Visual breeder and travel document authenticator

Henri Bouma [1*], Jorge Melo [1], Johan-Martijn ten Hove [1], Luca Ballan [1], Muriel van der Spek [1], Arthur van Rooijen [1], Fieke Hillerström [1], Dominic Wolbers [2], Niels Dolstra [3]

[1] TNO, The Hague, The Netherlands
[2] Immigration and Naturalization Service IND, Zwolle, The Netherlands
[3] Royal Netherlands Marechaussee, Schiphol/Ter Apel, The Netherlands

ABSTRACT

The use of AI technologies improves document authentication, which supports border guards and immigration services to fight document fraud, identity theft, illegal border crossing and illegal migration. This paper shows a novel application for the authentication of travel and breeder documents that preserves privacy during the process. The new capabilities include robust processing of images from mobile phones, federated-learning based training, data-driven discovery of new rules and knowledge-based tactical anomaly detection. The processing allows the tactical analysis of many data elements, such as consistency checks, multi-language support and validity of data elements.

Keywords: Document Authentication, Artificial Intelligence, Breeder documents, Travel documents.

1. INTRODUCTION

Document authentication is vital for border guards and immigration services since it helps prevent cross-border crime, ensures legal movement, combats identity fraud and enhances overall security. By verifying the authenticity of travel documents (e.g., passports), border guards can stop illegal activities like terrorism and smuggling, protect citizens and support automated systems for faster and more consistent checks. By verifying the authenticity of breeder documents (e.g., birth certificates), immigration authorities can prevent fraud, confirm the identity of applicants and ensure that only eligible individuals are granted visas, residency, or citizenship. This process helps maintain national security, supports legal compliance and it maintains the integrity of national borders.

Others already worked on the verification of well standardized passport, e.g., by using the machine readable zone (MRZ) and electronic information on the chip [14]. Earlier work already showed an initial flexible pipeline for the automatic authentication of less standardized documents, such as breeder documents [1][2]. However, this pipeline only included scans from document scanners, fixed number of data fields, a centralized training database and limited anomaly detection.

The novel contribution of this paper includes robust processing of images from mobile phones, federated-learning based training, flexible handling of tabular data with varying number of rows and hybrid anomaly detection that combines data-driven rule discovery with knowledge-based tactical anomaly detection. The new flexible pipeline can be applied travel, identity and breeder documents.

The method and results are described in Section 2, the workflow and graphical user interface (GUI) of the application is described in Section 3, evaluation by end users is described in Section 4 and finally, the conclusion is summarized in Section 5.

* henri.bouma@tno.nl

H. Bouma, J. Melo, J.-M. ten Hove, L. Ballan, M. van der Spek, A. van Rooijen, F. Hillerström, D. Wolbers, N. Dolstra, "Visual breeder and travel document authenticator", Artificial Intelligence for Security and Defence Applications, Proc. SPIE, vol. 13679, (2025). https://doi.org/10.1117/12.3069992

2. METHOD AND RESULTS

This section first presents the overall architecture in Section 2.1 and then clarifies various modules in other subsections.

2.1 Application architecture

The architecture and its mapping on hardware is shown in Figure 1. This figure shows the graphical user interface (GUI), command and control (C&C) layer, an internal database and document authentication components (federated or not). Note that the GUI does not directly access the database or components, but only through the C&C-layer. During training with federated learning (FL), central aggregation is used which runs on one of the computers. FL enables the training on datasets of multiple organizations while preserving the privacy by sharing only the AI-model updates and not the local data. This helps avoiding the cross-border sharing of personal data [15].

The workflow and relation between document-authentication components is shown in Figure 2. The system supports rapid automated processing and human verification. The modular approach enables human oversight. The output of each module is understandable and verifiable. The decision of the application (in the tactical rules) can be traced back to its cause.

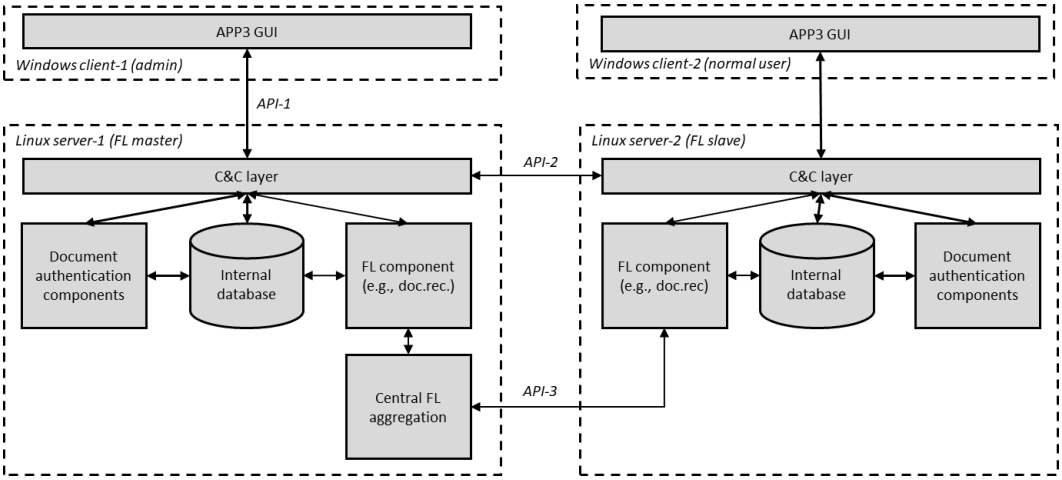


Figure 1: Application architecture mapped on hardware with a separation between GUI, database and analytics components.

