Comparing personal image collections with PICTuReVis

Citation for published version (APA):

van der Corput, P. N. A., & van Wijk, J. J. (2017). Comparing personal image collections with PICTuReVis. *Computer Graphics Forum*, 36(3), 295-304. <https://doi.org/10.1111/cgf.13188>

DOI:

[10.1111/cgf.13188](https://doi.org/10.1111/cgf.13188)

Document status and date:

Published: 01/06/2017

Document Version:

Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Comparing Personal Image Collections with PICTuReVis

Paul van der Corput¹ and Jarke J. van Wijk¹

¹Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands

Abstract

Digital image collections contain a wealth of information, which for instance can be used to trace illegal activities and investigate criminal networks. We present a method that enables analysts to reveal relations among people, based on the patterns in their collections. Similar temporal and spatial patterns can be found using a parameterized algorithm, visualization is used to choose the right parameters and to inspect the patterns found. The visualization shows relations between image properties: the person it belongs to, the concepts in the image, its time stamp and location. We demonstrate the method with image collections of 10,000 people containing 460,000 images in total.

Categories and Subject Descriptors (according to ACM CCS): H5.2 [Information Systems]: User Interfaces—Graphical user interfaces H5.2 [Information Systems]: User Interfaces—Interaction styles

1. Introduction

The popularity of small electronic devices like smart phones has stimulated people to capture their lives and their surrounding world in an unimaginably detailed way. Part of this data consists of digital images, which are taken, stored, and shared on a massive scale in the last decade. This data contains a wealth of information that can be used to learn about individuals, society, and the world. The extraction of this information can be challenging. Each image has associated metadata about who made it, where it was made, when it was made, what concepts are in the image, and possibly also tags. Still it is hard to pose database queries like: ‘what is the probability that two people know each other?’

In this paper we focus on finding relations between image collections to reveal relations between the people who own these collections. In contrast to social media, where people build their network explicitly and send messages to each other, relations between their image collections are only implicit. Imagine the case where a detective wants to investigate a criminal network, but only has confiscated data from individual mobile phones. It would make the investigation easier if relations between the collections can be found and shown effectively and efficiently. Our main goal is (1) to find people with similar patterns in this data and (2) if found, explain why these patterns are similar. The ability to reconstruct the whole network would be a bonus, but is outside the scope of this paper.

Previous work either visualizes a generic collection (composed of images from various owners) or a personal collection. In neither of these cases, the characteristics of the original owners are compared. To this end, we developed a visual analytics method that enables users to explore the characteristics of multiple personal image collections simultaneously, and to spot matching patterns between

them. This is done by showing the **people, images, concepts, time** and their **relations**; hence the name *PICTuReVis*.

The paper is structured as follows. We discuss related image collection visualizations and image classification methods in Section 2. We give a specific problem definition in Section 3, and present our approach in Section 4. Details about the implementation are described in Section 5. Our approach is demonstrated in Section 6 and we discuss its limitations and future directions in Section 7. Finally, we give our conclusions in Section 8.

2. Related work

The comparison of image collections involves a number of steps: derivation of visual descriptors, visualization of temporal and spatial patterns, visualization of their similarity, and the presentation of images and their metadata itself. In this section we discuss work related to these steps, and how we built upon that.

2.1. Computer vision and organizing images

Organizing images is important for the visualization of large image collections. A common approach is to first derive a set of features with a computer vision algorithm, and then use these features for similarity based layout techniques. For example, PEX-Image [ENP*09] considers images as points in a high-dimensional space and projects them on a 2D space using a Neighbor-Joining similarity tree. The Honeycomb Image Browser [PS10] displays images on hexagonal tiles, placing similar tiles close to each other. A tile can be expanded to show that subset in more detail. Image-Hive [TSLX12] summarizes collections using a constrained graph layout algorithm and a Voronoi scheme for local refinement to use

the space optimally. iMap [WRZ*15] uses a treemap-based layout to cluster and search images. Clustering is done using a weighted sum of partial visual and textual distances.

A recent breakthrough in the field of computer vision is the use of large, deep convolutional neural networks [KSH12]. Networks that are trained on the ImageNet [DDS*09] dataset enable classification with thousands of concepts that describe the image. Deep learning frameworks like Caffe [JSD*14] and Overfeat [SEZ*14] make it easier to use existing nets or train new ones and use these to classify images efficiently. We use this deep learning technology as a pre-processing step to derive 13,000 concepts as described in Section 3.2. For simplicity reasons and since our focus is more on the concept distribution, we use a sortable thumbnail grid so that users can quickly find images that match the selected concept.

2.2. Event visualization

Temporal data is ubiquitous and its visualization is studied for centuries [AMST11]. Section 3.1 explains why the temporal aspect is important for analyzing image collections. Images can be seen as events and can be placed on a time line, like the Time Line Browser [CK91] and LifeLines [PMR*96] show medical events and intervals on a time axis. Dörk et al. [DGWC10] visualize how topics in digital backchannels, which can contain images, evolve over time using a stream graph. EventAction [DPSS16] presents student events as boxes on a discrete time line for the purpose of comparison with students from earlier generations.

