Similarity Adaptation in an Exploratory Retrieval Scenario

Sebastian Stober and Andreas Nürnberger

Data & Knowledge Engineering Group
Faculty of Computer Science
Otto-von-Guericke-University Magdeburg, D-39106 Magdeburg, Germany
{Sebastian.Stober,Andreas.Nuernberger}@ovgu.de

Abstract. Sometimes users of a multimedia retrieval system are not able to explicitly state their information need. They rather want to browse a collection in order to get an overview and to discover interesting content. Exploratory retrieval tools support users in search scenarios where the retrieval goal cannot be stated explicitly as a query or user rather want to browse a collection in order to get an overview and to discover interesting content. In previous work, we have presented Adaptive SpringLens – an interactive visualization technique building upon popular neighborhood-preserving projections of multimedia collections. It uses a complex multi-focus fish-eye distortion of a projection to visualize neighborhood that is automatically adapted to the user's current focus of interest. This paper investigates how far knowledge about the retrieval task collected during interaction can be used to adapt the underlying similarity measure that defines the neighborhoods.

1 Introduction

Growing collections of multimedia data such as images and music require new approaches for exploring a collection's contents. A lot of research in the field of multimedia information retrieval focuses on queries posed as text, by example (e.g. query by humming and query by visual example) as well as automatic tagging and categorization. These approaches, however, have a major drawback – they require the user to be able to formulate a query which can be difficult when the retrieval goal cannot be clearly defined. Finding photos that nicely outline your latest vacation for a presentation to your friends is such a retrieval goal and underlining the presentation by a suitable background music cannot be done with query by example. In such a case, exploratory retrieval systems can help by providing an overview of the collection and let the user decide which regions to explore further.

When it comes to get an overview of a collection, neighborhood-preserving projection techniques have become increasingly popular. Beforehand, the objects to be projected have to be analyzed to extract a set of descriptive features. (Alternatively, feature information may also be annotated manually or collected from external sources.) Based on these features, the objects can be

M. Detyniecki et al. (Eds.): AMR 2010, LNCS 6817, pp. 144–158, 2012.

Fig. 1. Galaxy user-interface visualizing a photo collection with an object marked green in primary focus and two objects in secondary focus. (color scheme inverted for better printing).

compared – or more specifically: appropriate distance- or similarity measures can be defined. The general objective of the projection can then be paraphrased as follows: Arrange the objects (on the display) in such a way that neighboring objects are very similar and the similarity decreases with increasing object distance (on the display). As the feature space of the objects to be projected usually has far more dimensions than the display space, the projection inevitably causes some loss of information – irrespective of which dimensionality reduction techniques is applied. Consequently, this leads to a distorted display of the neighborhoods such that some objects will appear closer than they actually are, and on the other hand some objects that are distant in the projection may in fact be neighbors in feature space.

In previous work [10,13], we have developed an interface for exploring image and music collections using a galaxy metaphor that addresses this problem of distorted neighborhoods. Figure 1 shows a screenshot of the interface visualizing a photo collection. Each object is displayed as a star (i.e. a point) with its brightness and (to some extend) its hue depending on a predefined importance measure – e.g. a (user) rating or a view / play count. A spatially well distributed subset of the collection (specified by filters) is additionally displayed as a small image (a thumbnail or album cover respectively) for orientation. The arrangement of the stars is computed using multi-dimensional scaling (MDS) [5] relying on a set of descriptive features to be extracted beforehand. (Alternatively, feature

information may also be annotated manually or collected from external sources.) MDS is a popular neighborhood-preserving projection technique that attempts to preserve the distances (dissimilarities) between the objects in the projection. The result of the MDS is optimal w.r.t. the minimization of the overall distance distortions. Thus, fixing one distorted neighborhood is not possible without damaging others. However, if the user shows interest in a specific neighborhood, this one can get a higher priority and be temporarily fixed (to some extend) at the cost of the other neighborhoods. To this end, an adaptive distortion technique called SpringLens [4] is applied that is guided by the user's focus of interest. The SpringLens is a complex overlay of multiple fish-eye lenses divided into primary and secondary focus. The primary focus is a single large fish-eye lens used to zoom into regions of interest compacting the surrounding space but not hiding it from the user to preserve overview. While the user can control the primary focus, the secondary focus is automatically adapted. It consists of a varying number of smaller fish-eye lenses. When the primary focus changes, a neighbor index is queried with the object closest to the center of focus. If nearest neighbors are returned that are not in the primary focus, secondary lenses are added at the respective positions. As a result, the overall distortion of the visualization temporarily brings the distant nearest neighbors back closer to the focused region of interest. This way, distorted distances introduced by the projection can to some extend be compensated.

The user-interface has been evaluated in a study as reported in [9]. In the study, 30 participants had to solve an exploratory image retrieval task: Each participant was asked to find representative images for five non-overlapping topics in a collection containing 350 photographs. This was repeated on three different collections – each one with different topics (and with varying possibilities for interaction). The evaluation showed that the participants indeed frequently used the secondary focus to find other photos belonging to the same topic as the one in primary focus. However, some photos in secondary focus did not belong to the same topic. Thus, this paper aims to answer the question whether it is possible to automatically adapt the neighborhood index during the exploratory search process to return more relevant photos for the primary focus topic.

