DocTr: A Unifying Framework for Tracking Physical Documents and Organisational Structures

Sandra Trullemans, Ayrton Vercruysse and Beat Signer
Web & Information Systems Engineering Lab
Vrije Universiteit Brussel
Pleinlaan 2, 1050 Brussels, Belgium
{strullem,ayvercru,bsigner}@vub.ac.be

ABSTRACT
Despite major advancements in digital document management, paper documents still play an important role in our daily work and are often used in combination with digital documents and services. Over the last two decades, we have seen a number of augmented reality solutions helping users in managing their paper documents in office settings. However, since data is mainly managed at the application layer, the use of multiple document tracking setups results in fragmented and inconsistent tracking data. Furthermore, existing tracking solutions often focus on the tracking of paper documents in organisational structures such as folders or filing cabinets without taking into account the flow of documents across these organisational structures. We present the Document Tracking (DocTr) framework for unifying existing document tracking setups and managing document metadata across organisational structures. The DocTr framework has been implemented based on a user-centric requirements analysis and simplifies the development of interactive computing systems for personal cross-media information management.

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

Author Keywords
Document tracking; personal information management; organisational structures.

INTRODUCTION
Despite the predictions of the paperless office in the 1970s, we are still using paper documents in our daily work [29]. In many office environments digital as well as physical documents form a part of our workflows. It has further been shown that frequent tablet users often print digital documents in order to facilitate the task at hand [4]. Recent studies highlight that besides the personal preferences for paper documents, companies in various business sectors do not get rid of paper documents due to productivity or law reasons [11].

Since the early 1990s, applications and technologies have been developed which enable the tracking of paper documents in a user's working environment. Thereby, paper documents are tagged with two-dimensional tags such as barcodes or electronic markers like Radio Frequency Identification (RFID) tags. As an alternative to the explicit tagging of documents, computer vision techniques can be used to identify paper documents. These document tracking technologies enabled various research projects such as the DigitalDesk [35] project where paper is integrated with an interactive desk. Users can further be supported in organising and re-finding their documents via so-called Personal Information Management (PIM) solutions. There are a number of PIM solutions for managing paper documents which usually consist of a paper document tracking component and some digital search functionality. In addition to the absolute position of a document in space, some of these applications also track to which organisational structure (e.g. pile or folder) a document belongs to. Note that existing solutions are often limited to track documents within a specific organisational structure in order to provide users access to the corresponding digital document version. For example, the SIFT algorithm has been applied for tracking interactions with document piles [17] whereas in [28] the physical folders in a filing cabinet have been augmented based on visual tags and digital pen and paper technology.

In this paper, we use the term *tracking setup* for a particular setup including hardware as well as software components which enable the tracking of paper documents within a specific area based on tagging or some of the other technologies mentioned above. For example, a tracking setup might consist of a camera which is mounted above the desk in combination with some computer vision software or it can be a bookcase where individual books have been RFID tagged. While typical office environments consist of multiple organisational structures and paper documents naturally flow between these structures, existing solutions focus on tracking documents within single organisational structures based on dedicated technologies. Furthermore, any data management is typically delegated to the application layer, leading to fragmented and inconsistent tracking data when multiple tracking setups are used.

EICS'16, June 21–24, 2016, Brussels, Belgium

DOI: http://dx.doi.org/10.1145/2933242.2933254

We start by investigating the existing body of work in the domain of document tracking solutions. The detailed analysis of existing solutions in combination with an exploratory user study led to a number of user-centric as well as technical design requirements which are discussed in detail and form the basis of our unifying document tracking framework. We then describe the DocTr framework for integrating different document tracking setups and managing document flows across organisational structures. Furthermore, we illustrate how DocTr unifies data management for third-party applications based on the OC2 PIM framework [33]. A description of how third-party applications can profit from the DocTr framework is followed by a preliminary evaluation of the framework and some concluding remarks.

BACKGROUND

Given the fact that paper documents still play an important role in our daily work, various research efforts have been undertaken to integrate paper documents and physical storage artefacts (e.g. a filing cabinet) with digital information systems. Interactive tabletop surfaces are for example used in combination with cameras above the desk for augmenting paper documents with digital functionality. Thereby, paper documents are often tagged with a two-dimensional (2D) code as seen in DocuDesk [7]. A number of rapid prototyping frameworks have been developed to ease the use of tags for object recognition. For instance, the reacTIVision [14] framework detects fiducial markers (i.e. 2D tags with a specific pattern) within a well-defined surface area and notifies registered client applications about detected tags and their relative orientation. Note that the reacTIVision framework is widely used in tabletop and augmented desk applications including ObjectTop [16] and iCon [5].

The tagging of paper documents interrupts the user's regular tasks and therefore various computer vision-based document identification solutions have been developed as alternatives to tagging. However, due to changing environmental factors such as the lighting conditions, computer vision-based solutions are often more vulnerable to errors than tag-based solutions. Already in their seminal DigitalDesk system, Wellner [35] used feature extraction on captured document images in order to identify which document had been placed on an interactive desk. Nowadays, more advanced solutions, such as the Scale Invariant Feature Transformation (SIFT) [21] algorithm proposed by Lowe or the Speed Up Robust Features (SURF) [1] algorithm, are used for document recognition. The SIFT algorithm has, for example, been applied in FACT [20] to enable fine-grained cross-media interactions between paper documents and a laptop. In addition to paper documents also other artefacts can be recognised. The work of Matsushita et al. [23] tracks books in a bookshelf based on the SURF algorithm. Since paper documents often contain a number of unique text blocks, Optical Character Recognition (OCR) techniques can be applied to the text in order to identify a document and any written text can be transformed into a digital form [34].

The aforementioned tracking techniques have also been used for tracking paper documents in organisational structures such as piles or filing cabinets. A number of applications augment existing physical organisational structures with tracking functionality in order to provide a digital representation of a specific organisational structure. The work of Lawrie and Rus [19] describes an early application providing a digital representation of a physical filing cabinet. Paper documents in piles on a desk can be tracked by using computer vision techniques [17, 25]. Besides these traditional organisational structures, people also use other artefacts such as boxes and drawers for storing documents or other physical objects. DrawerFinder [18] provides tracking support for such organisational structures by tagging drawers or boxes and monitoring a user's interaction with a storage shelf containing these tagged boxes.

While the previously mentioned applications monitor one specific sort of organisational structure (e.g. a pile), during their lifecycle documents naturally move between different organisational structures. Therefore, there is a need to track documents across organisational structures. The Human-Centered Workplace system monitors printed documents across organisationla structures by augmenting them with a Quick Response (QR) code at printing time [6]. PaperSpace [31] uses a similar tracking approach but adds interactive printed buttons to paper documents. Users can, for example, point to a printed button to request the digital version of a document. While these applications track paper documents across organisational structures, they do not keep track of the internal state of these organisational structures. For example, while the Human-Centered Workplace system monitors paper documents in an office environment by using multiple cameras and provides applications the absolute position of tracked documents, it does not model the individual organisational structures (e.g. piles). In the context of the presented DocTr framework, SOPHYA [12] is the most relevant solution. The SOPHYA framework provides a way to track ordered collections of artefacts such as folders which have been augmented with electronic circuits. In addition, artefacts can be enhanced with re-finding user interfaces such as an LED strip in a bookshelf which lights up when a user searches for the corresponding artefact [10]. Although the SOPHYA framework is extensible to support various kinds of organisational structures, it is necessary that all tracked artefacts are augmented with electronic circuits which might not be an ideal solution for every office setting.

