Search and Explore Strategies for Interactive Analysis of Real-Life Image Collections with Unknown and Unique Categories

Gisolf, F.; Geradts, Z.; Worring, M.

DOI
10.1007/978-3-030-67835-7_21

Publication date
2021

Document Version
Final published version

Published in
MultiMedia Modeling

License
Article 25fa Dutch Copyright Act

Link to publication

Citation for published version (APA):
Gisolf, F., Geradts, Z., & Worring, M. (2021). Search and Explore Strategies for Interactive Analysis of Real-Life Image Collections with Unknown and Unique Categories. In J. Lokoč, K. Schoeffmann, X. Li, I. Patras, T. Skopal, V. Mezaris, & S. Vrochidis (Eds.), *MultiMedia Modeling: 27th International Conference, MMM 2021 : Prague, Czech Republic, June 22–24, 2021 : proceedings* (Vol. 2, pp. 244-255). (Lecture Notes in Computer Science; Vol. 12573). Springer. https://doi.org/10.1007/978-3-030-67835-7_21

Search and Explore Strategies for Interactive Analysis of Real-Life Image Collections with Unknown and Unique Categories

Floris Gisolf[1,2](✉), Zeno Geradts[1,3], and Marcel Worring[1]

[1] University of Amsterdam, Amsterdam, The Netherlands
[2] Dutch Safety Board, The Hague, The Netherlands
`f.gisolf@safetyboard.nl`
[3] Netherlands Forensic Institute, The Hague, The Netherlands

Abstract. Many real-life image collections contain image categories that are unique to that specific image collection and have not been seen before by any human expert analyst nor by a machine. This prevents supervised machine learning to be effective and makes evaluation of such an image collection inefficient. Real-life collections ask for a multimedia analytics solution where the expert performs search and explores the image collection, supported by machine learning algorithms. We propose a method that covers both exploration and search strategies for such complex image collections. Several strategies are evaluated through an artificial user model. Two user studies were performed with experts and students respectively to validate the proposed method. As evaluation of such a method can only be done properly in a real-life application, the proposed method is applied on the MH17 airplane crash photo database on which we have expert knowledge. To show that the proposed method also helps with other image collections an image collection created with the Open Image Database is used. We show that by combining image features extracted with a convolutional neural network pretrained on ImageNet 1k, intelligent use of clustering, a well chosen strategy and expert knowledge, an image collection such as the MH17 airplane crash photo database can be interactively structured into relevant dynamically generated categories, allowing the user to analyse an image collection efficiently.

Keywords: Image collections · Exploration · Search · Strategy · Interactive

1 Introduction

Human analysts can quickly grasp the meaning of a small set of complex images, but it is difficult for them to analyze a large, unorganized image collection. Images in image collections often are related to each other in many possible

J. Lokoč et al. (Eds.): MMM 2021, LNCS 12573, pp. 244–255, 2021.
https://doi.org/10.1007/978-3-030-67835-7_21

ways, such as in time, location, objects and persons, making the collections very complex with $O(n^2)$ relations, even for a relatively small number of images. There are highly successful automatic categorization methods, e.g. [14,21]. However, many real-life image collections contain image categories that are unique and have not been seen before by analyst nor by the machine. With no training data available, this prevents supervised machine learning to be effective in classification. The analyst can learn a new category with only a few examples, however, has a limited working memory and suffers from fatigue during repetitive tasks. Therefore, to categorize the collection and gain insight in an efficient way, interplay between the human expert and a machine is essential and for that a good understanding of the analytical process is important.

In [27], the process is modeled by the exploration-search axis. **Search** is a sequence of query-response pairs where both the analyst and the system have a fixed model of the data. **Exploration** is the process of the analyst uncovering some structure and points of interest within the image collection, where the analyst and the system work with a dynamic model of the data, which can change over time based on what is deemed relevant and what is not.