Visualizations such as LifeLines and EventAction show events of the same type as glyphs in the same row. This does not scale well to the number of image types we have, i.e., the 13,000 concepts. We visualize all events, i.e., images, of one person in a single row and only highlight images with selected concept(s) to enable the comparison of several people in one view. We use similarity and relevance scores [vdCvW16a] to make it easier to find interesting concepts.

2.3. Similarity visualization

The City Melange [ZRW15] is an interactive venue explorer that learns an SVM model based on the user's preferences and suggests similar people. After each iteration, example pictures of suggested people are projected on a map to refine the result. EventAction [DPSS16] uses the Euclidian distance to determine the best match between student events. Similan [WS09] uses a more advanced match & mismatch score that is more computationally expensive, but can deal with missing and extra events.

We are looking for similar people based on their images, which is similar to what City Melange aims at, except that City Melange does not take time into account. Like EventAction and Similan we use a distance measure to find and visualize the best match for a selected person. An efficient measure is required, because the number of data points can be large. Keogh and Kasetty [KK03] showed that the simple Euclidian distance can outperform more complex measures on generic time series, but we found that it is not a suitable comparison for irregular series such as image collections. In Section 5.1 we present an alternative that is more robust to noise.

2.4. Image metadata visualization

To inspect if and how personal image collections are related, we must look at the images themselves and their metadata. This metadata can be considered as multivariate data. The Semantic Image Browser [YFH*06] makes a two-dimensional MDS plot of images based on concept similarities, so similar images are placed close to each other. Matković et al. [MGF*09] use multiple coordinated views and brushing to show relations between attributes along with the corresponding images. MediaTable [dRWvW10] visualizes all image properties in one big table. Multimedia Pivot Tables [WKZ16] split the collection into rows and use columns to show the distributions of the subsets over the other dimensions, such as textual description, location, time, and concept. Microsoft PivotViewer [Mic] organizes images in a zoomable histogram. ICLIC [vdCvW16b] also uses a histogram or matrix as the main visualization, and additionally shows the characteristics of the user's selection in a separate attribute view.

The VaR display [YFH*06] makes it easy to find correlated concepts in two-dimensional space. Due to the high number of concepts, we present these in a sortable list so the user can quickly find the most interesting concepts. Pivot Tables and ICLIC are very flexible and so in theory should be able to show the distribution of images over time per person. However, only the distribution is not enough for comparisons: the type and location of the images need to match simultaneously. Section 5.2 presents a specific glyph design to highlight such overlapping patterns between users. Furthermore, the above mentioned visualizations are tailored towards finding similar images, but not to comparing people.

3. Problem analysis

Our main goal is to find a method that assists analysts, e.g., forensic experts and detectives, in tracing criminal activities. Possible evidence partly consists of digital images from (confiscated) smart phones, surveillance cameras, the internet, etc. When an incident needs to be investigated, evidence from such sources is collected to get a clear view on the events prior to the incident. The people involved and their relations are of special interest. With millions of images from hundreds of sources, this is challenging. Below are the characteristics of the data and the requirements for a method that should assist the analysts.

3.1. Image collection characteristics

There are major differences between images of one person and a collection that consists of images from various people. One of the strongest factors is the distribution of images over time. When many collections are aggregated, one may be able to observe day/night, weekend, and seasonal effects in the aggregated data. The pattern of a single person is typically irregular because images are made in bursts. If the time between two consecutive images is small, there is also a high probability that they share concepts and are made at the same location. These temporal properties are useful for the comparison of people: if (parts of) their patterns are the same, there is a chance that there is a relation between them.

Digital images often carry some metadata in addition to the raw

Figure 1: Person Image Concept Time Relation (PICTuRe) Diagram.

pixels. Two properties are usually available: the person (this can be the owner or photographer, but can also be characterized by the device, ip-address, or source in general), and a time stamp (for instance the time when the image was created, stored, or uploaded). We can also use computer vision algorithms to determine which concepts are in the image, i.e., the things that are visible in the image. Section 3.2 gives an overview of the possibilities with computer vision. Images also provide other information like location and EXIF data with camera settings, etc., but these may be altered, stripped of on purpose, or not be recorded at all. In this paper we focus mostly on the properties person, concept, and time. Figure 1 shows how these image properties are related and also indicates possible queries:

- who is active at time t ?
- when is person p active?
- who is interested in concept c ?
- what is person p interested in?
- when is concept c popular?
- what is popular at time t ?
- who is interested in concept c at time t ?
- when is person p interested in concept c ?
- what is person p interested in at time t ?

All these queries have a similar pattern: the analyst selects a combination of people, concepts, time intervals, and regions; and wants to see the effect on the other aspects. They form the building blocks for more complex queries like finding people with shared interests: one can first query for the interests of one person, select these, and then query for people who have similar interests.

3.2. Concepts and features

Computer vision algorithms can be used for automated annotation. The current state-of-the-art in computer vision is the use of large, deep convolutional neural networks. We distinguish two levels of annotation: concepts and features. Concepts are high-level descriptors of an image that can easily be described by one or a few words, for instance: ‘car’, ‘tree’, and ‘indoor’. Features are low-level descriptors that represent subtle patterns like color, shape, and texture. Although features are a powerful tool for a computer to characterize and compare images, they are difficult to describe. Hence, the concepts are well-suited as feedback for visualization and explicit queries, and the features are well-suited for automated analysis and

queries by example. The term concept in Figure 1 hence denotes only the high-level descriptors, not the low-level ones.