In an office environment, users have different kinds of organisational structures and paper documents can move between instances of these organisational structures. Such office environments might therefore require multiple of the tracking solutions presented in this section. In the remaining part of this paper we introduce DocTr, a framework that unifies existing tracking solutions and manages a document's lifecycle across different organisational structures.

DESIGN REQUIREMENTS

The development of the DocTr framework has been based on a set of user-centric and technical design requirements. These requirements have been derived from an exploratory user study as well as an investigation of the existing body of descriptive PIM research.

Exploratory User Study

In order to define a number of user-centric design requirements, we have conducted an exploratory user study. We interviewed eleven participants aged between 23 and 56. All participants used a significant amount of paper documents during their working activities. Furthermore, they had different professions such as secretary, social worker, middle manager and managing director.

Since most participants were not familiar with paper document tracking solutions, we first informed them about the existing body of related work. We introduced DocuDesk and DigitalDesk, tracking books in bookshelves, DrawerFinder, the Human-Centered Workspace and SOPHYA to the participants by showing them what they could do with these systems based on screenshots of the corresponding publications. Our introduction focussed on HCI and no details about technical aspects were provided. Furthermore, in order to minimise bias we explained the possible interactions in an objective and neutral manner. In a next step, we conducted a semi-structured interview to investigate issues with existing solutions and to explore opportunities for improvement. The main questions among others were: "*For which reasons would you consider to use one of the current solutions?*", "*Do you see shortcomings or disadvantages of these solutions?*", "*How do you see the future of tracking technologies?*", "*What should be the minimal functionality of a tracking solution?*" and "*Can you describe how the ideal tracking solution could be integrated in your working environment?*". Note that since the focus is on determining generic requirements, the given questions did not refer to any specific tracking systems.

Based on these interviews, we identified some shortcomings of existing solutions. First, participants mentioned that using a single technology is not desired due to the fact that their document organisation is very dynamic. Moreover, paper documents are often moved around. At the same time, they did not want to be overwhelmed by multiple isolated solutions (e.g. different applications for each tracking setup). Finally, our participants would not consider to invest in existing solutions due to a lack of integration with their current applications such as the File Explorer, Microsoft Word and Evernote.

User-centric Design Requirements

Based on the interviews and related descriptive PIM research, we derived the following user-centric design requirements in order to improve current paper document tracking solutions.

R1: Categories of Organisational Structures

In descriptive PIM research, organisational structures can be classified in three categories including files (e.g. filing cabinets), piles and mixtures (e.g. unordered documents in a letter tray) [22, 32]. Previous research argues that during re-finding activities in these different categories of organisational structures users rely on different cues (i.e. context, space or time). In a file structure, users will mostly use the context cue such as orienteering between documents which were used in the same task. In contrast, when a user searches information in a pile, they often try to recall the position of the required document in the pile (e.g. at the bottom or top) [36, 32]. In addition, our interviews show that users prefer a semantic description of a document's position such as "*the document is in the ring binder with the label Bills*". For seven out of the eleven participants, it would already be enough to just mention the category of the organisational structure containing the requested document, as mentioned by a participant: "*In most cases, this would trigger my spatial cue and then my memory will take it over. I do not think that I will often need the exact location*". In order to enable the design of PIM applications which provide support for one or multiple re-finding cues and to inform users about the categories of organisational structures, we have to integrate tracking support for the three categories of organisational structures.

R2: Flow of Documents and Organisational Structures

All participants identified the limited support for tracking paper documents across organisational structures as the main reason for the lack of acceptance of current tracking technologies. They do not see the benefits of installing different tracking setups with their individual applications. In contrast, they would see this as a burden as stated by a participant "*I think people are becoming crazy with all those different apps everywhere. If all these setups would again require more apps, I first have to find the right app. By then I have found my paper sheet!*". Besides the flow of paper documents across organisational structures, participants also highlighted that they often move the organisational structures themselves such as when creating a new pile out of two existing piles or reorganising ring binders. Therefore, a tracking solution should also foresee the tracking of the organisational structures themselves. In addition to the results of our interview, the flow of documents has also been investigated in descriptive PIM studies. It has been shown that paper documents can contain cold, warm or hot information [29]. So-called cold documents are often archived in a filing structure whereas piles seem to contain mostly hot documents (i.e. documents which are often used) [29, 32]. In addition, it has been shown that a user's personal information space contains a lot of cold documents which are rarely accessed or for which the user simply forgot that they exist [36]. By tracking paper documents across organisational structures and managing the flow of organisational structures themselves, users can be offered a unified solution and PIM systems can, for example, integrate the history of a paper document's flow and make users aware of rarely accessed paper documents.

R3: Custom Metadata

As mentioned before, most participants prefer a semantic description of a paper document's position. Therefore, tracking setups should be able to provide end users some semantic document metadata. In addition, the interviews revealed that users frequently would like to provide their own metadata about specific paper documents. Note that this custom metadata does not necessarily consist of positional data but can contain arbitrary annotations. Users prefer to add custom metadata as observed in current practise where they annotate documents with contextual information [29]. One cause can be seen in the fact that when applying the filing strategy, the overall context of a document is lost [13]. A flexible document tracking solution should therefore offer a mechanism for managing various kinds of positional as well as custom document metadata.

R4: Integration with Third-party Applications
Eight of the eleven participants preferred to have the document tracking metadata available in their frequently used tools. A number of participants mentioned that they would find it convenient to have a digital representation of paper documents and their metadata in applications such as Evernote where these paper documents could then be included in digital notes. Other participants proposed to just make the data available in the File Explorer in order that they could organise paper documents together with their digital media. These findings are in line with previous work. It has been observed that users hold on to their familiar tools when it comes down to PIM [3]. In addition, SOPHYA which is closely related to the presented work, aims for a decoupling of tracking technologies and third-part applications [12]. Therefore, it should be possible to integrate a tracking solution with third-party applications and these applications should have easy access to any document tracking metadata.

Technical Design Requirements

Besides the design requirements resulting from our exploratory user study, we identified a number of technical requirements for offering the functionality previously described in the user-centric design requirements.

R5: Unlimited Tracking Setups
While existing solutions focus on particular recognition techniques, developers should be enabled to easily combine existing tracking solutions in their applications. Since an interactive office environment is highly user and application dependent (i.e. users have different storage artefacts and custom office configurations), there needs to be a separation of concerns between the used tracking setups and the applications making use of paper document metadata. Moreover, specific tracking setups should only be responsible for detecting a paper document's metadata such as its position while the storage of this metadata should be managed by a central repository which is accessible by third-party applications. Note that this technical requirement will enable the tracking of paper documents across organisational structures (R2) and support an easy integration with third-party applications (R4).