The remaining paper is structured as follows: Section 2 outlines the experimental setup comprising the datasets, features and the definition of the distance facets. The adaptation method is covered by Section 3. The experiments are described in Sections 4 to 6. Section 7 draws conclusions.

2 Experimental Setup

2.1 Dataset

Four image collection were used during the study of which the first one (Melbourne & Victoria) is not considered here because it was only used for the introduction of the user-interface and has no topic annotations. All collections

Table 1. Annotated Photo collections and topics used in the user study

collection	topics (number of images)
Barcelona	Tibidabo (12), Sagrada Família (31), Stone Hallway in Park Güell (13), Beach & Sea (29), Casa Milà (16)
Japan	Owls (10), Torii (8), Paintings (8), Osaka Aquarium (19), Traditional Clothing (35)
Western Australia	Lizards (17), Aboriginal Art (9), Plants (Macro) (17), Birds (21), Ningaloo Reef (19)

were drawn from a personal photo collection of the authors.[1] Each annotated collection comprises 350 images scaled down to fit 600x600 pixels – each one belonging to at most one of five non-overlapping topics. Table 1 shows the topics for each collection. In total, 264 of the 1050 images belong to one of the 15 topics.

2.2 Features

For all images the MPEG-7 visual descriptors EdgeHistogram (EHD), ScalableColor (SCD) and ColorLayout (CLD) [8] were extracted using the Java implementation provided by the Caliph&Emir MPEG-7 photo annotation and retrieval framework [6].

The EHD captures spatial distributions of edges in an image. The images in divided into 4×4 sub-images. Using standard edge detection methods, the following 5 edge types are detected: vertical, horizontal, 45°, 135° and nondirectional edges. The frequency of these edge types is stored for each sub-image resulting in 16×5 local histogram bins. Further, a global-edge histogram (5 bins) and 13 semiglobal-edge histograms (13×5 bins) are directly computed from the local bins. The 13 semiglobal-edge histograms are obtained through grouping 4 vertical sub-images (4 columns), 4 horizontal sub-images (4 rows) and 4 neighbor sub-images (5 (2×2)-neighborhoods).

The SCD is based on a color histogram in the HSV color space with a fixed color space quantization. Coefficients are encoded using a Haar transform to increase the storage efficiency. Here, we use 64 coefficients which is equivalent to 8 bins for the hue (H) component and 2 bins each for the saturation (S) and the value (V) in the HSV histogram.

The CLD is also based on color histograms but describes localized color distributions of an image. The image is partitioned into 8x8 blocks and the average color is extracted on each block. The resulting iconic 8x8 "pixel" representation of the image is expressed in YCrCb color space. Each of the the components (Y, Cr, Cb) is transformed by an 8x8 discrete cosine transform (DCT). Finally, the DCT coefficients are quantized and zigzag-scanned. A number of low-frequency coefficients of each color plane is selected beginning with the DC coefficient.

[1] The collections and topic annotations are publicly available under the Creative Commons Attribution-Noncommercial-Share Alike license, `http://creativecommons.org/licenses/by-nc-sa/3.0/` – please contact `sebastian.stober@ovgu.de`

Those coefficients form the descriptor (we obtain 3 different feature vectors – one for each color component – by concatenating the coefficients). We have chosen the recommended setting of 6, 3, 3 for the Y, Cr, Cb coefficients respectively.

2.3 Distance Computation

Facet Definition. Based on the features associated with the images, *facets* are defined that refer to different aspects of visual (dis-) similarity:

Definition 1. *Given a set of features F, let S be the space determined by the feature values for a set of images I. A facet f is defined by a facet distance measure δ_f on a subspace $S_f \subseteq S$ of the feature space, where δ_f satisfies the following conditions for any $x, y \in I$:*

- $\delta(x, y) \geq 0$ *and* $\delta(x, y) = 0$ *if and only if* $x = y$
- $\delta(x, y) = \delta(y, x)$ *(symmetry)*

Optionally, δ is a distance metric if it additionally obeys the triangle inequality for any $x, y, z \in I$:

- $\delta(x, z) \leq \delta(x, y) + \delta(y, z)$ *(triangle inequality)*

Specifically, during the study, three facets were used – each one referring to a single one of the above MPEG-7 features in combination with the respective distance measure proposed in the MPEG-7 standard:

For comparing two CLDs ($\{Y^{(a)}, Cr^{(a)}, Cb^{(a)}\}$ and $\{Y^{(b)}, Cr^{(b)}, Cb^{(b)}\}$), the sum of the (weighted) euclidean color component distances is computed [7]:

$$\begin{aligned}\delta_{CLD}(a,b) = &\sqrt{\sum_i w_{yi}(Y_i^{(a)} - Y_i^{(b)})^2} + \sqrt{\sum_k w_{rk}(Cr_k^{(a)} - Cr_k^{(b)})^2} \\ &+ \sqrt{\sum_k w_{bk}(Cb_k^{(a)} - Cb_k^{(b)})^2}\end{aligned} \tag{1}$$

where Y_i, represent the ith luminance coefficient and Cr_k, Cb_k the kth chrominance coefficient. The distances are weighted appropriately, with larger weights given to the lower frequency components.[2]