Concepts and features can be derived from the images by the deep learning framework Caffe [JSD*14]. It has a pre-trained neural network and an image as input, and outputs vectors with concept and feature weights. The concept weights (also called concept scores) are typically the output of the network’s final layer and indicate how strong each concept appears in the image. Features are typically the states of an intermediate layer. We use the network trained by Mettes et al. [MKS16] to find 13,000 ImageNet [DDS*09] concepts. We also use the 1024 features from the average pool $7 \times 7/1$ layer from the network trained by Szegedy et al. [SLJ*15].

3.3. Requirements

Based on interviews with forensic experts about their use cases, we came to the following requirements for our new method. Given a reference person, the two main objectives are to enable analysts

- R1** to find people with similar image patterns, and
- R2** if such similarities are found, to understand these.

This may look simple, but with a large number of people, images, concepts, and a wide timespan, the analysis can become slow and cumbersome. We aim to do R1 and R2

- R3** within reasonable time for a realistic data set. By this we mean that first results should be visible within a few minutes for $\pm 1,000$ collections with in total $\pm 1,000,000$ images.

Some flexibility is required to support the main objectives, of which we see the following as crucial: enable analysts

- R4** to answer the queries implied by the diagram in Figure 1, and
- R5** to define custom concepts so they can find use case specific objects that are not pre-trained.

4. Workflow

The problem with (automatically) searching for people with similar image collections is that the definition of similarity is not fixed. It will depend on the use case, analyst’s expertise, data, etc. Given the dimensionality of the data, i.e., many images per person and many concepts per image, it is difficult or even impossible to determine similarity scores that everyone agrees upon. Hence, analysts have to be enabled to define their interests iteratively, guided by the patterns found. A summary of a base use case is schematically depicted in Figure 2 and has the following steps: (1) the analyst selects a reference person or a group of reference people; (2) the analyst views the characteristics of this person to see which concept(s), location(s), and time period(s) are useful to take into account for comparisons; (3) the analyst selects images according to the findings in the second step; (4) the system computes the similarity between the selected person and other people using only the selected images in the third step; (5) the analyst inspects people with the highest similarity scores and investigates their relation with the reference. Findings in the last step can be a reason to select another person and replay the workflow. These steps form one base use case, but analysts should not be restricted to a specific order.

Figure 2: Workflow diagram of a base use case. Big arrows show the overall direction, smaller arrows indicate correction steps.

A typical pattern would be to alternate Steps 2 and 3 several times before proceeding to Step 4, or, to start at Step 2 to find interesting people for Step 1.

We have built a system to support this workflow. Its main interface is shown in Figure 4 and the interactions are demonstrated in the accompanying video. The five steps are detailed out below together with pointers to the corresponding parts of the system. Details on the implementation can be found in Section 5.

Step 1: Select person. The analyst selects the person or people of interest in the person view, see Figure 4A. The base use case assumes that the user knows the people of interest beforehand. Alternatively, the user can start from Step 2 and explore the characteristics of the whole collection to find interesting people for comparisons.

Step 2: View characteristics. The selection of a person implies the selection of images belonging to that person, and the same holds for the selection of concepts, time intervals, and geographic regions. Inspection of the characteristics of these subsets helps users to define the relevant aspects for the person similarity score by selecting concepts, time intervals, and locations of interest. The relation diagram in Figure 1 can be used as a guide for this. The person view, time axis, and concept view in Figure 4A, B, and D respectively have a histogram showing the distribution of all images with on top a histogram of the selection. These can be used to answer the queries in Section 3.1.

Additionally, we adopted the idea of similarity and relevance scores used in I^F , F^I -Tables [vdCvW16a]. Whenever a person is selected, the system computes how similar each person is to that selection and how relevant each concept is for this person's images. Vice versa, the system computes how similar each concept is to the selected concept and how relevant each person is given that concept. This enables the analyst to quickly find interesting people and concepts to investigate next. Details on the computations are given in Sections 5.1 and 5.4.

Step 3: Select images. Images are selected by selecting their properties in the views using a faceted search approach: specific values for people, concepts, time intervals, and spatial areas can be selected. The global selection consists of images that have properties within the selected ranges. Similar to the Attribute

Figure 3: Schematic overview of the selection model: green areas show the selected image properties; gray bars and dots show the distribution of all images; yellow bars and dots show the distribution of only the selections made in the other views; black dots in the time-person chart show images with selected locations and concepts.

Explorer [TSWB94], the distribution of the global selection is projected on top of the distribution of all images using histograms. This enables analysts to see the characteristics of the selection and how that differs from the average. One problem with this model is that the context is lost when a selection is made in a view. If for instance a person is selected, the distribution of images in the selected time interval is not visible in the person view anymore. We solved this by projecting only the selections that are made in the other views. The selection model is depicted in Figure 3.

