Interactive Learning for Multimedia at Large

Omar Shahbaz Khan[1], Björn Þór Jónsson[1,4], Stevan Rudinac[2], Jan Zahálka[3], Hanna Ragnarsdóttir[4], Þórhildur Þorleiksdóttir[4], Gylfi Þór Guðmundsson[4], Laurent Amsaleg[5], and Marcel Worring[2]

[1] IT University of Copenhagen, Copenhagen, Denmark
[2] University of Amsterdam, Amsterdam, Netherlands
[3] Czech Technical University in Prague, Prague, Czech Republic
[4] Reykjavik University, Reykjavík, Iceland
[5] CNRS–IRISA, Rennes, France

Abstract. Interactive learning has been suggested as a key method for addressing analytic multimedia tasks arising in several domains. Until recently, however, methods to maintain interactive performance at the scale of today's media collections have not been addressed. We propose an interactive learning approach that builds on and extends the state of the art in user relevance feedback systems and high-dimensional indexing for multimedia. We report on a detailed experimental study using the ImageNet and YFCC100M collections, containing 14 million and 100 million images respectively. The proposed approach outperforms the relevant state-of-the-art approaches in terms of interactive performance, while improving suggestion relevance in some cases. In particular, even on YFCC100M, our approach requires less than 0.3 seconds per interaction round to generate suggestions, using a single computing core and less than 7GB of main memory.

Keywords: Large multimedia collections · Interactive multimodal learning · High-dimensional indexing · ImageNet · YFCC100M

1 Introduction

A dominant trend in multimedia applications for industry and society today is the ever-growing scale of media collections. As the general public has been given tools for unprecedented media production, storage and sharing, media generation and consumption have increased drastically in recent years. Furthermore, upcoming multimedia applications in countless domains—from smart urban spaces and business intelligence to health and wellness, lifelogging, and entertainment—increasingly require joint modelling of multiple modalities [20, 47]. Finally, users expect to be able to work very efficiently with large-scale collections, even with the limited computing resources they have at their immediate disposal. All these trends contribute to making scalability a greater concern than ever before.

User relevance feedback, a form of interactive learning, provides an effective mechanism for addressing various analytic tasks that require alternating between search and exploration. Figure 1 shows an example of such a relevance feedback

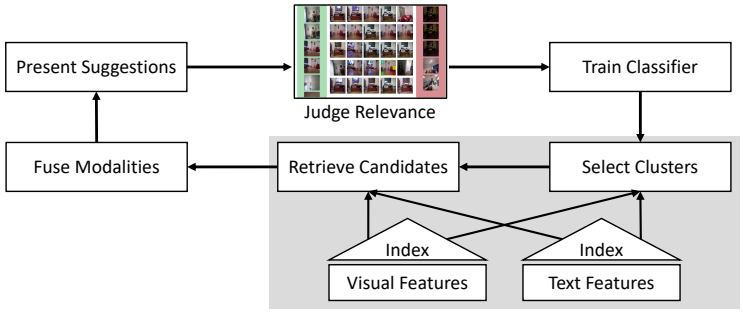

Fig. 1: An outline of the user relevance feedback approach proposed in this paper. The shaded area indicates that the traditional relevance feedback pipeline is enhanced with a novel query mechanism to a state-of-the-art cluster-based high-dimensional index.

process, where positive and negative relevance judgments from the user are used to train a classifier, which in turn is used to provide new suggestions to the user, with the process continuing until the user completes the interaction. There has been relatively little work on user relevance feedback and truly scalable and interactive multimedia systems in general in the last decade, however, which recently raised serious concerns in the multimedia community [39]. Clearly, the time has come to re-visit interactive learning with an aim towards scalability.

We propose Exquisitor, a highly scalable and interactive approach for user relevance feedback on large media collections. As illustrated in Figure 1, the proposed approach tightly integrates high-dimensional indexing with the interactive learning process. To the best of our knowledge, our approach is the first scalable interactive learning method to go beyond utilizing clustering in the preprocessing phase only. To evaluate the approach, we propose a new zero-shot inspired evaluation protocol over the ImageNet collection, and use an existing protocol for the large-scale YFCC100M collection. We show that our approach outperforms state-of-the-art approaches in terms of both suggestion relevance and interactive performance. In particular, our approach requires less than 0.3 seconds per interaction round to generate suggestions from the YFCC100M collection, using a single CPU core and less than 7GB of main memory.

The remainder of this paper is organized as follows. In Section 2, we discuss interactive learning from a scalability perspective, setting the stage for the novel approach. In Section 3, we then present the proposed approach in detail, and compare its performance to the state of the art in Section 4, before concluding.

2 Related Work

As outlined in the introduction, combining interactive learning with high dimensional indexing is a step towards unlocking the true potential of multimedia collections and providing added value for users. In this section we first describe the state of the art in interactive learning. Then, based on the identified ad-

vantages and limitations of interactive learning algorithms, we provide a set of requirements that high-dimensional indexing should satisfy for facilitating interactivity on extremely large collections. Finally, we use those requirements for reflecting on the state of the art in high-dimensional indexing.

