We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,300

Open access books available

172,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Invariant Image Retrieval: An Approach Based On Multiple Representations and Multiple Queries

Noureddine Abbadeni

College of Computer and Information Sciences, KSU

Riyadh, KSA

nabbadeni@ksu.edu.sa; noureddine.abbadeni@usherbrooke.ca

Abstract

We present an approach based on multiple representations and multiple queries to tackle the problem of invariance in the framework of content-based image retrieval. We consider the case of textures. This approach, rather than to consider invariance at the representation level, considers it at the query level. We use two models to represent the visual content of textures, namely the autoregressive model and a perceptual model based on a set of perceptual features. The perceptual model is used with two viewpoints: the original image's viewpoint and the autocovariance function viewpoint. After a brief presentation and discussion of these multiple representation models/viewpoints, we present the invariant texture retrieval algorithm. This algorithm considers results fusion (merging) at two different levels: 1. The first level consists in merging results returned by different models/viewpoints (representations) for the same query in one results list using a linear results fusion model; 2. The second level consists in merging each fused list of different queries into a unique fused list using a round robin fusion scheme. Experimental retrieval results and benchmarking based on the precision/recall measures applied on a large image database show interesting results.

1. Introduction

Content-based image retrieval (CBIR) is a research area that has been active for more than a decade and many research results and prototypes have been carried out since then (4), (1), (23), (26), (14), (27), (24), (11), (28). Despite the important number of works related to CBIR, the problem of invariance has not received sufficient attention yet. Invariance, a difficult problem in computer vision and image analysis, deals with, in the framework of image retrieval, the ability to retrieve all relevant images to a query image even if some of them have been transformed according to different geometric and photometric transformations such as rotation, scaling, illumination, viewpoint change and contrast change as well as non-rigid transformations (30). Some works in the literature have indicated that, even when a texture is transformed with some geometric or photometric transformation, human subjects still perceive it as the same texture (30), (21), (22). Two images with similar textures must still be similar, for some applications, if we rotate one texture or if we change the contrast of another texture for example. Therefore invariance might be an important problem depending on applications and users' needs.

This chapter is organized as follows: In section 2, related works are briefly presented and discussed; In section 3, the different content representation models/viewpoints are briefly presented and discussed; In section 4, the multiple queries approach is presented and discussed; In section 5, results fusion strategies are presented and discussed; In section 6, the invariant texture retrieval algorithm is applied to a large texture database and benchmarked using the well-known precision and recall measures; And, finally, in section 7, a conclusion and further investigations related to this work are briefly discussed.

2. Related work

Several works published in the literature deal with the problem of invariance, especially in the case of textures. These research works can be classified in two main categories (21), (30):

1. The first class of works considers invariance at the representation level, i.e. they try to develop a model which is invariant with respect to a limited number of transformations. A well-known example of such an approach is the rotation-invariant simultaneous autoregressive model (*RISAR*) proposed by Mao and Jain (25). Another example is the Wold-like model (23) which may present a good level of invariance with respect to translation, rotation and scaling. Other statistical, structural and model-based texture analysis methods which have some degree of invariance are discussed in (30). The main problem with this approach is its limited scope since it is very difficult to develop an invariant model that takes into account a large number of transformations. In general, models of this class consider one or two well-defined transformations and ignore the long list of all possible transformations. A more complete review of research works in this category can be found in (30).

2. The second class of invariant models consider invariance at the query level, rather than the representation level. Among the works that considered this approach, we cite (17) and (19). However, these works considered the problem of image retrieval in general and did not consider the problem of invariance in particular. We did not find in the literature any work that deals with image retrieval and explicitly considers invariance at the query level. In this approach, the problem is the number of queries and the number of representations needed for each query. Taken in an absolute way, there is no satisfying solution to this problem. However, in the framework of CBIR, we believe that a good solution to this problem must be based on users' needs. In fact, users' needs may play an important role in choosing the set of prototypes to represent an image. For example if a user is interested in finding the same texture with different orientations, we may use a restricted number of query images that can represent different orientations. If the user is interested in scaling, we may use a restricted number of query images that can represent different scales, etc.

The new invariant texture retrieval algorithm presented in this chapter fits within the second approach. We use multiple models/viewpoints to represent textural content of images. At the query level, we use multiple queries to represent user's need. We use appropriate results fusion models to merge results returned by each model/viewpoint and by each query. Results fusion is performed at two different levels (3): 1. The first level consists in merging results returned by different representations for the same query; 2. The second level consists in merging results returned by each query for the same user's need.