In case multiple items are selected in the same view, the system selects the union of images assigned to the items. By selecting two people, for instance, all images made by either of them are selected. Concept scores are real values, so finding images by concept requires the use of a threshold. Figure 4C shows a button to adjust this threshold to find an optimal balance between precision and recall. The image view in Figure 4F shows the selected images and enables the user to construct custom concepts by selecting positive and negative examples as detailed out in Section 5.5.

Step 4: Find similar people. At this point, a reference person has been selected in Step 1 and a subset of images has been found in Step 3. These are inputs for an algorithm, described in Section 5.1, that matches the temporal and spatial patterns of people with the reference person given the selected subset of images. The results are displayed in the similarity column of Figure 4A, and can be sorted to find the best matches.

Step 5: Inspect suggestions. The time-person chart, see Figure 4B, shows with glyphs when people made images and so enables the user to compare the temporal pattern of the reference person with the suggested people in Step 4. The glyphs are an abstract representation of the collection and are used to guide the user to the matching patterns. More details about the glyph design can be found in Section 5.2. Images that are represented by a glyph are shown in the image view after selecting the glyph. This enables the user to inspect whether this is a true match. This manual verification is necessary because matching concepts do not imply matching events and concept predictions are not always accurate.

5. PICTuReVis' detailed design

The system is composed of multiple connected views, which are detailed out below. The main design decision was to use a min-

Figure 4: Screenshot of the PICTuReVis system, which highlights a group of people that took pictures of ships at the Sail 2015 event in Amsterdam. (A) The person view shows a list of people, which can be sorted by name, count, similarity, and relevance. (B) The time-person chart shows the distribution of people's images over time. Glyphs indicate the selection and density of images, and highlight matching temporal and spatial patterns. (C) Controls for the concept threshold and a toggle between the time-person and map view. (D) The concept view shows a list of concepts and can be sorted the same way as the person view. (E) Controls for adjusting the time window w and distance threshold d . (F) The image view shows all images with the selected properties in the other views. (G) Controls that enable users to create custom concepts based on positive and negative example images. (H) A status bar gives information about the object under the cursor.

imalist interface to emphasize the base use case. We could have included elements to explore other metadata like EXIF data, but we focused first on support for the basic workflow.

5.1. The person view

The person view, depicted in Figure 4A, is a table that shows the names (or IDs) of the people, the number of images in their collection, the similarity score, and the relevance score. The tabular visualization makes it easy to sort by these statistics and hence to find the patterns as in Requirement R1. It also aligns with the visualization of temporal patterns (R2), discussed later. Image counts are shown by a histogram inspired by bullet graphs [Few13]. The gray bar in the background shows the total number of images, whereas the yellow bar in the foreground shows the number of selected images. The number of selected images is typically much less than the total, so the bars are normalized independently to increase the readability of the yellow bars. Interesting people can be found using the similarity and relevance columns, which are detailed out below.

Finding similar people. A core part of the workflow is searching for similar people given a selected person. This requires a similarity

measure that captures matching temporal (and spatial) patterns of the selected images (selected concept, time, location, etc.) between two people. Advanced measures are computationally too expensive to keep the system interactive for such large data. We therefore experimented with several simple measures like the Euclidian distance, where the time is subdivided into small time intervals and the distance is the sum of the squared differences in image count per interval. That approach appeared to be sensitive to noise in the data. We therefore defined a more robust measure, depicted in Figure 5. The key idea is simply to count the number of intervals where people took similar pictures at the same location.

Comparing the image densities over time appeared to be undesirable. A non-frequent photographer will never have a good match with a frequent photographer, although they may meet each other regularly. The frequent photographer can introduce noise by taking photos between their common events. To solve this, we use a time window w and a distance threshold d . Timestamps that are closer than $w/2$ are considered simultaneous, and locations that are closer than d are considered co-located. We only take into account the images that are selected, simultaneous, and (optionally) co-located.

Figure 5: Comparison between two people. Circles indicate non-selected images, dots indicate selected images, and lines between images indicate proximity. Only selected images that match within the time window and (an optional) distance threshold are taken into account. The number of matched intervals is the measure for similarity between the two people.

A second type of noise may occur if two people p_1 and p_2 have images of the same event, but one has many images, and the other only few. The question is if the quantity at the same event should be relevant for their similarity. Photos are often taken in bursts so this can have a serious impact on the score, also when the photos within a burst are almost identical. Hence we think that once an image of p_1 matches an image of p_2 , it should not matter anymore for their similarity if p_1 takes many more photos. This is achieved by sampling the images with time intervals of length w . The interval is matched if and only if it contains a matched image. The final similarity score is the count of matched intervals.

Finding relevant people. We consider people relevant if they made images in the selected time interval and location that score high on the selected concepts. The relevance of a person p is the maximum output score of the convolutional neural network for the selected concepts and selected images that belong to p .

5.2. The time-person chart

The main purpose of the time-person chart, depicted in Figure 4B, is to view the distribution of personal image collections over time and to spot relations between people. On the left side of the chart is the list of people and at the bottom is a time axis. The chart itself shows glyphs at the intersections between people and discrete intervals on the time axis indicating when a person made images.