Interactive Learning: Interactive learning has long been a cornerstone of facilitating access to document collections [1, 18, 27, 16] and it became an essential tool of multimedia researchers from the early days of content-based image and video retrieval [36, 15]. The most popular flavour of interactive learning is user relevance feedback that presents the user, in each interaction round, with the items for which the classification model is most confident [36]. User relevance feedback has frequently been used in the best performing entries of benchmarks focusing on interactive video search and exploration [28, 41]. However, those solutions were designed for collections far smaller than YFCC100M, which is the challenge we take in this paper. Linear models for classification, such as Linear SVM are still amongst the most frequent choices in relevance feedback applications [22, 31, 48] due to their simplicity, interpretability and explainability as well as the ability to produce accurate results with few annotated samples and scale to very large collections.

To the best of our knowledge, Blackthorn [48] is the most efficient interactive multimodal learning approach in the literature. Its efficiency is achieved through adaptive data compression and feature selection, multi-core processing, and a classification model capable of scoring items directly in the compressed domain. Compared to product quantization [17], a popular alternative optimized for k-NN search, Blackthorn was found to yield significantly more accurate results over YFCC100M with similar latency (1.2 seconds), while consuming modest computational resources (16 CPU cores with 5 GB of main memory).

Indexing Requirements: We have identified the following requirements for high-dimensional indexing to enhance the performance of interactive learning:

- **R1** *Short and Stable Response Time:* A successful indexing approach in interactive learning combines good result quality with response time guarantees [44].
- **R2** *Preservation of Feature Space Similarity Structure:* Since interactive classifiers compute relevance based on a similarity structure on the feature space, the space partitioning of the high-dimensional indexing algorithm must preserve this similarity structure.
- **R3** *k Farthest Neighbours:* Relevance feedback approaches typically try to inform the user by presenting the most confidently relevant items based on the judgments observed so far, which are the items farthest from the classification boundary. As results are intended for display on screen, the index should thus return k farthest neighbours (k-FN).

We are not aware of any work in the high-dimensional literature targeting approximate k-FN where the query is a classification boundary. We therefore next review the related work and discuss how well different classes of high-dimensional indexing methods can potentially satisfy these three requirements.

High-Dimensional Indexing Scalable high-dimensional indexing methods generally rely on approximation through some form of quantization. One class of methods uses scalar quantization. The NV-tree, for example, is a large-scale index that uses random projections at its core [25, 26], recursively projecting points onto segmented random lines. LSH is another indexing method that uses random projections acting as locality preserving hashing functions [2, 8]. Recently, multimedia researchers have considered hashing for multimedia applications, but typically at a much smaller scale than considered here [13, 29, 42]. LSH has been considered in the context of hyperplane-based nearest-neighbour queries [5, 45] and point-based farthest-neighbour queries [7, 32, 46], but not in the context of *hyperplane-based farthest-neighbour* queries. We argue that LSH and related methods fail to satisfy the three requirements above: they focus on quality guarantees rather than performance guarantees (**R1**); hashing creates "slices" in high-dimensional space, making ranking based on distance to a decision boundary difficult (**R2**); and they typically focus on ϵ-range queries, giving no guarantees on the number of results returned (**R3**).

A second class of methods is based on vector quantization, typically using clustering approaches, such as k-means, to determine a set of representative feature vectors to use for the quantization. These methods create Voronoï cells in the high-dimensional space, which satisfy **R2** well. Some methods, such as BoW-based methods, only store image identifiers in the clusters, thus failing to support **R3**, while others store the entire features, allowing to rank the results from the farthest clusters. Finally, many clustering methods seek to match well the distribution of data in the high-dimensional space. Typically, these methods end with a large portion of the collection in a single cluster, which in turn takes very long to read and score, thus failing to satisfy **R1** [12].

Product quantization (PQ) [17] and its variants [4, 10, 14] cluster the high-dimensional vectors into low-dimensional subspaces that are indexed independently. PQ better captures the location of points in the high-dimensional space, which in turn improves the quality of the approximate results that are returned. One of the main aims of PQ is data compression, however, and PQ-based methods essentially transform the Euclidean space, complicating the identification of furthest neighbours (**R2**). PQ-compression was compared directly with the Blackthorn compression method designed for interactive learning [48] and was shown as having inferior performance. The extended Cluster Pruning (eCP) algorithm [11, 12], however, is an example of a vector quantifier which attempts to balance cluster sizes for improved performance, thus aiming to satisfy all three requirements; we conclude that eCP is our prime candidate.

3 The Exquisitor Approach

In this section, we describe Exquisitor, a novel interactive learning approach that tightly integrates high-dimensional indexing with the interactive learning process, facilitating interactive learning at the scale of the YFCC100M image collection using very moderate hardware resources. Figure 1 shows an outline of

the Exquisitor approach. We start by considering the multimodal data representation and classifier, before describing the indexing and retrieval algorithms in separate sub-sections. To facilitate the exposition in this section, we occasionally use actual examples from the YFCC100M collection.