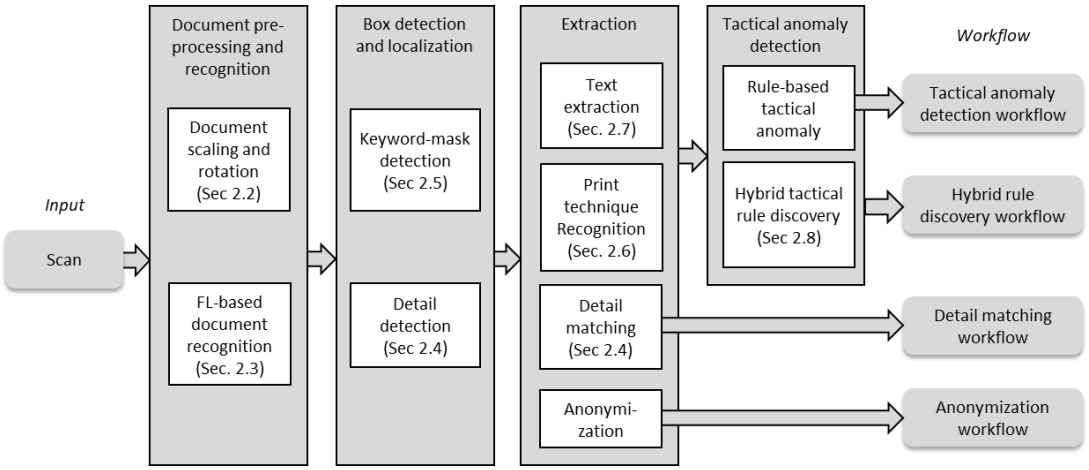


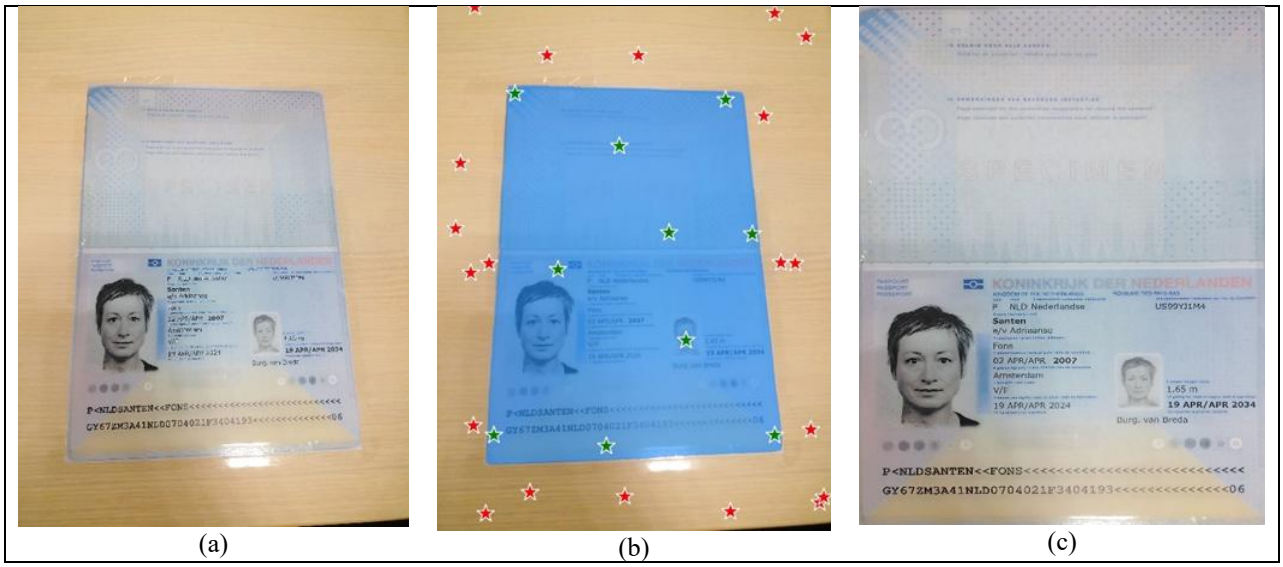
Figure 2: Workflow and relations between document-authentication components.

2.2 Document scaling and rotation for mobile phones

Inspecting documents captured by a smartphone rather than with a document scanner is relevant because it is more convenient since no scanner is required to store the documents. Nevertheless, the quality of the detected documents in the pictures may vary based on the smartphone type/settings, lighting conditions and angle of the camera and its distance to the document. To overcome these uncertainties, several measures are taken into account.

Several methods exist to detect a document in an image. One possible method uses background removal using a segmentation model [Gatis, 2020]. This method only separates foreground from background, but is therefore prone to the document being occluded or not completely visible in the picture. A different method is based on using conventional computer-vision techniques with OpenCV, such as the example in [10]. For example, the outline can be detected with edge detection. The contour formed by these edges that results in the largest rectangle in the image is likely to be the document. Nevertheless, this method is very prone to the different conditions under which the picture is taken, namely the busy background, lighting conditions and reflections.

Our solution is the following method. First the document is detected using CLIPSeg [3] which allows for textual guidance input such as “Document” and provides a heatmap of possible regions where the object is present. Using the heatmap, positive and negative keypoints are defined (“Document” *vs* not “Document”) and provided as input for the Segment Anything Model (SAM) which provides pixel-level segmentation [9]. This document is then converted (dewarped) into a rectangle using the contour around it. This is visualised in Figure 3. The advantages of SAM over the other two methods described above is that it still detects the document when it is occluded or not completely visible in the image and is not distracted by a busy background since it has a clear prompt to search for (result not shown). Depending on the image, the prompt could be “passport document”, “document” or “paper”. The output of SAM, the dewarped image, can be provided as input to the authentication modules the same way as an image from a document scanner would be provided.



(a) (b) (c)

Figure 3: SAM-based image warping: (a) photo of synthetically generated passport (b) output of SAM with random points (stars) and segmentation (blue region) (c) dewarped image.

This dewarped image can be applied to the document analysis modules. In Figure 4, the result on the Machine Readable Zone (MRZ) of another synthetically generated document is shown. The tactical rule concerning the MRZ checks if the extracted text is a valid MRZ. A visual examination shows that the extracted text is indeed the same as the original MRZ.

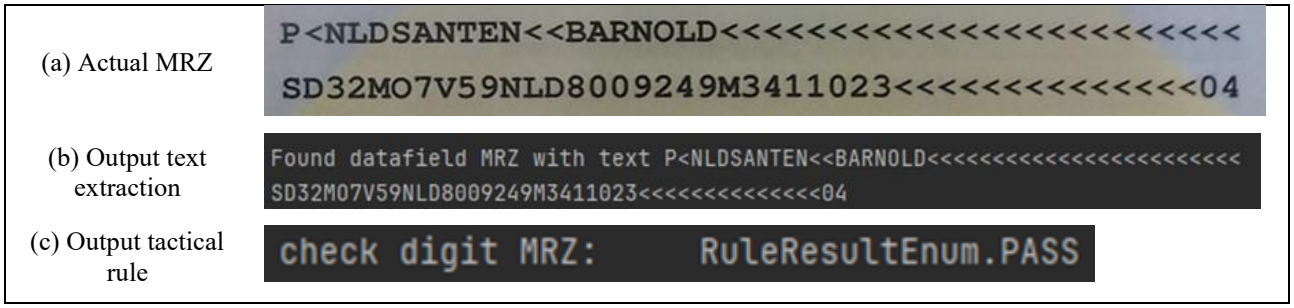


Figure 4: Output of the tactical rule regarding MRZ on an artificially generated passport: (a) cropped MRZ from the dewarped passport, (b) output from the text extraction, (c) output from the tactical rule.

2.3 Federated-learning based document recognition

Federated learning (FL) is important to support the cross-border collaboration on document authentication between border guards and immigration services from different countries. FL supports the this collaboration in the training phase of Artificial Intelligence (AI) models, because the sharing of personal data is minimised while the relevant data (e.g., trained models) is facilitated. FL differs from traditional central learning by training models across multiple decentralized devices or servers holding local data samples, rather than pooling data into a single central server.