Generally, some exploration needs to take place before the analyst can start searching for specific items of relevance. The individual components for exploring [12,13,18] and searching [17] exist and have been studied extensively. However, it remains unclear how a user can go from a complex, unstructured image collection to exploring the data, bringing structure to it, searching relevant items and ultimately gain insight. To model the analytical process, we should not only consider the individual components, but also a comprehensive strategy of exploration and search.

Categorizing relevant images is the umbrella task for the exploration-search axis [28] and allows the analyst to perform all other exploration and search tasks more efficiently. Browsing the data becomes more meaningful; the data can be summarized by its categories; and search tasks can be performed using the categories as a rough decomposition.

We develop and evaluate several explore and search strategies that allow the analyst to efficiently perform this umbrella task. A strategy is more successful when the analyst has to assess fewer images in order to find what she is looking for. A demonstration video, the code and the application[1] are available.

The unique opportunity of having access to and expert knowledge of a real life accident investigation photo database of the MH17 airplane crash on 17 July 2014 [4] is used to design and evaluate the proposed methods. To evaluate robustness of the methods an additional image collection is constructed using the Open Image Database [11] (OID).

The main contributions of this paper consist of (1) a method to explore and search through an image collection containing images of unknown and unique categories in an easy to use and intuitive way where the user can control the process using only a single parameter; (2) the evaluation of several explore and

[1] Demonstration video on https://youtu.be/73-ExDd2lco, code and application on https://tinyurl.com/imexMMM.

search strategies using an artificial user model to show that the right strategy can make a large difference in how many images the user has to inspect to find the relevant items; (3) user studies performed with investigators of the Dutch Safety Board (DSB) and Forensic Science students to verify the results obtained through the experiments with the simulated users.

2 Related Work

As there is no comparable method that covers both exploration and search and allows for structuring an image collection, this section mainly looks at the different components of the proposed method as discussed in the previous section, and into the expertise of humans versus machines.

Neural networks now perform with almost human accuracy on certain image classification tasks [17], but need many training examples. While one-shot and few-shot learning for machines has been studied, it is not yet on par with human capabilities [5,22,24]. It is thus necessary to combine human expertise with that of the computer in order to achieve insight in large image collections [27].

Zero-shot learning (ZSL) tries to tackle the problem of not having training data for new categories [25,29] through attributes, where a new category may consist of a combination of attributes already present in the training set. However, such extensive attributes need to be available for the categories needed by the analyst, which is a costly process. This makes ZSL unsuitable for the problem discussed in this paper.

Methods such as relevance feedback [30] and active learning [19] can help the analyst with finding new instances of a certain category through interaction. VITRIVR [7] is a complete search method for both images and video, but is less suitable for exploration. (Meta-)transfer learning [20,26] takes an existing neural network and uses additional training for new categories in order to classify images. These search methods do require that the analyst already knows what she is looking for, thus they need to be preceded by an exploration phase.

Worring et al. [23] proposed a framework to explore and visualize data, using content-based image features. However, they concentrate on the use of meta data and forensics, making it less suitable for other domains. If meta data is present at all, it is fairly unreliable and easily changed (on purpose or by accident). Furthermore, its main focus is on browsing of image collections, which is only part of the exploration-search axis. MediaTable [16] allows analysts to browse and categorize the data based on a variety of features, but it leaves the clustering and exploring mostly up to the user, which means it does not scale as well with more data. ImageX [10] uses a hierarchical graph, but offers limited search capabilities and no way to store any user progress or structure.

To evaluate algorithms, most papers use one of the in general very good publicly available databases. Life events such as VBS and LSC are becoming more commonplace too. However, in the case of real-life image collections that require expert knowledge to gain insight, such publicly available databases and events cannot completely capture the difficulties an analyst may have to deal with.

3 Proposed Method

The foundation of our proposed method consists of features extracted using a convolutional neural network (CNN); followed by clustering; exploration by the analyst, assigning clusters and images to categories as an interplay between expert and machine; and using the insight gained through exploration for searching additional relevant items Fig. 1. Other work [1,3] has shown that a pretrained CNN has a sufficient number of useful features to differentiate between images of unknown and unique categories. Resnet152 [9] is the neural network we used to extract 2048 features per image.