3.1 Media Representation and Classification Model

Similar to [48], we choose to represent each image with two semantic feature vectors, one for visual content using deep-learning-based feature vectors and the second for textual content by extracting LDA topics from any textual metadata associated with the images. Although more descriptive approaches for extracting text features exist, in this case the LDA is effective in yielding discriminative representation for different items.

Directly working with these representations, however, is infeasible. In our case, using 1,000 and 100 dimensions for the visual and text domains, respectively, the feature vectors would require 8.8KB of main memory per image, or around 880GB for the YFCC100M collection, which is far beyond the storage capacity of typical hardware. We use the data compression method presented in [48] that preserves semantic information with over 99% compression rate.

Consistent with the state of the art in user relevance feedback, the classifier used in Exquisitor is Linear SVM. The choice is further motivated by the algorithm's speed, reasonable performance and compatibility with the sparse compressed representation. Note that the choice of interactive classifier and features in each respective modality made in this paper is not an inherent setting of Exquisitor; they can be replaced as deemed fit. The choices made in this paper are in line with the choices made in the state of the art Exquisitor competes against (most notably [48]), providing a level field for experimental evaluation.

3.2 Data Indexing

The data indexing algorithm used in Exquisitor is based on the extended Cluster Pruning (eCP) algorithm [12]. As motivated in Section 2, the goal is to individually cluster each of the two feature representations with a vector quantizer, using a hierarchical index structure to facilitate efficient selection of clusters to process for suggestions. For each collection, cluster representatives are selected randomly and clusters are formed by assigning images to the nearest cluster based on Euclidean distance, computed efficiently directly in compressed space. The indexing algorithm recursively selects 1% of the images at each level as representatives for the level above, until fewer than 100 representatives remain to form the root of the index. As an example, the bottom level of the index for each modality in the YFCC100M collection consists of $992,066$ clusters, organized in a 3 level deep index hierarchy, which gives on average 100 images per cluster and per internal node. When building the indices, the average cluster size was chosen to be small, as previous studies show that searching more small clusters yields better results than searching fewer large clusters [11, 40].

3.3 Suggestion Retrieval

The retrieval of suggestions has the following three phases: identify b most relevant clusters, select r most relevant candidates per modality, and fuse modalities to retrieve k most relevant suggestions.[6]

Identify b Most Relevant Clusters: In each interaction round, the index of representatives is used to identify, for each modality, the b clusters most likely to contain useful candidates for suggestions. This search expansion parameter, b, affects the size of the subset that will be scored and can be used to balance between search quality and latency at run-time. All cluster representatives are scored by the interactive classifier and the b clusters farthest from the separating plane in the positive direction are selected as the most relevant clusters. In Section 4.3 we evaluate the effects of b on the YFCC100M collection.

We observe that with the YFCC100M collection, both modalities have 1-2 clusters that are very large, with more than 1M items. These clusters require a significant effort to process, without improving suggestion quality. In the experiments reported here, we have therefore omitted clusters larger than 1M.

Select r Most Relevant Candidates per Modality: Once the most relevant b clusters have been identified, the compressed feature vectors within these clusters are scored to suggest the r most relevant media items for each modality. The method of scoring individual feature vectors is the same as when selecting the most relevant clusters.

Some notes are in order here. First, in this scoring phase, media items seen in previous rounds are not considered candidates for suggestions. Second, an item already seen in the first modality is not considered as a suggestion in the second modality. Third, if all b clusters are small, the system may not be able to identify r candidates, in which case it simply returns all the candidates found. Finally, we observe that treating all b clusters equally results in an over-emphasis on items that score very highly in only one modality, but have a low score in the other modality. This can be troublesome if the relevant items have a decent score in both modalities. By segmenting the b clusters into S_c segments of size b/S_c this dominance can be avoided; we explore the impact of S_c in Section 4.3.

Modality Fusion for k Most Relevant Suggestions: Once the r most relevant candidates from each modality have been identified, the modalities must be fused by aggregating the candidate lists to produce the final list of k suggestions. First, for each candidate in one modality, the score in the other modality is computed if necessary, by directly accessing the compressed feature vector, resulting in $2r$ candidates with scores in both modalities.[7] Second, the rank of each item in each modality is computed by sorting the $2r$ candidates. Finally, the average rank is used to produce the final list of suggestions.

[6] In the case of unimodal retrieval, the latter two phases can be merged.

[7] To facilitate late modality fusion, the location of each feature vector in each cluster index is stored; each vector requires ∼800KB of RAM for the YFCC100M collection.

Multi-Core Processing: If desired, Exquisitor can take advantage of multiple CPU cores. With w cores available, the system creates w worker processes and assigns b/w clusters to each worker. Each worker produces r suggestions in each modality and fuses the two modalities into k candidates, as described above. The top k candidates overall are then selected by repeating the modality fusion process for the suggestions produced by the workers.

