

Inducing a Perceptual Relevance Shape Classifier.

Victoria J. Hodge
Dept of Computer Science
University of York
York, UK
+44 1904 433067
vicky@cs.york.ac.uk

John Eakins
Dept of Computer Science
University of York
York, UK
eakins@cs.york.ac.uk

James Austin
Dept of Computer Science
University of York
York, UK
+44 1904 432734
austin@cs.york.ac.uk

ABSTRACT

In this paper, we develop a system to classify the outputs of image segmentation algorithms as perceptually relevant or perceptually irrelevant with respect to human perception. The work is aimed at figurative images. We previously investigated human visual perception of trademark images and established a body of ground truth data in the form of trademark images and their respective human segmentations. The work indicated that there is a core set of segmentations for each image that people perceive. Here we use this core set of segmentations to train a classifier to classify closed shapes output from an image segmentation algorithm so that the method returns the image segments that match those produced by people. We demonstrate that a perceptual relevance classifier is attainable and identify a good methodology to achieve this. The paper compares MLP, SVM, Bayes and regression classifiers for classifying shapes. MLPs perform best with an overall accuracy of 96.4%.

Categories and Subject Descriptors

I.5.4 [Pattern Recognition] Applications - *Computer vision* I.5.1 [Pattern Recognition] Models - *Neural nets, Statistical* I.2.10 [Artificial Intelligence] Vision and Scene Understanding - *Perceptual reasoning, Representations, data structures, and transforms, Shape* I.4.6 [Image Processing And Computer Vision] Segmentation - *Edge and feature detection* I.4.7 [Image Processing And Computer Vision] Feature Measurement -- *Feature representation, Invariants, Moments, Size and shape.*

General Terms

Performance, Experimentation, Human Factors, Verification.

Keywords

Perceptual relevance, classification, image segmentation, perceptual classifier, human image segmentation.

1. INTRODUCTION

There has recently been tremendous growth in the storage of digital imagery producing a need for accurate and fast indexing

and retrieval systems. Examples of applications include archiving images or photographs, medical image analysis and trademark retrieval. In Content-based Image Retrieval (CBIR) the aim is to retrieve images from an image database that are similar to a query image. This process may be performed by matching the whole image as a single entity or matching components within each image [8]. In this work, we focus on component-based similarity matching of trademark images.

Our work forms part of the PROFI (Perceptually-Relevant Retrieval of Figurative Images) project [See section 6]. In PROFI, we aim to develop new techniques for the retrieval of figurative images (i.e. abstract trademarks and logos) from large databases and, in particular, aim to reproduce the matches that people find by manual methods on this task. The techniques are based on the extraction of perceptually relevant shape features and the matching of these features in the target image against features in the stored images. The first stage of this procedure is to identify the components present within an image. As our aim is to return the images from the automated system that people would say were similar, we believe that this segmentation process should reflect human perception and segmentation. The principal difficulty for image segmentation algorithms in the context of our work is the selection of parts that accurately reflect the image's appearance to a human observer.

To obtain a base line for the human performance on the task, we have previously conducted a set of experiments investigating human segmentation of trademark images [10,11]. The experimental results detailed in the two papers and outlined in section 2 concur with previous investigations such as [17] in that human image segmentation appears to follow a set of perceptual principles analogous to the Gestalt laws [15,25]. The experiments and analyses show that these Gestalt laws interact and possibly conflict as noted by [6]. The experiments also indicate that there are a core set of segmentations for each image perceived by two or more people along with a set of segmentations seen only by individuals. This core set of segmentations forms the ground truth for our evaluations into inducing a perceptual relevance classifier. It is vital for any computerised image segmentation algorithm to include a perceptual relevance classifier, effectively a global goodness score. This allows the algorithm to reduce the number of segmentations output and to focus on perceptually relevant shapes whilst, hopefully, discarding irrelevant segmentations.

The first stage is to identify the shapes present in an image. To do this we require a shape identification algorithm. In practice any closed shape identifier could underpin the procedure, such as region growing [28], watershed [2] or closed shape identification [1] provided the result of the algorithm may be represented by a

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CIVR'07, July 9-11, 2007, Amsterdam, Netherlands.

Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

list of boundary points to calculate the attributes used here. It is the classification process where we focus our research here, not the underlying shape identification algorithm. We use Saund's closed shape identification algorithm [22] here. It was developed for the sketch retrieval domain to identify shapes within human sketches but is equally applicable to the trademark retrieval domain. It aims to find closed shapes satisfying global criteria and has similarities to our aim of classifying perceptual relevance.

We aim to use our previous empirical evaluations of human perception to induce a classifier that classifies the closed shapes output by a closed shape identification algorithm as perceptually relevant (to keep) or perceptually irrelevant (to discard). This would effectively replace the global goodness measure used in Saund's method. To do this we require a set of attributes to represent each shape as a vector and a classifier to classify the shapes output for relevance.

To decide which attributes to use, this work takes its cue from the attributes elicited by Alwis [1], Chan & King [4], ARTISAN [8] and QBIC [9]. Alwis [1] has produced a trademark retrieval system, with many similarities to the work here, so the features he used are particularly relevant for the work here: circularity, aspect ratio, stuffedness, "right angledness", sharpness, complexity, directness and straightness. Chan & King [4] propose a method for feature weight assignment in a trademark system. They use invariant moments, Euler number, eccentricity, and circularity in their evaluations. ARTISAN [8] is "regarded as one of the most comprehensive trademark retrieval systems in the current literature" [13]. ARTISAN uses component-based matching so the features used by ARTISAN should be relevant: aspect ratio, circularity, convexity, the Fourier descriptors, the shape's area and the three 'natural' shape measures defined by Rosin [18]: rectangularity, triangularity and ellipticity for trademark retrieval. The IBM QBIC [9] system is one of the most ubiquitous image retrieval systems developed and has been used widely so the features used for matching should be relevant to our developmental system. The shape features used in QBIC consist of shape area, circularity, eccentricity and a set of algebraic moment invariants.

In this work, we analyse a series of common classifiers to verify that it is possible to classify perceptual relevance using human classifications and to pinpoint the classifier that achieves the highest recall accuracy while maintaining recall consistency. We assess four supervised learning classifiers: Naïve Bayes [14], Multi-Layer Perceptron (MLP) [19], Support Vector Machine (SVM) [24] and Regression [20]. Naïve Bayes is a simple statistical model linear classifier that often outperforms more sophisticated classifiers [26]. Standard regression is a statistical model linear classifier aimed at classification with numeric attributes such as we use here. Non-linear statistical model classifiers such as the MLP or SVM can model non-linear class boundaries and are usually robust to outliers in the data. These four classifiers together thus provide a broad cross-section of classifier technology.

In the remainder of this paper, we describe: our previous human segmentation experiments, the underlying closed shape identification algorithm that we used for our analyses, the 23 attributes used to represent each closed shape, the data used to perform our classification analyses, the four classifiers we have

evaluated for recall accuracy, the methodology we use for our evaluations, the results, analyses and conclusion inferred.






2. HUMAN PERCEPTION ANALYSES

To test the system, it has been necessary to collect ground truth data from human subjects on how individuals segment images – thus asking the question: "what are the human segmentation preferences?" The following explains how we collected this data and summarises the work published in [10,11].

In our human perception experiments, 53 subjects each received 32 trademark like images in a booklet. The subjects were requested to draw (using pen or pencil) their perceived segmentations of each image in turn on to the booklet. We collated the segmentations drawn by the subjects and produced a listing of all segmentations for each image in turn. For our work here, we only consider segmentations seen by 2 or more people which represent our core set of segmentations that the trademark system should output to represent each image.

Table 1 shows an example image and the human segmentations perceived for that image. The human subjects perceived four different segmentations – they comprised the following number of components (shapes): 5, 2, 3 and 1 components respectively. We identify these as the perceptually relevant components (shapes) for this image which the closed shape identification algorithm should ideally identify.

Table 1 Table showing an image (top row) and the four segmentations seen by 2 or more people for that image.