R6: Integration with a PIM Framework
In the PIM research field, frameworks such as Gnowsis [26], HayStack [15] and MyLifeBits [8] have been developed for document management in personal information spaces. Commonly, these systems manage personal documents in a central repository and provide metadata about digital documents which can be used in organising and re-finding activities. A long-term study by Sauermann and Heim [27] highlighted the promising future of such PIM frameworks. Nevertheless, most PIM frameworks only support digital media. By also integrating paper documents with these frameworks, users can be offered a unified cross-media PIM solution (R4). Second, by making use of a PIM framework, any paper document

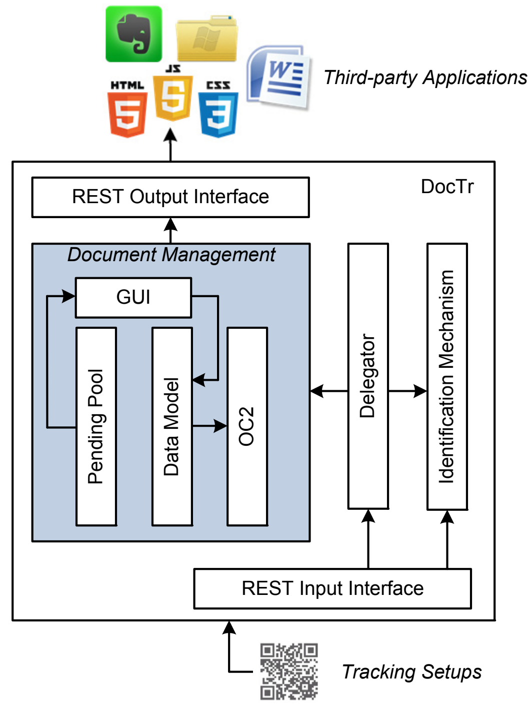

Figure 1: DocTr architecture overview

metadata can be unified with the metadata of digital documents. The integration with a PIM solution would further make it possible to let users and applications store their own custom metadata (R3) since most PIM frameworks offer this functionality. In addition, it is commonly accepted that there is a need for document management in office settings and that a PIM solution can be used for managing information about the three categories of organisational structures [32] (R1). Ultimately, the integration of a PIM framework will lead to a more unified personal information space.

R7: Managing Unique Identifiers
Due to the possibility of combining tracking setups with their own unique document identifiers, a general document tracking solution has to provide a mechanism for managing and mapping the different unique identifiers that might be assigned to a single paper document by the different tracking solutions. The challenge lies in the fact that unique identifiers can take different formats. For example, a document recognised by the SIFT algorithm will be identified by a set of extracted features while tag-based recognition frameworks use an integer or string as identifier. By supporting multiple unique identifiers for a single paper document or organisational structure, the flow of items across different organisation structures and tracking setups can be managed (R2).

DOCTR FRAMEWORK

We now present the DocTr framework which addresses the seven design requirements that have been defined in the previous section. As illustrated in Figure 1, the framework forms a middleware between multiple tracking setups and third-party applications that want to make use of a document's tracked metadata. The different tracking setups can be as simple as

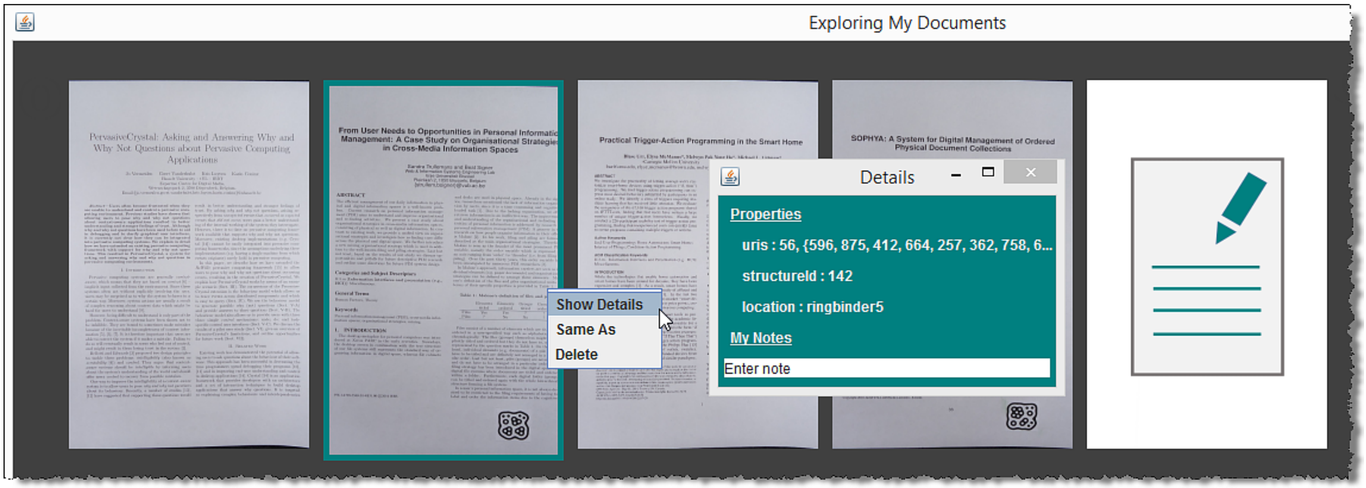

Figure 2: DocTr exploratory graphical user interface

listening to reacTIVision for detected tags or they might consist of complex software to extract paper document features via the SIFT algorithm.

There is a clear separation of concerns between the responsibilities of different tracking setups and the DocTr framework. The responsibility of a tracking setup is to determine the unique identifier of a traced paper document. In addition, a tracking setup is in charge of providing DocTr the digital representation of a monitored paper document. Subsequently, the DocTr framework makes these digital representations available to third-party applications. It is up to a specific tracking setup to define how this digital representation looks like. A representation could, for example, consist of an image of the paper document or a custom icon. In case no digital representation has been assigned to a monitored paper document, third-party applications will receive a default icon by the DocTr framework which is shown in DocTr's graphical user interface shown in Figure 2. When a tracking setup detects a paper document, it sends the document's unique identifier together with some derived metadata such as the documents position to DocTr. DocTr then checks whether the paper document is moving between organisational structures or whether it is a new paper document. In the case that this process does not lead to a clear result, the user is asked for clarification. Users can interact with DocTr via the exploratory graphical user interface shown in Figure 2 as well as the sidebar shown in Figure 3 where users can provide feedback about documents with open tracking issues.

The implementation of DocTr is based on a client-server architecture with the DocTr server using the component-based software architecture shown in Figure 1. Tracking setups can communicate with DocTr via the REST Input Interface while third-party applications can use the REST Output Interface to query metadata about tracked paper documents and organisational structures. The Document Management component is responsible for the data management of tracked documents and organisational structures while the Identification Mechanism is in charge of comparing unique document identifiers. The Document Management component includes the OC2 PIM framework [33] in order to fulfil the second technical requirement (R6). We have extended the OC2 data model to support different categories of organisational structures (R1) and enable the storage of custom metadata (R3). When DocTr cannot automatically determine whether a tracked entity has been moved, the corresponding document is forwarded to the Pending Pool component. Paper documents and organisational structures that have been added

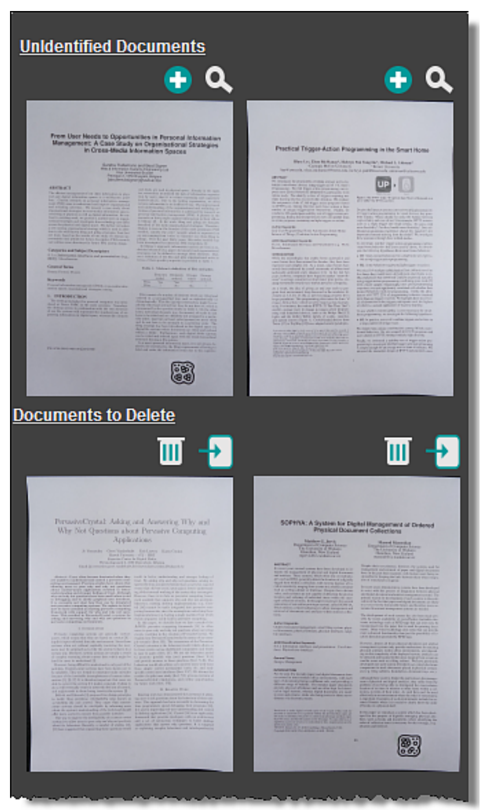

Figure 3: DocTr sidebar

to the Pending Pool wait for further user feedback such as a confirmation that a document has been moved if DocTr is not sure about its position or if a document has been placed in a non-trackable area. The communication between the different components is coordinated by the Delegator component.