The relation to current state-of-the-art is as follows. Earlier, it was already shown that document authentication is possible with trainable AI models [2]. But this is a system for one single user in one country. Our novelty is that we implement FL for the document-recognition module (Figure 2) and evaluate its performance on this task. Document recognition estimates for a scan what the country and document type are.

FL enables multiple parties to collaboratively train a model without sharing their data, effectively expanding the training data pool without exchanging actual data. This was be evaluated with random splits of data from one end user to mimic the use by multiple organizations. To illustrate its benefits, an experiment was conducted using the data of six document types, which originate from three countries, (i.e. 6 classes) with 1186 documents in total. Two configurations were explored. In the “Central” configuration, the entire dataset is used to train a model on a single client (or PC), representing the traditional training method. In the “Federated” configuration, two clients (or computers) each receive half of the data per class, representing a distributed approach in which two parties are working together to train a model without sharing their data. For each configuration, four variants were tested, each with a different number of images per class per client allocated to the training subset, with the remaining data used for the testing subset. Each variant was repeated ten times. The results are shown in Figure 5 and Table 1.

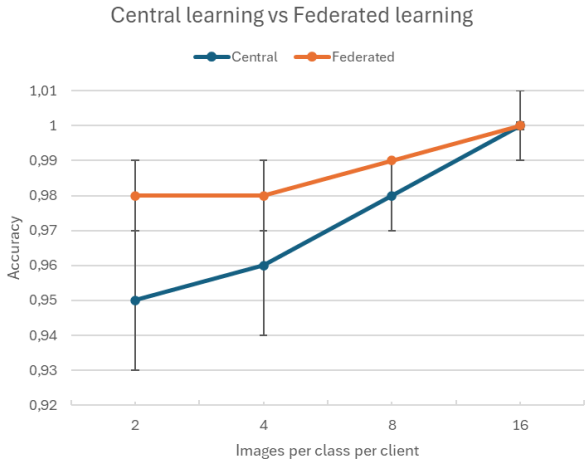


Figure 5: Federated learning obtains higher accuracy than central learning on a small dataset.

Table 1: Accuracy (mean ± std) for central learning and federated learning.

Images per class per client	Central (1 client)	Federated (2 clients)
2	0.95 ± 0.02	0.98 ± 0.01
4	0.96 ± 0.02	0.98 ± 0.01
8	0.98 ± 0.01	0.99 ± 0.00
16	1.00 ± 0.00	1.00 ± 0.01

The results indicate that the “Federated” setup consistently outperforms the “Central” setup until the final variant, where perfect scores are achieved with 16 images per class per client. This demonstrates the effectiveness of FL which effectively increases the available training data by enabling multiple parties to collaborate without sharing their data. As expected, the “Federated” setup, with two clients, effectively doubles the training pool compared to the single-client "Central" setup at each variant of the experiment, enhancing the model's performance. It is unexpected that in some cases the federate approach (e.g., 2x2 images) even performs better than the central approach with twice the data (e.g., 1x4). We assume that this is merely caused by the noise/variance in the experiments.

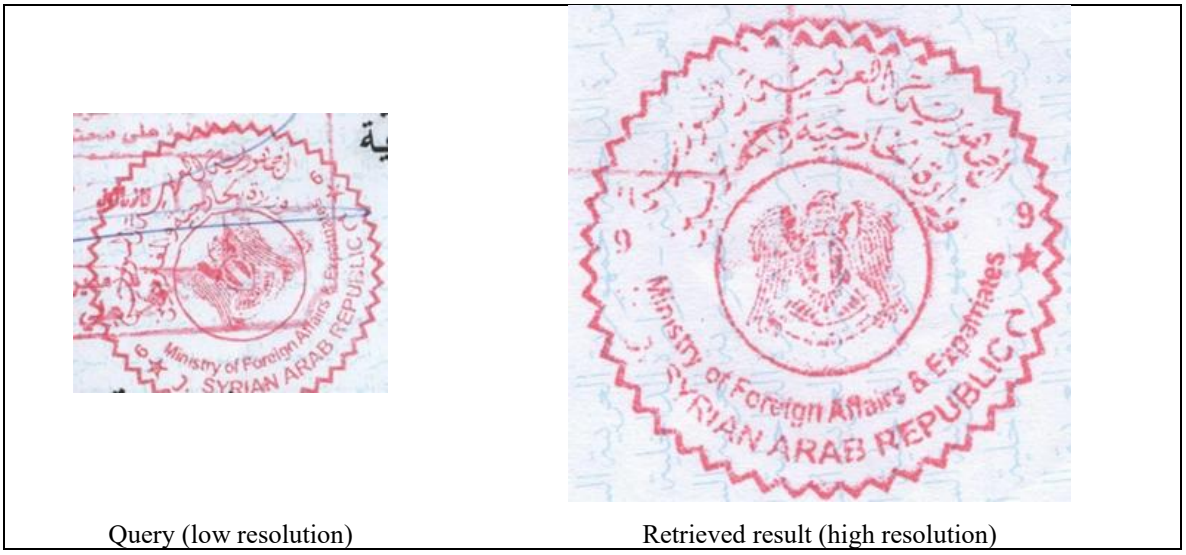
Future work will focus on two points. First, FL for other document-authentication modules, such as detail detection and detail matching. Second, increased security and efficiency during the training of models with secure sparse gradient aggregation (SSGA) [12]. Extended research showed that FL and SSGA can be applied to various computer-vision (CV) tasks, which are also relevant for document authentication [15]. Applying SSGA to the federated AI modules will further improve its usefulness for organizations from different countries.

2.4 Detail detection, matching and metadata extraction

The security features in breeder documents are limited. Stamps and signatures are therefore among the most important security features on breeder documents (e.g., birth certificates or marriage certificates). These must be compared with examples in the database to verify authenticity. Furthermore, detail detection is important for anonymization (e.g., faces) and tactical anomaly detection (e.g., comparing barcode information with text). The first step is the detection of details, which consists of two dedicated detectors for faces [5] and barcodes [16] and a retrainable detector for many other details, such as stamps, signatures and coat-of-arms [7][11]. The current state-of-the-art off-the-shelf detectors work good for persons and cars, but not for document details. Therefore, the retrainable detector is trained an finetuned on annotated document details. Metadata from barcodes is extracted for later tactical anomaly detection.

The second step is detail matching, which allows the comparison with examples in the database. The user can indicate whether two stamps are: exactly the same (identical), almost the same (similar), completely different, or unverified. This feedback is used to generate two graphs: one graph for identical details and one graph for similar details. The ‘same’ edges are used to create clusters and the ‘different’ edges are used to detects conflicts, which are inconsistencies in the user feedback. The clusters are used to train the detail matcher. Detail matching is implemented with Triplet-REID [8]. To minimize the annotation effort, post processing selects only details from the same country and one detail (the most similar) per cluster.

