USE OF ORTHOGONAL ARRAYS TO AID RELEVANCE FEEDBACK IN CONTENT BASED IMAGE RETRIEVAL SYSTEMS

Raghavendra Singh and Ravi Kothari

IBM India Research Laboratory
Indian Institute of Technology Delhi, India
{raghavsi, rkothari}@in.ibm.com

ABSTRACT

Supervised learning algorithms (relevance feedback (RF) algorithms) are often used in content based image retrieval (CBIR) systems to enhance interactive search and browsing of image databases. One of the issues associated with RF based CBIR systems is the lack of a large training set. Labeling of images is a time consuming activity and user's usually do not have the patience to provide feedback on a large set. Thus the challenge is to select a "good" small training set in order to improve the retrieval performance of CBIR systems. In this paper we propose to use orthogonal arrays (OA), popularly used in design of experiments, in order to select this set. The property of OA that make them useful for CBIR systems is that they can be designed with or without using any prior classification information. We show that the top k retrieval accuracy increases rapidly as the number of labeled samples increase.

1. INTRODUCTION

To enhance interactive search and browsing in content based image retrieval systems (CBIR) systems, supervised learning algorithms, i.e., relevance feedback (RF) algorithms, are often used [1]. In a RF based CBIR system, a user provides feedback on the retrieved images. This feedback, which is most often of the form "relevant" or "non-relevant", indicates content that is of interest to a user. Using this feedback, an RF algorithm refines the search and attempts to retrieve increasingly relevant matches. There has been considerable work on both CBIR and RF based CBIR systems. An extensive review of existing systems appears in [1].

The generalization performance of a supervised learning algorithm is strongly dependent on the algorithm and on the quality and amount of labeled training data available [2] (and references therein). However in many situations finding a large representative labeled training data set is not an easy task. For example in a RF based CBIR system, the training set consists of the initial query and the set of retrieved images on which the user has provided feedback. Thus initially the training set is one or two images, and at each iteration the set is augmented by a small number of images.

One approach to training supervised learning algorithms using a small training set is to design the training set using concepts from the field of design of experiments (DOE) [3]. The primary purpose of this field is to create designs such that the relationship between a univariate response from the experiment and several predictor variables may be accurately modeled with an *efficient number* of design points [3]. A design point is a combination of predictor variables set at specific values and a trial of an experiment entails measuring the response produced at this design point.

In a CBIR system setting the purpose is to find the relationship between the user's preferences (response) and a representation of the images (predictor variables). An experiment would thus consists of finding a efficient number of images (design points) on which to request feedback from the user. This selected set of images could be dependent on the model being assumed for the relationship between the predictor variables and the response. For example in *active learning*, a DOE concept that has been used in CBIR systems [4, 2], given the current classification information, the set of most informative samples, i.e., samples that would maximally affect the classification accuracy, is found. Though active learning gives promising results, it requires prior information (classification information) and is hence dependent on the classifier or model, being used [4, 2].

Even if a model can not be assumed, DOE concepts can be used to create designs, e.g., space-filling designs, which spread the design points evenly over the region of interest [5]. An example of a space-filling design is orthogonal arrays (OA) [6, 5]. Rather than taking all possible combinations of the variables at various levels, OA choses an "optimal" fraction of combinations; which not only takes care of the the effects of the individual variables on the outcome, but also how the variables interact. OA have been used extensively in physical experiments and in meta-modeling; see the book by Hedayat *et al.* [6] for more details. In settings where the number of predictor variables or/and the number of levels that each variable can take is large, an OA will have a substantial number of design points. For such scenarios arrays that contain near-orthogonal design points (NOA) can be designed such that the number of design points is limited [7].

In this paper we propose to use the concept of OA to identify a set of images for a RF based CBIR system. This set of images is provided to the user in order to be labeled by him/her. In a CBIR system setting, the predictor variable space is the low-level feature space, thus there are large number of features and each feature takes a large number of values. Rather than using orthogonal design points (which may be very large) we propose to use a small number of near-orthogonal design points that are "spread" in the feature space. We provide three different algorithms for finding near-orthogonal design points in a color histogram feature space; each algorithm leads to a different distribution of the design points. These algorithms can be easily extended to the case where more features, such as texture or edge histograms are included in the feature space. We compare these three algorithms in terms of both their space filling properties and also their retrieval performance. We show that the top k accuracy increases rapidly as the number of labeled samples increase. Though OAs have been used in designing physical experiments, to the best of our knowledge, this is the first use of OA in CBIR systems. Note that each predictor variable can take different number of levels as mixed OAs can be defined [7].