The proposed distance for the SCDs of images a and b is the l_1-norm on the coefficients $(c_1^{(a)}, \ldots, c_n^{(a)})$ $c_1^{(b)}, \ldots, c_n^{(b)})$ in the Haar-transformed histogram domain:

$$\delta_{SCD}(a,b) = \sum_{i=1}^{n} \left| c_i^{(a)} - c_i^{(b)} \right| \tag{2}$$

For the EHDs, the l_1-norm is used as well to compare the images a and b:

$$\delta_{EHD}(a,b) = \sum_{i=1}^{80} \left| h_i^{(a)} - h_i^{(b)} \right| + 5 \times \sum_{i=1}^{5} \left| g_i^{(a)} - g_i^{(b)} \right| + \sum_{i=1}^{65} \left| s_i^{(a)} - s_i^{(b)} \right| \tag{3}$$

[2] Note that the CLD component weights are applied to compute the distance for the CLD facet and thus are part of the facet's definition in contrast to the facet distance weights defined below that are used for the aggregation of different facets.

Here the h_i refer to the 5×16 histogram bin values, g_i to the 1×5 global-edge histogram bins (weighted by factor 5) and s_i to 13×5 semiglobal-edge histograms. All bin values are normalized.

Facet Distance Normalization. In order to be able to aggregate values from several facet distance measures, the following normalization is applied for all distance values $\delta_f(x, y)$ of a facet f:

$$\delta'_f(a, b) = \min\left\{1, \frac{\delta_f(a, b)}{\mu + \sigma}\right\} \tag{4}$$

where μ is the mean

$$\mu = \frac{1}{|\{(x, y) \in I^2\}|} \sum_{(x,y)\in I^2} \delta_f(x, y) \tag{5}$$

and σ is the standard deviation

$$\sigma = \sqrt{\frac{1}{|\{(x, y) \in I^2\}|} \sum_{(x,y)\in I^2} (\delta_f(x, y) - \mu)^2} \tag{6}$$

of all distance values with respect to δ_f. This truncates very high distance values and results in a value range of $[0, 1]$.

Facet Distance Aggregation. In order to computer the distance between images $a, b \in I$ w.r.t. to the facets $f_1, \ldots, f_l$, the individual facet distances $\delta_{f_1}(a, b), \ldots, \delta_{f_l}(a, b)$ need to be aggregated. Here, we use the weighted sum:

$$d(a, b) = \sum_{i=1}^{l} w_i \delta_{f_i}(a, b) \tag{7}$$

which is a very common weighted aggregation function that allows to control the importance of the facets $f_1, \ldots, f_l$ through their associated weights $w_1, \ldots, w_l$. Per default, all weights are initialized as $\frac{1}{l}$, i.e. considering all facets equally important.

3 Adaptation Method

Changing the weights of the facet distance aggregation function described in the previous section allows to adapt the distance computation to a specific retrieval task. This can already be done manually – e.g., using the slider widgets (hidden on a collapsible panel) in the graphical user interface. However, it is often hard to do this explicitly. Several metric learning methods have already been proposed that aim to do this automatically: Generally, the first step is to gather preference information – either by analyzing information created by the user such as already labeled objects or manual classification hierarchies (e.g. documents in

a folder structure) [1] or alternatively by interpreting user actions such as rearrangement of objects in a visualization [12], changing cluster assignments [1], sorting a result list [11] or directly giving similarity judgments [2]. In the second step, the gathered preference information is turned into similarity constraints. These constraints are used finally to guide an optimization algorithm that aims to identify weights that violate as few constraints as possible. At this point, several possibilities exists: E.g., [12] describes a quadratic optimization approach that is deterministic and has the advantage of gradual and more stable weight changes and non-negative, bounded weights. However, it cannot deal with constraint violations. The approaches presented in [1,2] rely on gradient descent and ensemble perceptron learning instead. These methods allow constraint violation but may cause drastic weight changes. Further, they do not limit the value range of the weights which can however be achieved by modifications of the gradient descent update rule as proposed in [2].

In this paper, we interpret the problem of adapting the distance measure as a classification problem as proposed in [2]: The required preference information is deduced from the topic annotation already made by the user. We assume that images of the same topic are visually similar and that the respective visual features are covered appropriately by the facets introduced in the preceding section. For any pair of images a and b annotated with the same topic T, we can demand that they are more similar (or have a smaller distance) to each other than to any other image c not belonging to T:

$$d(a,b) < d(a,c) \quad \forall (a,b,c) | a,b \in T \wedge c \notin T \tag{8}$$

where d is the aggregated distance function defined in Equation 7. This can be rewritten as:

$$\sum_{i=1}^{l} w_i(\delta_{f_i}(a,c) - \delta_{f_i}(a,b)) = \sum_{i=1}^{l} w_i x_i > 0 \tag{9}$$

with $x_i = \delta_{f_i}(a,c) - \delta_{f_i}(a,b)$. Using the positive example $(x, +1)$ and the negative example $(-x, -1)$ to train a binary classifier, the weights $w_1, \ldots, w_l$ define the model (hyperplane) of the classification problem. This way, basically any binary classifier could be used here. We apply the linear support vector machine algorithm as provided by LIBLINEAR [3] that is faster and generates better results than the gradient descent approach used initially. However, with this approach, a valid value range for the weights cannot be enforced. Specifically, weights can become negative. We added artificial training examples that require positive weights (setting a single x_i to one at a time and the others to zero), but these constraints can still be violated.