| | |
|--|---|
|  | |
|  |  |
|  |  |

3. CLOSED SHAPE IDENTIFICATION.

To identify the closed shapes in the image, we use Saund's method as pointed out above. This method requires an underlying algorithm to identify line segments in an image and the relationships between those line segments. Therefore, we initially find the edges in an image and subdivide these into constant curvature segments using the Sarkar & Boyer [21] edge detection algorithm and the Wuescher & Boyer [26] curve segmentation algorithm. These methods were selected as they had successfully been used in the trademark system developed by Alwis [1]. The Sarkar & Boyer method finds the edge lines in an image and splits

these lines into primitives. Wuescher & Boyer performs some aggregation of these primitives into more perceptually-oriented constant curvature segments and outputs these as a list of constant curvature segments. These segments provide the building blocks for our closed shape identifier. Our aim is to group these constant curvature segments using Gestalt like methods to produce a graph of segment relations which will underpin the Saund closed shape identification algorithm. To produce this graph we use the following methods. Each constant curvature segment becomes a node in the graph with two ends (first point (denoted as an x, y coordinate) and last point (also denoted as an x, y coordinate)). We find all segments that are end-point proximal. We use Lowe's method [16] to extract endpoint proximity by comparing two lines with lengths l_1 and l_2 or curves with perimeter lengths l_1 and l_2 . In the following, ($l=l_1$ if $l_1 < l_2$ else $l=l_2$). The distance between their endpoints is r . The inverse significance of endpoint proximity between them is $\frac{r^2 \rho}{l^2}$. The parameter ρ is a unit-less constant and may effectively be ignored (i.e. set to 1). So if: $\frac{r^2}{l^2} < \text{threshold}$

where threshold = 0.01 then the two endpoints are joined. This effectively joins the graph by linking the proximal end-points.

The Saund algorithm overlays this and focuses on managing the search of possible path continuations through the graph particularly where the graph nodes represent junctions (crossroads, t-junctions etc) of lines in the original image. The search is managed through the use of local criteria for prioritising the order in which paths are pursued. Saund has identified criteria (scores) for ranking possible paths through junctions based on observations. Path scores accumulate by multiplying junction preference scores as the path progresses.

The closed path search commences from each end (first and last) of each node (line segment) identified by the underlying Wuescher & Boyer algorithm. For each end (first then last) in turn, all possible paths are followed. This effectively forms a search tree with paths through the tree representing the paths of candidate shapes. As each leaf node in the tree is expanded, any new child nodes are compared with child nodes in the opposite side of the tree. If they are end-point proximal then a closed path has been identified and its nodes and pixels are added to the list of candidate paths. All closed paths exceeding a threshold score are thus stored as candidate paths. Saund terminates searching when a closed path score exceeds a pre-specified threshold. Saund accepts a closed path as a candidate if its cumulative junction score exceeds 0.6 or accepts the closed path and terminates search from the particular root node if the score exceeds 0.9. We do not terminate search if the score exceeds this threshold as we feel potential closed paths may be missed due to higher scoring and shorter paths terminating the search prematurely. Hence, all paths that exceed 0.6 are accepted as candidates but search continues.

Saund discards closed paths that are subsumed by other closed paths with higher scores. Hence, each new closed path is compared to all existing stored paths. If the new path is a subset (including the equivalence set) of an existing path but has lower score then the new path is discarded. If the new path has higher score than the existing saved path then the saved path is discarded.

3.1 Determining Good Shapes

In Saund's methodology, all paths accepted as candidates are then assessed for global figure goodness by awarding a score. In Saund's approach the score for each closed path is produced multiplicatively ($C*N*E$) where C, N and E are:

Compactness (C) - the ratio of [figure area: area of convex hull].

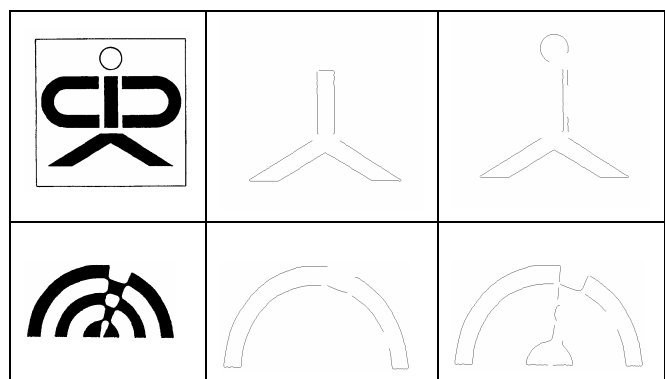
End-point distance (E) - calculated using $1 - d_e/p$ where d_e is the distance between endpoints of the path and p the path length.

Non-end-nearest-approach (N) which penalises paths where an endpoint terminates near the body of the path.

This method does not produce the perceptually relevant closed figures identified by our experiments. This is where our work has changed the method: by adding a perceptual classifier taught using the data from human experiments. Through a brief comparison, we identified that, of the three Saund attributes, only C (which we call areaScore) matches to some extent the human preferences from our experiments.