The learning algorithm that we are using in this paper has been proposed by us in [2]. In relation to the size of the image database, the size of the training set is insignificant, i.e., the number of labeled samples (in the training set) is much less than the number of unlabeled sample (in the rest of the database). In this algorithm the role of the labeled and unlabeled samples are reversed. Implicit class memberships are assigned to the samples in the unlabeled data set, such that a classifier trained on these implicitly labeled samples can classify the explicitly labeled samples, in the labeled data set, with maximum accuracy. A classifier is then constructed on the basis of the explicitly *and* implicitly labeled samples. In the proposed CBIR system a binary classifier, that can classify the images into relevant and non-relevant classes, is learned from the feedback provided by the user. Once the classifier is trained, the top-*k* most relevant images are retrieved from the database.

0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Table 1. OA with $N = 8$ run, $m = 4$ variables, $l = 2$ levels of each variable, and strength $t = 3$.

The paper is organized as follows. In section 2 we review some of the related work both on design of experiments and on CBIR systems. In section 3 we present the proposed RF algorithm, and in section 4 experiments and results are presented.

2. ORTHOGONAL ARRAYS

An orthogonal array of size N, input variables m, levels l, and strength s, denoted $OA(N, m, l, s)$, is a $m \times N$ matrix D of l symbols such that all the ordered s-tuples of the symbols occur equally often as column vectors of any $s \times N$ sub-matrix of D [6]. Every $OA(N, m, s, t)$ defines an N-trial factorial design for m variables each having s levels. An example of an OA is shown in Table 1.

Each row of the matrix corresponds to a design point while the column corresponds to a variable. An experiment is performed with each of the row settings, the outputs observed and inference drawn on the relationship between the predictor variables and the response. In this design if we pick *any* $s = 3$ columns, we see that each of the possible combination $000, 001, 010, 011, 100, 101, 110, 111$, does appear and appear the same number of times (once). Thus by basing the experiment on an OA of strength s we ensure that all possible combinations of up to s of the variables occur together equally often. This allows us to investigate not only the effects of the individual variables (or factors) on the outcome, but also how the variables interact. The smallest number of rows that can occur in an OA is dependent upon the number of variables, their levels, and the strength. The strength is at maximum set to four invoking the sparsity of effect principle [3].

In cases where the number of predictor variables is large and the number of levels is also large, the resulting OAs will have a large number of design points. In order to reduce the number of design points substantially, near-orthogonal arrays [7] can be used. In this array the design points are chosen such that they are J_2 optimal, i.e., let us define a $m \times N$ matrix $D = [d_{ik}]$. Let column k have s_k levels; for $1 \leq i,\ j \leq m$, let

$$\delta_{ij}(D) = \sum_{k=1}^{N} w_j \delta(d_{ik}, d_{jk}) \tag{1}$$

where $\delta(x, y) = 1$ if $x = y$ and 0 otherwise. Assuming $w_i = 1\ \forall i$, $\delta_{ij}(D)$ value measures the similarity between the ith and jth rows of D. Define

$$J_2(D) = \sum_{1 \leq i,\, j \leq m} [\delta_{i,j}(D)]^2 \qquad (2)$$

A design is J_2 optimal if it minimizes J_2. From the estimation point of view, the difference between NOA and OA is that in NOA all main effects are still estimable but some of them are partially aliased with others [7]. Xu [7] has proposed an algorithm for efficiently finding NOAs. In this paper we modify and apply this algorithm to identify near-orthogonal images in a low-level image feature space. This is discussed in the next section.

3. FINDING NEAR ORTHOGONAL IMAGES IN A CBIR SYSTEM

The purpose of the algorithms proposed in this section is to identify a set of near-orthogonal images in a region of interest of the feature space. These near-orthogonal images will then be provided to the user in order to be labeled; and based on this labeling the RF algorithm will learn the user's preferences. There are different ways in which the region of interest can be identified, e.g., the top k most relevant images, or the top k most informative images, etc; we will discuss this in more detail in the experiments section (sec. 4). Below we discuss three algorithms for identifying the near-orthogonal images. The input to these algorithms are the features of all the images in the region of interest, and the number of desired near-orthogonal images N. The output is the identified set of near-orthogonal images.

Algorithm 1 (Algo-1) In this algorithm we first find the unique values of each feature and sort these values. For each feature k the number of levels s_k is the number of unique values and a level is the index in the sorted value vector, e.g. if a feature has values $\{0.3\ 0.2\ 0.4\ 0.2\}$, then the number of levels is 3, and the levels associated with the values are $\{0\ 1\ 2\ 1\}$. For all the features in the region of interest a matrix X is formed, where x_{ij} is the level associated with the feature j of the image i. Initially a random row is selected from this matrix; then at each step a row that has the minimum J_2 distance from the already selected rows is chosen, till N rows are chosen.

Algorithm 2 (Algo-2) Algo-1 does not take into account the distribution of points in the region of interest; it assumes that the points are uniformly distributed over the space and finds the most dissimilar points in the space. However in an image database the feature space is not uniformly distributed; there are clusters and there are outliers. Algo-1 may often pick up outliers and the feedback on these outliers may not lead to good generalization performance of the learning algorithm. In this algorithm we set the weight w_i associated with ith row of X as the the inverse of the mean distance of the top 10-nearest neighbors of the row (image) i. Thus a row corresponding to an image in a cluster will have a lower weight, than a row corresponding to an outlying image. Algo-1 is run to find the N near-orthogonal images with this weighted J_2 metric.

Algorithm 3 (Algo-3) In this algorithm we first discretize the features in the region of interest such that each feature can take only a small finite number of levels. There are many feature discretization algorithms, for an extensive review see [8]. For each feature we have used a scalar quantizer with the property that if there are s_k bins in the kth quantizer, then each bin of the quantizer approximately quantizes $1/s_k$ of the points in the region of interest; i.e., each bin of the quantizer contains approximately equal number of points. These scalar quantizers will partition the n-dimensional feature space into (approximately) equi-probable clusters. The centroid of each cluster is found and mapped to the closest image in the region of interest. Then Algo-1 is run over the cluster centers to chose centers that minimize the J_2 distance.

4. EXPERIMENTS

In this paper we have tested our algorithm with the Corel image dataset. We randomly chose ten classes out of the dataset: *Tigers, Cities of Italy, Arabian Horses, English County Gardens, Cheetahs Leopards and Jaguars, Bald Eagles, Rome, Land of Pyramids, Ocean Life and Interior Design*. We have not tried to choose the set of 10 classes to maximize the inter-class separation, rather they were chosen at random. Each of these classes have about 100 images for a total of 963 images. For feature extraction we used a 27-bin HSV space color histogram. We mapped this 27 dimensional vector to a 8 dimensional space using principal component analysis [9].

Our first experiment is to compare the algorithms presented in section 3 using space filling measures. Comparisons were made with two types of common measures [5]: one which measure the average distance between design points Δ_{dd} and one which measures the average distance between a design point and the top-10 nearest neighbors of the design point (that are not in the design) Δ_{nd}. The results are shown in the Table 2, region of interest is the entire image database. Clearly Algo-2 is able to select reasonably dissimilar design points that also belong to a cluster.

	Δ_{dd}	Δ_{nd}
Algo-1	0.5397	0.0688
Algo-2	0.1887	0.0620
Algo-3	0.1602	0.0499

Table 2. Space filling measures for design points obtained through different algorithms.

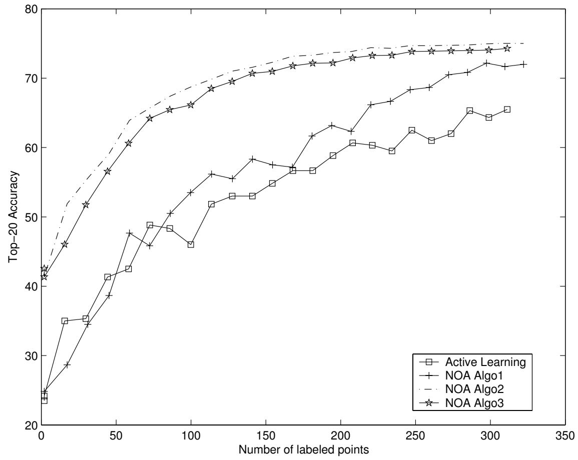

Fig. 1. Precision-Recall curve. The mean values across all classes are plotted. In *Active Learning* only images that are confusing are provided to the user to be explicitly labeled. In *Algo 1-Algo 3* near-orthogonal images are used along with active learning to populate the feedback set.

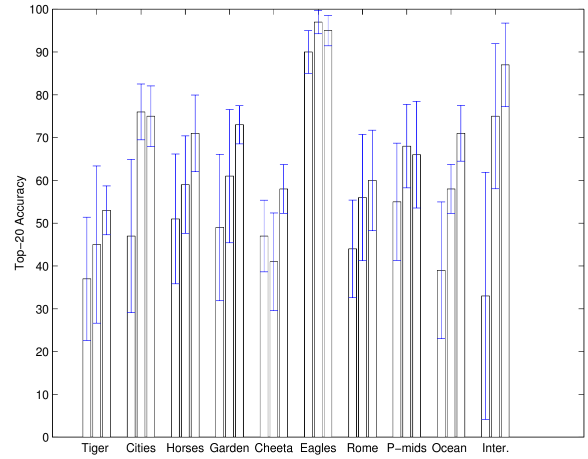

Fig. 2. Top-20 Accuracy for each class for the experiment that uses NOA (Algo-2) with active learning. The bar represents the mean result while the line segment on the bar shows the variance. The first bar for each class is obtained with 100 labeled points; similarly the second, third , and fourth are obtained using 200, 300, and 400 labeled points respectively.

For each retrieval experiment we randomly divide the data set into three sets, the labeled data set, the unlabeled data set and test data set. Samples from the unlabeled data set are chosen for explicit labeling and the retrieval performance is measured over the test data set. The experiments have been run multiple times for each class to ensure statistically correct results. The results are shown in Figs. 1-2. The results of the proposed classifier, without OA, have been provided in [2] and will not be duplicated here, except to mention that there is an improvement of about 10-40% over performance of a learning algorithm that uses only labeled images. Fig 1 shows the average (over 10 classes) top-20 accuracy of the RF algorithm when four different experiments are performed. In *Active learning* the top-20 most confusing samples are shown to the user at each iteration to the user to be labeled by him/her. Confusing is measured in terms of classification of a sample across a committee of classifiers [2]. In the other experiments near-optimal images selected by the different algorithms (presented in section 3) are chosen to be labeled by the user. For each of these experiments, the region of interest in the first iteration are all the images that any of the multiple classifiers [2] have found to be relevant. From the second iteration onwards, when the classifiers have some classification information, the region of interest are all the images that the multiple classifiers disagree on, i.e., find confusing. The results show that the retrieval performance of near-orthogonal images found by Algo-2 is the best. In Fig. 2 the top-20 results per class are shown for the experiment where NOA selected by Algo-2 are used.

In this paper we have shown that a "good" feedback set for RF based CBIR systems can be chosen using DOE concepts such as OA. This is particularly useful when there is little or no classification information available. We are currently working on a more extensive of experiments with a larger feature space.

5. REFERENCES

[1] X.S. Zhou and T.S. Huang, "Relevance feedback in image retrieval: a comprehensive review," *ACM Multimedia Systems Journal*, 2002.

[2] R. Singh and R. Kothari, "Relevance feedback algorithm based on learning from labeled and unlabeled data," in *In Proc. ICME03*, 2003.

[3] D.C. Montgomery, *Design and Analysis of Experiments*, John Wiley and Sons, 1997.

[4] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proc. of ACM Multimedia*, 2001.

[5] V. C. P. Chen, K-L Tsui, R. R. Barton, and M. Meckesheimer, "Design, modeling, and applications of computer experiments," *IIE Transacations*, 2003.

[6] A. S. Hedayat, N. J. A Sloane, and J. Stufken, *Orthogonal Arrays: Theory and Applications*, Springer-Verlag, 1999.

[7] H. Xu, "An algorithm for constructing orthogonal and nearly-orthogoanl arrays with mixed levels and small runs," *Technometrics*, vol. 44, pp. 356–368, 2002.

[8] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and unsupervised discretization of continuous features," *Proc. of 12th Int. Conf. on Machine Learning*, 1995.

[9] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.