The DocTr framework has been implemented in Java and uses a GlassFish[1] server together with the Atmosphere[2] framework providing an implementation of the JAX-RS recommendation for RESTful web services. In the following, we further elaborate on the individual DocTr components and their implementation.

DocTr Data Model and PIM Integration

We have chosen to use the OC2 PIM framework [33] for offering the necessary PIM functionality. The OC2 framework enables the linking of digital and physical documents via bidirectional navigational and structural links as defined by the RSL hypermedia metamodel [30]. Navigational links are used to express navigational paths between digital and physical documents while structural links provide a way to express structures within documents. Furthermore, OC2 allows us to store document metadata and define how relevant a document is in a given context or task. Last but not least, the OC2 PIM solution also offers some user management.

In order to support organisational structures (R1), custom metadata (R3) and multiple unique identifiers (R6), we have slightly extended the OC2 data model. An Entity-Relationship (ER) representation of the resulting extended data model with the shaded entities that have already been defined in OC2 is shown in Figure 4. In the OC2 model, objects can represent digital or physical objects as indicated with a disjoint constraint between the subclasses of `Object`. This allows us to create an instance of a physical object for each paper document. As mentioned before, a tracking setup is responsible to provide unique identifiers for their tracked paper documents and documents can be tracked by various tracking setups. Therefore, an `Object` can be associated with multiple tracking setups where each association between an object and a specific tracking setup can have multiple tracking properties. In this way, we can store custom metadata of a specific paper document for a specific tracking setup. Since the unique identifier of a paper document depends on the tracking setup monitoring the document, it is stored as a tracking property. The one-to-many constraints on the relationship between tracking properties and tracking setups enables the reuse of unique document identifiers across tracking setups. For example, a paper document might be identified by a fiducial marker which can of course be used by two or more tracking setups. Furthermore, it is necessary to manage multiple unique identifiers of paper documents with different formats (R7). Therefore, we introduced the `Comparator` entity. For a specific format, a comparator component will compare a given detected identifier against a set of unique identifiers. In order to allow the identification mechanism to compare unique identifiers, each tracking setup has to specify the appropriate comparator for its tracked paper documents.

[1] https://glassfish.java.net/
[2] https://github.com/Atmosphere/

Note that the tracking functionality is modelled at the level of an `Object`. In the future this allows us to reuse the DocTr data model for digital media such as the tracking of digital documents across cloud spaces.

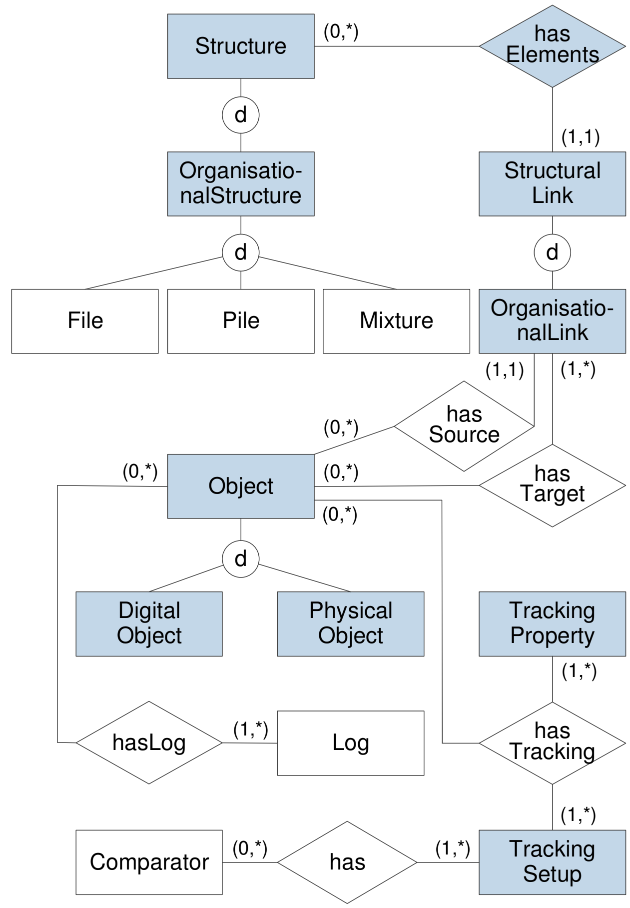

Figure 4: DocTr data model with highlighted OC2 entities

Besides offering this tracking setup functionality, we have to enable the storage of custom metadata (R3). Therefore, an object can be assigned some `Log` instances. A log takes the format of `(TimeStamp, Transactions)` where a transaction consists of an action and its value. Note that in contrast to other tracking applications, a timestamp can have multiple transactions. This enables applications to decompose an action in sub-actions. For example, applications can store one log entry when a paper document is removed from an organisational structure with the action 'remove' having a value 'pile1' and at the same time with an action 'byUser' with the name of the user who moved the paper document.

Organisational structures are stored as an organisational link with an `Object` entity as source and an ordered set of `Object` entities as target. The source object represents the organisational structure such as a pile while the targets of the organisational link represent the documents included in the organisational structure. Note that by modelling the organisational structure itself as an `Object` entity, the model's tracking extensions can also be applied to organisational structures. This is necessary since organisational structures are also dynamic in their position and a ring binder can, for example, be moved from a shelf to a desk. In addition, organisational structures have to be categorised in one of the three organisational structure types (R1). Since the OC2 data model already provides the notion of structures, we have introduced the `OrganisationalStructure` as a subclass of the `Structure` entity. In its turn, the `OrganisationalStructure` entity is subclassified into the `File`, `Pile` and `Mixture` organisational structures. The disjoint constraint indicates that an organisational structure can only take the form of one type of organisational structure. Finally, an organisational link has to be associated with an organisational structure.

Identification Mechanism

The identification mechanism is responsible for finding a `Physical Object` instance based on a detected unique identifier. Since physical objects can have multiple unique identifiers in different formats, we have introduced the `Comparator` components. As mentioned before, each tracking setup indicates the `Comparator` instance to be used when processing received unique identifiers. Currently, DocTr supports comparators for SIFT features, String comparators based on the Levenshtein distance as well as perfect matching for Strings and Integers. From our experience, these are the most commonly used formats in existing tracking setups. However, custom comparators can be provided by implementing the `compare()` method of the `Comparator` interface shown in Listing 1. The method has two parameters. The first parameter is a map with `Physical Object` as key entry and a list of `Object`s representing an object's unique identifier values. The second parameter is the received unique identifier (i.e. wrapped in an `Object` instance) which has to be compared against the values provided in the map instance. When the received unique identifier can be mapped to a physical object's identifier, the physical object is returned.