The output of our implementation of the Saund algorithm is therefore a list of candidate closed shapes found in the image. These closed shapes are the candidate shapes whose cumulative junction score exceeds 0.6. Each candidate shape is classified as relevant or irrelevant using our perceptual classifier and only shapes classified as relevant will be retained for further processing. The candidate closed shapes are represented by a list of x, y coordinates representing each point on the shape's boundary (in order with no gaps). Two example images with one relevant shape and one irrelevant shape identified by our implementation are shown in Table 2. The classifier should classify the relevant shapes as relevant and the irrelevant shapes as irrelevant thus slowing us to discard the irrelevant shapes from any further processing.

Table 2 Table listing 2 images (leftmost column) and two paths identified by the Saund algorithm for each image (one perceptually relevant (middle column) and one perceptually irrelevant (right column)).



3.2 Attributes

As outlined above, we use a classifier to determine which closed figures output from our implementation of the Saund algorithm are perceptually relevant. The classifier has to be trained on the data collected from our ground truth experiments described in Section 2. The selection of the attributes for the classifier is considered as follows.

Each output is a list of boundary points (x, y coordinates) of each closed shape. We produce various attributes from the boundary points thus representing each closed shape as a vector of 24 attributes: 23 attributes calculated from the list of boundary points (x, y coordinates) plus the class (perceptually relevant or irrelevant). Note that the 23 attributes are not independent of each other; many are closely related such as AreaRatio and Roughness; it is the job of the classifier to determine the optimum set of attributes. The following attributes are calculated:

Roughness = Perimeter / Convex Hull Perimeter

AspectRatio = Perimeter / Min. Area Bounding Box Perimeter

Stuffedness = Area / Min. Bounding Box Area

AreaRatio = Area / Convex Hull Area

GapScore = Max. Gap in Perimeter / Perimeter

Circularity = $4\pi * \text{Area} / \text{Perimeter}^2$

Eccentricity = $\frac{(M_{20}-M_{02})^2+4M_{11}^2}{(M_{20}+M_{02})^2}$ where M is calculated using

the centroid thus $M_{pq} = \sum_{i=0}^{N-1} (x_i - C_x)^p \times (y_i - C_y)^q$

Ellipticity = $\begin{cases} 16\pi^2 I_1 & \text{if } I_1 \leq \frac{1}{16\pi^2} \text{ where } I_1 = \frac{\mu_{20}\mu_{02} - \mu_{11}^2}{\mu_{00}^4} \\ \frac{1}{16\pi^2 I_1} & \text{otherwise} \end{cases}$

and $\mu_{pq} = \sum_x \sum_y (x_i - C_x)^p \times (y_i - C_y)^q \times f(x_i, y_i)$

Triangularity = $\begin{cases} 108I_1 & \text{if } I_1 \leq \frac{1}{108} \\ \frac{1}{108I_1} & \text{otherwise} \end{cases}$

Hu Moments [12] (from boundary points).

Fourier coefficients (from boundary points): The Fourier coefficients are the amplitude of the Fourier expansion of the cumulative angular bend around the shape's boundary points with $0 \leq k \leq 6$ here.

$$c[k] = \frac{\sqrt{\left(\sum_0^{n-1} \left(\theta_j \cos\left(\frac{-2\pi jk}{n}\right)\right)\right)^2 + \left(\sum_0^{n-1} \left(\theta_j \sin\left(\frac{-2\pi jk}{n}\right)\right)\right)^2}}{n}$$

3.3 Data Preparation

To allow the system to classify the data from these attributes, we first collected a set of data from the ground truth images. All images from the experimental set of 84 images [10,11] which contained texture were discarded as texture confuses shape identifiers and produces very poor segmentation results. The underlying line segmentation algorithm finds a large number of edges in texture data. To do this we discarded all images that produced more than 500 shapes as this is too many to process by hand. This left 48 images.

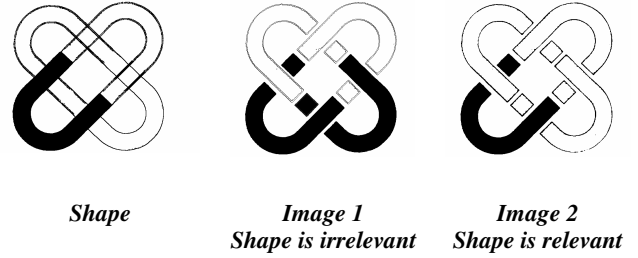


Figure 1. The leftmost shape (two interlocking loops) is relevant for the rightmost image but irrelevant for the middle image.