Clustering. The features are used to generate clusters. This initial clustering helps the analyst in identifying structure and relations within the image collection. The clustering algorithm needs to be fast, so that the user can quickly adjust the clustering results; any user set parameter should be intuitive; and larger, high quality clusters are preferred for faster analysis and reducing user fatigue.

For initial experimentation we used K-means and DBSCAN as a baseline, as they are established clustering algorithms. We also used DESOM [6] (self-organizing map) [2] and DCEC (convolutional autoencoder) [8] as recent methods. Despite recommended hyperparameters as well as others, DCEC performed not much above chance and needs long training time, making it unsuitable. DESOM performed similar to k-means but slower, despite k-mean's already significant computation time. Due to limited space, DCEC and DESOM results have not been included. DBSCAN was by far the fastest. It requires the user to set a distance (threshold) between 0 and 1, which we find more intuitive than the number of clusters. Unfortunately, DBSCAN was too sensitive to the distance function used and would either mark most images as outliers, or put most images in a single cluster for any threshold between 0 and 1.

As none of the clustering algorithms meets all requirements, we developed Correlation-based Clustering (CC, Algorithm 1), loosely based on DBSCAN. A distance matrix is calculated for all image feature vector pairs using correlation (to us the most intuitive distance metric). Next, the image that correlates above threshold t with most other images is used as the start of the first cluster. The highest correlating image is first added to the cluster. Then, the cluster center is recalculated and correlations for all images are recalculated with respect to this new center. The highest correlating image is added to the cluster. This is repeated until no image correlates with the cluster center above t. Of the remaining images not in a cluster, the image with most other images correlating above t is the start of the second cluster. This repeats until all images are clustered or until no image has a correlation above t. In the latter case, t is lowered by 0.1 and the process repeats until no images remain. This process means clusters can have arbitrary shapes and that early clusters will be of high quality, while later clusters will be of lower quality. We did not look into optimizing memory usage of the distance matrix, but CC can be batched with an additional step of merging clusters of different batches. For the remainder of the paper, k-means and CC were used as the clustering algorithms. In our method the user is presented with the images of one cluster at a time on a basic scrollable grid. For CC, clusters are shown in order of creation. This means the analyst sees the highest quality clusters first. Images within the cluster are sorted from most to least similar to the first image. For k-means cluster order is random, and images are shown in order of distance to the cluster center.

ALGORITHM 1: Correlation-based clustering

```
Calculate correlation matrix
while there are correlations between image pairs in Correlation Matrix do
    if max(Correlation Matrix) > threshold then
        initialize new cluster with $I_{most}$ // $I_{most}$ is the image that is
          correlated with most other images above threshold
        center = feature vector of $I_{most}$
        while max(Correlation(center, other images) > threshold do
            add $I_{max}$ to cluster // $I_{max}$ is the image that has the
              highest correlation with center
            center = average feature vector of images in cluster
        end
    else
        decrease threshold
    end
end
```

Buckets. An important part of the method is the set of buckets, in which the analyst can place images belonging to a self defined category. These buckets can be created on the fly and may change over time, as the insight of the analyst changes. Creation of buckets is part of exploration, where the data model is not fixed.

Strategies. The user needs a strategy to go from knowing nothing about an image collection given some basic structure through the clustering algorithm, to a structured, categorized image collection. The fewer images a user has to see to attain a high recall, the better the strategy. Evaluating the results of a strategy in such a way implicitly takes precision into account as well. The most basic strategy is browsing clusters and assigning images or clusters to buckets. This can take a long time, as relevant images may be located in clusters shown last. We therefore propose several explore and search methods. Methods were chosen based on common explore and search tasks [28]: structuring, summarization, sorting and querying. These methods are combined into strategies.

An artificial (simulated) analyst (AA) is used to evaluate all strategies, as this is infeasible with human analysts. The AA has access to the ground truth (the annotations of the images in the image collection). The goal is to categorize all images in the image collection (m) into predetermined buckets (one bucket for each class in the annotations). Each strategy (S) is repeated for each bucket (B), resulting in a total of $S * B$ simulations per image collection. For each simulation, the image collection is divided into relevant images $RI_{0,1,...n}$ (the images belonging to the current B), and irrelevant images $II_{0,1...m-n}$ (all images not belonging in B). For each simulation, the AA will see each image once. The recall of S is calculated after each image the AA has seen and then averaged for all B per S. Success of a strategy is determined by how many images need to be visually inspected to reach 80% recall of all RI over all B. 80% was chosen as it is the majority of images in a category, and should give the analyst a clear idea of the relevant images.

Each strategy consists of five choices. The ***first choice*** is between k-means or CC. The ***second choice*** is to set the threshold t or k for k-means. The AA uses 0.3 and 0.7 for t when the first choice is CC. If the first choice is k-means, k is based on the number of clusters generated by CC using these thresholds. All t between 0 and 1 with steps of 0.1 have been tested, 0.3 and 0.7 were chosen to show the difference between high and low t due to limited space in the paper.

Exploration. We propose two exploration methods. The AA chooses whether to use them or not. Using the overview is the ***third choice***. The overview is a grid showing one representative image for each cluster to quickly get an idea of the contents of the image collection, and allows the analyst to select relevant clusters to inspect further. The representative image is the image whose feature vector is closest to the average feature vector of the cluster. The ***fourth choice*** is to use sorting. Once at least one image is added to a bucket, the analyst can sort all clusters based on similarity to that bucket, making it more likely that clusters with relevant images show up first. *Search* There are several ways to search through an unannotated image collection. Most are based on similarity queries. The following is a list of search methods that we consider for our strategies and is the ***fifth choice***. The AA picks one or no search method. Search methods that rank images make use of the correlation matrix.

Expand Cluster/Bucket: A cluster with relevant images can be recalculated with a lower threshold, expanding the cluster with more images that may be relevant. Similarly, more images can be found using the images in a bucket. This can be repeated until no more relevant images are found. The AA reduces t by 0.2 for each expansion. If additional images found through the expansion contained at least 30% relevant images, expansion is repeated with a reduced threshold. *Query (external) image/bucket/part of image:* query (part of) an image or the average feature vector of multiple images to rank all images from most similar to least similar to the queried image (the analyst selects which part of an image she is interested in if required). Each query is processed until 8 II in a row are encountered. For Query Image the AA selects 10 random images, for Query Bucket, the query is repeated if at least 10 RI have been added. Query Part of Image is not used by the AA, as it would require bounding box annotations that are not available. Query External Image (image not in the image collection) is also not used as it would require human judgment to select a useful external image. *Select from projection:* the feature vectors are used to calculate a 2D representation of the features using UMAP [15]. The analyst can then select with a bounding box which images to view. For the AA, the UMAP 2D representation was normalized. The AA will calculate the modal image from the images in bucket within the 2D representation and draw a bounding box of 0.2 by 0.2, centered on the modal image. Images within the bounding box will then be assessed. Combining the baseline with these choices gives us our strategies. Figure 1 shows the complete method.