3. Multiple Representations

To represent texture content, we use two different models, the autoregressive model and a perceptual model based on a set of perceptual features (8). The autoregressive model used

is a causal simultaneous *AR* model with a non-symmetric half-plan (NSHP) neighborhood. The perceptual model is considered with two viewpoints: the original image's viewpoint and the autocovariance function (associated with original images) viewpoint. Each of the viewpoints of the perceptual model used is based on four perceptual features, namely coarseness, directionality, contrast and busyness. So, we have a total of three models/viewpoints; each model/viewpoint results in a vector of parameters of size four for a total of twelve parameters. Briefly, the autoregressive model is characterized, in particular, by a forecasting property that allows to predict the grey-level value of a pixel of interest in an image by using the grey-level values of pixels in its neighborhood. The autoregressive model, when used to model a textured image, allows to estimate a set of parameters (their number corresponds to the number of neighbors considered), each one corresponds to the contribution of its corresponding pixel in the forecasting of the pixel of interest (the total of contributions of all pixels in an image is 100%).

The perceptual model, which is perceptual by construction, is based on a set of four computational measures that simulate four perceptual features: coarseness, directionality, contrast, and busyness. Briefly, coarseness was estimated as an average of the number of extrema; Contrast was estimated as a combination of the average amplitude of the gradient, the percentage of pixels having their amplitude superior to a certain threshold and coarseness itself; Directionality was estimated as the average number of pixels having the dominant orientation(s); And finally, busyness was estimated based on coarseness. The computational measures proposed for each perceptual textural feature were evaluated by conducting a set of experimentations taking into account human judgments and using a psychometric method. Thirty human subjects were asked to rank a set of textures according to each perceptual feature. Then, for each perceptual feature, we consolidated the different human rankings into one human ranking using the sum of rank values. For each feature, the consolidated human ranking obtained was compared to the ranking given by the corresponding computational measure using the Spearman coefficient of rank-correlation. Experimental results showed very strong correspondence between the proposed computational measures and human rankings. Values of Spearman coefficient of rank-correlation r_s found were as follows: for coarseness, $r_s = 0.913$; for directionality, $r_s = 0.841$; for contrast, $r_s = 0.755$; and finally, for busyness, $r_s = 0.774$. Compared to related works, our results were found more satisfactory (8), (9).

These models and viewpoints were not constructed to be invariant to geometric and photometric transformations. The autoregressive model and the perceptual model based on the autocovariance function viewpoint have poor performance in the case of invariance due to the fact that both the autocovariance function and the autoregressive model are very sensitive to variance and transformations that may occur in a texture. The perceptual model based on the original image's viewpoint performs well in the case of invariance even if it was not constructed to deal with the problem of invariance. In fact, the measure of directionality proposed is not very sensitive to rotation since we compute a degree of directionality and not a specific orientation. Directionality is, however, sensitive to scaling since it is computed as a number of oriented pixels: the more a texture is coarse, the less is the number of oriented pixels and the less the texture is directional. The measure of coarseness proposed is robust with respect to rotation but it is sensitive to scaling. The same remarks given on coarseness are applicable to busyness. The measure of contrast proposed is, obviously, sensitive to illumination conditions. For more details on the perceptual model, refer to (8), (9) and (6); for more details on the autoregressive model, refer to (10) and (5).

4. Multiple Queries

In the framework of content-based image retrieval, we propose the use of multiple queries and representations to tackle the problem of texture invariance. That is, rather than trying to develop invariant content representation models, which may be very complex and of limited scope, we consider the problem of invariance at the query and the representation levels. Each user’s need is represented by a set of query images that are different from each other in terms of orientation, scale, contrast, etc. depending on the user’s needs. Search results returned for each query are fused to form a unique results list using appropriate results fusion models. For example, if a user wants to retrieve all textures that are visually similar to a given texture, even those not having the same orientation(s) as the given texture, we can represent this user’s need by a set of query images with different orientation(s) and fuse the results returned for each query image to form a unique results list that corresponds to the original user’s need. Results fusion returned for different queries allows to consider the fact that relevant images to a given query are not necessarily near the query in the feature space, especially in the case of invariance. In fact, since content representation models are not, in general, invariant with respect to different transformations, features computed on the same image, which has been rotated for example, are not the same as the original image and can be very different. So, such features are not in the neighborhood of the query image in the feature space and can be even located in disjoint regions that can be more or less far from each other in the feature space (17). The use of only one query image will not retrieve an important set of relevant images (depending on the degree of variance in the database) since, in this case, these relevant images are not necessarily near the considered query image in the feature space. Figure 1 shows the fact that many query images will search in different regions, that can be disjoint, in the feature space (19).

Fig. 1. Different regions for different queries in the same feature space. When query images are different, the corresponding results may be located in different regions in the feature space.

We should also mention that with multiple queries, there is some computational overhead since we use several queries for the same user's need. However, with such an approach, the size of the parameters vector does not change for images in the database. Only a user's need is represented with several queries, and thus with several parameters vectors. So, with such an approach, efficiency is not altered in an important way. Compared to the other approach of tackling invariance, i.e. representation-level invariance, and making abstraction of its difficulty and its limited scope to handle a wide range of transformations, the number of parameters is generally increased in an important way and this results generally in an important loss of efficiency.