4 Experiment 1: Assessing the Potential for Adaptation

The first question to be answered is: Given all knowledge about the topic assignments, how much can the performance be improved through adaptation of the facet weights? This gives us an estimation of the "ceiling" for the adaption

in simulation or application in real world. We consider the following three levels of adaptation:

1. **Topic-specific adaptation (`Topic`):** This is the most general form of adaptation. For each photo of the topic, a ranking of all photos in the collection is considered and constraints are derived that require images of the same topic to be ranked higher than others. The facet weights are then learned subject to the constraints from all rankings. This results in the highest number of constraints.
2. **Query-specific adaptation considering all photos from the topic (`Query_All`):** Here, facets weights are not adapted per topic but per query. To this end, only the single ranking for the query and the derived constraints are considered.
3. **Query-specific adaptation considering only the 5 nearest neighbors from the topic (`Query_5NN`):** This is a variation of the previous case with the difference that here only the 5 nearest neighbors from the topic are considered relevant instead of all images of the topic. This is the most specific adaptation with the lowest number of constraints.

In order to assess the retrieval performance, precision and recall were computed using each image that belongs to a topic as single query. Figure 2 shows the averaged recall-precision curves per topic for the three adaptation levels and the baseline (no adaptation). Retrieval performance varies a lot from topic to topic: For topics with high diversity that have several sub-clusters of similar images (e.g., "Sagrada Familia", "Lizards", "Birds"), it tends to be much worse than for rather homogeneous topics with only few outliers (e.g., "Stone Hallway...", "Owls", "Ningaloo Reef"). Generally, an improvement over the baseline can be observed but it is mostly marginal. This indicates that either the facets are unsuitable to differentiate relevant from irrelevant images or the small number of facets does not provide enough degrees of freedom for the adaptation.

`Query_5NN` provides the best adaptation in the low recall area whereas at higher recall `Query_All` performs better. This is not surprising considering the above mentioned diversity and resulting sub-clusters within the topics. The more specific the adaptation the more likely it will consider only neighbors of the same sub-cluster as relevant and thus show superior performance for these images while this overfitting leads to a penalty when trying to find other images of the same topic (in the high recall range). `Topic` only leads to (small) improvements on rather homogeneous topics such as "Paintings" and "Aboriginal Art". For the topics "Torii" and "Tibidabo", its precision is close to zero and significantly below the baseline. These topics were perceived as especially difficult by the participants of the study because they are very divers and share visual similarity only at a higher level of detail. E.g., the vermilion color of the Torii is very remarkable but the respective objects often cover only a small portion of the image.

Summarizing, it can be concluded that this setting does not have much potential for adaptation it thus is unsuitable for a simulation of user-interaction.