Listing 1: Comparator interface

```
interface Comparator

public PhysicalObject compare(
  Map<PhysicalObject, List<Object>>, Object);
```

When a tracking setup sends the unique identifier of a detected paper document to DocTr, the `Delegator` fetches the corresponding `Comparator` and queries the data model for all tracking properties where the value can be compared with the corresponding comparator. This set of unique identifiers is then compared to the received unique identifier by the `Comparator` instance. If there is a match, the physical object is returned to the `Delegator` which requests the document management component to update the data model with the received metadata. Otherwise, the `Delegator` forwards the unique identifier to the document management component where it is added in the pending pool.

Pending Pool

The Pending Pool component manages paper documents which could not be resolved by the identification mechanism. This can, for example, happen when a paper document that is uniquely identified by a tag is moved to another tracking setup which uses SIFT features for identification. Similarly, organisational structures such as a ring binder could, for instance, be identified via an RFID tag when placed on a desk while computer vision techniques might be used to detect the ring binder's label when stored in a shelf. Furthermore, a user can always throw away a document, give it to somebody else or place it in an unmonitored area.

Our solution for addressing undetected documents is inspired by context-aware applications which aim to involve the user in order to decrease the vulnerability to errors and increase user satisfaction [2]. We developed an end-user application where users can provide the necessary input to resolve issues with documents in the pending pool. In addition, DocTr offers a exploratory graphical user interface to navigate through the tracked paper documents and explore their metadata. The exploratory GUI presents the paper documents in a list showing the digital representation of the paper documents. In case that the tracking setup did not provide a digital representation, a default icon is used. Users can right click on a document in order to request a document's details as shown in Figure 2. The details panel includes the document's tracking properties such as the unique identifiers (`uris`) but also to which organisational structure (`structureId`) the document belongs to. Furthermore, users can add annotations to the document in the note section. When a document is added to the pending pool, the user will get a notification via the system tray. The system tray allows quick access to the DocTr sidebar shown in Figure 3. Unidentified paper documents are added to the sidebar's upper part. A user can click on the green plus button to indicate that the document is new or use the search button to open the exploratory GUI. After selecting a document, a user can inform DocTr that it is the same document as the one shown in the sidebar by selecting the `Same As` context menu. The pending pool will then add the new unique identifier to the selected document. When a paper document is removed from an organisational structure and cannot be allocated to another structure, the document is added to the lower part of the sidebar. A user can choose to delete a document (bin) or to add it to another organisational structure. In case that a document should be assigned to another organisational structure, a popup is shown with all organisational structures containing non-identified documents as well as an area where a user can indicate whether a document has been placed outside of a monitoring area or given to someone else. This metadata will be stored in the document's log. Depending on a user's actions, the pending pool will notify the Delegator component to proceed with the required actions such as adding the physical object instance as a target to the organisational link at a certain index of the given organisational structure.

HOW TO USE DOCTR

On the server side, we have foreseen two REST interfaces which can be used by client applications to communicate with DocTr as illustrated earlier in Figure 1. We distinguish between physical document tracking client applications and client applications making use of the tracking metadata in order to ease the development of interactive office environments. The REST Input Interface is meant for tracking setups which add information about documents in their monitored area while the REST Output Interface is used by third-party applications intending to use metadata about tracked paper documents and organisational structures.

Input Interface for Tracking Setups

The REST Input Interface includes endpoints for creating new documents as well as organisational structures and for updating metadata values. In addition, it can also be used to interact with the identification mechanism for comparing unique identifiers across tracking setups. First, a tracking setup has to register itself and define which comparator is required for its unique identifiers. In the simplest case, a tracking setup only monitors paper documents without any organisational structures. When a document's unique identifier (i.e. tag number or SIFT features) is detected in this simple case, the tracking setup sends the unique identifier together with additional log entries to DocTr by calling the corresponding endpoint for updating a physical document. The Delegator component requests the identification mechanism for the given physical object and checks whether the detected document is already waiting for allocation in the pending pool. Next, the Delegator updates the data model with the new knowledge. When a tracking setup monitors an organisational structure, it has to create this structure in DocTr in an initialisation phase and keep the returned identifier of the initialised organisational structure. When detecting a change to the organisational structure, the tracking setup has to notify DocTr by sending a JSON string including the identifier of the organisational structure, the index where the document has to be added and the derived unique identifier of the tracked physical document. When a document is removed from an organisational structure or monitoring area, the tracking setup has to call the remove endpoint with the identifier of the organisational structure and the document identifier. As previously mentioned, the document will be added to the pending pool if no other tracking setup recognised the removed document.

Output Interface for Third-party Applications

While the REST Input Interface is used by tracking setups, the REST Output Interface can be used by third-party applications to query DocTr's data model as well as to listen to updates to the data model. The interface provides endpoints for querying the data model entities and exploring documents of the pending pool. Furthermore, applications can establish a WebSocket connection to be notified when a data model entity has changed its value. A JSON message is sent over the WebSocket connection including the identifier of the entity and the new values such as a new location.

INTEGRATION WITH EXISTING SOLUTIONS

In order to illustrate how DocTr can be used in practice, we provide three use cases in different application domains. Note that these use cases have not yet been implemented but rather illustrate the potential and future directions in the development of interactive computing systems based on DocTr.

Integration with Office Environments

Normally, users have different storage artefacts and configurations of their office environments which require different tracking technologies. Based on the fact that DocTr helps integrating different tracking setups, an interactive office application no longer has to be based on a proprietary platform for managing different tracking setups. In practice, we could, for example, use the SOPHYA [12] framework for tracking physical folders in bookcases. In addition, a user might have a monitored area around their laptop on a desk which identifies paper documents based on fiducial markers as done in DocuDesk [7]. In their daily activities, a user might move a paper document from an augmented SOPHYA folder to their workplace around the laptop. DocuDesk and SOPHYA would only have to deal with four DocTr endpoints including the registration, the initialisation of organisational structures, as well as the adding and removal of endpoints. Of course, a tracking setup can always make more extensive use of DocTr. It could, for example, use DocTr to store log entries every time a user accesses a document or store so-called cross-media organisational structures such as a pile that contains physical as well as digital documents.

Integration with Third-party Applications

DocTr can also easily be integrated with third-party applications. For example, an Evernote extension can be developed which listens to DocTr for log updates of paper documents which a user has used in Evernote notes. In addition, Evernote can request the digital representation of paper documents in order to allow users to integrate paper documents in their Evernote notes. Similarly, the digital representation of a paper document can, for example, be used in the File Explorer in order that users can organise digital and physical documents in a unified way. This functionality can be implemented by invoking a WebSocket connection. DocTr will publish new updates of the data model in the form of JSON messages on this WebSocket channel. Third-party applications only have to process the received messages which contain the identifier of the document or organisational structure as well as any additional metadata.

Integration with Context-aware Frameworks

In the digital information space, metadata about digital documents such as the creation time and access frequencies are provided by most operating systems. Third-party applications can easily access this information. For example, recent research in context-aware applications such as CAAD [24] and Passage [9] use this metadata and other data extracted by text mining to determine a user's current activity. Due to the simultaneous use of digital and physical documents in our daily workflows, we can improve these activity recognition applications by providing metadata about paper documents in a similar way as done for their digital counterparts.

EVALUATION

The accuracy of DocTr depends on the accuracy of the used tracking setups and the implementation of the comparators and an evaluation of DocTr as a middleware is therefore only possible by integrating it in larger applications with a significant amount of different tracking setups. Therefore, we have

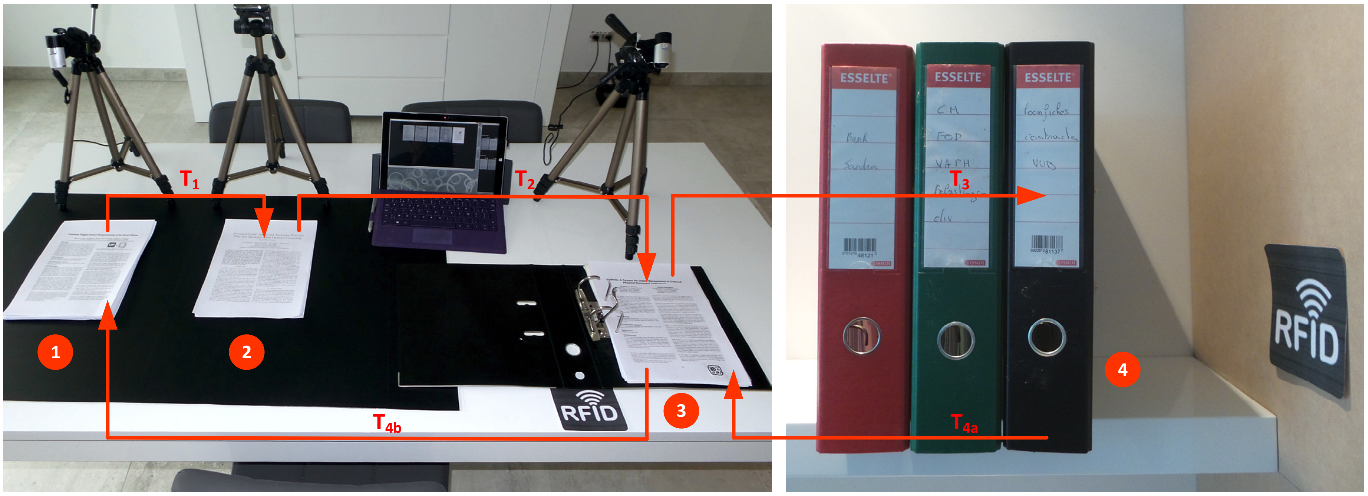

Figure 5: Evaluation setup

chosen for a preliminary evaluation which investigates how well DocTr can handle error-prone tracking setups such as the SIFT algorithm and how the behaviour changes over time when multiple unique identifiers are given to a document. The results show that even with error-prone tracking, the accuracy improves by letting users provide feedback about similar documents in the pending pool.

We have implemented the four tracking setups shown in Figure 5 in order to drive the evaluation. The first two tracking setups ① and ② monitor piles based on the SIFT algorithm. The third tracking setup ③ tracks the adding and removal of paper documents from ring binders on a desk surface. Finally, the last tracking setup ④ monitors a bookcase with ring binders. Note that the presented results are focussing on the features of DocTr and the accuracy of the document tracking depends on the used document tracking setups.

Methodology

The four tracking setups have been used to determine the accuracy of DocTr when moving paper documents across organisational structures with the same or different document identifiers formats as well as for the movement of the organisational structures themselves. Note that in the beginning of the evaluation, documents have been added to the pile monitored by the first tracking setup ①.

T_1: Moving a paper document between organisational structures with the same document identifier format
Tracking setups ① and ② both track physical documents based on the SIFT algorithm. We have used the SIFT Keypoint Detector Java library[3] to extract the SIFT features of documents. Furthermore, as shown in Figure 5 the two tracking setups use different cameras. Note that each task T_i consists of a subtask T_i^R which removes a document from a tracking setup and a subtask T_i^A adding the same document to another tracking setup. In this first task T_1 of our evaluation, documents are removed from the pile of tracking setup ① via T_1^R and added to the pile of tracking setup ② via T_1^A.

T_2: Moving a paper document between organisational structures with different document identifier formats
In order to evaluate the fact that a physical document can have more than one unique identifier across tracking setups, T_2^R moves the documents from the pile in tracking setup ② and T_2^A adds them to a ring binder monitored by tracking setup ③. Tracking setup ③ consists of RFID tagged ring binders, an RFID reader placed underneath the desk and a camera installed above the desk. When physical documents are moved from the pile of tracking setup ②, we added a fiducial marker to them. In the case a ring binder is placed within the range of the RFID reader, the application starts listening to the reacTIVision framework in order to detect fiducial markers. Finally, if any document is detected, the application requests DocTr to add the detected document to the ring binder organisational structure.

T_3: Moving an organisational structure
Since organisational structures can also be tracked by the DocTr framework, in task T_3 the ring binder was removed from the desk via T_3^R and added to the bookcase via T_3^A. Each shelf of the bookcase contains an RFID reader and before placing the ring binder on a shelf, it has to be scanned by the associated RFID reader.

T_4: Moving documents back to the first tracking setup
Finally, we moved the documents of the ring binder back to the pile of tracking setup ①. As part of task T_4, T_{4a}^R first moved the ring binder from the bookcase shelf and T_{4a}^A added it to tracking setup ③. The documents were then removed from the ring binder via T_{4b}^R and added to the pile by T_{4b}^A.

The procedure consisting of task T_1 to T_4 was repeated five times in order to present more accurate results. Furthermore, a total of 55 printed research papers from various conferences and journals were used. We decided to use research papers due to their high similarity when processed by computer vi-

[3] http://www.cs.ubc.ca/~lowe/keypoints

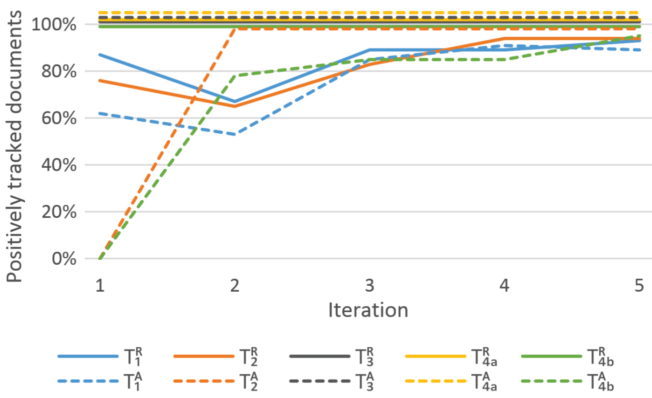

Figure 6: Percentage of positively identified documents

sion techniques which represented a challenge the used document identification techniques.

Results

As described in the methodology, in each task physical documents are first removed from an organisational structure and then added to another organisational structure and the two subtasks are monitored by different tracking setups. The results of the five iterations for the four described tasks T_1 to T_4 are shown in Figure 6. For each iteration and task T_i, we measured the amount of positively identified documents when removing them via subtask T_i^R from on organisational structure and adding them to another organisational structure via T_i^A.