5. Results fusion

Results fusion has been used in the information retrieval community for a long time now (12; 13; 20; 29). Results fusion concerns two levels: 1. results fusion from multiple models/viewpoints (representations) for the same query; 2. And, results fusion from different queries.

5.1 Multiple models/viewpoints fusion

In the invariant texture retrieval algorithm we are proposing, each query is represented by three different set of parameters corresponding to the different models/viewpoints. Results fusion returned by each of these models/viewpoints, for the same query, can be done using an appropriate fusion model. We have experimented many results fusion models. The model which gave the best results was the linear fusion model, denoted *FusCL*, and defined as follows:

$$FusCL_{ij} = \frac{\sum_{k=1}^K GS_{M_{ij}^k}}{K} \quad (1)$$

Equation (1) is based on the similarity value returned by the similarity measure $GS_{M_{ij}^k}$ (scores) and expresses the merging of results returned by different models/viewpoints M^k as an average of the scores obtained by an image in its different rankings corresponding to different models/viewpoints. K is the number of models/viewpoints considered, i is the query image and j corresponds to images returned for query i according to model/viewpoint M^k .

The *FusCL* model exploits two main effects: 1. The first effect known as the chorus effect in the information retrieval community (29), that is, when an image is returned as relevant to a query by several models/viewpoints, this provides stronger evidence of relevance than if it is returned by only one model/viewpoint; 2. The second effect called the dark horse effect (29), that is, when a model/viewpoint ranks exceptionally an image, which is not relevant to a query, in top positions, this can be attenuated by the fused model if the other models/viewpoints do not rank it in top positions (it is very rare for a non-relevant image to be ranked at top positions by several models/viewpoints).

5.2 Multiple queries fusion

In the case of multiple queries, we can use the same fusion models as in the case of multiple models/viewpoints fusion. However, those models did not give good results in practice since the chorus and dark horse effects are not very significant in the case of invariance. In fact, when using multiple queries, each query with a different orientation, scale, or contrast for example, the results returned for each query contain, actually, a little number of common

images. Models that can be used, in this case, are models able to exploit the skimming effect, i.e. take from each list (for each query), the best results. Among such models, we have the *FusMAX* model and the *FusRR* model.

FusMAX. We define the *MAX* model, denoted *FusMAX*, as follows:

$$FusMAX_{ij} = MAX(GS_{M_{ij}^k}) \quad (2)$$

The *MAX* model will consider the model/viewpoint that gives the highest value of similarity. $GS_{M_{ij}^k}$ is the similarity measure defined in section 6.2. The *MAX* model may give good results since it exploits the so-called skimming effect (29) which corresponds better to the case of invariance.

FusRR. Another fusion technique that can be used, since it exploits also the skimming effect, is a technique called round robin (we denote it *RR* or *FusRR*) (12), (13). The *RR* technique makes use of the rank of an image in the returned results list rather than the value of the similarity function. The *RR* technique simply considers images that are ranked in the first position in each list, corresponding to each query, and gives them all the same first position in the fused list, then taking images that are ranked in the second position in each list, corresponding to each query, and give them all the same second position in the fused list and so on. We can stop this process after a threshold of the similarity value and/or the rank or after having taken a sufficient number of images. Obviously, if we find the same image in different lists, which may occur occasionally, only one image is considered. Note that the *RR* technique is different from the *MAX* model. Experimental results will show that the *RR* fusion technique exploits the skimming effect in a more effective way than the *MAX* model does.

The *RR* technique, as mentioned, does not make use of similarity values. If necessary, we can use the rank given to images in the fused list and compute an artificial similarity value for each image in the fused list. To do so, we can use the rank-based similarity measure defined as follows (20):

$$Sim_{Rank} = 1 - \frac{Rank - 1}{N} \quad (3)$$

where *Rank* is the rank given to an image in the fused list and *N* is the number of images in the fused list. With such an artificial similarity function, an image ranked at position 1 will have $Sim_{Rank} = 1$, an image ranked at the last position will have Sim_{Rank} near 0 (actually $1/N$) and all other images will have an artificial similarity value between 0 and 1.

6. Application to image retrieval

6.1 Invariant texture retrieval algorithm

A general scheme of our invariant texture retrieval algorithm is given in figure 2. The algorithm is based on two levels of results fusion as we have already mentioned. The first level consists in fusing results returned by different models/viewpoints for the same query. The second level consists in fusing results returned for different queries. Multiple models/viewpoints fusion, for the same query, is based on fusion models exploiting the chorus and dark horse effects such as the *FusCL* model while multiple queries fusion is based on fusion models exploiting the skimming effect such as *FusMAX* and *FusRR* models.