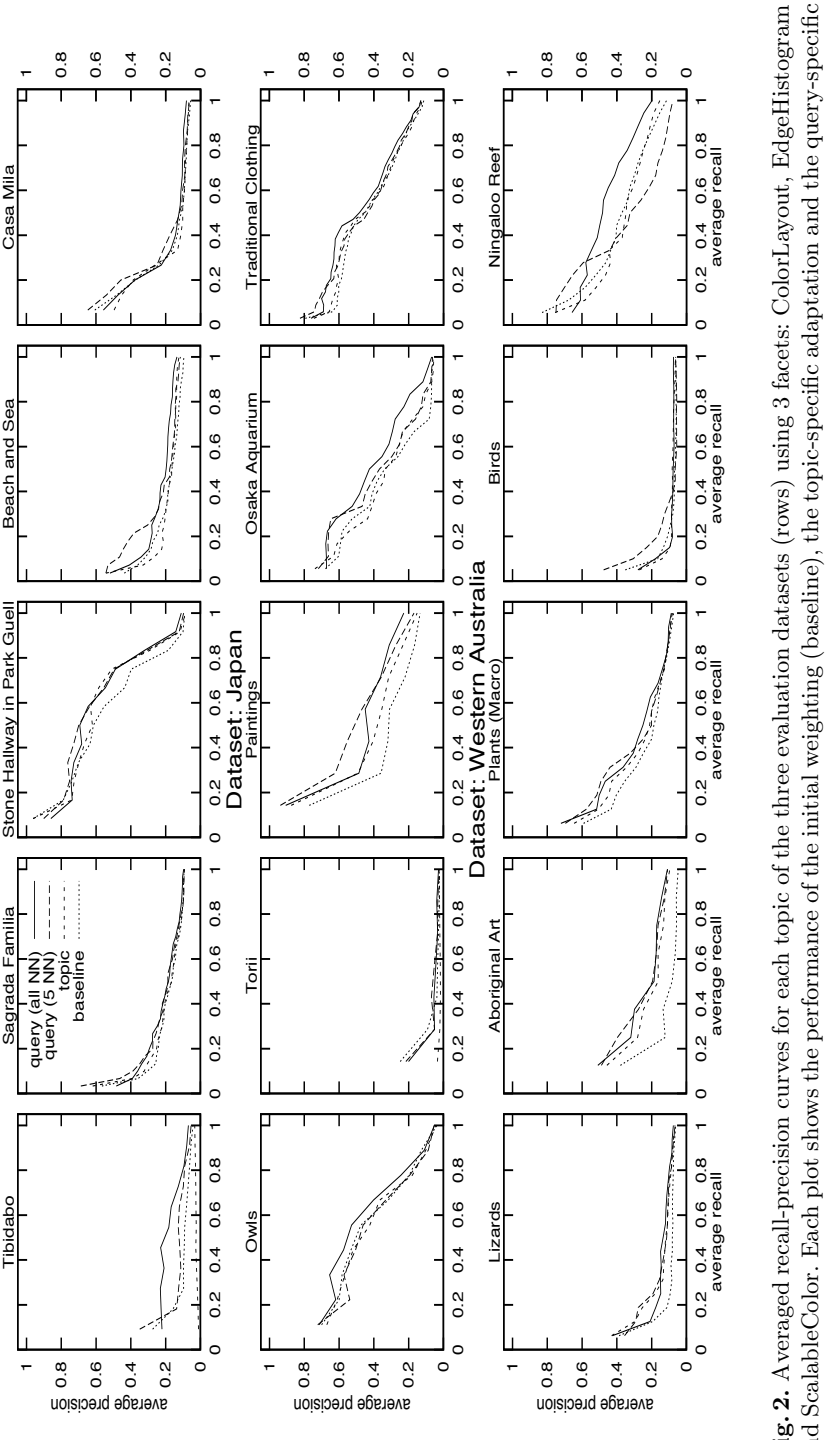

Fig. 2. Averaged recall-precision curves for each topic of the three evaluation datasets (rows) using 3 facets: ColorLayout, EdgeHistogram and ScalableColor. Each plot shows the performance of the initial weighting (baseline), the topic-specific adaptation and the query-specific adaptation (taking all or only the closest five images from the same topic).

The potential could be increased by adding more facets that cover different aspects of visual similarity which may help to differentiate the images of a topic from others. Alternatively, the information covered by the existing facets may in fact be already sufficient for differentiation but the comparison (i.e. the distance computation) takes place at a level that is too high to cover the inner-topic commonalities. For instance, the remarkable color of the Torii is captured by the SCD but in the current setting, we are only able to express that color in general is important for comparison but cannot stress this specific color. Similarly, it would make sense to emphasize vertical edges in the EHD for the Sagrada Família which is not possible either. In order to make such fine-grain adaptations possible, the respective sub-features currently hidden within the (high-level) facets need to be made visible for adaptation by becoming (low-level) facets themselves. This is covered by the next section.

5 Experiment 2: Extending the Number of Facets

As proposed in the previous section, this experiment investigates how the potential for adaptation can be increased by replacing a high-level facet by a set of low-level facets. Recall that a facet is defined by a set of features *and* a distance measure. Thus, in order to decompose a facet into sub-facets, it is not sufficient to just identify suitable subsets of the feature set (possibly splitting a feature further into sub-features). More importantly, it is also necessary to define appropriate distance measures that work on the feature subsets in a way that preserves the semantics of the original distance measure during aggregation by linear combination.

In the following, we consider the SCD as representative example for decomposing a facet. (For the EHD and CLD similar transformations can be done analogously using the histogram bins or the three coefficient vectors respectively as sub-features.) We choose the SCD because a finer differentiation in the color domain appears to be promising to increase performance on some of the difficult topics such as "Torii". The decomposition of the SCD is very straightforward as its distance measure (Equation 2) is itself an (equally) weighted sum of per-coefficient distances. The SCD-facet can therefore simply be replaced by 64 SCD-coefficient-facets – each one considering a different coefficient and using the absolute value difference as facet distance measure. As a result, the adaptation algorithm has now 63 more degrees of freedom. Figure 3 shows the performance for running the experiment described in the previous section in this modified setting. The performance improvement is now clearly evident throughout all topic for all three adaptation levels. `Query_All` outperforms the others significantly, often achieving maximum precision in the low and middle recall range. `Query_5NN` does well on the first ranks but its precision rapidly declines afterwards which is a nice indicator for the overfitting that takes place here. `Topic` lies somewhere in between the baseline and `Query_All` – except for "Osaka Aquarium" where it has almost zero precision. This is somewhat surprising as this topic is rather homogeneous. Examining the topic weights learned by the