The time axis. The aggregated distribution of images over time is shown similar to the person view. The distribution of selected images is projected on top of the rest to spot deviations from the global pattern. The time axis is divided into small intervals of which the duration depends on the zoom level so that the width in pixels remains constant. The same intervals are used for the glyphs. Although sorting items in the person, concept, and image view is powerful; sorting the time axis would be counterintuitive given the

(a) Preferred configuration. The disk size shows the density. Disks are transparent if all images are below the concept threshold. Disks are highlighted if the selected person has images with this concept simultaneously. Plus signs indicate co-located events.

(b) Alternative configuration with more detail. The height shows the density, and the intensity encodes the concept score distribution within the bars. Concepts and locations matching the selected person are marked with an underline and triangle respectively.

Figure 6: Possible configurations of visual encodings. (1) This person has the concept, but the selection does not. (2) Both people have the concept and were at the same location. (3) Both have the concept, but had a different location. (4) Both were at the same location, but only the selection has the concept. (5) This person has the concept, but the selection has no images in this interval.

natural ordering of temporal data. Sorting time intervals by image count is hence disabled.

Glyph design. The basic properties that the glyphs should encode are the number of images represented by the glyph and whether these images contain a selected concept. We experimented with several encodings, see Figure 6. We noticed that increasing the level of detail in the glyphs quickly reduces the readability. Showing the distribution of concept scores as in Figure 6b may seem useful, but users are typically only interested in the images that score above a certain threshold. We therefore have chosen for the least visually complex one as the final configuration, which is shown in Figure 6a. It is a disk of which the size encodes the density: the bigger, the more images in the interval. A logarithmic scale is used to prevent oversized glyphs. Glyphs are transparent when none of the images score above an adjustable concept threshold, this still gives users context information but puts the focus on the images of interest. Although the glyphs enable a step-by-step comparison between people, this appeared to be cumbersome for rows that are far apart. We solve this by highlighting the glyphs that match the selected person, i.e., all images that contribute positively to the similarity score. Images that were made within an adjustable geographic distance d of the selected person are indicated by plus signs.

(a) Images at July 5th 2015, (b) All locations of one Flickr user in the year 2015.

Figure 7: Screenshots of the map view. The orange dots are selected images, the background is a density plot of the other images.

5.3. The map view

Users can toggle between the time-person chart and a map view with the buttons in Figure 4C. This map view shows a density plot of all images in the background, and highlights the selected images as dots. Spatial patterns can be revealed as shown by the screenshots in Figure 7. By highlighting the images containing the concept ‘cycling’ at July 5th 2015, for example, we see the route of a Tour de France stage. Vice versa, by selecting regions in the map view, temporal patterns can be revealed. The user is enabled to specify a point and a radius for making spatial selections.

5.4. The concept view

Concepts are displayed in a table, see Figure 4D, which has the same columns as the person view. The counts column shows the distribution of all images over the concepts with gray bars in the background. The yellow bars in the foreground show the distribution of selected images.

Finding similar concepts. Concepts are considered similar if their scores are correlated for the images in the collection. This property can be found by calculating Pearson’s correlation coefficient, which is a measure for linear dependence between two variables. The linear dependence between two concepts c and d is

$$\frac{\sum_{i \in I} (c(i) - \mu_c)(d(i) - \mu_d)}{\sqrt{\sum_{i \in I} (c(i) - \mu_c)^2} \sqrt{\sum_{i \in I} (d(i) - \mu_d)^2}}, \quad (1)$$

where I is the whole image collection, and $c(i)$ is the concept score of c for an image $i \in I$. Small subsets of images lead to low quality similarity scores, so we decided to keep the concept similarity independent of the selected images, unlike the person similarity.

Finding relevant concepts. Concepts are relevant if they can be used to characterize a selection of images. A commonly used measure for this is the Information Gain [Qui86]. It is an indicator for

a concept’s ability to distinguish the selected images from the rest. The relevance or gain g of a concept c , given the selected subset I' of the collection I is given by

$$g(c) = H(a, b) - \sum_{j \in \{1, 2\}} \frac{a_j + b_j}{a + b} H(a_j, b_j), \text{ where} \quad (2)$$

$$H(a, b) = -\frac{a}{a+b} \log_2 \frac{a}{a+b} - \frac{b}{a+b} \log_2 \frac{b}{a+b}, \quad (3)$$

$a = |I'|$ (number of selected images), $b = |I - I'|$ (number of non-selected images), and subscripts 1, 2 indicate whether these contain concept c or not (i.e., score above the threshold set in Figure 4C).

5.5. The image view and custom concepts

The image view in the bottom of the interface, see Figure 4F, serves two purposes: (1) to show selected images, and (2) to build new concepts using the images as examples. The content of the image view is automatically updated when a selection in one of the other views is changed. There are some convenient controls for adjusting the size, change the order (by time, or selected concept), and an attribute can be selected to project metadata on the images, like the timestamp, title, owner, etc.