When having a look at the results of T_1^R, T_1^A and T_2^R, we can observe that in the first two iterations the identification mechanism could not accurately match the given unique identifiers based on SIFT. This might be caused by the error proneness of the used SIFT algorithm and the small threshold value defined in the SIFT comparator. In each iteration, the new unique SIFT identifier is manually added to non-recognised documents via the DocTr sidebar. As a result, the matching accuracy improved over the five iterations without reaching 100%. This might be caused by the fact that the two SIFT tracking setups both define their SIFT feature matrices which might slightly diverge from each other. Note that the amount of unmatched physical documents were caused by the inaccuracy of the used SIFT algorithm in the two tracking setups and by using research papers which looked very similar. It is known that the SIFT algorithm is not ideal for identifying similar documents [17] but DocTr can help to improve the accuracy of error-prone tracking setups via user feedback.

In T_2 documents are removed from the pile of tracking setup ② via subtask T_2^R and added to the ring binder ③ via subtask T_2^A. During the first iteration of T_2, documents were augmented with a tag before being added to the ring binder and the DocTr sidebar has been used to allocate the tag number to the document. As a consequence, none of the documents could be identified during this first iteration. The results of T_2^A further show that documents are always positively identified after the first iteration. Furthermore, removing the ring binder from the desk via subtask T_3^R, adding it to the shelf via T_3^A and placing it back to the desk via subtasks T_{4a}^R and T_{4a}^A is always positively tracked. Finally, the removal of documents from the ring binder via T_{4b}^R has also been tracked accurately. Nevertheless, when adding the documents back to the first pile of tracking setup setup ① via T_{4b}^A, due to SIFT-related issues we could observe the same behaviour as for the results of T_1 and T_2. Note that in the first iteration of T_{4b}^A no documents were identified by the SIFT comparator since tags have been added to the documents in the first iteration of T_2^A.

DISCUSSION AND FUTURE WORK

While physical document tracking solutions are recently gaining attention, existing solutions show some limitations. They often only track physical documents in one specific type of organisational structure such as piles or physical folders. In addition, the metadata management of tracked documents is forwarded to the application layer which leads to fragmentation and inconsistency when multiple tracking setups are used. In order to overcome these issues, we presented the DocTr framework which unifies existing tracking setups and provides extensive data management possibilities. With DocTr documents can be tracked across organisational structures regardless of the used tracking setups. In addition, we also support the traceability of the organisational structures themselves which is normally not supported by existing solutions. The DocTr framework currently offers a simple graphical user interface for exploring tracked documents. We plan to develop a more advanced GUI where users can organise their physical documents by also defining associations between them. Furthermore, in a long-term in-context evaluation we plan to investigate the usability of DocTr and gain insights about the trade-off between the effort it takes to maintain the pending pool and not up-to-date tracking metadata. Since physical documents are often used together with digital media, in the near future DocTr might be extended to also deal with the management of a user's digital documents. This would ultimately enable the unified organisation of physical and digital documents through the development of so-called cross-media PIM applications. Finally, there is also potential to further improve the identification mechanism when multiple versions of an identifier are used by removing the versions that have not been matched for a long time.

CONCLUSION

We have presented DocTr, a unifying framework for tracking physical documents and organisational structures across different tracking setups. DocTr deals with data provided by various tracking setups and makes this data available to third-party applications. The exploratory graphical user interface and DocTr sidebar further enable end users to explore tracked documents and provide feedback for documents that have not be recognised correctly. Our general DocTr framework offers an ideal platform for future research on interactive computing systems including interactive office environments and personal cross-media information management in particular.

ACKNOWLEDGEMENTS

The research of Sandra Trullemans is funded by the Agency for Innovation by Science and Technology in Flanders (IWT).

REFERENCES

1. Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. SURF: Speeded up Robust Features. In *Proceedings of European Conference on Computer Vision (ECCV 2006)*. Springer Berlin Heidelberg, 404–417. DOI: http://dx.doi.org/10.1007/11744023_32

2. Victoria Bellotti and Keith Edwards. 2001. Intelligibility and Accountability: Human Considerations in Context-Aware Systems. *Human-Computer Interaction* 16, 2–4 (2001), 193–212. DOI: http://dx.doi.org/10.1207/S15327051HCI16234_05

3. Ofer Bergman, Ruth Beyth-Marom, Rafi Nachmias, Noa Gradovitch, and Steve Whittaker. 2008. Improved Search Engines and Navigation Preference in Personal Information Management. *Transactions on Information Systems* 26, 4 (2008), 20. DOI: http://dx.doi.org/10.1145/1402256.1402259

4. Paris Buttfield-Addison, Christopher Lueg, Leonie Ellis, and Jon Manning. 2012. "Everything Goes Into or Out of the iPad": The iPad, Information Scraps and Personal Information Management. In *Proceedings of Australian Computer-Human Interaction Conference (OzCHI 2012)*. ACM, 61–67. DOI: http://dx.doi.org/10.1145/2414536.2414546

5. Kai-Yin Cheng, Rong-Hao Liang, Bing-Yu Chen, Rung-Huei Laing, and Sy-Yen Kuo. 2010. iCon: Utilizing Everyday Objects as Additional, Auxiliary and Instant Tabletop Controllers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2010)*. ACM, 1155–1164. DOI: http://dx.doi.org/10.1145/1753326.1753499

6. Amay Dighe and Annika Hinze. 2012. Human-Centred Workplace: Re-finding Physical Documents in an Office Environment. In *Proceedings of the Annual Conference of the New Zealand Chapter of the ACM Special Interest Group on Computer-Human Interaction (CHINZ 2012)*. ACM, 9–16. DOI: http://dx.doi.org/10.1145/2379256.2379259

7. Katherine M. Everitt, Meredith Ringel Morris, A.J. Bernheim Brush, and Andrew D. Wilson. 2008. DocuDesk: An Interactive Surface for Creating and Rehydrating Many-to-Many Linkages Among Paper and Digital Documents. In *Proceedings of IEEE International Workshop on Horizontal Interactive Human Computer Systems (TABLETOP 2008)*. IEEE, 25–28. DOI: http://dx.doi.org/10.1109/TABLETOP.2008.4660179

8. Jim Gemmell, Gordon Bell, Roger Lueder, Steven Drucker, and Curtis Wong. 2002. MyLifeBits: Fulfilling the Memex Vision. In *Proceedings of the 10th ACM international Conference on Multimedia (MULTIMEDIA 2002)*. ACM, 235–238. DOI: http://dx.doi.org/10.1145/641007.641053

9. Karl Gyllstrom. 2009. Passages Through Time: Chronicling Users' Information Interaction History by Recording When and What They Read. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI 2009)*. ACM, 147–156. DOI: http://dx.doi.org/10.1145/1502650.1502673

10. Matthew Jervis and Masood Masoodian. 2013. Visualization of Physical Library Shelves to Facilitate Collection Management and Retrieval. In *Proceedings of the Symposium on Engineering Interactive Computing Systems (EICS 2013)*. ACM, 133–138. DOI:http://dx.doi.org/10.1145/2494603.2480321

11. Matthew Jervis and Masood Masoodian. 2014. How Do People Attempt to Integrate the Management of Their Paper and Electronic Documents? *Aslib Journal of Information Management* 66, 2 (2014), 134–155. DOI: http://dx.doi.org/10.1108/AJIM-01-2013-0007