Our experiments indicated that there are a core set of segmentations for each image perceived by 2 or more people. From the chosen 48 images, we ran the closed shape identifier (described in section 3) and selected (by hand) shapes output by this algorithm that were perceptually relevant (matched shapes within the segmentation drawn by 2 or more human subjects) and shapes that were perceptually irrelevant (very dissimilar from the shapes drawn by the human subjects). We tried to balance the number of relevant with the number of irrelevant examples from each image although this is not always possible.

We note that for our analyses here, it is important to choose relevance/irrelevance carefully. We are representing the global picture; one shape that is relevant for one image may in fact be irrelevant for another similar image containing that shape as shown in figure 1. We took this into account when preparing our training/test sets for the classifiers and only selected shapes that were perceptually irrelevant across the board. The final classifier is a global classifier; it is not trained on a per image basis so we need the global relevance picture which needs careful consideration.

From the 48 images available, we used 29 images to produce an original training set comprising 435 records and 19 images to produce an original test set comprising 306 records giving a total data set size of 741 records. This represents all of the data we had available. We labelled these two data sets: **set1** and **set2** respectively. We then extracted the top half of the training set1 and the top half of the test set2 to produce **set3** comprising 371 records. When splitting the data sets in half, we ensured that all records from a particular image were kept together in one half or the other. **Set4** which contains 370 records is the bottom half of the training set1 and the bottom half of the test set2. **Set5** comprises 365 records and is the top half of the training set1 and the bottom half of the test set2 and finally we merged the bottom half of the training set1 and the top half of the test set2 to produce **set6** with 376 records. This subdivision allows us 6 runs of each classifier with a training set and a test set. In these analyses, standard x-fold cross validation is not feasible as the data set contains images that are variants of other images (altered according to Gestalt principles) so the constituents of the test and training sets must be considered carefully to prevent biasing and instability and we tried to prevent this by splitting the sets carefully. Also, we cannot split the records for each image, for example if image 1 produced 10 relevant and 10 irrelevant shapes then these must all be kept together in one set to prevent biasing

of the data. We are searching for a classifier that generalises well so all equivalent data must be together but image variants may be split in a considered way.

3.4 Classifiers

In the work we assess four classifiers to select the perceptually relevant and irrelevant shapes from the images, these are Naïve Bayes [14], MLP [19], SVM [24] and Regression [20]. These were selected as the most common methods used currently. The work aims to pinpoint the best (highest recall coupled with highest recall consistency) classifier for classifying the outputs of the segmentation algorithm. The Naïve Bayes does not require parameter setting so we do not tune that algorithm. We ran various configurations (as outlined below) of the MLP and SVM algorithm on all 6 train/test set combinations and selected the configuration for each classifier with the highest recall. All classifiers use identical sets and are free to choose any attributes from each training set in turn. The relatively small size of the data prevents us using train and validation sets prior to classifying a blind test set. The regression algorithm does allow tuning but is slow to run (up to 1 day) with some configurations. For this algorithm, we ran some initial analyses and selected the best performing (highest recall accuracy) configuration when classifying the train set (set1) only.

Naïve Bayes assumes that the attributes $X = \{x_1, x_2, x_3, \dots, x_d\}$ are independent to simplify the classification task by allowing the class conditional densities $p(x_k | C_j)$ to be calculated separately for each attribute. This assumption appears not to affect the posterior probabilities greatly, especially in areas near decision boundaries, thus, leaving the classification task unaffected. We use the Naïve Bayes C source code available from [3] running under Linux.

The MLP neural network is a feed forward topology with a single hidden layer comprising 23 input neurons, a hidden layer of neurons and a single output neuron. We have 23 neurons in the input as there are 23 attributes in the input data and a single output neuron to represent perceptual relevance for these analyses. Selecting the number of hidden neurons is important. We tried various settings to choose the optimal configuration of the MLP. We selected between 3 to 12 hidden neurons and ran each MLP on each of the 6 train/test set combinations (60 runs in total). An MLP with 4 hidden neurons produced the highest recall. We then ran the 4 hidden neurons MLP for between 1000 and 7000 epochs. The MLP recall percentage increases from 1000 to 2000 to 3000 training epochs. However, with 4000 epochs, recall accuracy degrades markedly and remains worse for both 5000 and 7000 epochs which indicates overtraining. Thus, 3000 epochs produced the best results coupled with 4 neurons in the middle (hidden) layer. Multi-layer networks use a variety of learning techniques; we use back-propagation where the output values are compared with the correct answer during network training to compute the value of the error-function. In our analyses here, we use the MLP C source code available from [3] running under Linux.

For the SVM, we use C++ source code (LibSVM) available from [5] running under Linux. We use the nu-SVC SVM type (where nu is related to the ratio of support vectors and the ratio of the training error) with radial basis function kernels ($\exp(-\gamma * |x_j - z|^2)$). All data attributes are scaled in the range [0, 1]. We used the script available with libSVM (grid.py) to select values for γ in the

kernel function. This recommended 1.0, 0.125 and 0.0625. We then ran the SVM on the 6 train/test set combinations with each of these three γ settings along with 0.5 and 0.25. A setting of 0.125 produced the highest overall recall. With γ set to 0.125 we tried various nu-values (0.1, 0.3, 0.4, 0.5 (default) and 0.6). 0.4 produced the highest recall figure.

For the regression analyses, we use the Sagata regression program available from [20] which provides proprietary regression algorithms. It runs under MS Windows XP and sits on top of MS Excel. Our preliminary analyses which involved generating the regression equation using the training set1 and then classifying the same training set1 indicated that the combination of selecting an initial set of attributes using the MinPress regression algorithm with default settings then using standard stepwise with order up to 2 attributes followed by Least Squares regression to select the equation coefficients produced the highest recall.

MinPress is similar to stepwise regression except that attributes are selected based on improvements in the Press statistic defined as:

$PRESS = \sum_{i=1, \dots, n} w_i [y_i - y^{(i)} \cdot est(x_i)]^2$ where $y^{(i)} \cdot est(x_i)$ is the prediction at the data point x_i . Inputs, classes, and weights for the x_i -th record are omitted. The same model is fitted to the data minus the x_i -th record. This fitted model is used to make a prediction for x_i . This is $y^{(i)} \cdot est(x_i)$.

Once we have used MinPress to select an initial set $\{S_1\}$, we supplement this set with a set of 2nd order attributes $\{S_2\}$ selected using standard stepwise regression [20].

We merge $\{S_1\}$ and $\{S_2\}$ giving the selected attributes $\{S\}$. We use Least Squares estimation (LSE) to select the regression equation coefficients: LSE derives the regression equation coefficients that minimize the sum of squared differences (residuals) between the regression equation predictions and the corresponding actual response (class) values (0 or 1 here).

3.5 Method

The SVM and Naïve Bayes are discrete classifiers; each record is classified as relevant or irrelevant so we use the classes $\{0, 1\}$. In contrast, the regression algorithm and MLP produce continuous classifications in the range $[0, 1]$. For classification (testing), we use a threshold value of 0.5 for the regression and MLP outputs. If the predicted output class score is >0.5 then the record is classified as relevant but if the output score value is ≤ 0.5 then we classify as irrelevant.

Each classifier is trained and tested with one pair of sets in turn and the outputs stored for recall accuracy calculation. Each classifier produces 6 separate equations/models for the data.

To measure success we recorded overall recall accuracy, (i.e., the number of perceptually relevant examples classified as perceptually relevant plus the number of perceptually irrelevant examples classified as perceptually irrelevant) and the recall accuracy for the perceptually relevant examples. False positives (irrelevant shapes classified as relevant) increase the amount of data to be processed which is a nuisance factor but less serious than false negatives (relevant shapes classified as irrelevant) which indicate missing perceptually relevant shapes.

For example, for the pair “Train using set1 + test using set2”, we train the classifier with the 435 records in set1 to produce a classifier model. For each of the 306 records in set2, we apply this classifier model to the record to produce a class prediction (perceptually relevant or perceptually irrelevant). We can then calculate the recall accuracy by counting the number of correct predictions and dividing this figure by the number of records in the test set. The set pairs are {train, test}: {set1, set2}, {set2 set1}, {set3, set4}, {set4, set3}, {set5, set6}, {set6 set5}

4. RESULTS & ANALYSIS

The recall accuracy for the four classifiers when run on each of the 6 train/test set combinations is listed in Table 3.

