A Multimedia Analytics Framework for Browsing Image Collections in Digital Forensics

Marcel Worring
Informatics Institute
University of Amsterdam
P.O. Box 94323
1098 XH Amsterdam
The Netherlands
m.worring@uva.nl

Andreas Engl
Informatics Institute
University of Amsterdam
P.O. Box 94323
1098 XH Amsterdam
The Netherlands
lynn@cephea.de

Camelia Smeria
Informatics Institute
University of Amsterdam
P.O. Box 94323
1098 XH Amsterdam
The Netherlands
camyyssa@gmail.com

ABSTRACT

Searching through large collections of images to find patterns of use or to find sets of relevant items is difficult, especially when the information to consider is not only the content of the images itself, but also the associated metadata. Multimedia analytics is a new approach to such problems. We consider the case of forensic experts facing image collections of growing size during digital forensic investigations. We answer the forensic challenge by developing specialised novel interactive visualisations which employ content-based image clusters in both the analysis as well as in all visualizations. Their synergy makes the task of manually browsing these collections more effective and efficient. Evaluation of such multimedia analytics is a notoriously hard problem as there are so many factors influencing the result. As a controlled evaluation, we developed a user simulation framework to create image collections with time and directory information as metadata. We apply it in a number of scenarios to illustrate its use. The simulation tool is available to other researchers via our website.

Categories and Subject Descriptors

H.3 [**Information storage and retrieval**]: Information Search and Retrieval

Keywords

Image Search, Information Visualization, Visual Analytics

1. INTRODUCTION

Searching for images in large collections is difficult as the information of interest might be captured in any of the various modalities associated with the image. The determining clues for the search might be in the content of the image, but might also be in its metadata. Furthermore, the best way to search is highly dependent on the task, ranging from seeking an abundance of results to little nuggets of information. To cater for all these tasks a flexible and interactive integral view of the different information channels is needed. In this paper we focus on the specific case of image collections and their metadata in the context of digital forensics, but many of the ideas transfer easily to other application fields in which along with images multiple channels of information play a role.

In digital forensics an investigator is often tasked to analyse a data source containing more than a million images and their metadata. In order to filter out files, the cryptographic hashes of all files are compared with large databases of known incriminating material as well as common benign files. The most famous reference set is the National Software Reference Library (NSRL) which contains hashes for over 31 million files [24]. With such an approach we can make a significant reduction in the number of images to consider for example used in XIRAF [1]. The more cautious offenders can easily circumvent such an attack as changing a single bit will radically alter the hash of a file. The police is therefore now moving to near copy detection to solve this [18]. This reduces the workload, but doesn't solve the real problem namely that the interest lies in finding evidence in unkwnown material. Apart from the content of the images there is a lot of information present in the metadata. It is e.g. rarely the case that a person got into possession of illicit image material randomly. Downloading and filesharing habits are often governed by usage patterns with varying complexity and we can use the MAC times (last modified, last accessed and created) to understand patterns of use. Similarly, the file path can be used to extract hierarchical information that can shed light on naming patterns and other organisational schemes. To perform digital forensics on such a collection, Richard et al. argue that this "not only requires better acquisition tools, but also better analysis tools", as "investigators must be relieved of manual, time-consuming tasks" [24]. We focus on automated methods to make the search in image collections easier by considering both the content and the metadata.

Along the dimension of varying degrees of automation there are essentially four different categories that arise.

Most traditional designs in forensics do not go beyond presenting images as a sequential list of files that have to be inspected one by one, thus "manual browsing" is still the dominant category.

To support browsing, methods in the second category fo-

MM'12, October 29–November 2, 2012, Nara, Japan.
Copyright 2012 ACM 978-1-4503-1089-5/12/10 ...$15.00.

cus on the metadata alone. CyberForensics Timelab [21] is an example of a system which is focused on time as metadata. Standard file explorers and text search systems can find relevant filenames.

The third category methods aim to arrange the images in a meaningful way. By doing so, images of a target class are closer together, thus finding at least one image of that class will make the process of finding related images easier. At the same time, images that are very different from the target class will have a low probability of relevant images in its vicinity, thus the surrounding region can be browsed with less attention and - consequently - faster. Quadrianto does so using a grid based approach [22] whereas Nguyen uses a similarity preserving projection ([19]) coupled with an approach to minimise the overlap between images.

The fourth and last category is Content-Based Image Retrieval and there is a wide variety of different approaches and features (see Datta et al. [8] for a recent survey). In theory this allows the highest possible degree of automation: given a perfect classifier for our target class we would not have to look at a single image anymore. In practice, however, humans still outperform machines in a classification task in terms of accuracy, and frequent misclassifications are not uncommon.

It should be clear that using traditional techniques on the problem of browsing large image collections is insufficient. Manual browsing is clearly too time-consuming, but each of the three automated categories by themselves do not provide a solution either. Visualisation-based browsing and content-based retrieval discard important meta-information, while retrieval purely based on meta-data doesn't consider the actual contents of images. Even in state-of-the-art research, full automation has not been achieved with any of the approaches. Interactive browsing techniques such as [31] are promising avenues to improve the seach performance, but they focus on the underlying algorithm only and ignore the visualization.

All of the three semi-automated techniques have their merits, but the references study them in isolation. The overarching aim of this paper is to combine the strength of each method in order to overcome the individual weaknesses. This means we need to communicate the result of content-based retrieval methods to a human operator who uses a visualisation-based browsing interface to make sense of them. Further, all of the meta-information available in the dataset has to be reflected by the design of the visualisations and accessible to the operator. Then, and only then, will we be able to take full advantage of the available data as well as computational and human resources. Research argues that a suitable framework for such an undertaking is Visual Analytics [15].

Visual Analytics is a relatively young field, and even the term has only been in use since "Illuminating the Path" [27] was published in 2005 [3]. Keim et al. [15] define it as "combining automated analysis techniques with interactive visualisations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets". From this definition it becomes clear that Visual Analytics puts a stronger emphasis on the human component than its sibling disciplines. It strives to combine the strengths of human and that of electronic data processing - a semi-automated analytical process. Even more recently visual analytics has been combined with multimedia analysis, coined as multimedia analytics [6][9]. This paper is the first to develop a framework in which multimedia analysis and visualizations work in conjunction to support the browsing of large visual collections in digital forensics.