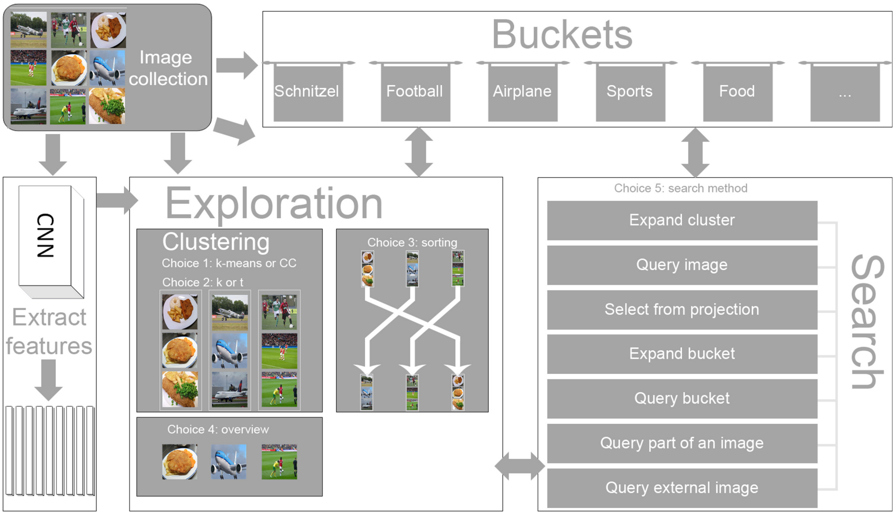

Fig. 1. Schematic overview of the method

Image Collections. To evaluate the strategies, this paper makes use of two image collections. The *MH17 image collection (MIC)*: the real-life accident investigation database [4] of the MH17 airplane crash. The MIC contains 14,579 images and contains classified information, thus only the first author, part of the DSB, had full access. Co-authors have seen examples of content. Images shown are from public sources. For evaluation, annotation used by the DSB was used to reflect the actual situation. Each image can be annotated with one or more of 27 categories; some examples are given in Fig. 2. *OID image collection (OIC)*: a selection of images from the Open Image Database (OID). OIC consists of 37 image categories with objects, locations and activities that were not present in the training set for the CNN. Only images verified by human annotators were used.

4 Evaluation and Discussion

Table 1 shows strategy 1 requires the analyst to assess 70–80% of all images to reach a recall of 80% for all categories. Adding a search method (strategies 2–6) can reduce this to around 50%. Adding sorting of clusters (strategies 7–12)

Table 1. Strategies and fraction of images the analyst has to assess to reach 80% recall on all categories. CC0.3 is CC with a threshold of 0.3, km0.3 is k-means with k equal to the number of clusters CC0.3 generated.

Strategy	Sorting	Overview	Search	Fraction of images seen at recall of 0.8							
				MH17				OID			
				CC0.3	CC0.7	km0.3	km0.7	CC0.3	CC0.7	km0.3	km0.7
1	No	No	None	0.68	0.74	0.80	0.82	0.85	0.85	0.81	0.79
2			Expand cluster	0.69	0.74	0.61	0.68	0.82	0.84	0.43	0.55
3			Query image	0.69	0.74	0.78	0.80	0.71	0.62	0.57	0.57
4			Projection	0.56	0.72	0.61	0.59	0.76	0.53	0.28	0.51
5			Expand bucket	0.67	0.74	0.70	0.82	0.78	0.84	0.56	0.77
6			Query bucket	0.68	0.74	0.78	0.80	0.81	0.67	0.68	0.45
7	Yes	No	None	0.58	0.66	0.57	0.55	0.62	0.51	0.24	0.29
8			Expand cluster	0.58	0.62	0.50	0.51	0.59	0.51	0.26	0.23
9			Query image	0.57	0.63	0.52	0.49	0.59	0.51	0.23	0.25
10			Projection	0.57	0.69	0.60	0.60	0.67	0.53	0.29	0.52
11			Expand bucket	0.56	0.66	0.57	0.55	0.59	0.51	0.23	0.28
12			Query bucket	0.54	0.66	0.71	0.55	0.59	0.50	0.23	0.17
13	No	Yes	None	0.60	0.54	0.49	0.60	0.62	0.36	0.21	0.13
14			Expand cluster	0.55	0.44	0.65	0.50	0.28	0.18	0.10	0.12
15			Query image	0.47	0.39	0.65	0.50	0.25	0.10	0.04	0.10
16			Projection	0.48	0.54	0.52	0.34	0.23	0.11	0.06	0.10
17			Expand bucket	0.62	0.42	0.63	0.60	0.56	0.35	0.11	0.12
18			Query bucket	0.40	0.28	0.55	0.47	0.52	0.10	0.06	0.09
19	Yes	Yes	None	0.46	0.28	0.26	0.25	0.15	0.10	0.05	0.09
20			Expand cluster	0.30	0.30	0.43	0.36	0.19	0.10	0.9	0.10
21			Query image	0.42	0.26	0.30	0.24	0.08	0.10	0.04	0.09
22			Projection	0.46	0.33	0.33	0.30	0.14	0.11	0.06	0.10
23			Expand bucket	0.52	0.28	0.40	0.25	0.15	0.10	0.07	0.09
24			Query bucket	0.33	0.24	0.26	**0.23**	0.09	0.10	**0.04**	0.09