4 Experimental Evaluation

In this section, we experimentally analyse the interactive performance of Exquisitor. We first outline the baseline comparison architectures from the literature. We then describe two detailed experiments. In the first experiment, we propose a new experimental protocol for interactive learning based on the popular ImageNet benchmark dataset, and show that a) the Linear SVM model is capable of discovering new classes in the data, and b) with high-dimensional indexing, performance is significantly improved. In the second experiment, we then use a benchmark experimental protocol from the literature defined over the YFCC100M collection, and show that at this scale the Exquisitor approach outperforms the baseline architectures significantly, both in terms of retrieval quality and interactive performance.

4.1 Baseline Approaches

In the experiments we compare Exquisitor with the following state-of-the-art approaches from the literature.

Blackthorn: To the best of our knowledge, Blackthorn [48] is the only direct competitor in the literature for interactive learning at the YFCC100M scale. Unlike Exquisitor, Blackthorn uses no indexing or prior knowledge about the structure of the collection, instead using data compression and multi-core processing for scalability.

kNN+eCP: This baseline is representative of pure query-based approaches using a k-NN query vector based on relevance weights [34, 23], an approach that was initially introduced for text retrieval [35] but has been adapted for CBIR with relevance feedback [37].

SVM+LSH, kNN+LSH: These baselines represent SVM-based and k-NN-based approaches using LSH indexing. We replace the eCP index with a multi-probing LSH index [30] using the FALCONN library [3].

All comparison architectures are compiled with g++. Experiments are performed using dual 8-core 2.4 GHz CPUs, with 64GB RAM and 4TB local SSD storage. Note, however, that even the YFCC100M collection requires less than 7GB of SSD storage and RAM, and most experiments use only a single CPU core.

While tuning LSH performance is difficult, due to the many parameters that interact in complex ways (L is the number of tables, B is the number of buckets in each table, and p is the number of buckets to read from each table at query time), we have strived to find parameter settings that a) lead to a similar cell size distribution as eCP and b) yield the best performance.

4.2 Experiment 1: Discovering ImageNet Concepts

Zero-shot learning is a method which trains a classifier to find target classes without including the target classes when training the model. Taking inspiration from zero-shot learning, the objective of this experiment is to simulate a user that is looking for a concept that is on their mind, but is not directly represented in the data; a successful interactive learning approach should be able to do this.

Image Collection: ImageNet is an image database based on the WordNet hierarchy. It is a well-curated collection targeting object recognition research as the images in the collection are categorized into approximately 21,000 WordNet synsets (synonym sets) [9]. The collection contains 14,198,361 images, each of which is represented with the 1,000 ILSVRC concepts [38]. Due to images being categorized into multiple WordNet synsets, the ImageNet collection contains duplicate images, each labelled differently, which can lead to false negatives.

Experimental Protocol: The protocol for the experiment is constructed by randomly selecting 50 concepts from the 1,000 ILSVRC concepts. For each concept a simulated user (henceforth called actor) is created, which knows which images belong to its concept and is charged with the task of finding items belonging to that concept. We have then created and indexed 5 different collections of visual features, where the feature value of the concepts belonging to 10 different actors have been set to 0 to introduce the zero-shot setting.

The workload for each actor proceeds as follows. Initially, 10 images from the concept and 100 random images are used as positive and negative examples, respectively, to create the first round of suggestions, simulating a situation where the exploration process has already started. In each round of the interactive learning process, the actor considers the suggested images from the system and designates images from its concept as positive examples, while 100 additional negative examples are drawn randomly from the entire collection. This is repeated for 10 interaction rounds, with performance statistics collected in each round. To combat the duplicate images problem, we first run the workload using the original data where the concepts are known in order to establish an upper bound baseline for each approach.

Results: Figure 2 compares the average precision across the 10 rounds for each of the approaches under study, for both the case when the concept is *known* (blue columns) and *unknown* (red columns). For Exquisitor and eCP+kNN, the search expansion parameter b is set to 256, while SVM+LSH and kNN+LSH have the following settings for the LSH index: $L = 10$, $B = 2^{14}$, and $p = 20$.

Overall, the figure shows that precision for the known case is nearly 50% on average for the SVM-based approaches, and only slightly lower for the k-NN-based approaches. When the feature value for the actor's concept is not known, however, the average precision drops only slightly for the SVM-based approaches, while the k-NN-based approaches perform very poorly. These results indicate that the Linear-SVM model is clearly superior to the k-NN approach.

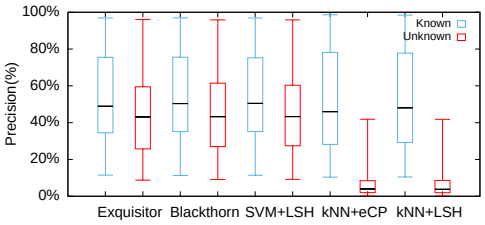

Fig. 2: Average precision per round across all ImageNet actors for each interactive learning approach. The blue columns depict the known case, while the red depict the unknown case.

Table 1: Average latency per interaction round across all ImageNet actors.

Approach	Latency
Exquisitor	0.008 s
Blackthorn (1w)	0.130 s
Blackthorn (16w)	0.017 s
SVM+LSH	0.008 s
kNN+eCP	0.008 s
kNN+LSH	0.004 s

Turning to the average time required for each iteration of the learning process, Table 1 compares the approaches under study. Overall, we note that the four approaches relying on high-dimensional indexing perform very well using a single computing core, requiring less than 10 milliseconds to return suggestions. At the moderate scale of the ImageNet collection, eCP and LSH perform similarly. Running Blackthorn with 16 cores is 2x slower, however, while running Blackthorn using a single core is about 16x slower.

As mentioned above, precision is impacted by the ImageNet collection itself containing duplicates. A visual inspection of the results of some of the worst-performing actors suggest that with known data, the majority of the non-relevant images are such duplicates. For the unknown case, a similar trend is seen for the SVM-based approaches, but not for the k-NN-based approaches, which clearly are unable to steer the query vectors for suggestions to a more relevant part of the collection. Figure 3 shows some examples of this, for the actor for concept "knee pad". As the figure shows, with any SVM-based approach the irrelevant images are also knee pads, but tagged to another related concept, while for the k-NN-based approach, no relevant images were found and the irrelevant images bear no relationship to knee pads.

4.3 Experiment 2: Performance at YFCC100M Scale

The goal of this experiment is to study the scalability of the Exquisitor approach, in comparison to the baseline approaches from the literature. To that end, we apply the only interactivee learning evaluation protocol from the literature that we are aware of at YFCC100M scale [48].

Collection: The YFCC100M collection contains 99,206,564 Flickr images, their associated annotations (i.e. title, tags and description), a range of metadata produced by the capturing device, the online platform, and the user (e.g., geo-location and time stamps). The visual content is represented using the 1,000 ILSVRC concepts [38] extracted using the GoogLeNet convolutional neural network [43]. The textual content is encoded by a) treating the title, tags, and

Fig. 3: Examples of relevant and irrelevant suggestions for different approaches for the ImageNet actor for the concept "knee pad".

description as a single text document, and b) extracting 100 LDA topics for each image using the gensim toolkit [33].

The YFCC100M collection, being large and uncurated, displays some interesting phenomena worth mentioning. First, a non-trivial proportion of images are a standard Flickr "not found" image.[8] A similar situation arises in the text modality, with many images lacking text information altogether, resulting in zero-valued vectors. Such images are essentially noise, potentially crowding out more suitable candidates. Second, with the collection being massive and the data being compressed and clustered, discriminativeness of feature vectors becomes a problem: non-identical images may be mapped to identical feature vectors.

Experimental Protocol: For this experiment we follow the experimental interactive learning protocol in [48]. This evaluation protocol is inspired by the MediaEval Placing Task [24, 6], in which actors simulating user behaviour look for images from 50 world cities.

To illustrate the tradeoffs between the interactive performance and result quality, we focus our analysis on precision and latency (response time) per interaction round. It is worth noting that due to both the scale of YFCC100M and its unstructured nature, precision is lower than in experiments involving small and well-curated collections.

Impact of Search Expansion Parameter: We start by exploring the impact of the search expansion parameter b for the eCP index. Figure 4 analyses the impact of b, the number of clusters read and scored, on the precision (fraction of relevant items seen) in each round of the interactive exploration. The x-axis shows how many clusters are read for scoring at each round, ranging from $b = 1$ to $b = 512$ (note the logarithmic scale of the axis), while the y-axis shows the average precision across the first 10 rounds of analysis. The figure shows precision for two Exquisitor variants, with $S_c = 1$ and $S_c = 16$. In both cases, only one worker is used, $w = 1$. For comparison, the figure also shows the average precision for Blackthorn, the state-of-the-art SVM-based alternative.

As Figure 4 shows, result quality is surprisingly good when scoring only a single cluster in each interaction round, returning about two-thirds of the pre-

[8] The image collection was actually downloaded very shortly after release, but already then this had become a significant issue.

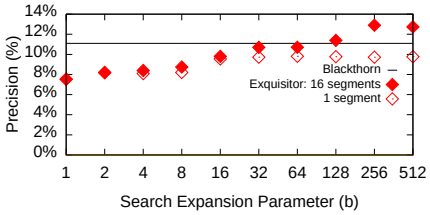

Fig. 4: Average precision over 10 rounds of analysis across all YFCC100M actors. Exquisitor: Varying b; $w = 1$; $S_c = 1, 16$.

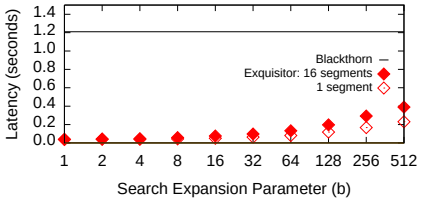

Fig. 5: Average latency over 10 rounds of analysis across all YFCC100M actors. Exquisitor: Varying b; $w = 1$; $S_c = 1, 16$.

cision of the state-of-the-art algorithm. As more clusters are considered, quality then improves further. As expected, dividing the b clusters into $S_c = 16$ chunks results in better quality, an effect that becomes more pronounced as b grows. In particular, with $b = 256$, Exquisitor returns significantly better results than Blackthorn. The reason is that by assigning the b relevant clusters to $S_c = 16$ segments, Exquisitor is able to emphasize the bi-modal media items as explained in Section 3.3. Note that as further clusters are added with Exquisitor ($b = 512$ and beyond), the results become more and more similar to the Blackthorn results.

Figure 5, on the other hand, shows the latency per interaction round. The figure again shows the two Exquisitor variants, with $S_c = 1$ and $S_c = 16$; in both cases, one worker is used, $w = 1$. For comparison, as before, it also shows the average latency for Blackthorn (with 16 CPU cores). Unsurprisingly, Figure 5 shows linear growth in latency with respect to b (recall the logarithmic x-axis). With $b = 256$, each interaction round takes less than 0.3 seconds with $S_c = 16$, and about 0.17 seconds with $S_c = 1$. Both clearly allow for interactive performance; the remainder of our experiments focus on $b = 256$. If even shorter latency is desired, however, fewer clusters can be read: $b = 32$, for example, also gives a good tradeoff between latency and result quality. This latency is produced using only a single CPU core, meaning that the latency is ∼4x better than Blackthorn, with 16x fewer computing cores, for an improvement of ∼64x, or nearly two orders of magnitude. With this knowledge we see b as a parameter that is determined by collection size and the task a user is dealing with, but, as a general starting point we recommend $b = 256$ for large collections.

Comparison: Figure 6 shows the tradeoff between result quality, measured by average precision across 10 rounds of interaction, and the average latency required to produce the suggestions in each round. For Exquisitor, the figure essentially summarizes Figures 4 and 5. For kNN+eCP, the dots represent the same b parameter values, while for the LSH-based approaches a variety of parameter values are represented. The figure clearly demonstrates that Exquisitor is the best approach in both precision and response time compared to all the baseline approaches, achieving better precision than Blackthorn, requiring less than 0.3 seconds compared to Blackthorn's 1.2 seconds. Both k-NN-bases approaches get

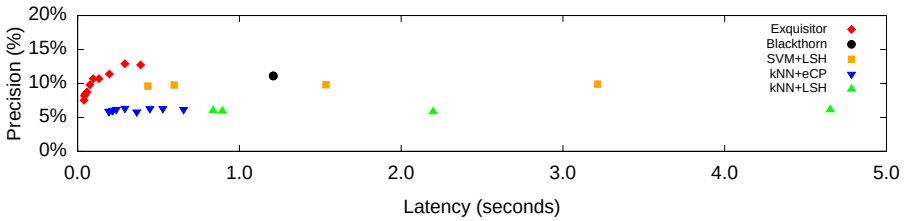

Fig. 6: Average precision vs. latency over 10 rounds of analysis across all YFCC100M actors. Exquisitor, kNN+eCP: $b = 1 - 512$. LSH: $L = 10$, $B = [2^{10}, 2^{18}]$, $p = [15, 40]$.

stuck at 6% which is to be expected since the k-NN query narrows down the scope of the search making it impossible to get out of local optima. SVM+LSH performs better, with precision nearly as good as Blackthorn and response time close to Exquisitor. Overall, however, Exquisitor performs better partly due to being able to utilize the SVM during cluster selection with k-FN queries, and partly due to the cluster segments allowing better multi-modal results.

5 Conclusions

In this paper, we presented Exquisitor, a new approach for exploratory analysis of very large image collections with modest computational requirements. Exquisitor combines state-of-the-art large-scale interactive learning with a new cluster-based retrieval mechanism, enhancing the relevance capabilities of interactive learning by exploiting the inherent structure of the data. Through experiments conducted on YFCC100M, the largest publicly available multimedia collection, Exquisitor achieves higher precision and lower latency, with less computational resources. Additionally, through a modified zero-shot learning experiment on ImageNet, we determine the Exquisitor approach to be excellent at solving cumbersome classification tasks. Exquisitor also introduces customizability that is, to the best of our knowledge, previously unseen in large-scale interactive learning by: (i) allowing a tradeoff between low latency (few clusters) and high quality (many clusters); and (ii) combatting data skew by omitting huge (and thus likely nondescript) clusters from consideration. Exquisitor has recently been used successfully in interactive media retrieval competitions such as the Lifelog Search Challenge [21] and Video Browser Showdown [19]. In conclusion, Exquisitor provides excellent performance on very large collections while being efficient enough to bring large-scale multimedia analytics to standard desktops and laptops, and even high-end mobile devices.

Acknowledgments: This work was supported by a PhD grant from the IT University of Copenhagen and by the European Regional Development Fund (project Robotics for Industry 4.0, CZ.02.1.01/0.0/0.0/15 003/0000470).

References

1. Allan, J.: Incremental relevance feedback for information filtering. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 270–278. ACM, New York, NY, USA (1996)
2. Andoni, A., Indyk, P.: Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In: Proceedings of the IEEE Symposium on the Foundations of Computer Science. pp. 459–468. IEEE Computer Society, Berkeley, CA, USA (2006)
3. Andoni, A., Indyk, P., Laarhoven, T., Razenshteyn, I., Schmidt, L.: Practical and optimal lsh for angular distance. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 1225–1233. Curran Associates, Inc. (2015)
4. Babenko, A., Lempitsky, V.S.: The inverted multi-index. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(6), 1247–1260 (2015)
5. Basri, R., Hassner, T., Zelnik-Manor, L.: Approximate nearest subspace search. IEEE Trans. Pattern Anal. Mach. Intell. **33**(2), 266–278 (2011)
6. Choi, J., Hauff, C., Laere, O.V., Thomee, B.: The placing task at mediaeval 2015. In: Proceedings of the MediaEval 2015 Workshop. CEUR, Wurzen, Germany (2015)
7. Curtin, R.R., Gardner, A.B.: Fast approximate furthest neighbors with data-dependent candidate selection. In: Proc. SISAP. pp. 221–235. Springer, Tokyo, Japan (2016)
8. Datar, M., Immorlica, N., Indyk, P., Mirrokni, V.S.: Locality-sensitive hashing scheme based on p-stable distributions. In: Proc. ACM Symposium on Computational Geometry. pp. 253–262. ACM, Brooklyn, NY, USA (2004)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255 (2009)
10. Ge, T., He, K., Ke, Q., Sun, J.: Optimized product quantization. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(4), 744–755 (2014)
11. Gudmundsson, G.Þ., Amsaleg, L., Jónsson, B.Þ.: Impact of storage technology on the efficiency of cluster-based high-dimensional index creation. In: Proc. International Conference on Database Systems for Advanced Applications (DASFAA). pp. 53–64. Springer, Busan, South Korea (2012)
12. Gudmundsson, G.Þ., Jónsson, B.Þ., Amsaleg, L.: A large-scale performance study of cluster-based high-dimensional indexing. In: Proc. International Workshop on Very-large-scale Multimedia Corpus, Mining and Retrieval (VLS-MCMR). pp. 31–36. ACM, Firenze, Italy (2010)
13. Hansen, C., Hansen, C., Simonsen, J.G., Alstrup, S., Lioma, C.: Unsupervised neural generative semantic hashing. In: Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 735–744. SIGIR'19, ACM, New York, NY, USA (2019)
14. Heo, J., Lin, Z., Yoon, S.: Distance encoded product quantization. In: Proceedings of the IEEE International Conference on Computer Vision & Pattern Recognition. pp. 2139–2146. IEEE Computer Society, Columbus, OH, USA (2014)
15. Huang, T., Dagli, C., Rajaram, S., Chang, E., Mandel, M., Poliner, G.E., Ellis, D.: Active learning for interactive multimedia retrieval. Proc. IEEE **96**(4), 648–667 (2008)

16. Iwayama, M.: Relevance feedback with a small number of relevance judgements: Incremental relevance feedback vs. document clustering. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 10–16. ACM, New York, NY, USA (2000)
17. Jégou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(1), 117–128 (2011)
18. Joachims, T.: A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning. pp. 143–151. ICML '97, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1997)
19. Jónsson, B.Þ., Khan, O.S., Koelma, D.C., Rudinac, S., Worring, M., Zahálka, J.: Exquisitor at the video browser showdown 2020. In: International Conference on Multimedia Modeling. pp. 796–802. Springer (2020)
20. Jónsson, B.Þ., Worring, M., Zahálka, J., Rudinac, S., Amsaleg, L.: Ten research questions for scalable multimedia analytics. In: International Conference on Multimedia Modeling. pp. 290–302. Springer (2016)
21. Khan, O.S., Jónsson, B.Þ., Zahálka, J., Rudinac, S., Worring, M.: Exquisitor at the lifelog search challenge 2019. In: Proceedings of the ACM Workshop on Lifelog Search Challenge. pp. 7–11. ACM (2019)
22. Kovashka, A., Parikh, D., Grauman, K.: Whittlesearch: Interactive image search with relative attribute feedback. International Journal of Computer Vision **115**(2), 185–210 (2015)
23. Lan, M., Tan, C.L., Su, J., Lu, Y.: Supervised and traditional term weighting methods for automatic text categorization. IEEE transactions on pattern analysis and machine intelligence **31**(4), 721–735 (2008)
24. Larson, M., Soleymani, M., Serdyukov, P., Rudinac, S., Wartena, C., Murdock, V., Friedland, G., Ordelman, R., Jones, G.J.F.: Automatic tagging and geotagging in video collections and communities. In: Proceedings of the 1st ACM International Conference on Multimedia Retrieval. pp. 51:1–51:8. ACM, New York, NY, USA (2011)
25. Lejsek, H., Ásmundsson, F.H., Jónsson, B.Þ., Amsaleg, L.: NV-Tree: An efficient disk-based index for approximate search in very large high-dimensional collections. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(5), 869–883 (2009)
26. Lejsek, H., Jónsson, B.Þ., Amsaleg, L.: NV-Tree: nearest neighbors at the billion scale. In: Proceedings of the ACM International Conference on Multimedia Retrieval. ACM, Trento, Italy (2011)
27. Lewis, D.D., Schapire, R.E., Callan, J.P., Papka, R.: Training algorithms for linear text classifiers. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 298–306. SIGIR '96, ACM, New York, NY, USA (1996)
28. Lokoč, J., Bailer, W., Schoeffmann, K., Muenzer, B., Awad, G.: On influential trends in interactive video retrieval: Video browser showdown 2015–2017. IEEE Transactions on Multimedia **20**(12), 3361–3376 (2018)
29. Lu, X., Zhu, L., Cheng, Z., Nie, L., Zhang, H.: Online multi-modal hashing with dynamic query-adaption. In: Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 715–724. ACM, New York, NY, USA (2019)

30. Lv, Q., Josephson, W., Wang, Z., Charikar, M., Li, K.: Multi-probe lsh: efficient indexing for high-dimensional similarity search. In: Proceedings of the 33rd international conference on Very large data bases. pp. 950–961. VLDB Endowment (2007)
31. Mironică, I., Ionescu, B., Uijlings, J., Sebe, N.: Fisher kernel temporal variation-based relevance feedback for video retrieval. Computer Vision and Image Understanding **143**, 38–51 (2016)
32. Pagh, R., Silvestri, F., Sivertsen, J., Skala, M.: Approximate furthest neighbor with application to annulus query. Inf. Syst. **64**, 152–162 (2017)
33. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010)
34. Robertson, S.E., Spärck Jones, K.: Simple, proven approaches to text retrieval. Tech. rep., University of Cambridge, Computer Laboratory (1994)
35. Rocchio, J.J.: Relevance feedback in information retrieval. Tech. rep., University of Harvard, Computer Laboratory (1965)
36. Rui, Y., Huang, T.S., Mehrotra, S.: Content-based image retrieval with relevance feedback in MARS. In: Proc. International Conference on Image Processing (ICIP). pp. 815–818. IEEE Computer Society, Santa Barbara, CA, USA (1997)
37. Rui, Y., Huang, T.S., Mehrotra, S.: Content-based image retrieval with relevance feedback in mars. In: Proceedings of International Conference on Image Processing. vol. 2, pp. 815–818. IEEE (1997)
38. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (Dec 2015)
39. Schoeffmann, K., Bailer, W., Gurrin, C., Awad, G., Lokoč, J.: Interactive video search: Where is the user in the age of deep learning? In: Proc ACM Multimedia. pp. 2101–2103. ACM, Seoul, Republic of Korea (2018)
40. Sigurðardóttir, R., Hauksson, H., Jónsson, B.Þ., Amsaleg, L.: The quality vs. time tradeoff for approximate image descriptor search. In: Proc. IEEE EMMA workshop. IEEE, Tokyo, Japan (2005)
41. Snoek, C., Worring, M., de Rooij, O., van de Sande, K., Yan, R., Hauptmann, A.: Videolympics: Real-time evaluation of multimedia retrieval systems. IEEE MM **15**(1), 86–91 (2008)
42. Sun, C., Song, X., Feng, F., Zhao, W.X., Zhang, H., Nie, L.: Supervised hierarchical cross-modal hashing. In: Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 725–734. ACM, New York, NY, USA (2019)
43. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proc. IEEE CVPR. pp. 1–9. IEEE Computer Society, Boston, MA, USA (2015)
44. Tavenard, R., Jégou, H., Amsaleg, L.: Balancing clusters to reduce response time variability in large scale image search. In: International Workshop on Content-Based Multimedia Indexing. IEEE, Madrid, Spain (2011)
45. Vijayanarasimhan, S., Jain, P., Grauman, K.: Hashing hyperplane queries to near points with applications to large-scale active learning. IEEE Trans. Pattern Anal. Mach. Intell. **36**(2), 276–288 (2014)
46. Xu, X., Bao, J., Yao, B., Zhou, J., Tang, F., Guo, M., Xu, J.: Reverse furthest neighbors query in road networks. J. Comput. Sci. Technol. **32**(1), 155–167 (2017)

47. Zahálka, J., Worring, M.: Towards interactive, intelligent, and integrated multimedia analytics. In: Proc. of the IEEE Conference on Visual Analytics Science and Technology (VAST). pp. 3–12. Paris, France (2014)
48. Zahálka, J., Rudinac, S., Jónsson, B.T., Koelma, D.C., Worring, M.: Blackthorn: Large-scale interactive multimodal learning. IEEE Transactions on Multimedia **20**(3), 687–698 (2018)