Fig. 2. General scheme of our invariant texture retrieval algorithm. The algorithm has 2 levels of fusion. Each query is represented by 3 different models/viewpoints: *AR*, *PCP*, and *PCP – COV*. The first level of fusion consists in merging the retrieval results returned by each model/viewpoint, for the same query, using the *FusCL* model. The second level of fusion consists in merging the retrieval results for each query in one final fused list of results using either the *FusMAX* model or the *FusRR* model.

6.2 Similarity measure

The similarity measure used is based on Gower coefficient of similarity (18) we have developed in our earlier work (7). The non-weighted similarity measure, denoted *GS*, can be defined as follows:

$$GS_{ij} = \frac{\sum_{k=1}^n S_{ij}^{(k)}}{\sum_{k=1}^n \delta_{ij}^{(k)}} \quad (4)$$

where $S_{ij}^{(k)}$, the partial similarity, is the score of comparison of images i and j according to feature k and $\delta_{ij}^{(k)}$ represents the ability to compare two images i and j on feature k . $\delta_{ij}^{(k)} = 1$ if

images i and j can be compared on feature k and $\delta_{ij}^{(k)} = 0$ if not¹. $\sum_{k=1}^n \delta_{ij}^{(k)} = n$ if images i and j can be compared on all features $k, k = 1..n$. Note that, in our case, we will use all features; thus, $\delta_{ij}^{(k)} = 1$ for all k, i, j .

$S_{ij}^{(k)}$ is defined as follows:

$$S_{ij}^{(k)} = 1 - \frac{|x_{ik} - x_{jk}|}{R_k} \quad (5)$$

where x_{ik} represents the value of feature k of image i . R_k represents a normalization factor. R_k is computed on the database considered for experimentations and is defined as follows :

$$R_k = \text{Max}(x_{ik}) - \text{Min}(x_{ik}) \quad (6)$$

The weighed version of the similarity measure can be defined as follows :

$$GS_{ij} = \frac{\sum_{k=1}^n w_k S_{ij}^{(k)}}{\sum_{k=1}^n w_k \delta_{ij}^{(k)}} \quad (7)$$

where w_k corresponds to the weight associated with feature k .

We have used, in the case of the *AR* model, the inverse of variance of each estimated parameter k computed on the considered experimental database as a weight for this parameter. In the case of the perceptual model, we have used two approaches of weighting: 1. In the first approach, as in the case of the *AR* model, we used the inverse of variance of each feature as a weight for this feature; 2. In the second approach, we used the Spearman coefficients of rank-correlation found for each perceptual feature (section 2) as a weight for this feature. The use of the inverse of the variance as a weight is motivated by the fact that the discrimination capacity of that feature increases when the variance is small and decreases when the variance is large. Therefore, using the inverse of variance as a weight will give more importance to features having a more important discrimination capacity (16). The use of Spearman rank-correlation coefficient as a weight is motivated by the fact that we want to give more importance to perceptual features that are highly correlated with human judgments. Note that the Spearman coefficient of rank-correlation is independent from the image database considered for benchmarking while the inverse of variance is dependent on this image database.

6.3 Benchmarking database

For experimental results and benchmarking, we have used an image database coming from Ponce's group at UIUC.² This database contains 22 classes of 40 640x480 images each class for a total of 880 images.³ Images within the same classes have been taken with different photometric and geometric conditions. In each class there is a high degree of variance between

¹ Note that, in our case, we are able to compare all images on all features. Thus, $\delta_{ij}^{(k)}$ will never be 0. The formula is a general expression of the similarity measure and can be used in other cases where images or objects cannot be always compared on all features.

² http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database

³ Note that, originally, this database contains 25 classes. However, during the conversion of images from their original JPEG format to PGM format, we have found some errors in some images in classes 17, 18 and 23. We have decided to eliminate these 3 classes and consider only 22 classes rather than 25 classes.

images in orientation, scale, contrast as well as non-rigid deformations. A sample from this database is given in figure 3.⁴

Fig. 3. Sample of images from the Ponce's group texture database: 4 sample images from 5 different classes T05, T06, T10, T15, and T20.

In experimental results, the *AR* model with an *NSHP* (non-symmetric half-plan) neighborhood weighted with the inverse of variance of each feature gave the best results among the different variants of the *AR* models. Both the *PCP* model (perceptual model based on original images) and the *PCP – COV* model (perceptual model based on the autocovariance function) in their weighted version using Spearman coefficient of rank-correlation gave the best results among the different variants of the perceptual model. In the rest of this chapter, we consider

⁴ http://www-cvr.ai.uiuc.edu/ponce_grp/data/texture_database/samples

these three best versions for results fusion:

- **AR** or **AR NSHP-V**: The autoregressive model weighted with the inverse of variance.
- **PCP** or **PCP-S**: The perceptual model based on the original image’s viewpoint and weighted with Spearman coefficients of rank-correlation.
- **PCP-COV** or **PCP-COV-S**: the perceptual model based on the autocovariance function view-point and weighted with Spearman coefficients of rank-correlation.

