

# A Ground-truth Training set for Hierarchical Clustering in Content-based Image Retrieval

D.P. Huijsmans, N. Sebe and M.S. Lew

LIACS, Leiden University, P.O. Box 9512, 2300 RA Leiden, The Netherlands  
huijsman,nicu,mlew@liacs.nl

**Abstract.** Progress in Content-Based Image Retrieval (CBIR) is hampered by the absence of well-documented and validated test-sets that provide ground-truth for the performance evaluation of image indexing, retrieval and clustering tasks. For quick access to large (tenthousands or millions of images) digital image collections a hierarchically structured indexing or browsing mechanism based on clusters of similar images at various coarse to fine levels is highly wanted. The Leiden 19th-Century Portrait Database (LCPD), that consists of over 16,000 scanned studio portraits (so-called Cartes de Visite CdV), happens to have a clearly delineated set of clusters in the studio logo backside images. Clusters of similar or semantically identical logos can also be formed on a number of levels that show a clear hierarchy. The Leiden Imaging and Multimedia Group is constructing a CD-ROM with a well-documented set of studio portraits and logos that can serve as ground-truth for feature performance evaluation in domains beside color-indexing. Its grey-level image lay-out characteristics are also described by various precalculated feature vector sets. For both portraits (near copy pairs) and studio logos (clusters of identical logos) test-sets will be provided and described at various clustering levels. The statistically significant number of test-set images embedded in a realistically large environment of narrow-domain images are presented to the CBIR community to enable selection of more optimal indexing and retrieval approaches as part of an internationally defined test-set that comprises test-sets specifically designed for color-, texture- and shape retrieval evaluation.

## 1 Introduction

So far a lot of effort in the CBIR community has been put into features, metrics and ranking (for our own effort see for instance [metric98] and [perfHuijsmans97]) but comparatively little effort has been put into performance evaluation (a theoretical example is [Dimai99] and a practical one [HP98]). The main obstacle for application of a sound statistical approach is not the lack of theory (see for instance [DeVijver82]), but missing ground-truth. Validated test sets like the Brodatz textures are hard to find; in most practical studies unvalidated ad hoc test cases are used that miss any ground. As to our knowledge no ground-truth test-set for hierarchical clustering exists in the CBIR domain.

Any content-based image retrieval task comes down to finding the right balance between grouping and separating images in like and unlike clusters. No general applicable way of image similarity clustering can be devised that provides the same answer to different vision tasks; the semantic clustering of human subjects may well overlap badly between different vision tasks and even be conflicting (one persons signal is the other persons noise and vice versa). This means that in a generally applicable image retrieval setting a learning or optimization stage like the one in [Kittler96] will be needed regularly or even on a search or goal image basis. Although the black box neural net approach would be a logic choice in this case, we prefer the use of a statistical approach, leaving most of the controls to us.

Image retrieval user interfaces should therefore present the user with tools to (fine)tune the indexing and retrieval mechanisms whenever the semantics of a task have changed. As databases get larger and larger an hierarchical clustering phase also becomes indispensable. Theory developed in the sixties and seventies of the last century (see [Hartigan75] and [survey83]) are being revived in efforts to visualize the information structure of very large information systems like Internet (for a recent overview see [specialissue98]). In a general CBIR system tools must be provided to spot the right information given a learning or design set or during an interactive dialog with the user to tune the indexing, retrieval and clustering methods for the task presented.

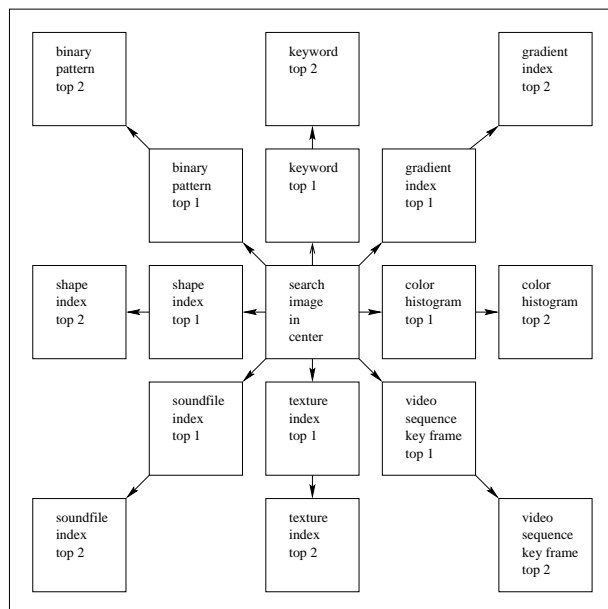
In our group two roads are explored to support the (fine)tuning phase:

- well-documented test-sets that provide ground-truth for (fine)tuning the feature selection in a large-scale well-defined static application using sound statistics.
- interactively used relevance feedback for small-scale (coarse) tuning within a dynamic search environment.

In this paper a well-documented ground-truth set of studio logos and portraits at various fine and coarse clustering levels for the first approach is described.

The relevance feedback approach is the topic of a recently started PhD research project. For relevance feedback a set-up was devised that should minimize the time spent in the learning stage. To quickly select and set weights for the contribution of specific indexes during retrieval, the user will be presented with a multi-dimensional ranking GUI (like the eight ranking lists in figure 1 around the central present best search or goal image, which might even be a random image at start). Images chosen for positive feedback will lead to the inclusion of specific feature vectors for the final one-dimensional ranking based on a weighted combination of indexes. Not only image features but real multi-media indexing using indexes for associated annotation and sound as well can be taken into account whenever feedback indicates this would be appropriate.

In addition, clustering may be a necessary tool for suppressing duplicates or near-duplicates in any interactive ranking stage; especially at the initial search phases and on Internet where many near-copies of original images might otherwise clutter the top of ranking lists.



**Fig. 1.** Learning phase GUI with multi-dimensional ranking lists around present goal or search image: relevance feedback helps forming the specific classifier combination for final goal image delivery

## 2 Evaluation of content-based search for cluster members

### 2.1 (*Index, Retrieval*) pair performance

In general, CBIR approaches can be characterized by a specific (*index, retrieval*) pair used to produce a linear ranking  $R_{i,r}$ : *index* stands for any *feature – vector* used to characterize content (from the raw digital pixel values to scale space, affine transform invariants, wavelet coefficients, etc.), whereas *retrieval* stands for any *distance – measure* ( $L_1, L_2, Mahalanobis$ , etc.) calculated from (part of) the feature vector elements and used for sorting the similarities into a linear ranking order  $R_{f,d}$ . So:  $R_{i,r} = R_{f,d}$ .

**Perfect ranking results** For a cluster of  $m$  members embedded in a database of  $n$  images, a perfect ranking result would mean that the cluster members occupy the first  $m - 1$  positions among the  $n - 1$  ranked with each of the cluster members used in turn as the search image. When the feature vector of the search image itself is present in the database or compared with itself as well the first  $m \in [1, n]$  positions would be occupied. So the ideal rank  $R_{i,d}$  for a cluster of  $m$  images within a database of  $n$  images ( $n \geq m$ ) would be irrespective of database size:

$$R_{i,d} = R_{i,r} = R_{f,d} = m/2 \quad (1)$$

**Imperfect ranking results** In general, ranking results will show a less than ideal situation of dispersed cluster members; each cluster member  $i \in [1, m]$  when using cluster member  $j \in [1, m]$  as the search image, will end up at rank  $k \in [1, n]$ . Let  $m_{ijk}$  denote this rank. The average rank for a particular (*feature – vector, distance – measure*) combination  $R_{f,d}$  is obtained by averaging over  $i$  and  $j$ :

$$\left(\sum_i \sum_j m_{ijk}\right)/(m \cdot m) = R_{f,d} \in [m/2, n - (m/2)] \quad (2)$$

Dividing  $R_{id}$  by  $R_{f,d}$  gives a normalised performance measure  $P_{f,d}$ :

$$R_{id}/R_{f,d} = P_{f,d} \in [(m/2)/(n - (m/2)), 1]; P_{f,d} \in (0, 1] \quad (3)$$

for with  $n \geq m$ ,  $(m/2)/(n - (m/2)) \in (0, 1]$ ;  $\lim_{n \rightarrow \infty} P_{f,d}(n) = 0$  and  $P_{f,d}(n) = 1$  for  $n = m$ .

A plot of this performance against database size  $n$  will give a clear indication of the ranking strength of the (*index, retrieval*) pair. The ideal (*index, retrieval*) pair would show as a straight line at  $P_{f,d}(n) = 1$  irrespective of  $n$ . Less ideal performances all start off at  $P_{f,d}(m) = 1$ , but will gradually fall away towards 0 for growing  $n$ . When more than 1 cluster is used to evaluate the performance of the (*index, retrieval*) pair a weighted average of the individual cluster performances can be used instead (using the cluster sizes as weights).

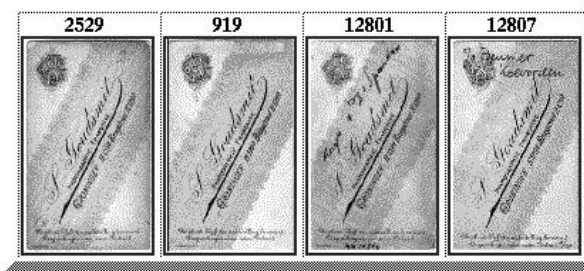
In reality the best performing (*index, retrieval*) combination will be the one that occupies the largest area under the  $P_{f,d}(n)$  graph and thus performances of specific (*index, retrieval*) or (*feature – vector, distance – measure*) combinations can be compared using the single normalized qualifier:

$$\left(\int_m^n P_{f,d}(n) dn\right)/(n - m) = A_{i,r} = A_{f,d} \in (0, 1] \quad (4)$$

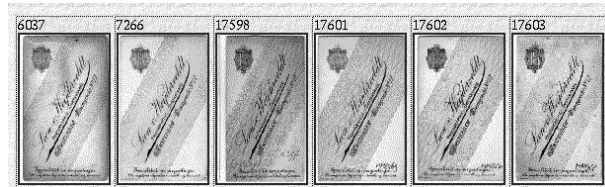
### 3 Cluster membership

#### 3.1 Definition of a cluster

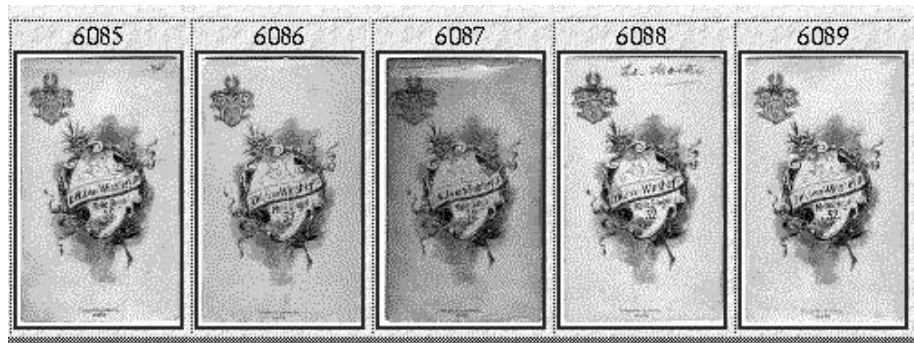
What specifically defines a cluster is often a very subjective grouping (when done by humans) or numerical harness (like a k-means clustering) without being useful in general. Only in specific situations like (near-)copies can clustering be considered a well-defined task. The studio logos in the Leiden 19th-Century Portrait Database and the doubles and triples that exist from some portraits (that were once manufactured by the dozen from an identical glass negative) are cases of fine-level clustering that can be considered to be quite objective clusters. See for instance examples from identified cluster members in figures 2, 3, 4 and 5. Clearly members of these clusters were produced from the same printing plate made from a more or less artistic design.



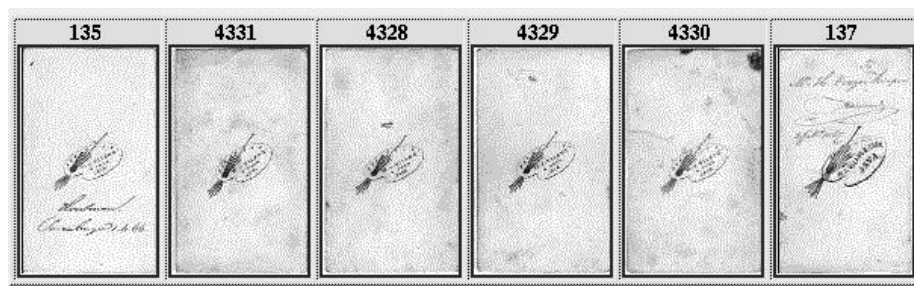
**Fig. 2.** Lowest level cluster 378 in LCPD showing annotation noise in c and d



**Fig. 3.** Lowest level cluster studio weydeveldt showing level noise and shift noise in c



**Fig. 4.** Lowest level cluster of artistic design studio winsheym showing level noise in c and annotation noise in d



**Fig. 5.** Lowest level cluster 976 in LCPD showing annotation noise in a and f

### 3.2 Spread within a cluster: sources of noise

Several effects at the time of production and during conservation have altered the appearance of the images and are to be seen as added noise causing spread within clusters. Main causes for the studio logos of changed appearance are:

- *level – noise*, change in background intensity due to either manufacture or differences in daylight exposure;
- *contrast – noise*, different intensity distribution within normalised intensity range (bleached or faded copies)
- *rotation – noise*, mostly slight (but sometimes 180 degrees) misrotations that remain after normalizing orientation using the portrait on the front side
- *shift – noise*, small misalignments that remain after normalisation
- *condition – noise*, some images are better preserved, in terms of scratches, dirt etc., than others
- *annotation – noise*, the addition of annotation (see figure 5) like person depicted, date of exposure and last but not least collection identifiers;
- *clip – noise*, change in size due to cutting off edges

These various noise sources that are found to be active within LCPD are representative for most cases of noise found in image collections. Some of the noise sources are symmetric, others are asymmetric in their effect upon noise distributions. *Level – noise*, *rotation – noise*, and *shift – noise* are examples of noise distributions symmetric around a mean noise level, whereas *contrast – noise*, *condition – noise*, *annotation – noise* and *clip – noise* show asymmetric distributions. Recognizing noisy members as belonging to a cluster is easier in the presence of symmetric than asymmetric noise. By working in gradient space many induced lighting variations can be minimized. By working with indexes obtained from low-resolution averages *condition – noise* can be kept small. For the clusters defined in the LCPD set *annotation – noise* will be the most difficult to cope with, for in gradient space its effect is even enlarged.

### 3.3 Representative member of a cluster

To represent clusters at a higher (coarser) retrieval level the concept of a *cluster – representative* becomes important. What is the most representative member according to the different noise sources? The following list indicates the representativity in terms of a statistical *MIN*, *MEAN*, *MEDIAN* or *MAX* value of the associated noise distributions:

- *level – noise*, best by *MEDIAN*
- *contrast – noise*, best by *MAX*
- *rotation – noise*, best by *MEAN*
- *shift – noise*, best by *MEAN*
- *condition – noise*, best by *MEDIAN*
- *annotation – noise*, best by *MIN*
- *clip – noise*, best by *MIN*

One way to automatically select a representative of same images for a given (*feature – vector, distance – measure*) is to take the highest ranking individual member, when ranking with  $n = m$ , for this identifies the member of the cluster closest to all the other members. However this can only be done easily when cluster membership is established. Especially in small clusters the members closest to all the others in the cluster may not be the one a user would pick as a representative. Because picking a *cluster – representative* may not be easy, part of the clustering effort will be devoted to providing hand-picked ground-truth for that task as well.

### 3.4 Outliers of a cluster

Due to one or more noise sources the feature vector characterizations and distance measures obtained from them will be more or less successful in clustering like images without wandering into nearby clusters. Clustering and choosing a representative from a cluster will be greatly enhanced when cluster members with the biggest noise contribution can be detected at an early stage and suppressed during specific stages. The next list tries to indicate those outliers per noise source:

- *level – noise*, worst by *MIN,MAX*
- *contrast – noise*, worst by *MIN*
- *rotation – noise*, worst by *MIN,MAX*
- *shift – noise*, worst by *MIN,MAX*
- *condition – noise*, worst by *MIN,MAX*
- *annotation – noise*, worst by *MAX*
- *clipnoise*, worst by *MAX*

For a given (*feature – vector, distance – measure*) there is an easy way to isolate outliers automatically (again only when cluster membership is established!). By setting a threshold distance in the ranking results in case  $n = m$  (ranking only applied to cluster members) outliers can easily be identified. However substantial overlap between clusters may remain for a certain threshold setting.

## 4 A hierarchy of clusters: superclusters

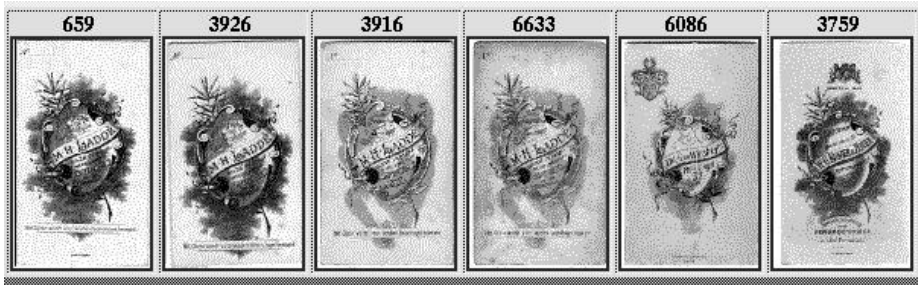
Although clustering above the lowest (fine)level clustering level in LCPD will be less objective we felt a strong need to define ground-truth for a hierarchy of cluster levels, to support initial browsing and to control the degree of likeness allowed in ranking lists.

### 4.1 Similar: between same and unlike

Grouping images for display purposes will have both a lower threshold on likeness (members treated as the same, undistinguished given the high amount of likeness) and an upper threshold on likeness (members still seen as similar but distinguishable).



**Fig. 6.** Supercluster of similar diagonal design with emblem top left while suppressing the printed character information



**Fig. 7.** Supercluster of same artistic design while suppressing printed character information



**Fig. 8.** Semantic OCR-level cluster of studio B. Bruining in Leiden when suppressing artistic design elements

## 4.2 Opposing views: OCR versus design

Our portrait database with scanned studio logos proves to present a particularly nice example of the different ways its information is indexed by various user groups: the content can be divided into three categories:

- printed characters (OCR recognizable part)
- artistic design
- added annotation (usually handwritten characters)

Collections can be characterized by the main key for sorting and storing these portraits: most collections (institutional and private) use the studio information (printed characters) as the main index; one private collector uses the artistic design as the main index; one institutional collection uses the added annotation (person depicted) as the main key. The studio and design index offer opposing views of the information contained in logos: the studio index demands complete suppression of artistic design elements, whereas the design index demands complete suppression of the printed character parts. For the extraction of information for the studio index from the printed characters a spotting method like Optical Character Recognition (OCR) is needed to suppress the more dominant artistic design signal; for the extraction of the design index it suffices to extract features from low resolution copies of the images. Annotation can be spotted by recording the difference of annotated backsides with an annotation-free cluster representative.

The LCPD directory (at <http://ind156b.wi.leidenuniv.nl:2000/>) uses the studio index as the main key and will use the design index as a second key.

Figures 6 and 7 show examples in our testsets of superclusters of artistic design elements. Figures 9 and 8 illustrate the effect of clustering when all the non-OCR recognizable patterns are treated as noise and part of the OCR recognizable information is used to form high-level semantic clustering of studio logos. Within the LCPD directory four clustering layers on the basis of printed characters (OCR recognizable information) is used to form superclusters in LCPD: photographer, photographer plus city, photographer plus city plus street, photographer plus city plus street plus streetnumber.

## 4.3 Supercluster in binarized gradient space

Most lighting noise sources can be effectively suppressed by extracting features from binarized gradient images. For photographs this transformation has a particularly attractive side-effect: positives and negatives of the same scene become highly alike, which makes it easy to trace back prints to original negatives; also in the LCPD studio logos many designs exist in both positive (black characters on light background) and negative (white characters on dark background) versions (see figure 9 a and b). In the LCPD ground-truth cluster definitions positive and negative versions are clearly indicated. An example of a supercluster based on gradient features is shown in figure 10.



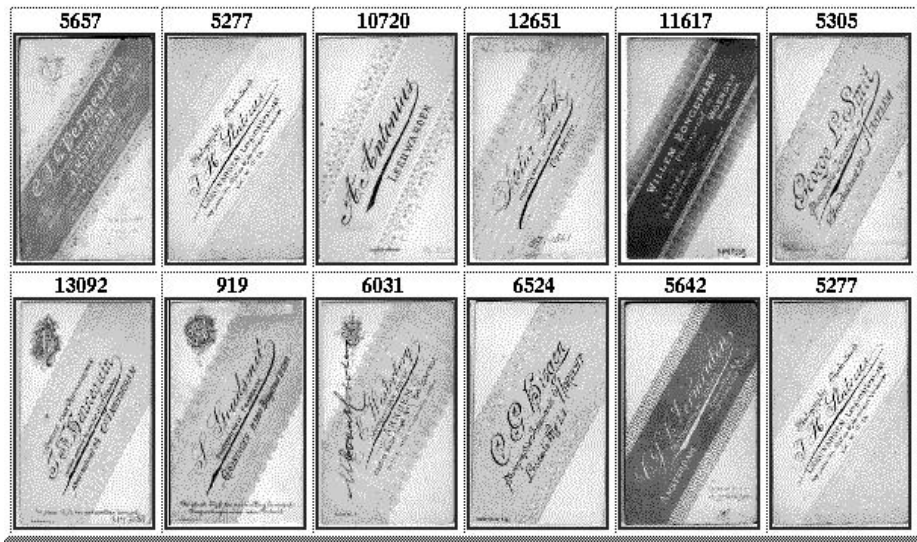
**Fig. 9.** Semantic OCR-level supercluster of studio B. Bruining (Arnhem and Leiden combined) while suppressing artistic design elements

## 5 CD-ROM with LCPD ground-truth testsets for hierarchical clustering

Apart from an already available CD-ROM with about 500 testset pairs of portraits (originally contact-printed from a same glass negative) embedded in about 10,000 studio portraits and with a number of feature vector sets produced in December 1998, the ground-truth clustering effort undertaken for this paper will lead to a large test-set of about 2000 validated clusters and superclusters obtained from the logos at the back of 16,500 LCPD studio portraits. Researchers that would like to use this material can contact the first author in order to obtain a copy of the cluster CD-ROM.

## 6 Acknowledgements

We gratefully acknowledge support by a grant from Philips Research Laboratories in the Netherlands that made the construction of this ground-truth test-set for hierarchical clustering and performance evaluation of indexing and retrieval from large image databases possible.



**Fig. 10.** Super cluster of diagonal design in gradient space where black on white (positives) and white on black (negatives) differences disappear

## References

- [specialissue98] Murtagh, F. (ed.): Special Issue on Clustering and Classification. The Computer Journal **41-8** (1998)
- [survey83] Murtagh, F.: A Survey of Recent Advances in Hierarchical Clustering Algorithms. The Computer Journal **26** (1983) 354–359
- [Hartigan75] Hartigan, J. A.: Clustering Algorithms. Wiley (1975)
- [Dimai99] Dimai, A.: Assessment of Effectiveness of Content Based Image Retrieval Systems. Conf. Proc. Visual'99 LNCS **1614** (1999) 525–532
- [HP98] Ma, W., Zhang, H.: Benchmarking of Image Features for Content-based Retrieval. IEEE (1998) 253–257
- [DeVijver82] DeVijver, P.A., Kittler, J.: Pattern Recognition A Statistical Approach. Prentice-Hall (1982)
- [Kittler96] Kittler, J., Hatef, M., Duin, R.P.W.: Combining Classifiers. IEEE Proc ICPR'96 (1996) **2B** 897–901
- [metric98] Sebe, N., Lew, M., Huijsmans, D.P.: Which Ranking Metric is Optimal? With Applications in Image Retrieval and Stereo Matching. Conf Proc ICPR'98 (1998) 265–271
- [perfHuijsmans97] Huijsmans, D.P., Lew, M.S., Denteneer, D.: Quality Measures for Interactive Image Retrieval with a Performance Evaluation of Two 3x3 Texel-Based Methods. Conf. Proc. ICIAP'97 LNCS **1311** (1997) 22–29