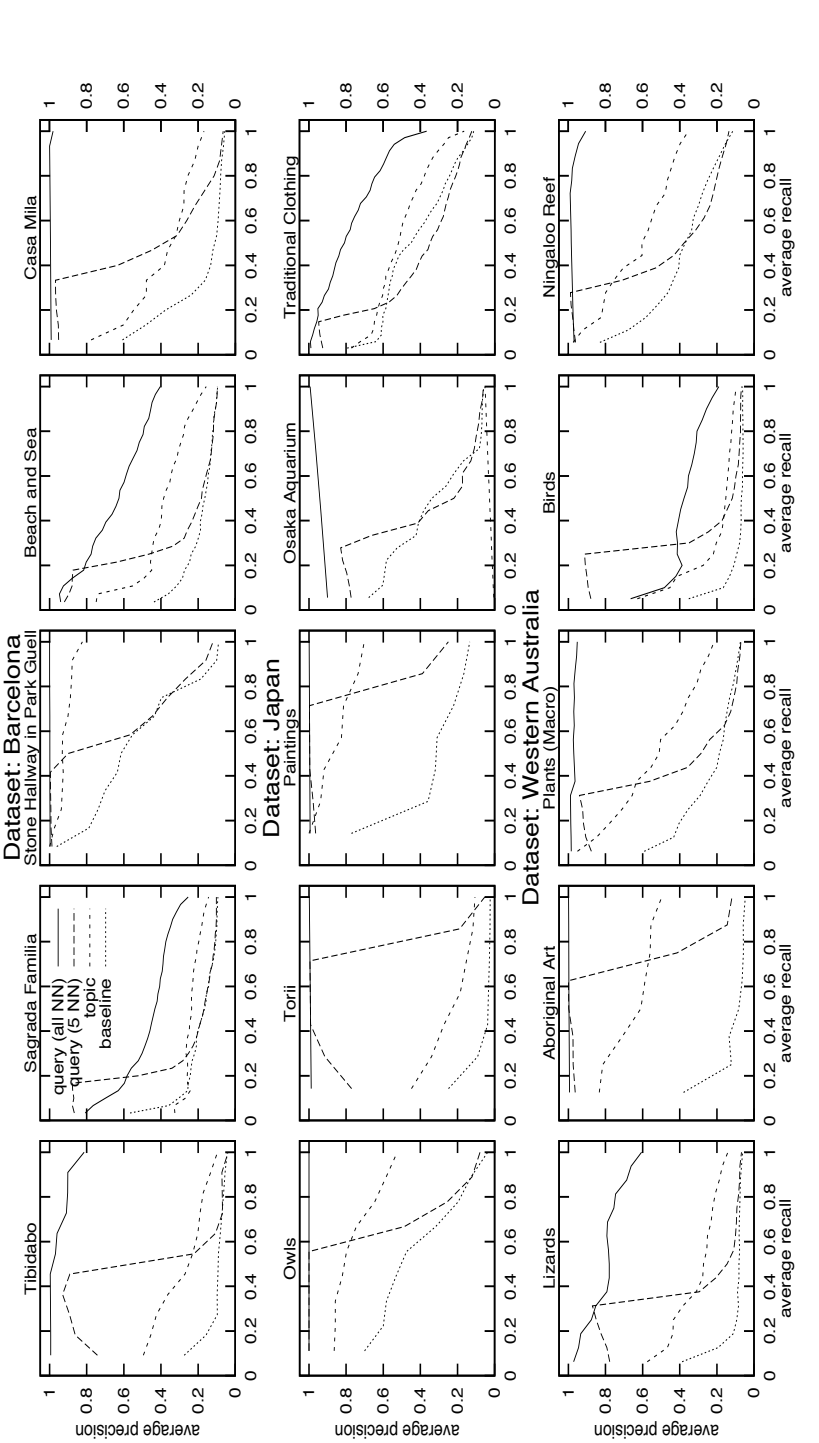

Fig. 3. Averaged recall-precision curves for each topic of the three evaluation datasets (rows) using 66 facets: ColorLayout (CLD), EdgeHistogram (EHD) and 64 bins of ScalableColor (SCD). Each plot shows the performance of the initial weighting (baseline), the topic-specific adaptation and the query-specific adaptation (taking all or only the closest five images from the same topic).

adaptation algorithm reveals that the weight are in the range $[-3.7, 3.6]$ with many negative values. This is caused by a known shortcoming of the adaptation method that cannot prohibit negative weights (see Section 3). Other weightings contain negative weights as well but are less extreme.

6 Experiment 3: Simulated User-Interaction

This experiment aims to simulate user-interaction as observed during the study. The question to be answered is whether an automatic weight adaptation could have helped the user in finding more relevant images for a topic through the secondary focus of the SpringLens. Figure 4 shows the outline of the simulation approach for a single session, i.e. finding five relevant images for a specific topic.

```
simulateSession(seed image seed, relevant images RELEVANT)
  initialize ANNOTATED ← {seed}
  repeat
    ANNOTATED ← ANNOTATED ∪ findNextImage(ANNOTATED, RELEVANT)
    adapt weights
    evaluate
  until |ANNOTATED| = 5

findNextImage(annotated images ANNOTATED, relevant images RELEVANT)
  // try to find a relevant image in secondary focus
  for all query ∈ ANNOTATED do
    NN ← getKNearestNeighbors(query, 5)
    for all neighbor ∈ NN do
      if neighbor ∈ RELEVANT \ ANNOTATED then
        return neighbor
      end if
    end for
  end for
  // fallback: query with the newest annotated image
  newest ← newestIn(ANNOTATED)
  RANKING ← getKNearestNeighbors(newest, ∞)
  for all neighbor ∈ RANKING do
    if neighbor ∈ RELEVANT \ ANNOTATED then
      return neighbor
    end if
  end for
```

Fig. 4. Outline of the simulation algorithm

A first relevant image is required as *seed* for the simulation. After each additional relevant image, the weights are adapted considering the three levels of adaptivity introduced in Section 4 and evaluated afterwards. The simulated user's strategy to find the next relevant image relies on the secondary focus that contains the five nearest neighbors of the image in primary focus. (This is the same setting as in

the user study.) Directing the primary focus onto already annotated images, the users tries to find another relevant image in secondary focus. If this fails, he looks in the surrounding of the most recently annotated image (*newest*) – a region that is most likely not fully explored yet. This is simulated by going through the full similarity ranking of all images using *newest* as query and picking the first relevant image that has not been annotated yet. (As all topics contain more than five relevant images, this fallback strategy never fails.) To generate the ranking and for finding the five nearest neighbors, the `Query_All` weights are used that performed best in the previous experiments. (For the first query with the *seed*, no adaptation can be made because at least two annotated images are required.)

Figure 5 shows the performance after each iteration averaged over all *seed* images for each topics. W.r.t. the user's retrieval goal – finding five images for each topic – two performance value are of interest: the precisions at rank 5 and the number of new relevant images in secondary focus. The *precisions at rank 5* refers to the portion of relevant images in secondary focus because it contains the five nearest neighbors. Looking only at this value, the adaptation increases performance significantly for `Query_All` and `Query_5NN` – both having identical values. For `Topic`, there is still an improvement in most cases. However, looking at the precision values, it has to be taken into account that with each iteration more relevant images are known to the user – and thus to the adaptation algorithm. It would be easy for an adaptation algorithm to overfit on this information by always returning simply the already annotated images as nearest neighbors. This way, a precision of 4.0 could be easily reached after four iterations. While such an adaptation could help to re-discover images of a topic, it is useless for the considered task of finding new relevant images. This issue is addressed by the *number of new images in secondary focus* performance measure. The values reveal that for the hard topics like "Torii" or "Tibidabo" the adaptation indeed leads to the above described overfitting and does not help much to find new images of the same topic. However, for about two thirds of all topics the adaptation turns out to be helpful as between 1 and 5 new relevant images can be found in the secondary focus. This value naturally decreases with each iteration as previously new images become annotated. Further, overfitting may also be involved to some extend but this cannot be measured.

7 Conclusion

We conducted three experiments to answer the question whether and how much automatic similarity adaptation can help users in an exploratory retrieval scenario where images are to be annotated with topic labels. As visual descriptors the EdgeHistogram, ScalableColor and ColorLayout Descriptors from the MPEG-7 specification were used. Similarity was adapted by changing the weights for several distance facets. The first experiment revealed that the initial setting – weighting three facet (one for each visual descriptor used) did not provide enough degrees of freedom for the adaptation approach. Decomposing the ScalableColorcDescriptor into its bins introduced additional low-level facets and increased

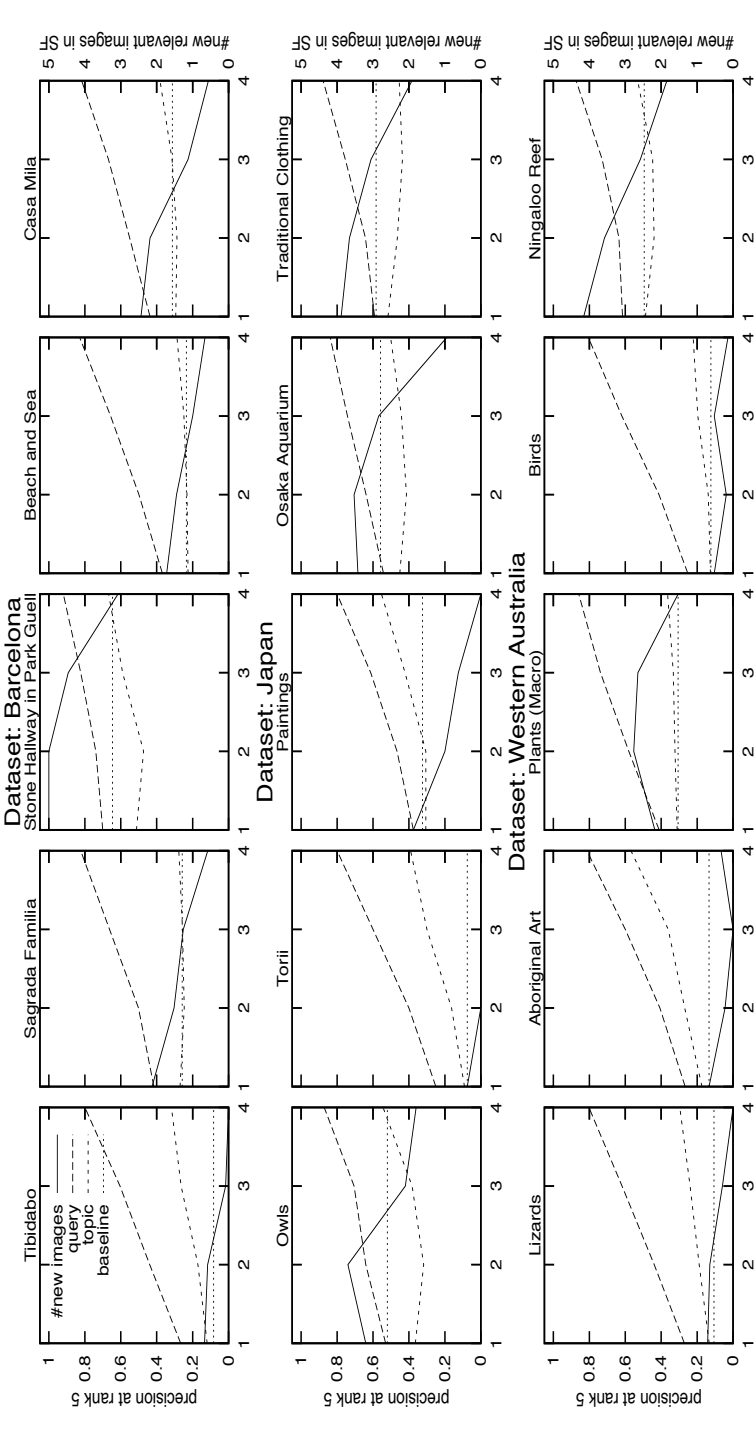

Fig. 5. Change of the performance over the course of the simulation (iterations) for each topic of the three evaluation datasets (rows) using 66 facets. Values are averaged over all possible seed images of the topic. Each plot shows the precision at rank 5 (left y-axis) for the initial weighting (baseline), the topic-specific adaptation and the query-specific adaptation. The solid line shows the average number of new relevant images in secondary focus (right y-axis) for the query-specific adaptation.

the potential for adaptation significantly as shown the second experiment. In the third experiment, user-interaction was simulated and the quality of the adaptation evaluated. It can be concluded the adaptation is useful in the considered retrieval scenario. The proposed decomposition approach is likely to work for other complex feature descriptors beyond those covered here. However, this still needs to be investigated more thoroughly.

Acknowledgments. This work was supported in part by the German National Merit Foundation and the German Research Foundation (DFG).

References

1. Bade, K., Nürnberger, A.: Creating a cluster hierarchy under constraints of a partially known hierarchy. In: Proc. of 8th SIAM Int. Conf. on Data Mining (2008)
2. Cheng, W., Hüllermeier, E.: Learning similarity functions from qualitative feedback. In: Althoff, K.-D., Bergmann, R., Minor, M., Hanft, A. (eds.) ECCBR 2008. LNCS (LNAI), vol. 5239, pp. 120–134. Springer, Heidelberg (2008)
3. Fan, R., Chang, K., Hsieh, C., Wang, X., Lin, C.: Liblinear: A library for large linear classification. The Journal of Machine Learning Research 9, 1871–1874 (2008)
4. Germer, T., Götzelmann, T., Spindler, M., Strothotte, T.: Springlens: Distributed nonlinear magnifications. In: Eurographics 2006 - Short Papers (2006)
5. Kruskal, J., Wish, M.: Multidimensional Scaling. Sage, Thousand Oaks (1986)
6. Lux, M.: Caliph & Emir: MPEG-7 photo annotation and retrieval. In: Proc. of 17th ACM Int. Conf. on Multimedia (2009)
7. Manjunath, B.S., Ohm, J.-R., Vasudevan, V.V., Yamada, A.: Color and texture descriptors. IEEE Transactions on Circuits and Systems for Video Technology 11, 703–715 (1998)
8. Martinez, J., Koenen, R., Pereira, F.: MPEG-7: The generic multimedia content description standard, part 1. IEEE MultiMedia 9(2), 78–87 (2002)
9. Stober, S., Hentschel, C., Nürnberger, A.: Evaluation of adaptive springlens – a multi-focus interface for exploring multimedia collections. In: Proc. of 6th Nordic Conference on Human-Computer Interaction, NordiCHI 2010 (2010)
10. Stober, S., Hentschel, C., Nürnberger, A.: Multi-facet exploration of image collections with an adaptive multi-focus zoomable interface. In: Proc. of 2010 IEEE World Congress on Computational Intelligence (WCCI 2010) (2010)
11. Stober, S., Nürnberger, A.: Towards user-adaptive structuring and organization of music collections. In: Detyniecki, M., Leiner, U., Nürnberger, A. (eds.) AMR 2008. LNCS, vol. 5811, pp. 53–65. Springer, Heidelberg (2010)
12. Nürnberger, A., Stober, S.: User modelling for interactive user-adaptive collection structuring. In: Boujemaa, N., Detyniecki, M., Nürnberger, A. (eds.) AMR 2007. LNCS, vol. 4918, pp. 95–108. Springer, Heidelberg (2008)
13. Stober, S., Nürnberger, A.: Musicgalaxy – an adaptive user-interface for exploratory music retrieval. In: Proc. of 7th Sound and Music Conference, SMC 2010 (2010)