Fig. 4. Search results returned for query T20-01 using the *PCP* model weighted with Spearman coefficients. T20-01 is the query image. Results are given in a decreasing order according to their similarity value.

6.4 Experimental results

When considering different models/viewpoints separately, the *PCP* model (based on the original image’s viewpoint) is the best model since, as we mentioned earlier, it exhibits some robustness in the case of invariance comparatively to the other models/viewpoints. In the following figures, we show some examples of results obtained with the *PCP* model weighted with Spearman coefficient of rank-correlation (see Figures 4, 5 and 6).

When examining these results, we can point out that the *PCP* model, with only 4 parameters, gave good results even if it was not constructed originally to be invariant with respect to some transformations. However, using one model/query gives in some cases bad results, that is some images not similar to the query image are returned with a high similarity value. The other two models, the autoregressive model and the *PCP – COV* (perceptual model based on

the autocovariance function viewpoint), have not good performance (more details are given in the following figures and tables).

Fig. 5. Search results returned for query T05-01 using the *PCP* model weighted with Spearman coefficients. T05-01 is the query image. Results are given in a decreasing order according to their similarity value.

Fig. 6. Search results returned for query T24-01 using the *PCP* model weighted with Spearman coefficients. T24-01 is the query image. Results are given in a decreasing order according to their similarity value.

6.5 Precision and recall measures

Precision and recall measures are common standard techniques to benchmark search relevance in information retrieval systems in general Dun. 97. Precision, which can be defined as the number of relevant and retrieved images divided by the number of retrieved images, measures the ability of the model to reject non-relevant images with respect to a query. Recall, which can be defined as the number of relevant and retrieved images divided by the number of relevant images in the database for the considered query, measures the ability of a model to retrieve all relevant images. Precision and recall measures are computed for each query at each position. Then, average precision and average recall are computed respectively as an average across a set of representative queries. Two complementary graphs are then given: 1. Average precision as a function of the number of retrieved images (positions) allowing to see the ability of the model to reject non relevant images as the model retrieves more images; 2. Average recall as a function of the number of retrieved images (positions) allowing to see the

ability of the model to retrieve all relevant images as the model retrieves more images.

Benchmarking we have done, based on the precision-recall measures, concerns both the use of one query as well as the use of multiple queries (see figure 7):

- When considering one query, we have considered the best version of each separated model/viewpoint. Then we have fused these best multiple models/viewpoints, using the *CL* results fusion model, for the same query.
- When considering multiple queries, based on the fused models/viewpoints for each query, we have fused the results of 4 queries using the *MAX* model and the *RR* model as described earlier in this chapter. Query images were selected randomly and correspond to images 1, 11, 21 and 31 from each class.

Figure 7 shows precision and recall graphs for different models, both separated and fused, by considering one query and multiple queries. When examining this figure, we can see that the *PCP* model weighted with Spearman coefficients performs better than the other separated models. Fusing different models/viewpoints for the same query did not achieve an important improvement in performance since two of the three models/viewpoints considered, namely the *ARNSHP – V* and the *PCP – COV* model, have rather poor performance in the case of invariance as explained earlier in this chapter. Fusing multiple queries using both the *MAX* and *RR* models allow improvement in performance. While the *MAX* model allows an average improvement in performance, the *RR* model allows a significant improvement in performance measured in terms of precision and recall. Remember that both the *MAX* and the *RR* models exploit the skimming effect while the *CL* model exploits both the chorus and dark horse effects. Thus, in the case of invariance, the skimming effect, in a multiple queries approach, is more important than both the chorus and dark horse effect. These conclusions can be also drawn by examining tables 1, 2, 3, 4, 5 and 6 in the next subsection.

(a)

(b)

Fig. 7. Precision and recall graphs for different separated models (1 query) and fused model (4 queries): (a) Precision as a function of the number of retrieved images and (b) Recall as a function of the number of retrieved images. The fused *RR* model is clearly the best model.

6.6 Retrieval rate per class and average retrieval rate

Tables 1, 2, 3, 4 and 5 give the retrieval rate (recall) per class at positions 40, 80, 120 and 160 according to the *PCP*, the *PCP – COV*, the *ARNSHP – V*, the *MAX* and the *RR* models respectively. Table 6 gives the average retrieval rate at positions 40, 80, 120 and 160 across all the database according to a selection of separated and fused models. Examination of these different tables allows to draw the same conclusions as in the precedent subsection in which we showed precision and recall graphs for the different separated and fused models. We can particularly notice, again, the good performance of the multiple queries approach using the *RR* fusion model.