In the top of the image view (Figure 4G) are controls to add custom concepts. The user is enabled to give the new concept a name and assign images as positive or negative examples, which is done using a left-click and right-click on images in the image view respectively. Training a new neural network would require many examples and too much processing time to be interactive, so we apply a technique called *transfer learning*. Inspired by Razavian et al. [RASC14], we use the low-level features as described in Section 3.2 and the examples provided by the users to train a linear support vector machine (SVM) [CL11]. The trained SVM is a weighted sum of the support vectors found by the algorithm, i.e., a weight vector that has the same dimension as the feature vectors. Now the new concept score of an image is the dot product of its feature vector with the trained weight vector. This score is normalized so that it falls in the same range as the pre-computed concepts. Custom concepts are visible in the concept view and can be used as normal concepts. Each time the user adds a new example, the scores are automatically updated. The similarity column can be used to find (pre-defined) concepts that are correlated with the custom concept.

5.6. Complexity and scalability

Each image is accompanied by a 13,000 dimensional concept vector and a feature vector of 1,024 dimensions. A collection of 5×10^5 images like in the use case of Section 6 would hence contain 7×10^9 floating point values. We use low resolution images in JPEG format that are on average 100 kB, which adds up to 50 GB of image data. Normal computers are clearly not capable of naively processing this amount fast enough to support interactive visualization. Here are some techniques we used to solve this problem.

Figure 8: Steps of an example use case. (A) One person is selected. (B) The characteristics of this person are inspected. The time axis shows a typical irregular pattern. The map shows that images were mostly made in the same part of the country. The top relevant concepts are related to cycling. (C) Inspection of the best matching people on the concepts 'cycling' and 'costume', the top relevant concept not related to cycling. By selecting different concepts we find different matching people. The glyphs and the corresponding images indicate a cluster of people who regularly visit fantasy events.

Derivation of image descriptors does not need to be interactive, and is therefore done in a pre-processing step that takes about a day. An LRU cache is used to display images quickly and to reduce disk access. Concept vectors mainly consist of values close to zero. We compress these vectors by only keeping the (for instance 100) highest values and their indices in memory. Besides saving memory, this also reduces computational effort. Correlations between one concept and all others can be computed by a single pass through the compressed data, which takes about 0.4 seconds on a normal laptop. The time needed for computing person similarity scores depends on the distribution of images over time. If two people have bursts in the same time window, this requires a many-to-many comparison for finding a pair of images that were made near each other. More advanced data structures may be required to handle this worst-case scenario. The average time to compute all similarity scores is typically within a second.

Next, the visual complexity plays also a role here. Our experience with a test data set of 460,000 images reveals that after selecting a concept and an appropriate threshold, the time-person chart is sparse (not so many bright glyphs). The most interesting people are usually in the top 10 after sorting the person view by similarity. But, dependent on the image set, this could be different in other cases, for instance when all images are highly similar.

6. Evaluation and use case demonstration

PICTuReVis has been evaluated at the Netherlands Forensic Institute and at a Dutch police department to test how the method can assist their analysts. The evaluation was performed by five analysts. They were provided with a ± 30 minute introduction to the system and a user manual before they used the system to explore some of their data sets. Since this data is highly confidential, we were not allowed to watch the analysts in action. Instead, they sent their feedback to us after the evaluation.

Analysts in this field often use forensic systems like Hansken [vBvEvB*15] and VizX2 [Ziu] or custom scripts for categorizing and searching within large image collections. Participants of the evaluation reported the following advantages with respect to existing systems: there is a time axis that clearly shows peak moments; it is easy to create new concepts, even with only a few examples; it shows relations between photos, concept, and time; and no programming experience is required. The system is furthermore useful for finding subsets of the data for detailed inspection. The glyphs are easy to interpret and it is easy to select photos from a specific area. On the contrary, all features are in the same window, so it looks complex at first sight. The most difficult part is to interpret the similarity and relevance columns. The time between the earliest photo and latest photo can be decades, which can make the time axis difficult to operate. All analysts who used PICTuReVis

agreed that it led to new insights in the data and that they would recommend it to their colleagues.

Since it is not possible to publish the use cases of the expert evaluations, we demonstrate the method on a publicly available subset of Flickr [Yah16] with images that were taken in The Netherlands in the year 2015. This subset contains more than 460,000 images made by over 9,600 different people. Their names are anonymized for privacy reasons. Figure 8 shows an example use case that follows the steps of the workflow diagram in Figure 2. Below are descriptions of two variations with a different starting point, which are also demonstrated in the accompanying video.

Begin with selecting a location. The analyst is interested in the characteristics of the city Utrecht. After selecting this city in the map view, the timeline shows an image burst that indicates an important event in the first weekend of July. The type of event can be found by sorting the concept view by relevance. Images made at the selected location and time interval are best described by the concept ‘cycling’. That weekend was the Grand Depart of the Tour de France in Utrecht. The stage of July 5th is visible in the map view, see Figure 7a. People who visited this event can be found by sorting the person view by frequency, i.e., the number of images people made with the concept ‘cycling’ and have the selected place and time. The analyst can proceed by comparing these people like in Figure 8.

Begin with selecting a concept. The analyst is looking for train spotters. This is achieved by looking for related concepts, the concept ‘train’ is available for instance. The similarity column in the concept view shows that there are also more specific types of trains available. Train spotters are the top relevant people given the selected concept. The analyst uses the glyphs of the time-person chart and the image view and sees that some train spotters come together regularly to take photos. They are photographing the same trains at the same location. One of these trains appears to be popular for a larger group. People who have seen this train can be found in two steps. First a custom concept is created based on positive and negative example images. After that, the custom concept is selected and the top relevant people all have this train in their collection.