Experiments are performed various documents with various details. The detection and matching appear to work robustly independent of the scan resolution. An example is shown in Figure 6. Future work could include a quantitative analysis of the matching quality.



Query (low resolution) Retrieved result (high resolution)

Figure 6: Matching two similar details despite difference in resolution.

2.5 Keyword-mask detection on tabular data with variable rows

A pairwise keyword-mask detection approach was proposed earlier [1] to detect textual information and process it in a structured way. In this approach, the keywords are related to labels that are consistently present on all documents of the same type (e.g., "Last name") and the masks contains unique information for a specific document (e.g., "Smith"). The approach was extended to handle also tabular information (e.g., for bank statements or family registries) with a variable amount of rows. In the new approach, keywords are assigned to a word in the header row. To minimize user interaction, one large box is generated for each column to cover multiple rows. An example of the keywords and masks are shown in Figure 7 and the application automatically determines the rows inside the masks (Figure 8).

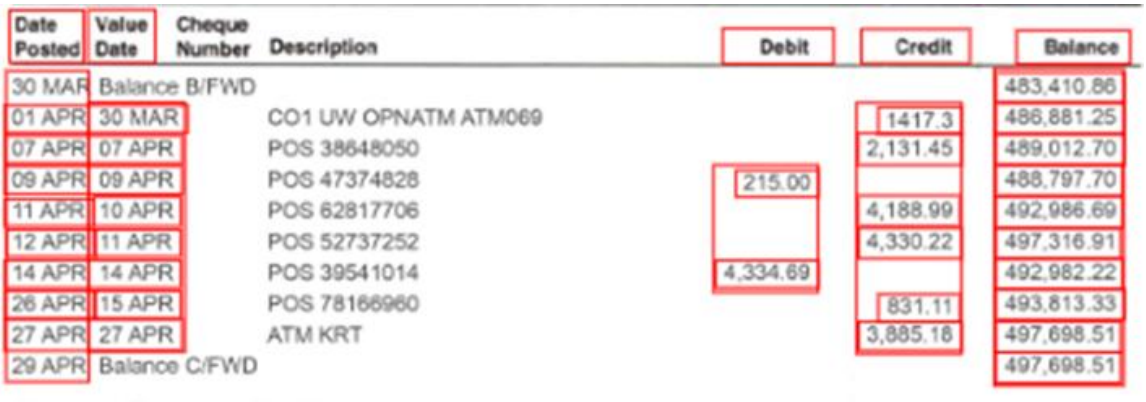
Statement of Account

Account Name			Account Number			
HENRI BOUMA			891293893			
Transaction Details						
Date Posted	**Value Date**	**Cheque Number**	**Description**	**Debit**	**Credit**	**Balance**
30 MAR	Balance B/FWD					483,410.86
01 APR	30 MAR		CO1 UW OPNATM ATM069		1417.3	486,881.25
07 APR	07 APR		POS 38648050		2,131.45	489,012.70
09 APR	09 APR		POS 47374828	215.00		488,797.70
11 APR	10 APR		POS 62817706		4,188.99	492,986.69
12 APR	11 APR		POS 52737252		4,330.22	497,316.91
14 APR	14 APR		POS 39541014	4,334.69		492,982.22
26 APR	15 APR		POS 78166960		831.11	493,813.33
27 APR	27 APR		ATM KRT		3,885.18	497,698.51
29 APR	Balance C/FWD					497,698.51

Account Summary As At 29 APR 2024

Opening Balance	-	Total Debits	+	Total Credits	=	Closing Balance
483,410.86		4,549.69		18,837.34		497,698.51

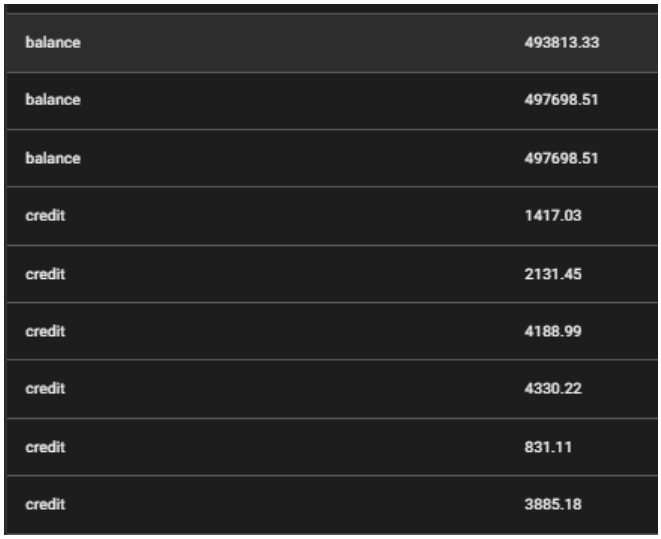
Figure 7: Example of an artificially generated bank statement. The words in the table header are detected as keywords and the dates/values in each column are detected as masks. Each mask covers multiple rows.



Date Posted	Value Date	Cheque Number	Description	Debit	Credit	Balance
30 MAR	Balance B/FWD					483,410.86
01 APR	30 MAR		CO1 UW OPNATM ATM069		1417.3	486,881.25
07 APR	07 APR		POS 38648050		2,131.45	489,012.70
09 APR	09 APR		POS 47374828	215.00		488,797.70
11 APR	10 APR		POS 62817706		4,188.99	492,986.69
12 APR	11 APR		POS 52737252		4,330.22	497,316.91
14 APR	14 APR		POS 39541014	4,334.69		492,982.22
26 APR	15 APR		POS 78166960		831.11	493,813.33
27 APR	27 APR		ATM KRT		3,885.18	497,698.51
29 APR	Balance C/FWD					497,698.51

Figure 8: The separate rows are detected inside the masks.

Finally, the textual values (dates or numbers) are extracted and stored in a structured way with a relation to the header keyword and the row (Figure 9). Tactical rules can check consistency in the tables. Most dates and numbers are extracted correctly, but in this case, an error occurs because a number is incorrectly extracted ("1417.03" should have been "1417.3").



balance	493813.33
balance	497698.51
balance	497698.51
credit	1417.03
credit	2131.45
credit	4188.99
credit	4330.22
credit	831.11
credit	3885.18

Figure 9: Text values are extracted and stored in a structured way with a relation to the header keyword and the row.

2.6 Printing-technique recognition

Printing-technique recognition plays a critical role in document forensics, particularly in document authentication. Recognizing the printing process used on different document elements – stamps, signatures, foreground / background elements – can help detect forgeries and unauthorized alterations. Traditionally, this task has been addressed through hand-crafted features such as gray-level co-occurrence matrices (GLCM) and local binary patterns (LBP) [2]. More recently, deep learning (DL) approaches have gained traction, particularly convolutional neural networks (CNNs) trained directly on scanned document details. The default CNN approach would be a to use a single small patch as input for a classifier and combining multiple classifier outputs in a region with max voting [2]. In this subsection, we introduce a novel pipeline combining segmentation and multi-patch CNN classification for improved printing-technique recognition.

The new mask-based patching approach consists of four steps: segmentation, patch selection, multi-patch CNN-input generation and classification. The first step is segmentation, which generates a binary mask of a detail (e.g., a stamp) to support effective patch selection. Its implementation is based on SAM2 [13]. To improve on SAM2's acclaimed out-of-the-box performance and be able to perform fine-grained segmentation of complex details on high-resolution document scans, the model was finetuned on a small detail segmentation dataset. The second step is patch selection, which is guided

by the segmentation and supports the classifier to focus on the most relevant parts of a detail. A fixed number of patches is extracted in such a way that they are centered on mask portions belonging to the actual detail and avoiding inclusion of irrelevant patches. At training time, the process of extracting mask-centered patches is randomized, guaranteeing as a side-effect a good level of image augmentation, without compromising fine-grained texture details relevant for classification. The third step is CNN-input generation to prepare the input of the classifier. In this step N x N patches are rearranged in a grid-like input to the classifier (Figure 10), without resizing or scaling, thereby avoiding the introduction of artifacts. The final step is classification. A VGG16 was used for classification. For each detail type and printing technique analyzed in this study, a binary classifier is built and tested. For example, in case of stamp classification, a binary classifier is defined to recognize stamp-ink versus other. And for background, a binary classifier is defined for offset versus other. This maximizes the amount of data available during training and is applicable to the real scenario where a specific document type may typically expose, for a certain detail, one printing technique. Anything that deviates from that (i.e. "other") should be determined accordingly.

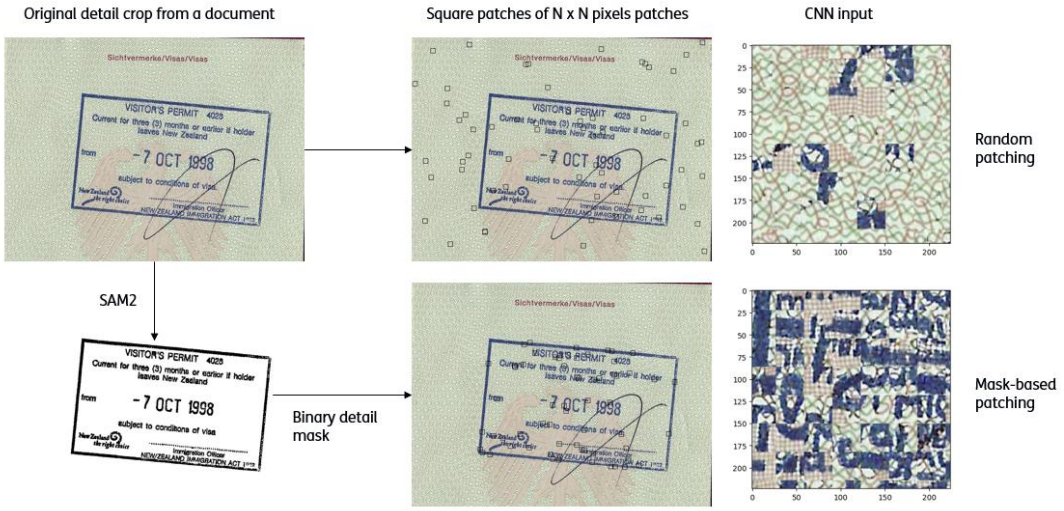


Figure 10: Random patching (top) versus mask-based patching (bottom). The example highlights the difficulties given by random patching on a high-resolution scan, when poor cropping leaves background and other irrelevant objects (e.g. signature) within the space occupied by the detail of interest. Especially in these cases, mask-based patching is promising, retaining relevant parts of the detail for classification.

Experiments have been carried out on two diversified sets of document details and relevant printing techniques. The quantitative evaluation of the methodology was conducted by performing N=5 training runs of each of the aforementioned binary classifiers. Dataset A includes five printing techniques and three detail types: document numbers, signatures and stamps (Table 2). Dataset B is focused on stamps and allows training of three main printing techniques (Table 3). Dataset A is more precisely annotated in terms of detail bounding boxes and generally contains less clutter (e.g. diverse background or partially overlapping elements). Dataset B is more diverse, being built over a larger set of document types and contains harder samples. Separate subsets are used for training (80%), validation (10%) and testing (10%). Furthermore, a balanced sampler is used at training time, to avoid under-sampling of less frequent printing techniques.

Table 2: Frequency of printing technique per detail in the balanced dataset of Dataset A. Each printing technique – detail type combination was used to train a [label] vs. 'other' classifier.

	Handwritten	**Stamp Ink**	**Letterpress**	**Inkjet**	**Toner**
Signature	100	0	0	86	100
Stamp	0	100	0	100	100
Number	0	0	100	100	99

Table 3: Frequency of printing technique per detail in the dataset of Dataset B. In particular, 'Dye Sublimation' and 'Dry Stamp' samples are considered insufficient to train a binary classifier and were therefore only added as label 'other' when training for recognition of the remaining printing techniques.

	Inkjet	Stamp Ink	Laser	Dye Sublim.	Dry stamp
Stamp	161	48	38	2	5

The results are shown in Table 4. On Dataset A, the main improvement is given by the multi-patch input generation approach, compared to the single-patch max-voting method adopted earlier. A jump in accuracy is shown on almost every combination of detail type and printing technique. Furthermore, a smaller, not significant improvement is given in this case by the mask-based patch selection. This is probably due to the fact that each detail crop already contains mainly relevant information and less clutter, so the random-patch selection seems sufficient. On Dataset B, instead, mask-guided patch selection is fundamental to yield consistently higher performances, i.e. the single-patch and random multi-patch methods achieve lower accuracy scores on the test split.

Table 4: Evaluation (mean accuracy rate over N runs) of binary classifiers. Performances are computed on the test split. Note: mask-guided patching was applied exclusively to stamp printing techniques. N/A: not applicable.

Dataset	Detail type	Classifier type (* vs. other)	Single-patch classification and max voting	Multi-patch classification with random patching	Multi-patch classification mask-guided patching
A	Stamp	Stamp ink	0.83	0.95	**0.96**
A	Stamp	Inkjet	0.77	0.91	**0.92**
A	Stamp	Toner	0.85	**0.96**	**0.96**
A	Signature	Handwritten	0.86	**0.96**	N/A
A	Signature	Inkjet	0.91	**0.99**	N/A
A	Signature	Toner	0.89	**0.96**	N/A
A	Number	Letterpress	**0.86**	0.83	N/A
A	Number	Inkjet	0.87	**0.92**	N/A
A	Number	Toner	0.80	**0.87**	N/A
B	Stamp	Stamp Ink	0.55	0.58	**0.67**
B	Stamp	Inkjet	0.75	0.81	**0.85**
B	Stamp	Laser	0.59	0.66	**0.77**

Overall, the multi-patch mask-guided approach, built on two novel techniques, has complementary beneficial effects, as shown on Dataset A and Dataset B, respectively. This is especially visible according to the characteristics and the amount of training data available. The number of training examples was very limited, especially in Dataset B. It is therefore expected that an increase in training data will lead to much higher performance (e.g., towards a mean accuracy of 96%, as shown on Dataset A).

2.7 Text extraction

Optical character recognition (OCR) typically consists of two aspects, text detection, which localizes the text in an image and text recognition, which predicts the text that is written in the detected region. We use the OCR engine EasyOCR [4] which implements both steps since it readily supports more than 80 languages/scripts (e.g., Latin, Arabic, Farsi, Chinese, Cyrillic, Hindu, Japanese, Korean, Thai, etc.) and uses a permissive license. In addition, it publishes training code that can be used to fine-tune the model locally on private data. We use the trained models from EasyOCR for the different languages and scripts. However, in the challenging use case of Arabic dates with watermarks in the background, the off-the-shelf model fails to produce sensible results (Figure 11). To improve the performance, we use the off-the-shelf model as starting point and fine-tune it for this specific task. Since manually labelling images of Arabic dates is time-consuming, we generate synthetic training samples instead. We first create crops of backgrounds of different real documents featuring different watermarks and colors. Then, we randomly select one of these backgrounds and print an arbitrary date on top of it. This enables the creation of 1000 annotated samples in less than a minute. During fine-tuning, we freeze the pre-trained

backbone of the model to keep its strong generalization performance and only update the weights of the final layers of the model. These steps significantly improve performance on Arabic dates (see Figure 11).

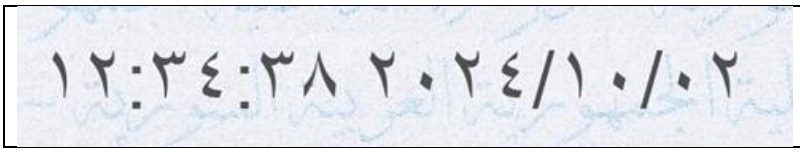


Figure 11: Image of an Arabic date (left) and the text predicted by the off-the-shelf model and the fine-tuned model (right). The prediction of the latter is correct.

2.8 Hybrid tactical anomaly detection

Tactical anomaly detection in breeder documents assists in fast assessment of these documents. Currently two types of methods are available to detect tactical anomalies. Rule based methods work in a top-down manner where a user dictates rules that define the anomalies. These rules are applied in the analysis of the documents, to detect the anomalies based on extracted information from text, barcodes, tables, MRZ-regions, etc. Secondly, anomalies can be detected using bottom-up data-driven methods. These methods extract patterns from the historical data and search for documents that deviate from this historical pattern. These deviating documents will be further inspected by the operator to validate the Genuity of the document. The use of rules in the top-down anomaly detection results in a transparent and flexible approach. Transparent since the rules used in the system are interpretable by humans, making it clear which steps resulted in the document being detected as an anomaly. Flexible, since the rules can be easily updated, solely based on expert knowledge, without the requirement of additional training examples. The data-driven approaches allow the discovery of new types of anomalies, yet unknown to the operator. However, simply presenting the documents that are deviating from the historical pattern, without any explanation, reduces the transparency of the results. To be able to discover a new type of anomaly, while maintaining the transparency of the results, hybrid tactical anomaly detection methods can be applied. These methods combine the bottom-up and top-down approach. A data-driven method is used to discover new types of anomalies from historical data, which are thereafter characterized in rules that can be applied for rule-based anomaly detection. The rules can be revised or rejected by the user to maintain transparency and flexibility. Our novel contribution is a group of three different rule discovery approaches to support the detection of tactical anomalies in documents.

Three different rule discovery methods are integrated into the hybrid anomaly detection: one for numerical outliers, one for categorical outliers and one for temporal changes.

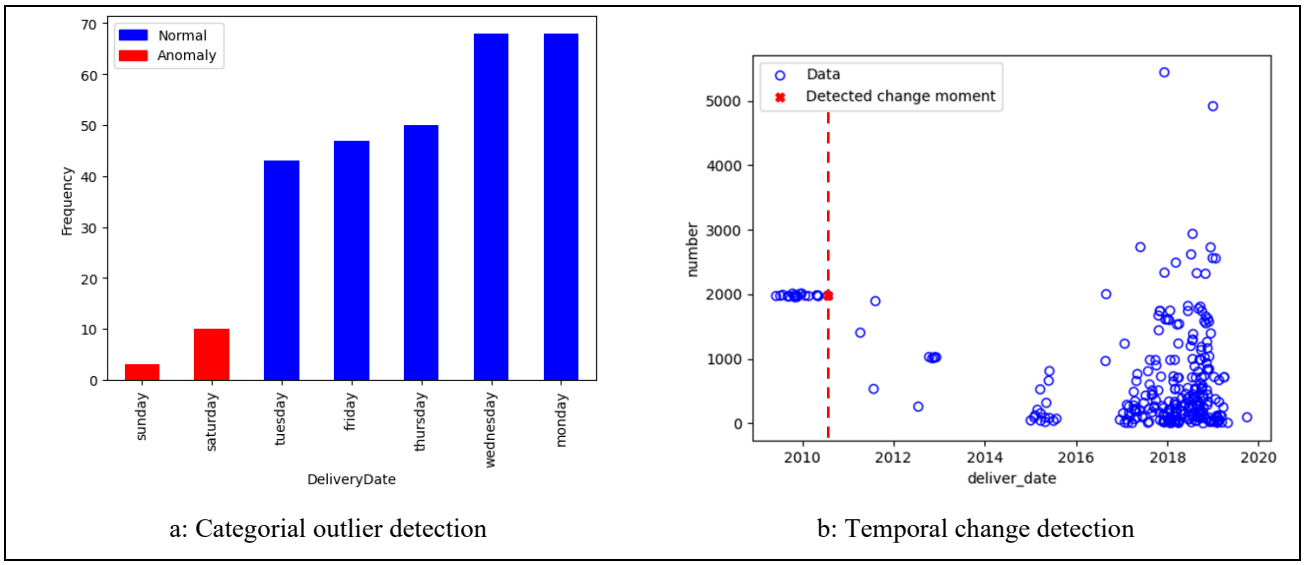
1. *Numerical outlier rule discovery*: Rules for extracted information that is numeric, are discovered by a numerical outlier detection based on mean and standard deviation. All the data for a single data type (for example personal number) is collected and processed. Data is considered as an outlier when the value is more than twice the standard deviation deviating from the mean (Z-score > 2). Based on this assumption, automatic rules for the upper and lower bound of the numeric values can be constructed.

2. *Categorical outlier rule discovery*: The outlier detection based on standard deviation and mean cannot directly be applied on categorical data. First numeric properties has to be calculated from the categorical data. Therefore we first count the occurrences of the values in the data. The occurrences distribution is used to characterize the data. The intuition that we use is that when a categorical value occurs less than expected by looking at the distribution of the occurrences, it can be suggested as an anomalous value. Only a one-sided test is used, looking at few occurrences, since a high number of occurrences is not anomalous. First the occurrences are filtered, to remove occurrences from the occurrence table that deviate more than one standard deviation from the mean. This is required because often one or a few categorical values occur very often, having a high influence on the mean. Thereafter the anomalous categorical values are extracted; a categorical value is stated as anomalous when it deviates more than two times the standard deviation from the mean (Z-score > 2). An example is shown in Figure 12b.

3. *Temporal change detection*: The two methods so far focused on anomalies in single column values. However, there are cases where anomalies occur in the combination of multiple column values. One example of this is a temporal change over time, for example when the construction of document numbers changes after a certain date, due to administrative issues. For these cases we developed a temporal change detection. The temporal change detection searches for moments in time for which the pattern in the data changes. It works by looping over a date column and looking whether the data can be separated by learning a classifier. The intuition behind this idea is

that when a change in the data pattern has occurred, the data before and after the moment of change could be easily separated into two classes (before/after). The algorithm loops over a date column from early dates to the latest date. In every step, an SVM classifier is learned on the data before the current date and several data points after the current date. This learned SVM classifier is thereafter used to classify the same data points as it is trained on. When the balanced accuracy (classification accuracy compensated for class imbalance) is around 0.5, the classifier was not able to learn a useful pattern from the data. When the balanced accuracy is higher than 0.9, the classifier could separate the data before and after the current date, indicating a change has detected. If a change is detected, the data points before the change are removed from the data and the change detection is applied again. This allows to detect multiple changes after each other. An example is shown in Figure 12b.

Future work could include a quantitative analysis.



a: Categorial outlier detection

b: Temporal change detection

Figure 12: Examples of categorial outlier detection (a: outliers in the weekend) and temporal change detection (b: change moment indicated as dashed line).

3. WORKFLOWS AND GRAPHICAL USER INTERFACE

The application supports multiple workflows. First, the normal and main workflow for document experts to detect tactical anomalies in the document is described in Sec. 3.2. After that, the other supporting workflows are shortly mentioned.

3.1 Main workflow for tactical anomaly detection

The overview of the GUI for the main workflow is shown in Figure 13. The GUI consists of the following seven elements:

1. Hamburger menu icon: For accessing to other workflows.
2. File handling buttons: For loading new documents or accessing documents in the database.
3. Document scan preview: For human inspection.
4. Processing pipeline widget: For automated processing (see Figure 2 for an overview of the modules).
5. Document rotation widget: For human verification of the document rotation.
6. Verification tab sections: Labels to access the verification tab content (element 7).
7. Verification tab section content: For human verification of the modules.

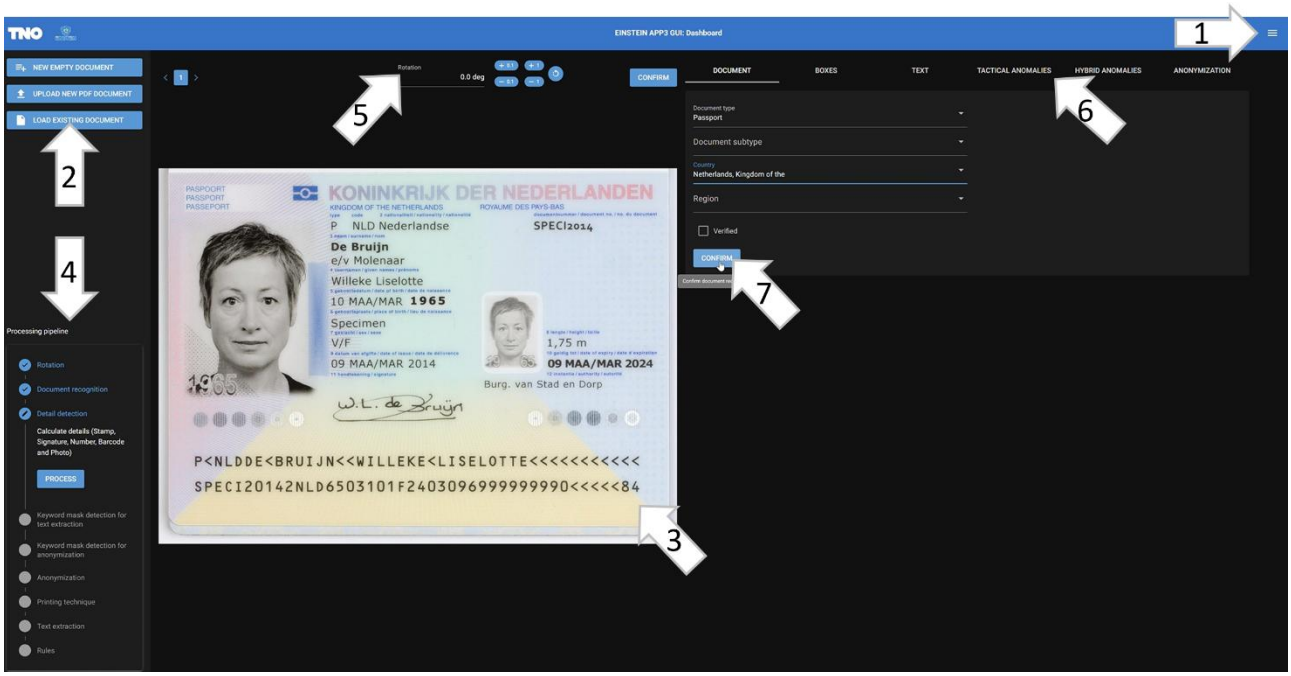


Figure 13: Overview of the APP3 GUI.

The human can decide to first activate all automatic processing steps (element 4 on the GUI) and then do the human verification (elements 5, 6, 7). This gives the user the freedom to jump directly to the output of tactical anomaly detection and optionally traceback certain findings. Alternatively, the user can go through the application step-by-step; first running and verifying one module before going to the next module.

The workflow first performs document rotation and document recognition, then detection of boxes from details and keyword-mask pairs, text extraction and finally tactical anomaly detection. An example of detected boxes from details (e.g., photo and signature) and keyword-mask pairs (e.g., 'Name' and "De Bruijn") are shown in Figure 14.

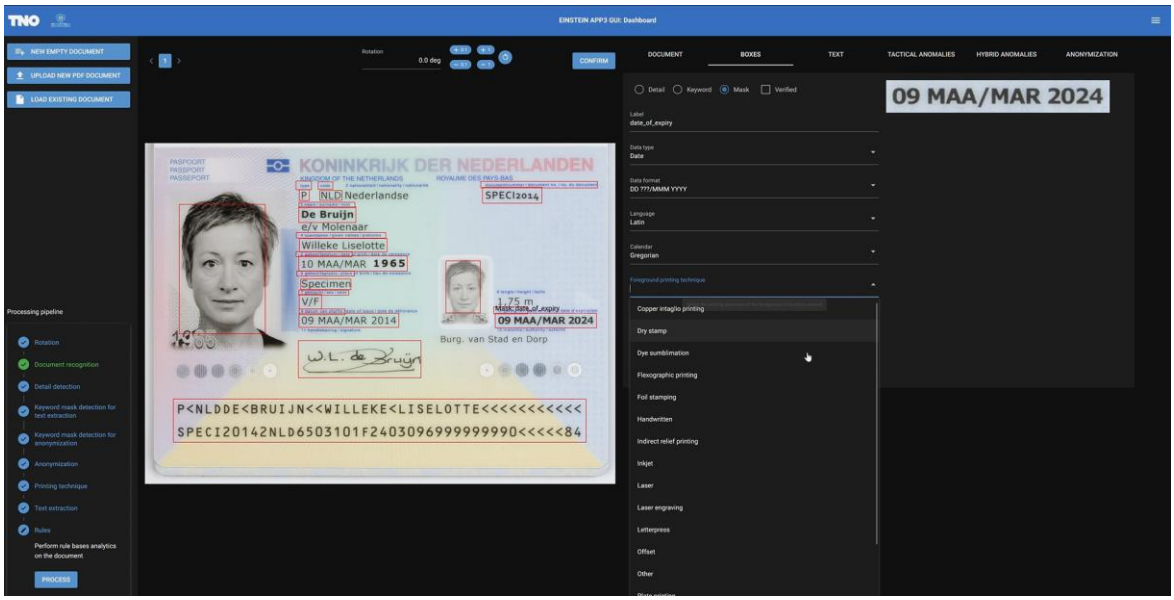


Figure 14: GUI with detected boxes from details (e.g., photo) and keyword-mask pairs ('Name' + 'De Bruijn') and masks (e.g., MRZ region).

3.2 Other supporting workflows

Examples of other supporting workflows are the following:

- Data-driven tactical rule discovery in a collection of documents and tactical rule editing.
- Anonymization of a document (to enhance privacy).
- Training of AI-modules and AI-model management (including analysis of train, validation and test data and performance metrics)
- User/role management and user login (to guarantee that the user can only view documents from their own organization and to support logging of user actions for accountability).
- Troubleshooting and access to logfiles (to provide remote support).

4. EVALUATION BY END USERS AND FUTURE WORK

An earlier version of the application was tested by end users and they provided feedback on the key strengths and lessons learned. Furthermore, plans for future work were prioritized.

The key strength and achievements of the application is that it is a complete application with many advanced capabilities to detect document fraud. The primary workflow to detect tactical anomalies is intuitive with a clear structure. The automated processing is shown on the left of the GUI (structured top to bottom) and human verification is shown at the top of the GUI (structured left to right). The system is flexible and allows humans to verify the results and insert new knowledge.

Lessons learned and recommendations are the following. User training appears essential. The basic steps are intuitive, but the application has many advanced capabilities which can only be understood after appropriate user training. After automated processing, the related human verification pane should pop-up automatically. Furthermore, several minor issues were documented related to the GUI.

Plans and priorities for improvement are related to the improvement of printing-technique recognition, a modified pipeline for improved document rotation, document recognition and less manual interaction, improving hybrid anomaly detection, extending federated learning for other AI-modules and addressing many minor GUI related issues.

5. CONCLUSION

In this paper, we showed enhanced document authentication with robust processing of images from mobile phones, federated-learning based training, flexible handling of tabular data with varying number of rows and hybrid anomaly detection which combines data-driven rule discovery with knowledge-based tactical anomaly detection.

ACKNOWLEDGEMENTS

Funded by the European Union

The EINSTEIN project (G.A. 101121280) was funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

[1] Bouma, H., Van Mil, J., ten Hove, J. M., Pruim, R., van Rooijen, A., et al., (2022, October). Combatting fraud on travel, identity, and breeder documents. In Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies VI (Vol. 12275, pp. 63-72). SPIE.

[2] Bouma, H., Reuter, A., Brouwer, P., et al., (2021, September). Authentication of travel and breeder documents. In Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies V (Vol. 11869, pp. 38-53). SPIE.

[3] ClipSeg, URL: https://huggingface.co/docs/transformers/model_doc/clipseg

[4] EasyOCR, URL: https://github.com/JaidedAI/EasyOCR

[5] Face, URL: https://pypi.org/project/face-recognition/

[6] Gatis, D., Docscan. Github. https://github.com/danielgatis/docscan

[7] He, K., Zhang, X., e.a., “Deep Residual Learning for Image Recognition,” arXiv:1512.03385, (2015).
[8] Hermans, A., Beyer, L., & Leibe, B., “In defense of the triplet loss for person re-identification,” arXiv:1703.07737, (2017).
[9] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., ... & Girshick, R., “Segment anything.” Proc. IEEE ICCV, 4015-4026 (2023)
[10] Olufemi, V., Buiding your document scanner. Medium, (2022). URL: https://medium.com/@victorolufemi/build-a-document-scanner-with-opencv-ff9f645a4085
[11] Ren, S., He, K., Girshick, R., Sun, J., “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”, IEEE Trans. Pattern Analysis and Machine Intelligence 39, 1137-1149 (2017).
[12] Rooij, M. van, van Rooij, S., Bouma, H., & Pimentel, A. (2022, November). Secure Sparse Gradient Aggregation in Distributed Architectures. In Internet of Things: Systems, Management and Security (IOTSMS). IEEE.
[13] SAM2, URL: https://github.com/facebookresearch/sam2
[14] Sinha, A., “A survey of system security in contactless electronic passports,” Journal of Computer Security, 19(1), 203-226 (2011).
[15] Spek, M. van der, van Rooijen, A., & Bouma, H. “Secure sparse gradient aggregation with various computer-vision techniques for cross-border document authentication and other security applications.” Artificial Intelligence for Security and Defence Applications II, Proc. SPIE Vol. 13206, 121-134 (2024)
[16] Zxing, URL: https://pypi.org/project/zxing-cpp/