Class	P40	P80	P120	P160
T01	.325	.45	.575	.65
T02	.225	.4	.5	.625
T03	.1	.225	.3	.425
T04	.4	.525	.6	.65
T05	.525	.675	.7	.725
T06	.375	.6	.725	.8
T07	.475	.625	.75	.825
T08	.525	.675	.775	.9
T09	.225	.45	.575	.6
T10	.4	.65	.825	.875
T11	.2	.425	.55	.625
T12	.2	.375	.475	.55
T13	.475	.7	.825	.925
T14	.275	.375	.45	.55
T15	.15	.275	.425	.55
T16	.425	.55	.65	.725
T19	.075	.175	.375	.5
T20	.65	.875	.975	1
T21	.35	.45	.5	.55
T22	.175	.275	.4	.475
T24	.625	.85	.925	.925
T25	.1	.15	.175	.3

Table 1. Retrieval rate for each class at positions 40, 80, 120, and 160 according to the *PCP – S* model (1 query). With the *PCP – S* model, results are acceptable for some classes and not acceptable for others. For example, for class T24, at position 40, we have retrieval rate of 0.625 while for class T03, the retrieval rate is 0.1 at position 40. For positions 80, 120 and 160, results are improved in an important way for some classes.

Class	P40	P80	P120	P160
T01	.075	.2	.225	.3
T02	.15	.2	.25	.3
T03	.1	.225	.3	.35
T04	.05	.1	.125	.15
T05	.1	.175	.2	.3
T06	.15	.25	.325	.4
T07	.325	.475	.65	.725
T08	.35	.55	.7	.725
T09	.05	.1	.125	.15
T10	.125	.2	.3	.4
T11	.075	.1	.2	.275
T12	.15	.25	.45	.6
T13	.15	.25	.35	.4
T14	.075	.075	.125	.125
T15	.15	.2	.275	.35
T16	.1	.15	.15	.175
T19	.025	.1	.175	.2
T20	.5	.75	.8	.875
T21	.25	.35	.5	.5
T22	.05	.175	.225	.275
T24	.05	.1	.175	.3
T25	.15	.175	.2	.275

Table 2. Retrieval rate for each class at positions 40, 80, 120, and 160 according to the *PCP – COV – S* model (1 query). With the *PCP – COV – S* model, results are not acceptable for most of classes. For example, for class T20, at position 40, we have retrieval rate of 0.5 while for class T04, the retrieval rate is 0.05 at position 40. For positions 80, 120 and 160, results are improved for some classes only. Note that the results of the *PCP – COV – S* model are clearly less good than the results of the *PCP – S* model.

Class	P40	P80	P120	P160
T01	.175	.425	.5	.525
T02	.375	.425	.475	.475
T03	.05	.1	.1	.15
T04	.4	.475	.625	.625
T05	.225	.35	.5	.6
T06	.175	.2	.25	.25
T07	.05	.1	.15	.175
T08	.2	.375	.4	.45
T09	.275	.525	.775	.875
T10	.325	.45	.55	.7
T11	.325	.5	.675	.8
T12	.275	.3	.325	.4
T13	.3	.375	.5	.6
T14	.125	.225	.3	.375
T15	.1	.275	.4	.45
T16	.4	.65	.75	.775
T19	.125	.25	.325	.375
T20	.275	.475	.6	.7
T21	.025	.025	.025	.025
T22	.125	.225	.25	.35
T24	.35	.55	.625	.675
T25	.15	.2	.275	.325

Table 3. Retrieval rate for each class at positions 40, 80, 120, and 160 according to the *ARNSHP – V* model (1 query). With the *AR-NSHP-V* model, results are not acceptable for most of classes. For example, for class T04, at position 40, we have retrieval rate of 0.4 while for class T03, the retrieval rate is 0.05 at position 40. For positions 80, 120 and 160, results are improved for some classes only. Note that the results of the *AR-NSHP-V* model are clearly less good than the results of the *PCP – S* model.

Class	P40	P80	P120	P160
T01	.375	.575	.675	.75
T02	.375	.55	.65	.725
T03	.35	.475	.6	.6
T04	.25	.325	.425	.525
T05	.425	.6	.675	.725
T06	.325	.425	.55	.675
T07	.275	.375	.6	.7
T08	.4	.6	.65	.7
T09	.5	.6	.775	.85
T10	.55	.7	.8	.85
T11	.675	.825	.95	.95
T12	.575	.8	.95	.95
T13	.5	.675	.825	.875
T14	.35	.525	.725	.825
T15	.4	.55	.625	.725
T16	.55	.725	.85	.925
T19	.35	.5	.6	.7
T20	.725	.85	.95	.95
T21	.2	.325	.45	.525
T22	.2	.475	.55	.575
T24	.35	.4	.475	.575
T25	.275	.375	.475	.5

Table 4. Retrieval rate for each class at positions 40, 80, 120, and 160 according to the *MAX* model (4 queries) applied on the fused *CL* model (1 query). With the fused model *MAX* model, results are clearly much better than the cases where we used one model/query. For most classes, the retrieval rate, at positions 80, 120 and 160, is improved in an important way.