12. Matthew G. Jervis and Masood Masoodian. 2010. SOPHYA: A System for Digital Management of Ordered Physical Document Collections. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction (TEI 2010)*. ACM, 33–40. DOI: http://dx.doi.org/10.1145/1709886.1709894

13. William Jones and Jaime Teevan. 2007. *Personal Information Management*. University of Washington Press.

14. Martin Kaltenbrunner and Ross Bencina. 2007. reacTIVision: A Computer-Vision Framework for Table-Based Tangible Interaction. In *Proceedings of the International Conference on Tangible and Embedded Interaction (TEI 2007)*. ACM, 69–74. DOI: http://dx.doi.org/10.1145/1226969.1226983

15. David Karger, Karun Bakshi, David Huynh, Dennis Quan, and Vineet Sinha. 2005. Haystack: A Customizable General-Purpose Information Management Tool for End Users of Semistructured Data. In *Proceedings of the 1st Biennial Conference on Innovative Data Systems Research (CIDR 2005)*. 13–26.

16. Mohammadreza Khalilbeigi, Jürgen Steimle, Jan Riemann, Niloofar Dezfuli, Max Mühlhäuser, and James D. Hollan. 2013. ObjecTop: Occlusion Awareness of Physical Objects on Interactive Tabletops. In *Proceedings of the ACM international Conference on Interactive Tabletops and Surfaces (ITS 2013)*. ACM, 255–264. DOI: http://dx.doi.org/10.1145/2512349.2512806

17. Jiwon Kim, Steven M. Seitz, and Maneesh Agrawala. 2004. Video-Based Document Tracking: Unifying Your Physical and Electronic Desktops. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST 2004)*. ACM, 99–107. DOI:http://dx.doi.org/10.1145/1029632.1029650

18. Mizuho Komatsuzaki, Koji Tsukada, and Itiro Siio. 2011. DrawerFinder: Finding Items in Storage Boxes Using Pictures and Visual Markers. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI 2011)*. ACM, 363–366. DOI: http://dx.doi.org/10.1145/1943403.1943466

19. Dawn Lawrie and Daniel Rus. 1999. A Self-Organized File Cabinet. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM 1999)*. ACM, 499–506. DOI: http://dx.doi.org/10.1145/319950.320056

20. Chunyuan Liao, Hao Tang, Qiong Liu, Patrick Chiu, and Francine Chen. 2010. FACT: Fine-Grained Cross-Media Interaction With Documents Via a Portable Hybrid Paper-Laptop Interface. In *Proceedings of the International Conference on Multimedia (MM 2010)*. ACM, 361–370. DOI: http://dx.doi.org/10.1145/1873951.1874001

21. David Lowe. 1999. Object Recognition From Local Scale-Invariant Features. In *Proceedings of the International Conference on Computer Vision (ICCV 1999)*. IEEE, 1150–1157. DOI: http://dx.doi.org/10.1109/ICCV.1999.790410

22. Thomas W. Malone. 1983. How People Organize Their Desks? Implications for the Design of Office Information Systems. *ACM Transactions on Office Information Systems (TOIS)* 1, 1 (1983), 99–112.

23. Kazuhiro Matsushita, Daisuke Iwai, and Kosuke Sato. 2011. Interactive Bookshelf Surface for In Situ Book Searching And Storing Support. In *Proceedings of the Augmented Human International Conference (AH 2011)*. ACM. DOI: http://dx.doi.org/10.1145/1959826.1959828

24. Tye Rattenbury and John Canny. 2007. CAAD: an Automatic Task Support System. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI 2007)*. ACM, 687–696. DOI: http://dx.doi.org/10.1145/142750.143007

25. Jan Riemann, Mohammadreza Khalilbeigi, Niloofar Dezfuli, and Max Mühlhäuser. 2015. StackTop: Hybrid Physical-Digital Stacking on Interactive Tabletops. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI EA 2015)*. ACM, 1127–1132. DOI: http://dx.doi.org/10.1145/2702613.2732742

26. Leo Sauermann, Gunnar Aastrand Grimnes, Malte Kiesel, Christiaan Fluit, Heiko Maus, Dominik Danish Heim, DominHeim, Benjamin Horak, and Andreas Dengel. 2006. Semantic Desktop 2.0: The Gnowsis Experience. In *Proceedings of the International Semantic Web Conference (ISWC 2006)*. Springer Berlin Heidelberg, 887–900. DOI: http://dx.doi.org/10.1007/11926078_64

27. Leo Sauermann and Dominik Heim. 2008. Evaluating Long-Term Use of the Gnowsis Semantic Desktop for PIM. In *Proceedings of the International Semantic Web Conference (ISWC 2008)*. Springer Berlin Heidelberg, 467–482. DOI: http://dx.doi.org/10.1007/978-3-540-88564-1_30

28. Thomas Seifried, Jervis Matthew, Michael Haller, Masood Masoodian, Masoodian, and Nicolas Villar. 2008. Integration of Virtual and Real Document Organization. In *Proceedings of the International Conference on Tangible and Embedded Interaction (TEI 2008)*. ACM, 81–88. DOI: http://dx.doi.org/10.1145/1347390.1347410

29. Abigail Sellen and Richard Harper. 2001. *The Myth Of The Paperless Office*. MIT Press.

30. Beat Signer and Moira C. Norrie. 2007. As We May Link: A General Metamodel for Hypermedia Systems. In *Proceedings of the 26th International Conference on Conceptual Modeling (ER 2007)*. Springer Berlin Heidelberg, 359–374. DOI: http://dx.doi.org/10.1007/978-3-540-75563-0_25

31. Jeff Smith, Jeremy Long, Tanya Lung, Mohd Anwar, and Sriram Subramanian. 2006. PaperSpace: A System for Managing Digital and Paper Documents. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI EA 2006)*. ACM, 1343–1348. DOI: http://dx.doi.org/10.1145/1125451.1125700

32. Sandra Trullemans and Beat Signer. 2014a. From User Needs to Opportunities in Personal Information Management: A Case Study on Organisational Strategies in Cross-Media Information Spaces. In *Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2014)*. IEEE, 87–96. DOI: http://dx.doi.org/10.1109/JCDL.2014.6970154

33. Sandra Trullemans and Beat Signer. 2014b. Towards a Conceptual Framework and Metamodel for Context-Aware Personal Cross-Media Information Management Systems. In *Proceedings of the International Conference on Conceptual Modeling (ER 2014)*. Springer International Publishing, 313–320. DOI: http://dx.doi.org/10.1007/978-3-319-12206-9_26

34. Pierre Wellner. 1991. The DigitalDesk Calculator: Tangible Manipulation on a Desk Top Display. In *Proceedings of the Symposium on User Interface Software and Technology (UIST 1991)*. ACM, 27–33. DOI: http://dx.doi.org/10.1145/120782.120785

35. Pierre Wellner. 1993. Interacting with Paper on the DigitalDesk. *Commun. ACM* 36, 7 (1993), 87–96. DOI: http://dx.doi.org/10.1145/159544.159630

36. Steve Whittaker and Julia Hirschberg. 2001. The Character, Value and Management of Personal Paper Archives. *Transactions on Computer-Human Interaction* 8, 2 (2001), 150–170.