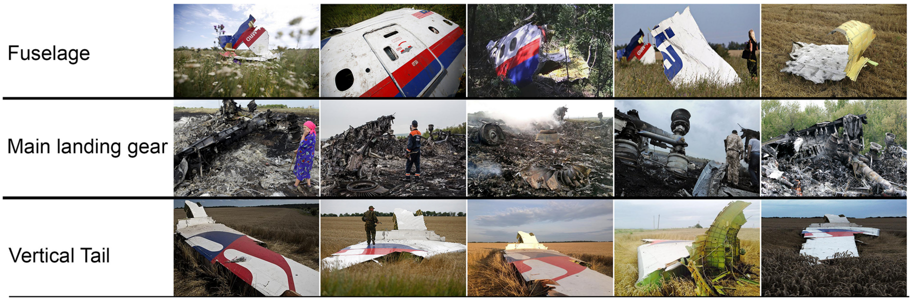

Fig. 2. Some examples of the categories in the MH17 image collection

reduces the number of images that have to be assessed and likely results in a better browsing experience. Using the overview (strategies 13–18) has a stronger impact. Combining all choices (strategies 19–24) means the user only has to assess about a quarter of the MIC to find 80% of all images of a specific category, with querying the bucket being the best search method. For OIC the user only

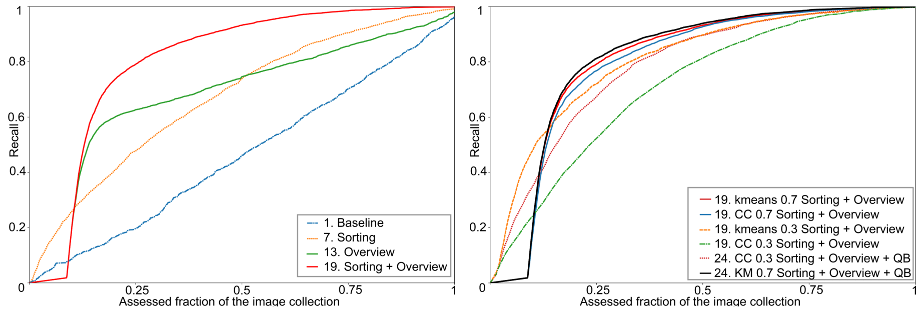

Fig. 3. Recall vs. # of images assessed of strategies employed on the MIC. (left) Exploration strategies are shown, using k-means with $k = 1221$, which has the best performance on the MIC as shown in Table 1. (right) The difference between k-means and CC, the effects of high and low thresholds, and the effect of the best performing search method. The graph of strategies using the overview have a distinct shape at the start; this is where the user assesses the representative images of the clusters.

has to assess 4% of the image collection using strategy 24. The table shows the MIC is more difficult than the OIC, with users having to assess a fraction at least 5 times as large to reach the same results. While k-means performs better than CC on OIC, the difference on the MIC is minimal and depends on the applied strategy. Figure 3 shows the complete curve of recall versus images assessed.

Table 1 shows that increasing the number of clusters achieves better results for MIC, as smaller clusters are generally more precise. However, when using strategy 13 with $k \geq 2000$ for k-means recall will increase less fast than with fewer than 2000 clusters, because the overview requires the user to view a representative image, which counts towards the images seen. And while the AA does not get tired of assessing many small clusters, it is our experience that for a human analyst this increases fatigue compared to fewer, larger clusters.

Clustering 10,000 images on an Intel Xeon E3-1505M v5 using CC ($t = 0.5$) takes 12.7 s (2.6 s for the matrix and 10.1 s for clustering), resulting in 449 clusters. The Scikit implementation of k-means took approximately 349.2 s with $k = 449$.

4.1 User Experiments

To validate whether the results obtained with our method through the AA are actually useful for human analysts, we performed two user experiments. One with a group of domain experts, another with a group of Forensic Science students (to-be domain experts).

Domain Expert User Experiment. The main goal of the domain expert user experiment is to find out whether the experts think the clusters provide value for their work. The strategy used in this user experiment resembles mostly strategy 1. An application was built around the method (see footnotes). Four accident investigators from the DSB were given 2 hours to organize the MIC as they would in an investigation. Two users were blindly assigned clustering results for the MIC from CC and two users k-means. t for CC was set to 0.5 resulting in 1150 clusters, which was also used as k for k-means. Consensus among expert users was that the method as implemented in the application worked, that clusters were of usable and of high quality, and that it would significantly increase their efficiency when working with large image collections. In most cases it was clear why certain images were in a cluster, giving confidence in the method.

Students User Experiment. The second user experiment involved 16 Forensic Science students who had access all explore and search functions in Sect. 3. In pairs they were asked to perform 3 tasks in 1 hour with a written report about their approaches.

Find Image of Old Blue Car. Strategies: Query an external image they found on the internet of a blue car; use the overview to find a cluster with cars, then query images with cars to find the blue car; find an image of a wheel and query that part of the image to find the blue car. A group using the external image query was the fastest to find the image.

Find 40 Images That Best Summarizes the Image Collection. Evaluated by measuring the minimum distance of the features of all images to the features of the 40 selected images. Strategies: change t such that only 40 clusters were generated, then choose 1 image of each cluster; look through the overview from the default t and pick 40 images; browse through clusters and select 40 images. Adjusting t gave the best results.

Find the Food Item Most Common in the Image Collection and Find the Most Images with This Item. 8 different food items were present. 4 of 6 groups found the right food group. Strategies: all used the overview to find food clusters and divided them over several buckets. Finding more items was done through querying individual items or buckets. Best group recalled 83% with a false positive rate of 0.1. Groups not identifying the correct food item both chose the second most common food item.

The experiment showed that the proposed method was fairly intuitive to use, since all groups managed to use several explore and search functions and combine them to complete the tasks.

5 Conclusion

In this paper, a method for analyzing image collections is proposed to help expert analysts. The method covers the exploration-search axis, allowing the analyst to quickly find relevant images in an image collections containing unique, never seen before image categories. For this, the analyst only needs to set a single intuitive parameter. A clustering algorithm (CC) was designed to meet the criteria of serving the expert user at interactive speed, with priority on showing large and high quality clusters. Quantitative results are similar to k-means, with much lower computation time. We show how effective explore and search strategies can greatly reduce the number of images the analyst needs to see in order to find relevant images. Through the user experiment it is demonstrated that with combining expert knowledge, computer vision and the right strategy, the analyst has the tools needed to face the challenges posed by the ever increasing amount of data in a complex environment, such as a real life accident investigation database.

References

1. Babenko, A., Slesarev, A., Chigorin, A., Lempitsky, V.: Neural codes for image retrieval. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 584–599. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_38
2. Barthel, K.U., Hezel, N.: Visually exploring millions of images using image maps and graphs, pp. 251–275. John Wiley and Sons Inc. (2019)
3. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: European Conference on Computer Vision (2018)
4. Dutch Safety Board: Investigation crash mh17, 17 July 2014, October 2015
5. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Trans. Pattern Anal. Mach. Intell. **28**, 594–611 (2006)
6. Forest, F., Lebbah, M., Azzag, H., Lacaille, J.: Deep embedded SOM: joint representation learning and self-organization. In: ESANN 2019 - Proceedings, April 2019
7. Gasser, R., Rossetto, L., Schuldt, H.: Multimodal multimedia retrieval with Vitrivr. In: Proceedings of the 2019 on International Conference on Multimedia Retrieval, ICMR 2019, pp. 391–394. Association for Computing Machinery, New York (2019)
8. Guo, X., Liu, X., Zhu, E., Yin, J.: Deep clustering with convolutional autoencoders. In: Liu, D., Xie, S., Li, Y., Zhao, D., El-Alfy, E.S. (eds.) ICONIP 2017. LNCS, vol. 10635, pp. 373–382. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70096-0_39
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, June 2016
10. Hezel, N., Barthel, K.U., Jung, K.: ImageX - explore and search local/private images. In: Schoeffmann, K., et al. (eds.) MMM 2018. LNCS, vol. 10705, pp. 372–376. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73600-6_35
11. Krasin, I., et al.: OpenImages: a public dataset for large-scale multi-label and multi-class image classification (2017). https://github.com/openimages
12. Kratochvíl, M., Veselý, P., Mejzlík, F., Lokoč, J.: SOM-hunter: video browsing with relevance-to-SOM feedback loop. In: Ro, Y.M., et al. (eds.) MMM 2020. LNCS, vol. 11962, pp. 790–795. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-37734-2_71
13. Leibetseder, A., et al.: LifeXplore at the lifelog search challenge 2019. In: Proceedings of the ACM Workshop on Lifelog Search Challenge, pp. 13–17. Association for Computing Machinery, New York (2019)
14. Liu, C., et al.: Progressive neural architecture search. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11205, pp. 19–35. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_2

15. McInnes, L., Healy, J., Melville, J.: UMAP: uniform manifold approximation and projection for dimension reduction (2018)
16. de Rooij, O., van Wijk, J.J., Worring, M.: MediaTable: interactive categorization of multimedia collections. IEEE Comput. Graph. Appl. **30**(5), 42–51 (2010)
17. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
18. Schoeffmann, K.: Video browser showdown 2012–2019: a review. In: 2019 International Conference on Content-Based Multimedia Indexing (CBMI), pp. 1–4 (2019)
19. Settles, B.: Active learning literature survey. Computer Sciences Technical report 1648, University of Wisconsin-Madison (2009)
20. Sun, Q., Liu, Y., Chua, T.S., Schiele, B.: Meta-transfer learning for few-shot learning. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019
21. Touvron, H., Vedaldi, A., Douze, M., Jégou, H.: Fixing the train-test resolution discrepancy. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
22. Wang, Y., Chao, W.L., Weinberger, K.Q., van der Maaten, L.: SimpleShot: revisiting nearest-neighbor classification for few-shot learning (2019)
23. Worring, M., Engl, A., Smeria, C.: A multimedia analytics framework for browsing image collections in digital forensics. In: Proceedings of the 20th ACM International Conference on Multimedia, MM 2012, pp. 289–298. ACM, New York (2012)
24. Yan, M.: Adaptive learning knowledge networks for few-shot learning. IEEE Access **7**, 119041–119051 (2019)
25. Yang, G., Liu, J., Xu, J., Li, X.: Dissimilarity representation learning for generalized zero-shot recognition. In: Proceedings of the 26th ACM International Conference on Multimedia, MM 2018, pp. 2032–2039. Association for Computing Machinery, New York (2018)
26. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Proceedings of the 27th International Conference on Neural Information Processing Systems, NIPS 2014, vol. 2. pp. 3320–3328. MIT Press, Cambridge (2014)
27. Zahálka, J., Worring, M.: Towards interactive, intelligent, and integrated multimedia analytics. In: 2014 IEEE Conference on Visual Analytics Science and Technology (VAST), pp. 3–12, October 2014
28. Zahálka, J., Rudinac, S., Worring, M.: Analytic quality: evaluation of performance and insight in multimedia collection analysis. In: Proceedings of the 23rd ACM International Conference on Multimedia, MM 2015, pp. 231–240. ACM, New York (2015)
29. Zhang, Z., Saligrama, V.: Zero-shot learning via joint latent similarity embedding. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6034–6042, June 2016
30. Zhou, X.S., Huang, T.S.: Relevance feedback in image retrieval: a comprehensive review. Multimedia Syst. **8**(6), 536–544 (2003)