Class	P40	P80	P120	P160
T01	.425	.675	.95	1
T02	.4	.7	.9	1
T03	.425	.675	.9	1
T04	.275	.525	.7	.9
T05	.55	.9	1	1
T06	.275	.5	.65	.8
T07	.425	.725	.9	1
T08	.55	.8	1	1
T09	.725	1	1	1
T10	.625	.9	1	1
T11	.675	1	1	1
T12	.675	.95	1	1
T13	.625	1	1	1
T14	.35	.6	.8	1
T15	.35	.625	.75	.825
T16	.65	.95	1	1
T19	.4	.55	.65	.775
T20	.875	1	1	1
T21	.375	.475	.525	.625
T22	.375	.5	.6	.7
T24	.45	.775	1	1
T25	.425	.6	.75	.875

Table 5. Retrieval rate for each class at positions 40, 80, 120, and 160 according to the *RR* model (4 queries) applied on the fused *CL* model(1 query). With the fused model *RR* model, results are clearly much better than the cases where we used one model/query. For most classes, the retrieval rate, at positions 80, 120 and 160, is improved in an important way and is close to 1. Note that the results of the fused *RR* model are clearly much better than the fused *MAX* model.

Model	P40	P80	P120	P160
<i>ARNSHP – V</i>	.219	.34	.426	.485
<i>PCP – S</i>	.331	.489	.593	.67
<i>PCP – COV – S</i>	.145	.234	.31	.37
<i>MAX (4 Queries)</i>	.408	.557	.674	.735
<i>RR (4 Queries)</i>	.495	.747	.867	.932

Table 6. Average retrieval rate at positions 40, 80, 120, and 160 according to different separated and fused models. The fused *RR* model is clearly the best model.

7. Conclusion

In this chapter, we presented a multiple models/viewpoints and multiple queries approach to tackle the problem of invariant image retrieval, in the case of textures. Three content representation models/viewpoints were used: the autoregressive model and a perceptual model, based on a set of perceptual features such as coarseness and directionality, used with two viewpoints: the original image’s viewpoint and the autocovariance function viewpoint. Each of these models/viewpoints results in a parameters vector of size four. These models/viewpoints were not built to be invariant with respect to geometric and photometric transformations.

Experimental results and benchmarking conducted on a large database, using precision and recall measures, show that, when fusing the results returned with each model/viewpoint for the same query, using the *CL* fusion model, retrieval effectiveness are not significantly improved and are quite similar to the case when only the perceptual model based on the original image's viewpoint was used. When fusing results returned by multiple queries, using both the *MAX* and the *RR* fusion models, retrieval effectiveness is significantly improved especially with the *RR* fusion model.

The choice of appropriate queries is an open question. In this work, we have chosen multiple queries in a random way. We think that, if this choice can be done using some procedure taking into account users' needs, search effectiveness can be further improved.

Acknowledgments

The author acknowledges the financial support of the research center at the college of computer and information sciences (CCIS) at King Saud University(KSU), Riyadh, KSA.

8. References

- [1] Datta R., Joshi D., Li J., and Wang J.Z. Image Retrieval: Ideas, Influences, and Trends of the New Age. *ACM Transactions on Computing Surveys*, 40(2), 60 pages, 2008.
- [2] Abbadeni, N. and Alhichri, H. Low-level invariant image retrieval based on results fusion. *Proceedings of the IEEE ICME, Hannover-Germany*, June 2008.
- [3] Abbadeni N. An Approach Based on Multiple Representations and Multiple Queries for Invariant Image Retrieval. *Proceedings of the International Conference on Visual Information Systems, VISUAL' 2007: 570-579, Shanghai, 2007.*
- [4] Lew M., Sebe N., Djeraba C., and Jain R. Content-Based Multimedia Information Retrieval: State of the art and challenges. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 26 pages, 2006.
- [5] Abbadeni N. Perceptual interpretation of the estimated parameters of the autoregressive model. *Proceedings of the IEEE ICIP, Genoa, Italy, 2005.*
- [6] Abbadeni N. Perceptual image retrieval. *Proceedings of the International conference on visual information systems (VISUAL'05), Amsterdam, Netherlands, 2005.*
- [7] Abbadeni N. A new similarity matching measure: application to texture-based image retrieval. In: *Proceedings of the 3rd International Workshop on Texture Analysis and Synthesis (held in conjunction with IEEE ICCV), Nice, France, October 2003.*
- [8] Abbadeni N, Ziou D, Wang S. Computational measures corresponding to perceptual textural features. In: *Proceedings of the 7th IEEE International Conference on Image Processing, Vancouver, BC, September 10-13 2000.*
- [9] Abbadeni N, Ziou D, Wang S. Autocovariance-based perceptual textural features corresponding to human visual perception. In: *Proceedings of the 15th IAPR/IEEE International Conference on Pattern Recognition, Barcelona, Spain, September 3-8 2000.*
- [10] Abbadeni N. Recherche d'images basée sur leur contenu. Représentation de la texture par le modèle autorégressif. Research report (in French), NO. 216, 50 p., University of Sherbrooke, 1998.
- [11] Antani S., Kasturi R., Jain R. A Survey on the Use of Pattern Recognition Methods for Abstraction, Indexing and Retrieval of Images and Video. *Pattern Recognition*, 35(2002):945-965. 2002.