7. Discussion

7.1. Requirements

We reflect on the requirements given in Section 3.3. The similarity measure for people gave convincing suggestions in our experiments (R1), in the sense that suggestions share patterns (in time, concept, and location) with the person of interest. The glyphs in the time-person chart highlight the matched images, which can be inspected and compared in the image view (R2). The computations have a low algorithmic complexity, so they can be done interactively on a modern laptop. We used a laptop with a 2.7 GHz CPU and 8 GB memory, and could work smoothly with a collection size of 460,000 images. The time to do one cycle in the workflow is typically below 5 minutes in our experiments (R3).

The histograms in the person-, concept, and time view support the queries implied by the PICTuRe diagram. By selecting people, we see when they are active on the timeline and which concepts

they are interested in via the concept view. The other queries are done in a similar way (R4). Custom concepts can be added to the concept view by assigning positive and negative examples in the image view (R5).

7.2. Limitations

An important difficulty we faced was the high dimensionality of the concept vector. We are simply not able to show all concepts at once and at the same time keeping the time-person chart clear. The use of location information in the similarity measure seems to be powerful. We however observed many false positive matches in densely populated areas, where there is a bigger chance that images were made at roughly the same time and location.

7.3. Future work

Now that we are able to spot relations between personal image collections, a natural next step would be to visualize their network. This could be challenging because the number of nodes can be large and the number of (weighted) edges can be huge. A simple alternative could be to show a two-dimensional MDS plot of the people using the distance measure in Section 5.1, in order to find clusters. A second direction would be to generalize the visualization to other types of media, such as text messages, music, or paintings. The person view would then contain authors, composers, or artists. Other neural networks can (be constructed to) derive concepts, styles, or topics for these alternative domains. A possible application could be to explore how artists influence each other by analyzing their styles over time. Finally, we could use more meta-data, for instance visual descriptors that support face recognition or detection of (near) duplicates. Duplicate detection can be helpful for seeing how an uploaded image spreads over a network.

8. Conclusions

We presented a method and demonstrated a prototype for finding people with similar image collections. In contrast to related work in image collection visualization, we show the patterns of people’s collections and enable the user to compare these. The core contribution to this is a metric for similarity between people, based on concepts, temporal, and geospatial aspects, and a glyph based interactive visualization to enable analysts to inspect the patterns found in detail. The main difficulty is the dimensionality of the concept vector, which forces the user to choose the parameters of the similarity metric carefully. Our experiments showed that the system enables the user to find convincing suggestions for people in reasonable time.

Acknowledgments

We thank Jan Zahálka and Marcel Worring (University of Amsterdam) for our discussions on this topic. We also thank the experts from the Netherlands Forensic Institute and the Dutch Police for participating in the evaluation and for their feedback. This work is part of the Open Technology Program with project number 12540, which is financed by the Netherlands Organization for Scientific Research (NWO).

References

- [AMST11] AIGNER W., MIKSCH S., SCHUMANN H., TOMINSKI C.: *Visualization of Time-Oriented Data*, 1st ed. Springer-Verlag London, 2011. 2
- [CK91] COUSINS S. B., KAHN M. G.: The Visual Display of Temporal Information. *Artificial Intelligence in Medicine* 3, 6 (1991), 341–357. 2
- [CL11] CHANG C.-C., LIN C.-J.: LIBSVM: A Library for Support vector Machines. *ACM Transactions on Intelligent Systems and Technology* 2 (2011), 27:1–27:27. 7
- [DDS*09] DENG J., DONG W., SOCHER R., LI L.-J., LI K., FEI-FEI L.: ImageNet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (June 2009), pp. 248–255. 2, 3
- [DGC10] DÖRK M., GRUEN D., WILLIAMSON C., CARPENDALE S.: A visual backchannel for large-scale events. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (Nov. 2010), 1129–1138. 2
- [DPSS16] DU F., PLAISANT C., SPRING N., SHNEIDERMAN B.: EventAction: Visual Analytics for Temporal Event Sequence Recommendation. In *Proceedings of the IEEE Visual Analytics Science and Technology (to appear)* (2016). 2
- [dRWvW10] DE ROOIJ O., WORRING M., VAN WIJK J. J.: MediaTable: Interactive Categorization of Multimedia Collections. *Computer Graphics and Applications, IEEE* 30, 5 (Sept. 2010), 42–51. 2
- [ENP*09] ELER D. M., NAKAZAKI M. Y., PAULOVIH F. V., SANTOS D. P., ANDERY G. F., OLIVEIRA M. C. F., BATISTA NETO J., MINGHIM R.: Visual Analysis of Image Collections. *The Visual Computer* 25, 10 (2009), 923–937. 1
- [Few13] FEW S.: *Information Dashboard Design: Displaying data for at-a-glance monitoring*, 2nd ed. Analytics Press, 2013. 5
- [JSD*14] JIA Y., SHELHAMER E., DONAHUE J., KARAYEV S., LONG J., GIRSHICK R., GUADARRAMA S., DARRELL T.: Caffe: Convolutional Architecture for Fast Feature Embedding. In *Proceedings of the 22nd ACM International Conference on Multimedia* (Nov. 2014), pp. 675–678. 2, 3
- [KK03] KEOGH E., KASETTY S.: On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration. *Data Mining and Knowledge Discovery* 7, 4 (2003), 349–371. 2
- [KSH12] KRIZHEVSKY A., SUTSKEVER I., HINTON G. E.: ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems* 25, Pereira F., Burges C. J. C., Bottou L., Weinberger K. Q., (Eds.). Curran Associates, Inc., 2012, pp. 1097–1105. 2
- [MGF*09] MATKOVIĆ K., GRAČANIN D., FREILER W., BANOVA J., HAUSER H.: *Large Image Collections – Comprehension and Familiarization by Interactive Visual Analysis*. Springer Berlin Heidelberg, 2009, pp. 15–26. 2
- [Mic] MICROSOFT LIVE LABS: PivotViewer. <http://www.microsoft.com/silverlight/pivotviewer/>. 2
- [MKS16] METTES P., KOELMA D. C., SNOEK C. G. M.: The ImageNet Shuffle: Reorganized Pre-training for Video Event Detection. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval* (June 2016), pp. 175–182. 3
- [PMR*96] PLAISANT C., MILASH B., ROSE A., WIDOFF S., SHNEIDERMAN B.: Lifelines: Visualizing personal histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (1996), pp. 221–227. 2
- [PS10] PLANT W., SCHAEFER G.: Image Retrieval on the Honeycomb Image Browser. In *17th IEEE International Conference on Image Processing (ICIP)* (Sept. 2010), pp. 3161–3164. 1
- [Qui86] QUINLAN J.: Induction of Decision Trees. *Machine Learning* 1, 1 (1986), 81–106. 7
- [RASC14] RAZAVIAN A. S., AZIZPOUR H., SULLIVAN J., CARLSSON S.: Cnn features off-the-shelf: An astounding baseline for recognition. In *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops* (June 2014), pp. 512–519. 7
- [SEZ*14] SERMANET P., EIGEN D., ZHANG X., MATHIEU M., FERGUS R., LECUN Y.: OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. In *International Conference on Learning Representations (ICLR 2014)* (Apr. 2014), pp. 1–16. 2
- [SLJ*15] SZEGEDY C., LIU W., JIA Y., SERMANET P., REED S., ANGUELOV D., ERHAN D., VANHOUCHE V., RABINOVICH A.: Going Deeper with Convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2015), pp. 1–9. 3
- [TSLX12] TAN L., SONG Y., LIU S., XIE L.: ImageHive: Interactive Content-Aware Image Summarization. *IEEE Computer Graphics and Applications* 32, 1 (2012), 46–55. 1
- [TWE94] TWEEDIE L., SPENCE B., WILLIAMS D., BHOGAL R.: The Attribute Explorer. In *Conference Companion on Human Factors in Computing Systems* (Apr. 1994), pp. 435–436. 4
- [vBvEvB*15] VAN BEEK H., VAN EIJK E., VAN BAAR R., UGEN M., BODDE J., SIEMELINK A.: Digital Forensics as a Service: Game on. *Digital Investigation* 15 (2015), 20–38. 8
- [vdCvW16a] VAN DER CORPUT P., VAN WIJK J. J.: Exploring Items and Features with IF,FI-Tables. *Computer Graphics Forum* (2016). 2, 4
- [vdCvW16b] VAN DER CORPUT P., VAN WIJK J. J.: ICLIC: Interactive Categorization of Large Image Collections. In *2016 IEEE Pacific Visualization Symposium (PacificVis)* (Apr. 2016), pp. 152–159. 2
- [WKZ16] WORRING M., KOELMA D., ZAHÁLKA J.: Multimedia Pivot Tables for Multimedia Analytics on Image Collections. *IEEE Transactions on Multimedia* 18, 11 (Nov. 2016), 2217–2227. 2
- [WRZ*15] WANG C., REESE J. P., ZHANG H., TAO J., GU Y., MA J., NEMIROFF R. J.: Similarity-Based Visualization of Large Image Collections. *Information Visualization* 14, 3 (2015), 183–203. 2
- [WS09] WONGSUPHASAWAT K., SHNEIDERMAN B.: Finding Comparable Temporal Categorical Records: A Similarity Measure with an Interactive Visualization. In *IEEE Symposium on Visual Analytics Science and Technology* (2009), pp. 27–34. 2
- [Yah16] YAHOO: Flickr. <https://www.flickr.com/>, 2016. 9
- [YFH*06] YANG J., FAN J., HUBBALL D., GAO Y., LUO H., RIBARSKY W., WARD M. O.: Semantic Image Browser: Bridging Information Visualization with Automated Intelligent Image Analysis. In *IEEE Symposium on Visual Analytics Science and Technology* (Oct. 2006), pp. 191–198. 2
- [Ziu] ZIUZ FORENSIC: VizX2. <http://forensic.ziuz.com/en>. 8
- [ZRW15] ZAHÁLKA J., RUDINAC S., WORRING M.: Interactive Multimodal Learning for Venue Recommendation. *IEEE Transactions on Multimedia* 17, 12 (Dec. 2015), 2235–2244. 2