Section 2 will highlight similar or related systems, both in terms of the underlying problem as well as systems that use approaches that might potentially be relevant for the task at hand. Section 4 presents our visualisation system for browsing image collections which integrates a set of advanced visualizations with content based analysis into a multimedia analytics solution. To evaluate the system we have developed tools to simulate users which are made available to the community. These are defined in Section 5. Section 6 walks through a series of scenarios that serve as use-cases to test the system. Finally section 7 summarises the results.

2. RELATED WORK

From the thus far mentioned prior art one can identify three different approaches to visually analysing large image collections. Although time-based visualisations have a long and successful history in research, the application of time as a primary or auxiliary dimension for purposes of digital forensics is surprisingly shallow. The arguably most prevalent category of techniques in the context of using visualisations to browse image collections is visualising the images on a 2D plane, usually underlying a layouting algorithm that maximises visibility and/or groups similar images closer together. The third category of systems arranges images in a network or tree in order to make relationships (e.g. similarity) between the images verbose. This section shows novel, related achievements in research in each of these three areas.

2.1 Time-based Visualisation Systems

In [14] it is argued that time is an especially relevant dimension in almost all visualisations since we are actively living in space and time. As such, all information has at least some relation to time which is especially true for digital files: all major disk formatting tables include time stamp information. Unsurprisingly, visualisations that are centred on the time dimension are a pervasive topic in research.

The influential Continuum [2] added a lot of new insights to timeline visualisations. By using histograms as an overview Continuum scales to an arbitrary number of items since time intervals for each histogram bin can be chosen independently from the data. The detail view is comprised of bars where related information is aggregated into a single item. In its essence, each bar in Continuum represents a cluster of information. Cluster Calendar View [29] makes the use of clusters even more explicit. Using a simple bottom-up clustering algorithm, it uses two visualisations, a calendar and a timeline. The calendar shows dominant clusters for any given day by colour-coding them. The timeline displays the averaged data for all clusters across a periodic time interval. This effectively condenses the data in order to emphasise trends and outliers, e.g. revealing differences between workdays and weekends.

In forensic investigations [7] time has rarely been used as a primary dimension in forensic visualisation systems. We highlight two systems. Zeitline [4], allows hierarchical organisation of events by grouping discrete events into what they call "super events", such as the installation of a program which is comprised of many read and write events of various files. These events are simply displayed in a hierarchical tree view, thus the visualisation capabilities are very limited. The CyberForensic TimeLab [21] is motivated by the lack of overview in Zeitline. TimeLab's viewer is comprised of a vertically stacked list of histograms, where each row corresponds to a specific source. The x-axis represents time, thus histogram bars show activity within a certain time interval. When a selection is made, a detail view shows a textual list of events.

2.2 2D Information Landscapes

Nguyen et al. [19] demonstrate a system for image annotation that uses a combination of projection techniques to map a large number of images into two-dimensional space. A point cloud is used as an overview over the entire image collection, whereas a number of representative images are shown in the central view. Each of these represent one cluster and are sampled using k-means, picking the images closest to the cluster centroids as representatives. Once the user selects an image he/she descends into the corresponding cluster and is able to annotate it.

Zooming as an interaction technique has also been explored by Girgensohn et al.'s MediaGLOW, [11] a visualisation system that aims to support users in organising their personal photo collection. Much like Nguyen et al.'s system [19] it projects cluster representatives (clusters are called "stacks") onto an information landscape by minimising the energy of a fully-connected, spring-based graph, with edges corresponding to the similarity between stacks of images. Analogous to the idea of zoom modifiers in Zoomable Object-oriented information Landscapes [13], MediaGLOW's zoom keeps the size of images constant while performing a geometric zoom, with the result that images that partially overlap gradually become visible. The combination with a timeline histogram provides both an overview in time and in space as well as a selection mechanism on either dimension.

TimeScape [23] uses an information landscape in combination with a time-based organisational approach. This spatio-temporal personal information management system enables users to exclusively use their desktop to spatially lay out their documents. Effectively this replaces hierarchical organisation with a temporal one. TimeSpace [17] extends this idea by partitioning the desktop into "multiple activity-oriented virtual workspaces".

Unlike time-based visualisations, there is very little consent on both the organisation as well as the interaction techniques for images in information landscapes. These systems are very usage specific, and their purposes range from full personal information management down to very specific tasks such as image annotation or browsing photo collections. As a consequence the layout of the images is often context specific or even entirely positioned by the user, but in most cases there is an underlying graph algorithm that uses some measure of similarity in order to cluster related images.

2.3 Graph-based Image Visualisation Systems

The distinction between images as leaves of a tree or nodes in a network, to that of projection into an information landscape is sometimes a shallow one as many of these projections use the very same graph-based layouting algorithms to position the images in space. Explicitly drawing the connections between individual items enables the user to reason about the relationship between items and to reflect on the resulting layouts rather than being forced to take it for granted. Trees and networks also allow a set of operations that would otherwise be impossible, such as "cutting" a connection in order to filter a branch or sub-network.

Chen et al. [5] developed an image retrieval technique based on Pathfinder networks that were originally devised to analyse proximity data in psychology. Using a similarity score based on colour, shape and texture features, Pathfinder networks are constructed in order to group images with similar appearance. Although the reference introduces a nice visualization, it doesn't allow for interaction.

Vizster [12], derives its network structure from a social network, yet still qualifies in this context as images are chosen to represent profiles. Vizster employs a force-directed layouting strategy which automatically groups users into communities, or, in more general terms, nodes into clusters.

PEx-Image [10] is a tool that creates a static projection of a collection of images. What is novel here is that the underlying structure is a rooted tree, not a network, more specifically a phylogenetic tree originally used for visualising evolutionary relationships and thus more commonly seen in bio-informatics. Using the neighbour-joining algorithm with a radial layout and subsequently applying a custom force-based algorithm to minimise the tree's energy, PEx-image accomplishes very compact, clustered graphs.

3. TIME, EVENTS, AND CLUSTERS

While most concepts will be defined when introduced, the notion of events will appear throughout the paper in various different contexts. As events are time based let us first define a timestamp:

- *Timestamp:* a specific moment in time, here we consider it to consist of the Microsoft Windows modified, accessed, created metadata.

- *Event:* a temporal abstraction of an image collection i.e. an aggregation of several images along a common, time-based criterion.

As an example, an event can be "a batch operation (e.g. a download of several images) that was started on March 23rd at 8:00 am". It could also be the subsequent operation where the user moved the same images to a different folder. Chow et al. [7] created a large list of such events that in an investigation are identifiable by looking at the associated timestamps of the images only.

Clusters are a similar notion as events, but not based on time, but aggregation based on visual content:

- *Cluster:* a group of visually similar images.

4. CLUSTER-BASED VISUALISATION

We will first define how to represent clusters across the visualisation and then consider the interaction design of the visualisation system by looking at each visualisation in turn.

4.1 Cluster Representation

Different visualisations often require different ways of depicting primitives. In a timeline where time is mapped to the Y axis and the primitives encode duration, they need to be stretched horizontally to indicate the start and end.

In the graph visualisation, we can arbitrarily assign a representation for the node, but using the same bars as in the timeline hardly makes sense as the X axis encodes spatial information. Yet the primitives in either visualisation are not only of the same type, they are two different views on the same data.

Ideally we want to be able to link the spatial (similarity) information to the temporal given those two visualisations. If we can design the primitive representation in such a way that every single primitive in the visualisation is discernible along at least one of the visual channels, the user can simply single out any primitive by searching for the same properties of the channel in the other visualisation. As long as the representation is identical we don't even have to know the exact channel that distinguishes an item, our bias in the primary visual cortex is adjusted automatically along all of the channels.

First, however, we need to solve the problem of primitives having different representations in the timeline and graph visualisation. This turns out to be reasonably straightforward: since primitive representations in the graph are arbitrary - vertices have no visual encoding themselves - we can simply choose them to have icon representations, where each image cluster is assigned a unique icon. Regardless of what we pick for the timeline primitives (e.g. lines or bars) we can associate them by visually linking them with the same icons.

To choose the icons, Colin Ware [30] gives an excellent overview on how Visual Search is optimised along the channel of colour and elementary shape. He suggests that the two primary considerations for picking colours are visual distinctness and learnability. The opponent-process theory gives us a set of strong hues that are visually very distinct. We can follow these up by colours with relatively consistent names, such as Orange, Purple, Grey or Pink. We can sacrifice some of the discernibility along the colour channel by shifting the colours towards a more aesthetic colour scheme. We can recover and even improve much of the learnability aspect by introducing objects that are close to the original shapes by having some real-world meaning, while improving the overall design at the same time. Once the location of an object has been learnt, spatial memory ensures that it can easily be found again. Figure 1, 2 and 3 show how the cluster icons are used in each of the visualizations with the aim to visually link them.

4.2 Timeline Visualisation

Primitives are structured and arranged. In addition to bars that encode the temporal dimension on the horizontal axis, each cluster or event contains a histogram. A histogram bin shows the activity for a fixed time segment. The icon that represents the cluster is superimposed on the bar to aid Visual Search of a cluster in related visualisations.

The timeline aims at revealing trends that are either self-evident by looking at the temporal information itself, such as an increased amount of activity during a certain period of time, or are revealed in conjunction with other dimensions, such as a directory that has seen many changes recently. The first activity is supported on a largely behavioural level. We have learnt to read Gantt-like diagrams and histograms so well that there is little reflection needed in order to spot trends, outlier, activity spikes or other emerging patterns. To see relations with other channels, we need a wider arsenal of interactive tools to facilitate reflection.

Filter-Select-Highlight

The Visual Analytics Mantra [16] compels us to "show the important", and the timeline is designed to support that proposition by encoding events and activity in a straightforward way. But it has to be backed up with zooming and filtering capabilities to support the full Visual Analytics process. Besides highlighting and selecting clusters with the mouse by means of hovering and clicking respectively, the timeline additionally enables temporal highlighting and selections. Histogram bins can be hovered to highlight contained images. Alternatively, a time period can be selected by clicking anywhere to set a start point, dragging and finally releasing the mouse button to complete the selection (see figure 1).

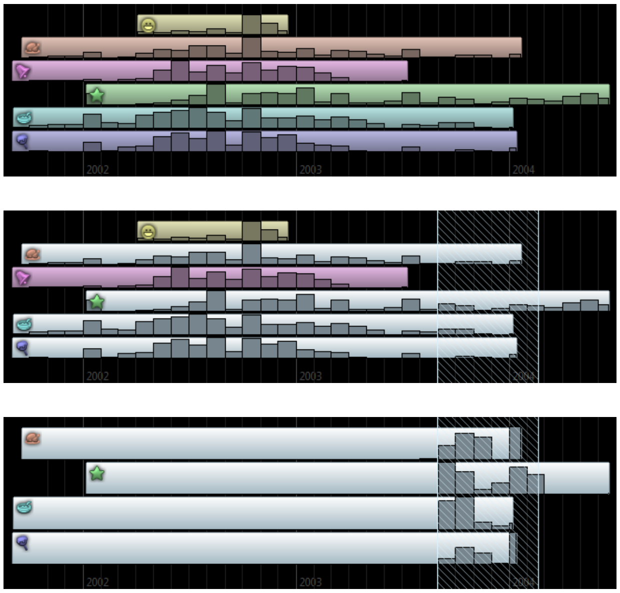


Figure 1: The timeline before (top) and after (middle) selecting a time period. Turning the selection into a filter (bottom) re-evaluates the histograms and strips empty clusters.

At any point, a selection (which can be composed of criteria along multiple dimensions) can be turned into a filter. If a filter contains a criterion other than a set of clusters, the histogram is updated to reflect the distribution of the remaining images in the data set (see figure 1).

It was Shneiderman [25] who noted that "[users] have highly varied needs for filtering". We use his principle of "dynamic queries", or "direct-manipulation queries". Graphically manipulating objects on the screen, e.g. by dragging a rectangle in the timeline, is linked to a certain filter criterion. Using "rapid, incremental and reversible actions" along with immediate feedback, users can build up complex filtering conditions composed of temporal and spatial conditions without ever having to explicitly formulate a query. Note that selection and filtering conditions do not have to be formulated in one visualisation alone - they can be freely combined across the system.

4.3 Graph Visualisation

The graph visualisation shows a network of image clusters, the topology of which is constructed using Pathfinder networks [5], whereas the layout is calculated using a force-based algorithm. Each node in the network represents an image cluster. The edges between clusters encode their similarity.

The aim of the graph is to provide an overview over a part of the spatial dimension of our image collection. This is accomplished on several levels. After the initial layout is calculated it is purposely held fixed throughout the remaining interaction steps (unless the user explicitly wishes to re-arrange the nodes). On a pre-attentive level this helps the user to form a spatial memory of the image clusters.

On the reflective level it allows reasoning about the clustering, and thus implicitly about the image collection. This level differs depending on what clustering the user chooses. For similarity-based clusters it will reveal how diverse the image collection is. If the collection consists only of holiday photos, most of which are taken outdoors, it will result in a more densely collected network, for example. A time-centred clustering, on the other hand, might put holiday trips undertaken at different times in different clusters, possibly connecting those that show similar sceneries or were edited or browsed in close temporal proximity. Figure 2 shows an example of a Pathfinder network.

Filter-Select-Highlight

A cluster is highlighted when hovering it with the mouse. By clicking it, it becomes selected. Both highlighted and selected clusters are marked with a distinct, coloured frame. By using a modifier key, clusters can be added or removed from the current selection. Furthermore, multiple clusters can be selected by click-dragging with the mouse, which creates a selection rectangle.

Clusters are visible as long as there is still an unfiltered image in the data source, but they will disappear once all images have been filtered. Any selection can be turned into a filter as well, either filtering out selected or unselected clusters.

Detail-on-demand: Peeking

With clustering methods using a variety of different criteria, reasoning about the image collection becomes increasingly complicated. Even though the detail visualisation allows us to browse the contents of a cluster, depending on the intra-cluster ordering of images it can still be a time-consuming process to obtain a sense of what a certain cluster encompasses. For that reason the graph implements an additional operation called “peeking”, or “peek zoom”.

Peeking is initiated by highlighting a cluster and then using the mousewheel to zoom into it. A pseudo-geometric zoom animation is displayed which will reveal a limited number of cluster representatives; images that best summarise the cluster. The distance between these clusters is used to construct an ad-hoc network of the representatives. The choice of representatives and their distances is specific to the clustering algorithm.

The zooming uses a variety of techniques. Fixed-shape zooming is used to retain the size of the clusters. Once zoomed in, the focused cluster is semantically enhanced by showing its representatives, an operation known as “semantic zooming”. Clusters connected by at least one edge are projected onto a circle around the zoomed cluster, and thus the distance is no longer correct, but the angle is retained. This provides a sense of context to the zoomed view. The illusion of a geometric zoom is accomplished by moving unconnected clusters out of the view. The movement vector is calculated by taking the vector from the centre of the visualisation to each cluster, normalising it and multiplying it by a constant. Refer to figure 2 for “peeking” as it is used in practice.

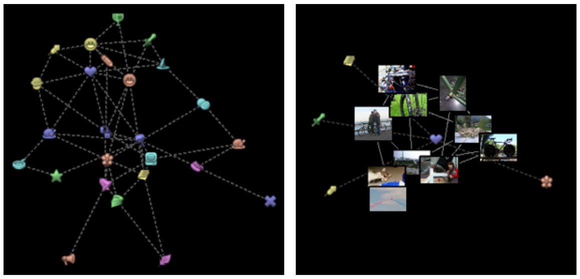


Figure 2: A pathfinder network based on visual similarity (left) and illustration of a zoom-in on a cluster showing representative images (right).

4.4 Hierarchy Visualisation

The purpose of the hierarchy visualisation is to connect the clusters and individual images back to their physical location on the disk, and thus complements the spatial dimension. The major difference to the spatial information provided in the graph visualisation is that the file structure has a natural hierarchy. The heatmap uses greyvalues, as these are the best in conveying different values, is used to provide a visual representation of the connection between the different directories and the derived clusters. This is illustrated in figure 3.

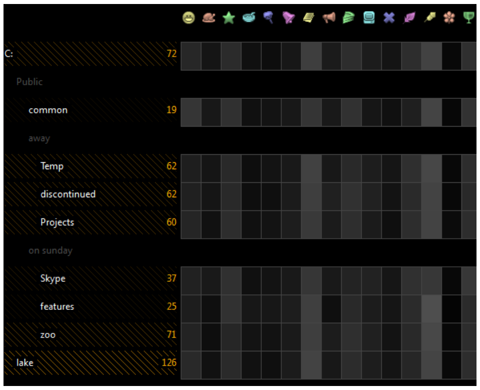
Filter-Select-Highlight

Even without any interaction, the hierarchy visualisation offers an easy to understand overview of all the folders on the hard drive that contain images, and how they are nested. It is the integration into the system, however, that truly makes this visualisation a good fit.

Highlighting works in various ways. Upon hovering an image with the cursor anywhere in the visualisation, the background of the corresponding folder is coloured. If a cluster is highlighted instead, the colour spreads across all the directories that contain images that are contained in this cluster. An exact number is shown next to each folder to indicate the number of hits. Additionally, the intensities of the background colouring is scaled by the number of hits over the maximum number of hits (meaning the folder that has the highest number, not the sum of all of them; see figure 3). This redundant encoding helps the user to quickly determine the dominant folders for a given cluster.

Directories within the visualisation itself can be highlighted as well. This will cause all other visualisations to both highlight the images that are contained in this folder, as well as the clusters that contain one or more of these images.

The hierarchy visualisation - just like the timeline and the graph visualisations - can be used as an entry point for the detail visualisation by selecting a directory. Cluster



C: 72
Public
common 19
away
Temp 62
discontinued 62
Projects 60
on sunday
Skype 37
features 25
zoo 71
lake 126

Figure 3: A heatmap showing the distribution of images in directories over the different clusters to aid the process of finding the directories with most candidate matches or suspicious distribution.

selections are shown analogous to cluster highlights, except encoded in a different colour.

Once all images of a certain directory are filtered out, the corresponding entry in the directory tree disappears as well.

4.5 Detail Visualisation

The detail visualisation is responsible for allowing the user to browse the actual images exhaustively. For reasons of simplicity and familiarity the detail visualisation is a simple grid of images (see figure 4).

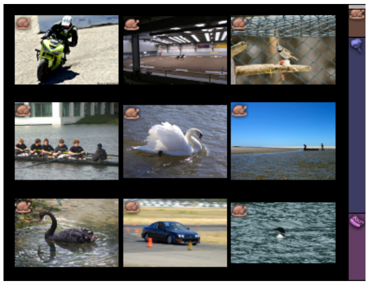


Figure 4: A selection of three clusters shown in the detail visualisation. Icons on top of the images identify the cluster each image is part of.

To ensure a connection between the detail view and the clusters as well as providing some limited sense of overview without having to resort to other visualisations, the detail visualisation introduces an enhanced version of a regular scrollbar. Just like a traditional scrollbar it shows the fraction and position of the actual visible region in relation to the entire set of selected images. It improves this concept by additionally plotting the different clusters as coloured regions - the order of the images as shown in the grid is partitioned into clusters - which can be clicked in order to jump between clusters. The icon representation of each cluster is embedded into the region as well. Refer to figure 4 for an example of the scrollbar.

5. A SIMULATION ENVIRONMENT

Evaluating complex user interfaces is a notoriously difficult problem. In complex multimedia analytics scenarios, the functionality of the interface is typically very rich and it is difficult and not even desirable to steer and formalize the users' quest for information. Therefore, it is unfeasible to test various usage scenarios with formal user studies which are typically meant to test small variations of the same interface. Benchmarks are another option and TRECvid has played an important role in developing interactive retrieval methods [26]. But there we have also seen that comparing the merits of different methods is difficult, even while the solutions are very much driven by the given benchmark task. North even advocates that for visualization driven systems benchmarks should be abandoned completely and that the focus should shift to the insight gained by using the interface [20] by freely employing its functionality in complex tasks. This is indeed the ultimate aim of any multimedia analytics interface, but before that the developers themselves should be able to flexibly test different scenarios, different visualizations and algorithms and see whether the expected patterns indeed are visible and whether they have the potential to support the search process.

5.1 Actions and events

For the digital forensics case, a diverse range of disk images with large image collections would be required as well as semantic information on the images (such as concepts present in the image). Unfortunately the research data available for digital forensics comes nowhere close to meeting any of these requirements, making the evaluation of the interface unfeasible using existing disk images. We have therefore developed a user simulation framework to create and manipulate usage patterns building upon the rules defined in [7]. The simulation provides us with a meta-data structure (consisting of the Microsoft Windows modified, accessed and created timestamps and the file paths) for the images and therefore it is only concerned with user patterns that directly influence time, paths and actions or events [1].

The actions (e.g. moving files from one location to another, scanning the files) are referred to as high level actions and represent the usage model that generates a certain pattern (based on the different rules defined in [7]). They generally refer to batch operations (i.e. operations applied to more than one file) or operations applied to multiple files in quick succession. Low-level actions represent actions on single files. Patterns are timestamp relationships on one or multiple files that can be observed on the final disk image. The distinction between high level and low level operations is important as the simulation builds timestamp information using basic actions (according to high level actions), while a forensic investigation starts from the timestamp information and (attempts to) derive high level actions or in general terms events based on it.

5.2 User Profiles

The sequence of actions is steered by a user profile, representing all characteristics that differentiate the model behind a system, or disk image, from another where the term refers both to user specific and system specific characteristics. The

[1] The Python based simulation environment we developed is available for download via www.science.uva.nl/˜worring/forensics-simulation

model refers to trends in the usage of a system e.g. a user that downloads frequently files from the Internet, as well as system specific details e.g. the download speed. One user model, as it is probabilistic, generates a different output disk image every time it is used. Users are characterized along the following dimensions:

Spatial user patterns: The level of semantic organization on the system based on the directory tree:

- **The organized user**: has a directory for most of the important concepts he is interested in and puts images in their proper directory when entered into the system.
- **The unorganized user**: utilizes one general directory to put everything in, or the directory structure of the images is random, with no direct relation to semantics.

Temporal user pattern: The extent to which semantics and time are correlated and secondly the regularity of use:

- **The single-task user**: works with images containing different contents at different moments in time without mixing them. Generally significant periods of time pass when switching between different concepts. Batch operations are applied to files belonging to the same semantic concept.
- **The multi-task user**: The multi-task user handles images with different semantic contents at the same moment in time, including during the same system operation.

The second categorization leads to:

- **The sporadic user**: The sporadic user initiates events in bursts, with significant time intervals in between.
- **The regular user**: The regular user manipulates images at relatively regular intervals in time e.g. every day or once every 6 days.

The Time Section

For user actions time is modeled as a series of Gaussian distributions to choose the days containing actions as well as the time during the day the actions are performed. System actions, are statically defined as a list of timesteps.

Timestamp variation for multiple files included in a batch operation (for each action) is modeled as well using a Gaussian distribution. The mean of the action is already known (the timestamp for the action chosen) and therefore only the variation specific to each action needs to be defined.

User action timestamps are generated in several steps: choosing the days containing actions, choosing the time of the day that (batches of) actions are performed at, and choosing individual timestamps for each of the files that will be part of the action - denoted as scrambling (needed for batch operations).

Actions are chosen according to a Markov model, independent of path or timestamp for the action. The states of the Markov Model are represented by the low-level actions (operations on individual files) corresponding to the high-level actions in the simulation. The propabilities generating various sequences are defined in a user profile which are characteristic for the specific types of users identified. An example Markov model and user profile is shown in figure 6.3.

The Path Section

This section is in charge of building the directory tree to be used for the file operations. Each file instance consists of a file name and path (on the simulated system) and the three timestamps (modified, accessed and created) and is responsible for updating them on the occurrence of a low-level action.

The decision of what directory and what files are chosen is done based on the action and, optionally, the semantics set for the action. Semantics are modeled as a Markov Model. Whether file semantics play a role in choosing the files depends on sthe user profile.

The tree is built starting from each partition (as the root node) and building the directory structure level by level. At each level we decide whether to build the level or stop according to the height probability. Building the level consists of choosing a number of directories again according to a Gaussian distribution.

Choosing files for the actions

The directory for the action is chosen according to a Gaussian distribution from the directories previously selected. The number of files that the action will be applied on is determined according to a Gaussian distribution specific to each action (in accordance with the number of files in the directory).

6. EXPERIMENTS

The dataset on which we perform experiments is the Visual Object Classes Challenge 2010 [2] with around 10.000 images. As a surrogate class for illegal material, we consider the horse category containing around 350 images as our target. As features, we use a method from van de Sande [28]. We compute a visual codebook, consisting of 64 elements based on a Harris-Laplace point detector and C-SIFT descriptors. For a more detailed explanation of the algorithms the reader is directed to the reference. As similarity function we use the Euclidean distance between the vectors. The images are clustered into a fixed number of clusters (15 in the following experiments) using a standard k-means algorithm. For each cluster a set of representative images is computed.

We consider two different use-cases to exemplify the usage of our visualisation system. Even though the organisational scheme of the user that has created this image collection is unknown in a real investigation and determined by examining results of different clustering schemes, we shall assume it is prior knowledge for the purpose of these experiments.

6.1 The Organised User

For our first experiment we consider the rather simple case of an organised user. This type of user neatly puts images of the same category into the same directory, an overview of the steps in the investigation are presented in figure 5.

6.2 Single-task User

Let us move to a more challenging scenario. In this case the user is rather disorganised - or at least the user's sense of organisation does not match the clustering, which is a very likely scenario in practice. More importantly, the user's organisational scheme doesn't categorise the target images

[2] `http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2010/`

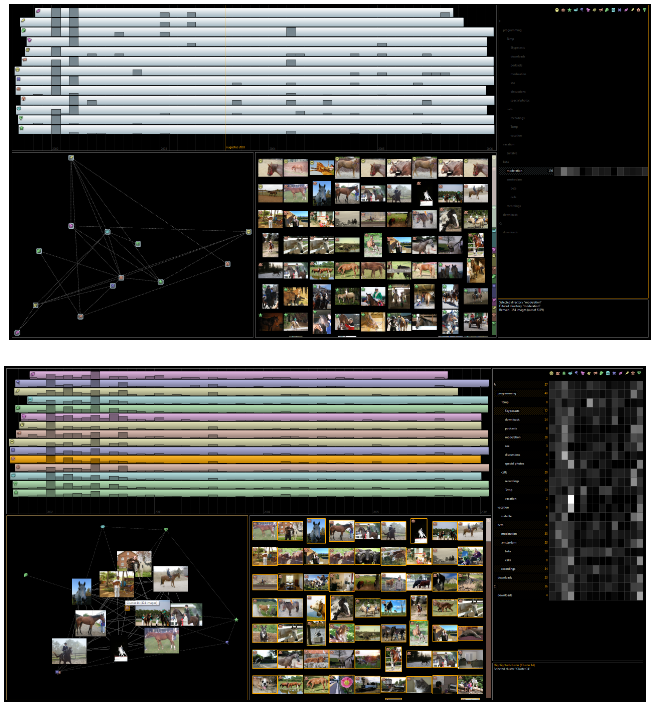


Figure 5: Screenshots from the investigation. (Top) As we are expecting an organized user we start by checking all directories one-by-one viewing one page of results in the detail view. When we hit the "moderation" directory we observe several target images. The heatmap reveals that "cluster 14" is represented most in this directory. Zooming into the cluster indeed reveals several target images among the representatives as well as in the detail view and also reveals several target images in various other directories.

as it was the case previously. Ideally k-means would put all target images into a single cluster, but practically that is a very unlikely outcome. However, unless the clustering fails completely, the resulting clusters will have varying densities of target images. We can now verify that the user is in fact disorganised in the sense that k-means doesn't match his organisational scheme. Hovering any cluster shows that the cluster images are seemingly randomly spread across all directories (see figure 6) so we should resort to time as a starting point.

An investigator can now proceed to find more images that might have been modified at a different time by successively repeating this strategy. At the end of each cycle, all images that have been seen so far (the image was visible in the detail visualisation) can be filtered out, supporting a systematic approach to an exhaustive search.

6.3 Exploration Strategies

As a final experiment we consider a more general strategy to use the visualisation system and compare its performance across several user types. Typically the challenge lies in finding a suitable entry point into the visualisation. The overview provided by the timeline, the graph and the hierarchy visualisation may give the investigator an idea about the type of user he is dealing with. We will propose a strategy that is independent of this, however, and can be used as a fallback.

First we need to find some initial images belonging to the target class. This is accomplished in three ways: either peeking into clusters in the graph visualisation, clicking through the clusters systematically in the timeline or graph and examining them in the detail visualisation, or checking in turn each directory in which this cluster is dominant. If none of these yield a single image of the target class, the preprocessing step (clustering) step needs to be revisited.

Once one or more target images have been found perform the following steps:

1. Roughly memorise the timestamp of the image. This is accomplished by hovering the image and reading the highlight label in the timeline visualisation.

2. Continue browsing the cluster/directory for more images using the detail visualisation.

3. If we find enough target images

   (a) filter using the timeline if the timestamps are similar

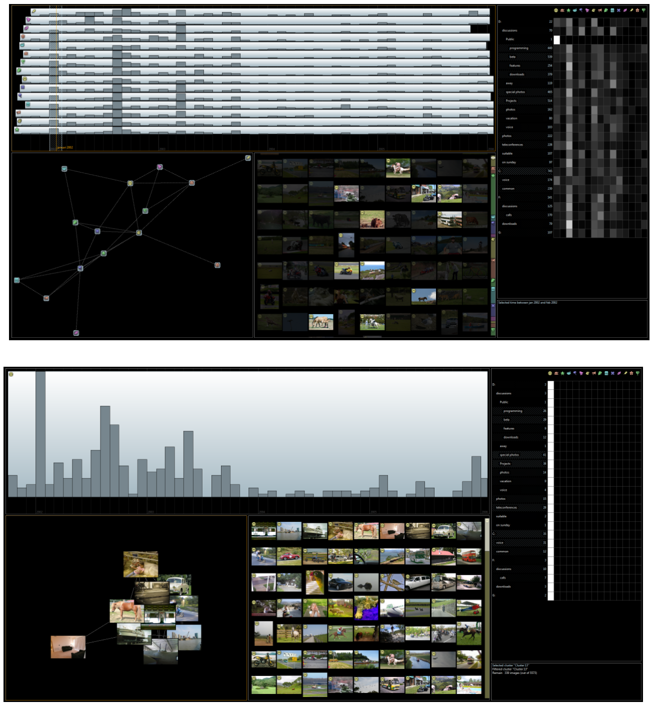
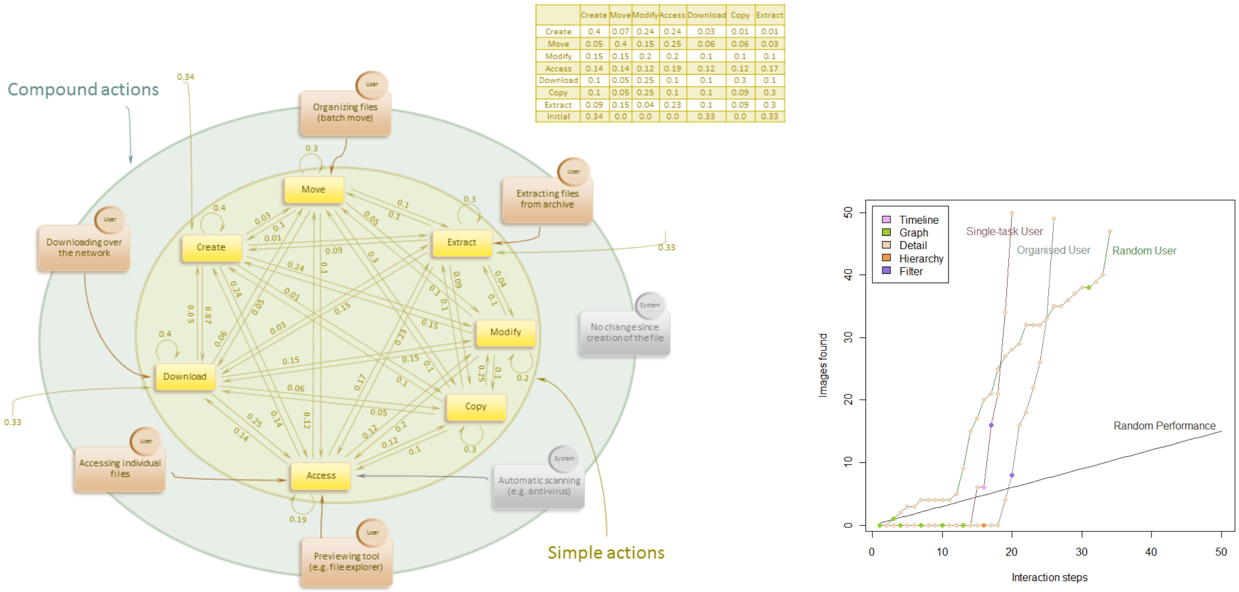


Figure 6: The second scenario. (Top) Inspecting images by looking at small intervals of the timeline reveals that that there are several target images with a timestamp around januari 2002. Looking at the cluster icons reveals that those are mostly coming from one cluster. Inspecting this cluster yields several additional target images at other moments in time (bottom). By focussing on one cluster the distribution of images over time becomes more apparent.



	Create	Move	Modify	Access	Download	Copy	Extract
Create	0.4	0.07	0.24	0.24	0.03	0.01	0.01
Move	0.05	0.4	0.15	0.25	0.06	0.06	0.03
Modify	0.15	0.15	0.2	0.2	0.1	0.1	0.1
Access	0.14	0.14	0.12	0.19	0.12	0.12	0.17
Download	0.1	0.05	0.25	0.1	0.1	0.3	0.1
Copy	0.1	0.05	0.25	0.1	0.1	0.09	0.3
Extract	0.09	0.15	0.04	0.23	0.1	0.09	0.3
Initial	0.34	0.0	0.0	0.0	0.33	0.0	0.33

Figure 7: Left: Markov model determining the sequence of actions on the system. The numbers on the arrows represent arbitrary probabilities of transitioning between two states. The three arrows starting in the exterior of the model represent the prior probabilities for the Markov model. Right: Performance of a custom exploration strategy on various instantiations of the model using the visualisation system against unaided random browsing (grey line). At each interaction step, a circle colour-codes the visualisation that was used for the specific interaction step.

(b) filter using the hierarchy visualisation if images fall in the same directory

(c) continue browsing the cluster otherwise

4. If the cluster/directory does not yield any more images within a reasonable amount of interaction steps, choose another and go to step 1 of this procedure.

Using this guideline we revisited the user types from the previous two scenarios, as well as a "random" user, who is following no particular scheme and is thus considerably more difficult. Time is used for comparing performance and is discretised into interaction steps. Examining a four-by-four grid of images counts as a single step, as does changing clusters, directories, making a selection or filtering. Peek-zooming in and out counts as a single step as well. Highlighting images is not considered for reasons of simplicity.

The experiments stop once at least 40 images belonging to the target class have been found. The results are summarised in figure 6.3.

Despite the lucky start for the random user where a good cluster was found almost instantly, initial performance of the visualisation system is typically worse since many interaction steps are used to find the clusters with a high density of target images. Once found, performance is better than random by a wide margin. Although the user profiles tested in this paper are fairly limited and more specialised real-world cases will certainly require adaptation, we are confident that the outlined exploration strategy provides a solid basis for using the visualisation system in other scenarios as well.

7. CONCLUSION

The emerging field of Multimedia Analytics where content analysis and visualization seamlessly merge into a framework to support the task of the user in an optimal way has high potential. Especially in the context of digital forensics where large collections of images need to be examined only a combined approach can truly support the investigative process. A characteristic of these complex tasks is that the content of the images is only one clue, whereas the metadata provides other interesting entries into the dataset.

In this paper, we presented a framework for browsing images and their metadata (in this case time and directory information) in which multimedia analysis is performed to derive content based clusters as basis for exploration. A timeline, graph, and directory visualization provide different ways to find patterns in the data. The highly interactive visualizations are visually connected through carefully chosen, easy to remember, icons and through the various interactions.

As multimedia analytics solutions are difficult to evaluate, we have developed a probabilistic tool to create disk images, in particular creating different timestamps and directory structures and content using Markov models to capture different user profiles. The tool is made available to other researchers so they can pursue different solutions to the same type of tasks. In this paper a limited set of simulated disk images and scenarios are illustrated. They show the possibilities of the developed framework. By looking at various simulated disk images and by considering the generic type of patterns they create in the various visualizations provides us insight in how to extend the framework into a solution where the system is proactively highlighting patterns of interest.

8. REFERENCES

[1] W. Alink, RAF Bhoedjang, PA Boncz, and AP De Vries. XIRAF-XML-based indexing and querying for digital forensics. *Digital Investigation*, 3:50–58, 2006.

[2] P. André, M.L. Wilson, A. Russell, D.A. Smith, and A. Owens. Continuum: designing timelines for hierarchies, relationships and scale. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, pages 101–110. ACM, 2007.

[3] G. Andrienko et.al. Space, time and visual analytics. *International Journal of Geographical Information Science*, 24(10):1577–1600, 2010.

[4] F. Buchholz and C. Falk. Design and implementation of Zeitline: a forensic timeline editor. In *Digital Forensics Research Workshop*. Citeseer, 2005.

[5] C. Chen, G. Gagaudakis, and P. Rosin. Similarity-based image browsing. In *Proceedings of the 16th IFIP World Computer Congress. International Conference on Intelligent Information Processing*. Citeseer, 2000.

[6] N.A. Chinchor, J.J. Thomas, P.C. Wong, M.G. Christel, and W. Ribarsky. Multimedia analysis + visual analytics = multimedia analytics. *Computer Graphics and Applications, IEEE*, 30(5):52 –60, sept.-oct. 2010.

[7] KP Chow, F.Y.W. Law, M.Y.K. Kwan, and P.K.Y. Lai. The Rules of Time on NTFS File System. In *Systematic Approaches to Digital Forensic Engineering, 2007. SADFE 2007. Second International Workshop on*, pages 71–85. IEEE, 2007.

[8] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, 40(2):1–60, 2008.

[9] O. de Rooij, M. Worring, and J. J. van Wijk. Mediatable: Interactive categorization of multimedia collections. *IEEE Computer Graphics and Applications*, 30(5).

[10] D.M. Eler, M.Y. Nakazaki, F.V. Paulovich, D.P. Santos, G.F. Andery, M.C.F. Oliveira, J. Batista Neto, and R. Minghim. Visual analysis of image collections. *The Visual Computer*, 25(10):923–937, 2009.

[11] A. Girgensohn, F. Shipman, L. Wilcox, T. Turner, and M. Cooper. MediaGLOW: organizing photos in a graph-based workspace. In *Proceedings of the 13th international conference on Intelligent user interfaces*, pages 419–424. ACM, 2009.

[12] J. Heer and D. Boyd. Vizster: Visualizing Online Social Networks. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, page 5. IEEE Computer Society, 2005.

[13] H.C. Jetter, W.A. König, J. Gerken, and H. Reiterer. ZOIL-a cross-platform user interface paradigm for personal information management. In *In" Personal Information Management: PIM 2008", CHI 2008 Workshop*, 2008.

[14] V. Kaptelinin and M. Czerwinski. *Beyond the Desktop Metaphor: Designing Integrated Digital Work Environments*. MIT Press, Cambridge, Mass., 2007.

[15] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann, editors. *Mastering the Information Age Solving Problems with Visual Analytics*, chapter Data Mining, pages 39–56. Eurographics Association, 2010.

[16] D. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler. Challenges in visual data analysis. In *Information Visualization, 2006. IV 2006. Tenth International Conference on*, pages 9–16. IEEE, 2006.

[17] A. Krishnan and S. Jones. TimeSpace: activity-based temporal visualisation of personal information spaces. *Personal and Ubiquitous Computing*, 9(1):46–65, 2005.

[18] H. Lejsek et.al. Videntifier forensic: large-scale video identification in practice. In *Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence*, MiFor '10, pages 1–6, 2010.

[19] GP Nguyen and M. Worring. Interactive access to large image collections using similarity-based visualization. *Journal of Visual Languages & Computing*, 19(2):203–224, 2008.

[20] C. North. Toward measuring visualization insight. *Computer Graphics and Applications, IEEE*, 26(3):6 –9, may-june 2006.

[21] J. Olsson and M. Boldt. Computer forensic timeline visualization tool. *Digital Investigation*, 6:78–87, 2009.

[22] N. Quadrianto, K. Kersting, T. Tuytelaars, and W.L. Buntine. Beyond 2d-grids: a dependence maximization view on image browsing. In *ACM International Conference on Multimedia Information Retrieval*, 2010.

[23] J. Rekimoto. TimeScape: a time machine for the desktop environment. In *CHI'99 extended abstracts on Human factors in computing systems*, pages 180–181. ACM, 1999.

[24] G.G. Richard III and V. Roussev. Next-generation digital forensics. *Communications of the ACM*, 49(2):76–80, 2006.

[25] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–, Washington, DC, USA, 1996. IEEE Computer Society.

[26] A.F. Smeaton, P. Over P, and W. Kraaij. Evaluation campaigns and TRECvid. In *Proceedings of Multimedia Information Retrieval*, 2006.

[27] J.J. Thomas, K.A. Cook, Institute of Electrical, and Electronics Engineers. *Illuminating the path: The research and development agenda for visual analytics.* IEEE Computer Society, 2005.

[28] K.E.A. Van De Sande, T. Gevers, and C.G.M. Snoek. Evaluating color descriptors for object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1582–1596, 2010.

[29] J.J. Van Wijk and E.R. Van Selow. Cluster and calendar based visualization of time series data. In *Information Visualization, 1999.(Info Vis' 99) Proceedings. 1999 IEEE Symposium on*, pages 4–9. IEEE, 1999.

[30] C. Ware. *Visual thinking for design.* Morgan Kaufmann Pub, 2008.

[31] X-Y. Wei X-Y. and Z-Q. Yang. Coached active learning for interactive video search. In *Proceedings of ACM Multimedia*, 2011.