- [12] Belkin NJ, Cool C, Croft WB, Callan JP. The effect of multiple query representation on information retrieval performance. In: Proceedings of the 16th International ACM SIGIR Conference, pp. 339-346, 1993.
- [13] Berretti S, Del Bimbo A, Pala P. Merging results for distributed content-based image retrieval, *Multimedia Tools and Applications*, 24:215-232, 2004.
- [14] Del Bimbo A. Visual information retrieval. Morgan Kaufmann Publishers, San Francisco, CA, 1999.
- [15] Dunlop MD. Time, relevance and interaction modeling for information retrieval. In: Proceedings of the International ACM SIGIR Conference, pp. 206-213, Philadelphia, USA, 1997.
- [16] Flickner M., Sawhney H., Niblack W. et al. Query by image and video content: the QBIC system. *IEEE Computer*, 28(9):23-32, 1995.
- [17] French JC, Chapin AC, Martin WN. An application of multiple viewpoints to content-based image retrieval, In: Proceeding of the ACM/IEEE Joint Conference on Digital Libraries, pp. 128-130, May 2003.
- [18] Gower JC. A general coefficient of similarity and some of its properties. *Biometrics Journal*, 27:857-874, December 1971.
- [19] Jin X. and French JC. Improving Image Retrieval Effectiveness via Multiple Queries. *ACM MMDB Workshop*, pp. 86-93, New Orleans, Louisiane, USA, novembre 2003.
- [20] Lee JH. Analysis of multiple evidence combination, In: Proceedings of the ACM SIGIR Conference, pp. 267-276, Philadelphia, PA, USA, 1997.
- [21] S. Lazebnik, C. Schmid and J. Ponce. A Sparse Texture Representation Using Local Affine Regions. *Beckman CVR Technical Report*, NO. 2004-01, University of Illinois at Urbana Champaign (UIUC), 2004.
- [22] Lazebnik, S., Schmid, C. and Ponce, J. Sparse Texture Representation Using Local Affine Regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265-1278, 2005.
- [23] Liu F, Picard RW. Periodicity, directionality and randomness: Wold features for image modeling and retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):722-733, July 1996.
- [24] Lu Y, Hu C, Zhu X, Zhang H, Yang Q. A unified framework for semantics and feature based relevance feedback in image retrieval systems. In: Proceedings of the 8th ACM International Conference on Multimedia, pp. 31-37, Marina Del Rey, CA, 2000.
- [25] Mao, J. and Jain, AK. Texture Classification and Segmentation Using Multiresolution Simultaneous Autoregressive Models. *Pattern Recognition*, 25(2):173-188, 1992.
- [26] Rui Y, Huang TS., Chang S-F. Image retrieval: current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10:39-62 1999.
- [27] Smeulders A. W. M., Worring M., Santini S., Gupta A., Jain R. Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349-1380, 2000.
- [28] Smith MA. and Chen T. Image and Video Indexing and Retrieval Handbook of Image and Video Processing (Second Edition), Pages 993-1012, 2005.
- [29] Vogt CC, Cottrell GW. Fusion via a linear combination of scores. *Information Retrieval Journal*, 1:151-173, 1999.
- [30] J. Zhang and T. Tan. Brief Review of Invariant Texture Analysis Methods. *Pattern Recognition*, 35:735-747, 2002.

IntechOpen

IntechOpen

Multimedia

Edited by Kazuki Nishi

ISBN 978-953-7619-87-9

Hard cover, 452 pages

Publisher InTech

Published online 01, February, 2010

Published in print edition February, 2010

Multimedia technology will play a dominant role during the 21st century and beyond, continuously changing the world. It has been embedded in every electronic system: PC, TV, audio, mobile phone, internet application, medical electronics, traffic control, building management, financial trading, plant monitoring and other various man-machine interfaces. It improves the user satisfaction and the operational safety. It can be said that no electronic systems will be possible without multimedia technology. The aim of the book is to present the state-of-the-art research, development, and implementations of multimedia systems, technologies, and applications. All chapters represent contributions from the top researchers in this field and will serve as a valuable tool for professionals in this interdisciplinary field.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Noureddine Abbadeni (2010). Invariant Image Retrieval: An Approach Based On Multiple Representations and Multiple Queries, Multimedia, Kazuki Nishi (Ed.), ISBN: 978-953-7619-87-9, InTech, Available from: <http://www.intechopen.com/books/multimedia/invariant-image-retrieval-an-approach-based-on-multiple-representations-and-multiple-queries>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen