. Unlike approaches requiring predefined activity sets, RAVEE supports unbiased, data-driven event detection through unsupervised extraction of actual process behavior. Therefore, RAVEE enhances decision support by ensuring that insights are based on actual observed behavior rather than expectations, enabling the identification of deviations, unexpected steps, and emerging patterns. Following a DSR methodology, we derived DOs from literature and expert input to guide development. We validated the RAVEE through semi-structured interviews and tested a prototypical instantiation on two public datasets, demonstrating its capability to generate unbiased event logs from unstructured video data across different environments and actors. A case study further supported the practical applicability of the artifact.

This work contributes to the scientific discourse by providing an RA for fully automatic, unsupervised extraction of process-relevant activity instances from video data, which are later enriched to construct valid event logs. The RA provides a flexible framework with variation points for different use cases and thus serves as a foundation for future approaches and application-specific design variants. Unlike related work, our RAVEE does not rely on predefined activity sets, reducing bias in the identification of relevant process steps and enabling objective, data-driven process discovery. This, in turn, fosters process transparency, strengthens decision support and facilitates optimization. The accompanying prototype further lowers barriers for adoption by demonstrating how the RA can be instantiated across diverse applications.

6.2. Limitations and future research

Despite its potential, our RA has several limitations that should be addressed in future research and real-world implementations. First, privacy concerns were not addressed. Since video data may contain sensitive information about recorded individuals, future implementations must incorporate mechanisms for data anonymization and secure handling to ensure compliance with privacy regulations. Second, we focused solely on video data. While video data alone provides valuable insights, integrating additional (un-)structured data sources — such as audio recordings or sensor data — could enhance process understanding and improve overall analysis. Third, our prototype omits certain components of the RAVEE: the active learning loop, the Investigation Area Segmenter, and the Label Suggester. The active learning loop enables automatic labeling based on previously collected information, reducing manual effort and allowing for multi-granularity analysis. The Investigation Area Segmenter facilitates detecting multiple parallel process steps within a single frame. Without it, our implementation assumes only one activity is executed per frame, limiting its applicability in environments where multiple processes occur simultaneously. Future work should incorporate these components to improve efficiency and analytical depth. Fourth, the RA was not tested in a real-world deployment. While our datasets simulate realistic conditions (e.g., varying backgrounds, different actors, and environmental noise), and we conducted a case study on one of the datasets — leveraging the insights of one of its authors to gather in-depth expert feedback — this setting does not fully reflect the complexities of a live operational environment. Although the case study allowed us to derive rich, context-specific insights — particularly regarding output quality and potential improvements to the labeling workflow — it does not support statistical generalization. Future work should broaden the empirical foundation through additional deployments or case studies to assess long-term robustness, resilience to data noise, and adaptability to unseen environments, thereby further validating our approach. Additionally, the RAVEE is not suitable for all use cases. Processes with highly variable motion sequences across instances pose challenges for clustering, as the approach relies on visual similarities to group events. Similarly, complex or visually indistinct movements may hinder accurate event detection. However, this limitation is inherent to process mining, which is generally most meaningful for processes with recurring, structurally similar instances. Finally, since our primary contribution is the RA rather than the instantiated prototype, we did not aim for an optimized implementation. Runtime performance or efficiency (e.g., through active learning or optimized labeling processes) were not primary goals but represent important avenues for refinement.

Several avenues for future research emerge from this work. Privacy-preserving mechanisms must be developed to address the use of video data containing personally identifiable information. Technologies such as anonymizing video cameras (e.g., [70]) offer promising solutions but require further evaluation to assess their impact on event detection performance and overall process mining outcomes. While this study focuses on event extraction in the XES format, future work could explore its extension to the Object-Centric Event Log (OCEL) standard. OCEL enables many-to-many relationships between events and objects, offering richer representations of complex processes [71]. Applying unsupervised event detection in this context could yield deeper insights into interactions between multiple entities within a process, thereby enhancing decision support in object-centric environments. Furthermore, the development and dissemination of video datasets related to process mining would significantly benefit the field. These datasets should represent various activity instances in realistic environments. Moreover, integrating additional (un)structured data sources — such as sensor logs or audio streams — with video data presents an opportunity to construct more comprehensive representations of underlying processes. An approach that creates a comprehensive process representation can thus generate significant added value. Additionally, the inherent uncertainty associated with video-based analysis requires further research. Uncertainty may arise from probabilistic object detection or conflicting information between data sources. Approaches could use process trees that provide uncertain process paths with the corresponding probabilities or methods that integrate process experts. Sixth, handling short or subtle activities remains a challenge, particularly at higher abstraction levels. As proposed by the dataset author in our case study, integrating hierarchical clustering presents a promising approach to enable seamless investigation of specific process segments within the software demonstrator. This would allow users to begin with a coarse-grained process view and progressively refine specific segments through drill-down analysis, thereby enhancing granularity control and supporting more flexible, user-driven exploration of complex process behavior. Finally, future work should focus on improving the efficiency and scalability of the RAVEE and its instantiation. Optimizing event detection and labeling algorithms and leveraging hardware acceleration could enhance performance. Refining feature extraction and clustering mechanisms would also improve adaptability across diverse process environments.

CRediT authorship contribution statement

Niklas Wördehoff: Writing – original draft, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Andreas Egger:** Writing – original draft, Validation, Project administration, Methodology, Conceptualization. **Wolfgang Kratsch:** Writing – review & editing, Validation, Supervision, Methodology. **Fabian König:** Writing – review & editing, Validation, Methodology, Conceptualization. **Maximilian Röglinger:** Writing – review & editing, Validation, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partly supported by the Bavarian Research Foundation (Bayerische Forschungsstiftung) [grant number AZ-1550-20].

Data availability

All research data and code are available in our GitHub repository that is referenced in the paper.

References

[1] W. van der Aalst, A. Adriansyah, A.K.A. de Medeiros, F. Arcieri, T. Baier, et al., Process mining manifesto, in: Business Process Management Workshops, Springer, Berlin, Heidelberg, 2012, pp. 169–194, http://dx.doi.org/10.1007/978-3-642-28108-2_19.

[2] W. van der Aalst, Academic view: Development of the process mining discipline, in: Process Mining in Action, Springer, Cham, 2020, pp. 181–196, http://dx.doi.org/10.1007/978-3-030-40172-6_21.

[3] W. van der Aalst, ICPM keynote on 20 years of process mining research: Accomplishments, challenges, and open problems, 2019, URL https://www.pads.rwth-aachen.de/cms/PADS/Der-Lehrstuhl/Aktuelle-Meldungen/~gedil/ICPM-Keynote-on-20-Years-of-Process-Mini/.

[4] V. Mehta, M. Kerremans, C. Tornbohm, F. Biscotti, Forecast analysis: Process mining, worldwide, 2021, URL https://www.gartner.com/en/documents/4007520.
[5] A. Konrad, Celonis raises $1 billion at $11 billion valuation, making it New York's —And Germany's — Most valuable startup, 2021, URL https://www.forbes.com/sites/gradsoflife/2022/03/31/3-ways-companies-can-advance-diversity-equity–inclusion-beyond-their-own-four-walls/?.
[6] W. van der Aalst, Data science in action, in: Process Mining: Data Science in Action, Springer, Berlin, Heidelberg, 2016, pp. 3–23, http://dx.doi.org/10.1007/978-3-662-49851-4_1.
[7] C. Taylor, Structured vs unstructured data, 2021, URL https://www.datamation.com/big-data/structured-vs-unstructured-data/.
[8] W. Kratsch, F. König, M. Röglinger, Shedding light on blind spots – developing a reference architecture to leverage video data for process mining, Decis. Support Syst. 158 (2022) 113794, http://dx.doi.org/10.1016/j.dss.2022.113794.
[9] D. Beverungen, J.C.A.M. Buijs, J. Becker, C. Di Ciccio, W.M.P. van der Aalst, et al., Seven paradoxes of business process management in a hyper-connected world, Bus. Inf. Syst. Eng. 63 (2) (2021) 145–156, http://dx.doi.org/10.1007/s12599-020-00646-z.
[10] R.B. Banziger, A. Basukoski, Chaussalet Thierry, Discovering business processes in CRM systems by leveraging unstructured text data, in: 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems, HPCC/SmartCity/DSS, 2018, pp. 1571–1577, http://dx.doi.org/10.1109/HPCC/SmartCity/DSS.2018.00257.
[11] Y. Nakayama, M. Mori, Y. Naruse, H. Morikawa, The process discovery approaches for decision making in sales activities, in: 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems, ISIS, 2018, pp. 1394–1399, http://dx.doi.org/10.1109/SCIS-ISIS.2018.00217.
[12] D. Jlailaty, D. Grigori, K. Belhajjame, Mining business process activities from email logs, in: 2017 IEEE International Conference on Cognitive Computing, ICCC, 2017, pp. 112–119, http://dx.doi.org/10.1109/IEEE.ICCC.2017.28.
[13] M.L. van Eck, N. Sidorova, W. van der Aalst, Enabling process mining on sensor data from smart products, in: 2016 IEEE Tenth International Conference on Research Challenges in Information Science, RCIS, 2016, pp. 1–12, http://dx.doi.org/10.1109/RCIS.2016.7549355.
[14] F. Leotta, M. Mecella, D. Sora, Visual process maps: a visualization tool for discovering habits in smart homes, J. Ambient. Intell. Humaniz. Comput. 11 (5) (2020) 1997–2025, http://dx.doi.org/10.1007/s12652-019-01211-7.
[15] S. Knoch, N. Herbig, S. Ponpathirkoottam, F. Kosmalla, P. Staudt, D. Porta, P. Fettke, P. Loos, Sensor-based human–process interaction in discrete manufacturing, J. Data Semant. 9 (1) (2020) 21–37, http://dx.doi.org/10.1007/s13740-019-00109-z.
[16] G. Elkoumy, S.A. Fahrenkrog-Petersen, M.F. Sani, A. Koschmider, F. Mannhardt, S.N. von Voigt, M. Rafiei, L. von Waldthausen, Privacy and confidentiality in process mining: Threats and research challenges, ACM Trans. Manag. Inf. Syst. 13 (1) (2022) 1–17, http://dx.doi.org/10.1145/3468877.
[17] A. Abbad Andaloussi, F. Zerbato, A. Burattin, T. Slaats, T.T. Hildebrandt, B. Weber, Exploring how users engage with hybrid process artifacts based on declarative process models: a behavioral analysis based on eye-tracking and think-aloud, Softw. Syst. Model. 20 (5) (2021) 1437–1464, http://dx.doi.org/10.1007/s10270-020-00811-8.
[18] S. Knoch, N. Herbig, S. Ponpathirkoottam, F. Kosmalla, P. Staudt, P. Fettke, P. Loos, Enhancing process data in manual assembly workflows, in: Business Process Management Workshops, Springer International Publishing, 2019, pp. 269–280, http://dx.doi.org/10.1007/978-3-030-11641-5_21.
[19] Q. Zhou, W. Suraworachet, M. Cukurova, Detecting non-verbal speech and gaze behaviours with multimodal data and computer vision to interpret effective collaborative learning interactions, Educ. Inf. Technol. 29 (2024) 1071–1098, http://dx.doi.org/10.1007/s10639-023-12315-1.
[20] M. Fichtner, S. Jablonski, A specification of how to extract relevant process details to improve process models, in: J. Filipe, M. Śmiałek, A. Brodsky, S. Hammoudi (Eds.), Enterprise Information Systems, Springer Nature Switzerland, Cham, 2023, pp. 391–414.
[21] A. Lepsien, A. Koschmider, W. Kratsch, Analytics pipeline for process mining on video data, in: C. Di Francescomarino, A. Burattin, C. Janiesch, S. Sadiq (Eds.), Business Process Management Forum, Springer Nature Switzerland, 2023, pp. 196–213.
[22] R. Andrews, C.G.J. van Dun, M.T. Wynn, W. Kratsch, M. Röglinger, A.H. ter Hofstede, Quality-informed semi-automated event log generation for process mining, Decis. Support Syst. 132 (2020) 113265, http://dx.doi.org/10.1016/j.dss.2020.113265.
[23] K. Peffers, T. Tuunanen, M.A. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, J. Manage. Inf. Syst. 24 (3) (2007) 45–77, http://dx.doi.org/10.2753/MIS0742-1222240302.
[24] C. Sonnenberg, J. Vom Brocke, Evaluation patterns for design science research artefacts, in: M. Helfert, B. Donnellan (Eds.), Practical Aspects of Design Science, Springer Berlin Heidelberg, 2011, pp. 71–83, http://dx.doi.org/10.1007/978-3-642-33681-2_7.
[25] L. Reinkemeyer, Process mining in a nutshell, in: Process Mining in Action, Springer, Cham, 2020, pp. 3–10, http://dx.doi.org/10.1007/978-3-030-40172-6_1.
[26] A. Burattin, F.M. Maggi, A. Sperduti, Conformance checking based on multi-perspective declarative process models, Expert Syst. Appl. 65 (2016) 194–211, http://dx.doi.org/10.1016/j.eswa.2016.08.040.
[27] M. La Rosa, M. Dumas, A.H. ter Hofstede, J. Mendling, Configurable multi-perspective business process models, Inf. Syst. 36 (2) (2011) 313–340, http://dx.doi.org/10.1016/j.is.2010.07.001.
[28] W.M.P. van der Aalst, Operational support, in: Process Mining: Data Science in Action, Springer Berlin Heidelberg, Berlin, Heidelberg, 2016, pp. 301–321, http://dx.doi.org/10.1007/978-3-662-49851-4_10.
[29] L. Reinkemeyer, How to get started, in: Process Mining in Action, Springer, Cham, 2020, pp. 11–14, http://dx.doi.org/10.1007/978-3-030-40172-6_2.
[30] M.J. Jans, From relational database to valuable event logs for process mining purposes: a procedure, 2017, http://dx.doi.org/10.13140/RG.2.2.11343.69289.
[31] W. van der Aalst, Getting the data, in: Process Mining: Data Science in Action, Springer Berlin Heidelberg, 2016, pp. 125–162, http://dx.doi.org/10.1007/978-3-662-49851-4_5.
[32] W. van der Aalst, A. Weijters, Process mining: a research agenda, Comput. Ind. 53 (3) (2004) 231–244, http://dx.doi.org/10.1016/j.compind.2003.10.001.
[33] S. van Zelst, F. Mannhardt, M. de Leoni, A. Koschmider, Event abstraction in process mining -literature review and taxonomy, Granul. Comput. 6 (2021) http://dx.doi.org/10.1007/s41066-020-00226-2.
[34] C. Gunther, H. Verbeek, XES - Standard Definition, in: BPM Reports, BPMcenter.org, 2014.
[35] R. Szeliski, Computer Vision: Algorithms and Applications, Springer, London, 2010, http://dx.doi.org/10.1007/978-1-84882-935-0.
[36] D.A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Prentice Hall Professional Technical Reference, 2002.
[37] V. Wiley, T. Lucas, Computer vision and image processing: A paper review, Int. J. Artif. Intell. Res. 2 (1) (2018) 22, http://dx.doi.org/10.29099/ijair.v2i1.42.
[38] P. Sinha, B. Balas, Y. Ostrovsky, R. Russell, Face recognition by humans: Nineteen results all computer vision researchers should know about, Proc. IEEE 94 (11) (2006) 1948–1962, http://dx.doi.org/10.1109/JPROC.2006.884093.
[39] G. Gerig, W. Kuoni, R. Kikinis, O. Kübler, Medical imaging and computer vision: An integrated approach for diagnosis and planning, in: H. Burkhardt, K.H. Höhne, B. Neumann (Eds.), Mustererkennung 1989, vol. 219, Springer Berlin Heidelberg, 1989, pp. 425–432, http://dx.doi.org/10.1007/978-3-642-75102-8_64.
[40] J. Janai, F. Güney, A. Behl, A. Geiger, Computer vision for autonomous vehicles: Problems, datasets and state of the art, Found. Trends® Comput. Graph. Vis. 12 (1–3) (2020) 1–308, http://dx.doi.org/10.1561/0600000079.
[41] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G.V. Hernandez, L. Krpalkova, D. Riordan, J. Walsh, Deep learning vs. Traditional computer vision, in: Advances in Computer Vision, Springer International Publishing, 2020, pp. 128–144, http://dx.doi.org/10.1007/978-3-030-17795-9_10.
[42] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge, Int. J. Comput. Vis. (IJCV) 115 (3) (2015) 211–252, http://dx.doi.org/10.1007/s11263-015-0816-y.
[43] J. Teuwen, N. Moriakov, Chapter 20 - convolutional neural networks, in: G. Fichtinger, D. Rueckert, S.K. Zhou (Eds.), Handbook of Medical Image Computing and Computer Assisted Intervention, Academic Press is an imprint of Elsevier and Credo Reference, 2022, pp. 481–501, http://dx.doi.org/10.1016/B978-0-12-816176-0.00025-9.
[44] M.S. Nixon, Feature Extraction and Image Processing for Computer Vision, fourth ed., Academic Press, an imprint of Elsevier, 2020, http://dx.doi.org/10.1016/C2011-0-06935-1.
[45] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[46] M.S. Sarfraz, N. Murray, V. Sharma, A. Diba, L. van Gool, R. Stiefelhagen, Temporally-weighted hierarchical clustering for unsupervised action segmentation, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, 2021, http://dx.doi.org/10.1109/cvpr46437.2021.01107.
[47] J. Chaki, N. Dey, A Beginner's Guide to Image Preprocessing Techniques, in: CRC Focus Series, CRC Press, Taylor & Francis Group, Boca Raton, FL, 2019, http://dx.doi.org/10.1201/9780429441134.
[48] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, A. Courville, Describing videos by exploiting temporal structure, in: 2015 IEEE International Conference on Computer Vision, ICCV, 2015, pp. 4507–4515, http://dx.doi.org/10.1109/ICCV.2015.512.
[49] H. Wang, A. Klaser, C. Schmid, C.-L. Liu, Action recognition by dense trajectories, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2011, pp. 3169–3176, http://dx.doi.org/10.1109/CVPR.2011.5995407.
[50] H. Wang, C. Schmid, Action recognition with improved trajectories, in: 2013 IEEE International Conference on Computer Vision, IEEE, 2013, pp. 3551–3558, http://dx.doi.org/10.1109/ICCV.2013.441.
[51] J. Sánchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the Fisher vector: Theory and practice, Int. J. Comput. Vis. 105 (3) (2013) 222–245, http://dx.doi.org/10.1007/s11263-013-0636-x.

[52] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, Work. Stat Learn. Comput. Vis. ECCV (2004).
[53] D. Oneata, J. Verbeek, C. Schmid, Action and event recognition with fisher vectors on a compact feature set, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1817–1824.
[54] A. Shmilovici, Support vector machines, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Scholars Portal, 2005, pp. 257–276, http://dx.doi.org/10.1007/0-387-25465-X_12.
[55] C. Lea, M.D. Flynn, R. Vidal, A. Reiter, G.D. Hager, Temporal convolutional networks for action segmentation and detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, vol. 9, IEEE, 2017, pp. 21–37, http://dx.doi.org/10.1109/cvpr.2017.113, 1.
[56] S. Gregor, A.R. Hevner, Positioning and presenting design science research for maximum impact, MIS Q. 37 (2) (2013) 337–355, http://dx.doi.org/10.25300/MISQ/2013/37.2.01.
[57] M. Galster, P. Avgeriou, Empirically-grounded reference architectures, in: Proceedings of the Joint ACM SIGSOFT Conference – QoSA and ACM SIGSOFT Symposium – ISARCS on Quality of Software Architectures – QoSA and Architecting Critical Systems – ISARCS, Association for Computing Machinery, 2011, pp. 153–158, http://dx.doi.org/10.1145/2000259.2000285.
[58] B. Kitchenham, S.M. Charters, Guidelines for Performing Systematic Literature Reviews in Software Engineering, Tech. Rep. 2, Keele University and Durham University Joint Report, 2007, URL https://www.elsevier.com/_data/promis_misc/525444systematicreviewsguide.pdf.
[59] W. van der Aalst, Process Mining: Data Science in Action, second ed., Springer Berlin Heidelberg, 2016, http://dx.doi.org/10.1007/978-3-662-49851-4.
[60] G. Kerpedzhiev, U. König, M. Roeglinger, M. Rosemann, An exploration into future business process management capabilities in view of digitalization: Results from a delphi study, Bus. Inf. Syst. Eng. 63 (2021) http://dx.doi.org/10.1007/s12599-020-00637-0.
[61] A. Koschmider, M. Aleknonytė-Resch, F. Fonger, C. Imenkamp, A. Lepsien, K. Apaydin, D. Janssen, D. Langhammer, T. Ziolkowski, Y. Zisgen, Process mining for unstructured data: Challenges and research directions, in: Modellierung 2024, Gesellschaft für Informatik e.V., 2024, http://dx.doi.org/10.18420/MODELLIERUNG2024_012.
[62] A. Koschmider, F. Mannhardt, T. Heuser, On the contextualization of event-activity mappings, in: F. Daniel, Q.Z. Sheng, H. Motahari (Eds.), Business Process Management Workshops, in: Lecture Notes in Business Information Processing, vol. 342, Springer International Publishing, 2019, pp. 445–457, http://dx.doi.org/10.1007/978-3-030-11641-5_35.
[63] M.D. Myers, M. Newman, The qualitative interview in IS research: Examining the craft, Inf. Organ. 17 (1) (2007) 2–26, http://dx.doi.org/10.1016/J.INFOANDORG.2006.11.001.
[64] A. Joshi, S. Kale, S. Chandel, D.K. Pal, Likert scale: Explored and explained, Br. J. Appl. Sci. Technol. 7 (4) (2015) 396–403, http://dx.doi.org/10.9734/BJAST/2015/14975.
[65] Y. Ben-Shabat, X. Yu, F.S. Saleh, D. Campbell, C. Rodriguez-Opazo, H. Li, S. Gould, The IKEA ASM dataset: Understanding people assembling furniture through actions, objects and pose, 2023, http://dx.doi.org/10.48550/arXiv.2007.00394.
[66] T. Fehrer, A. Egger, D. Chvirova, J. Wittmann, N. Wördehoff, W. Kratsch, M. Röglinger, Business Processes in IT Asset Management Multimedia Event Log, 2024, http://dx.doi.org/10.6084/m9.figshare.25246291.
[67] T. Tuunanen, R. Winter, J. vom Brocke, Dealing with complexity in design science research: A methodology using design echelons, MIS Q. 48 (2024) 427–458, http://dx.doi.org/10.25300/MISQ/2023/16700.
[68] B. Settles, Active Learning Literature Survey, Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009, URL http://digital.library.wisc.edu/1793/60660.
[69] Docker, Docker documentation, 2022, URL https://docs.docker.com/.
[70] BAUTA GmbH, Blind sensors from BAUTA, 2024, URL https://bauta.io/en/.
[71] A.F. Ghahfarokhi, G. Park, A. Berti, W.M.P. van der Aalst, OCEL: A standard for object-centric event logs, in: L. Bellatreche, M. Dumas, P. Karras, R. Matulevičius, A. Awad, M. Weidlich, M. Ivanović, O. Hartig (Eds.), New Trends in Database and Information Systems, Springer International Publishing, 2021, pp. 169–175, http://dx.doi.org/10.1007/978-3-030-85082-1_16.

Niklas Wördehoff started his studies in Industrial Engineering in October 2017 at the University of Augsburg, where he obtained his Bachelor's degree in 2021. From 2021 to 2024, he then studied in the Industrial Engineering Masters program at the University of Augsburg. Niklas also gained practical experience at Rolls-Royce Motor Cars (part of the BMW Group) in Munich. At the FIM Research Center for Information Management and the Branch Business & Information Systems Engineering of the Fraunhofer FIT, he started as a student research assistant in March 2018 and is a doctoral candidate since June 2024. Within his research activities, Niklas Wördehoff mainly focuses on data-driven business process management with a special focus on process mining.

Andreas Egger studied Business Administration at the University of Ulm. He then enrolled in the elite graduate program Finance & Information Management at the Technical University of Munich, the University of Augsburg, and the University of Bayreuth. Andreas gained practical experience at BMW Group and MAN Truck&Bus AG in Munich. For a research project on process mining and robotic process automation, he visited Queensland University of Technology (QUT) in Brisbane/Australia. At the FIM Research Center for Information Management and the Branch Business & Information Systems Engineering of the Fraunhofer FIT, he started as a student research assistant in May 2019 and is a doctoral candidate since May 2020. Within his research activities, Andreas Egger mainly focuses on data-driven business process management and especially on unstructured data in process mining.

Wolfgang Kratsch studied Computer Science and Information Management at the University of Augsburg. Wolfgang gained his PhD at the University of Bayreuth in December 2020. In 2020, Wolfgang also co-founded the Fraunhofer spinoff credium. He also co-founded the Fraunhofer Center for Process Intelligence (CPI). Since 2022, Wolfgang holds the professorship for Applied AI in Digital Value Creation at the Technical University of Applied Sciences Augsburg and is additionally affiliated as Director of the FIM Research Center for Information Management. He works in leading position with the Branch Business & Information Systems Engineering of the Fraunhofer FIT. Most of Wolfgang's work centers around process mining and process analytics.

Fabian König started his studies in Business & Information Systems Engineering in October 2014 at the University of Augsburg, where he obtained his Bachelor's degree in 2017. From 2017 to 2020, he studied in the elite graduate program Finance & Information Management at the Technical University of Munich, the University of Augsburg, and the University of Bayreuth. At the FIM Research Center for Information Management and the Branch Business & Information Systems Engineering of the Fraunhofer FIT, he started as a student research assistant in May 2015, as a doctoral candidate in 2020 and gained his PhD at the University of Bayreuth in January 2024. Within his research activities, Fabian König mainly focuses on data-driven business process management, as well as the application of artificial intelligence in process mining.

Maximilian Röglinger holds the Chair of Information Systems and Business Process Management at the University of Bayreuth and is Adjunct Professor in the School of Management at Queensland University of Technology. Maximilian serves as Academic Director of the FIM Research Center for Information Management, where he heads the business process management (BPM) group. Maximilian also works as deputy head of the Branch Business & Information Systems Engineering of the Fraunhofer FIT. Most of Maximilian's work centers around BPM, customer relationship management, and digital transformation. He publishes in journals like Business & Information Systems Engineering, Business Process Management Journal, Decision Support Systems, Journal of the Association for Information Systems, and Journal of Strategic Information Systems. Maximilian is highly engaged in projects with companies such as Deutsche Bahn, Deutsche Bank, Hilti, Infineon Technologies, Radeberger, and Siemens. Maximilian earned his PhD at the University of Augsburg and holds a Diploma in Business and Information Systems Engineering from the University of Bamberg.