From Table 3, the MLP algorithm has the highest overall recall coupled with the highest recall accuracy for perceptually relevant and perceptually irrelevant shapes by a considerable margin. It also has consistently high recall. The regression algorithm produces the second highest recall figures with the SVM third highest overall. The Naïve Bayes performs worst except for correctly classifying the perceptually irrelevant shapes where it is third best.

It is noted that the size of the training set can have an adverse affect on classifier recall accuracy. The MLP performs worst on the smallest set2 for training with the largest set1 for testing combination and conversely performs best when training on the largest set1 and testing with the smallest set2. The overall recall drops from 98% to 94% so the MLP may be adversely affected by training set size. When the SVM trains on the larger set1 and classifies the smaller set2 it produces 93% recall accuracy. Conversely, when the SVM trains on the smaller set2 and classifies the larger set1 the SVM produces 80% recall accuracy. However, the SVM suffers its worst performance when training with set5 and testing with set6 so we cannot say conclusively whether it is adversely affected by training size at this stage. The Naïve Bayes also suffers a performance drop when using the smallest training set2 compared with the largest training set1 but similarly, Naïve Bayes suffers its worst performance when training with set6 and testing with set5 so again, we cannot say conclusively whether it is adversely affected by training size at this stage. The regression algorithm does not suffer a significant overall performance drop when comparing the largest and smallest training sets.

It is possible to look at the weights in an MLP to see which attributes are being used to classify the shapes. Roughness is weighted consistently highly (either +ve or -ve). AspectRatio, Stuffedness, AreaRatio, GapScore, Circularity, Eccentricity, Ellipticity & Triangularity are generally weighted high. The Fourier descriptors are occasionally weighted highly and the Hu moments are generally weighted low. Roughness indicates how convex a shape is. A high score indicates that the shape fills its convex hull and thus the shape is convex. Conversely, a low score indicates a concave shape. This corresponds with visual observations of the results of our experiments: for images comprising flood-filled regions in particular, convex shapes tend to be perceptually relevant. Where concave shapes are relevant (generally more so for line-based images or thin regions from our experiments) the MLP may use a combination of the other attributes to achieve the correct classification. AreaRatio is very similar to Roughness so we would not expect both to score highly.

Circularity, Ellipticity, Triangularity and Stuffedness (with AspectRatio closely related to Stuffedness) all define specific shapes (circle, ellipse, triangle, and rectangle) and hence their applicability varies. GapScore is often weighted highly indicating that shapes with gaps vary in perceptual relevance from shapes without gaps in their perimeter. Eccentricity measures the regularity of a shape and we would expect regular shapes to be more perceptually relevant than irregular shapes. This hypothesis is borne out with the attribute’s relatively high weighting.

It is interesting to consider the speed of training, as this indicates the utility of the method for practical applications. For the implementations of the four classifiers used here, the Naïve Bayes trains fastest, followed by the SVM and the MLP all of which train much faster than the regression program. For the data set combination set6 training and set5 testing, the MLP trains in 2.0 seconds, the SVM in 0.4 seconds and the Naïve Bayes trains in 0.3 seconds all running on a 3.4GHz Pentium PC with 2GB RAM running Linux. The regression program takes 40m 24s (2424 seconds) to complete the three steps of regression training on the same data set pair running on a dual 2.8GHz AMD Athlon PC with 3GB RAM running MS Windows XP with the regression program running on top of MS Excel. Obviously, we are comparing slightly different machines and different operating systems (3 C++ algorithms running under Linux on 3.4GHz Pentium PC and one Windows application running on dual 2.8GHz AMD Athlon PC with MS Windows XP) but the time difference between the C++ algorithms and the regression algorithm is still significant if speed is the overriding criterion for the user.

5. CONCLUSION

The work has shown that it is possible to train a classifier to select perceptually relevant closed figures from a segmented image, effectively capturing the segments that humans see in images. Our work has shown that the MLP network can be trained to achieve this with 96.4% recall accuracy overall. The MLP has the highest recall for the important category: the perceptually relevant examples, where it achieves 97.4% accuracy. It is noted that the training time for the MLP is 2 seconds compared to 0.3 seconds for the Naïve Bayes which trains fastest. Although the MLP is slower, the training time is still fast. Therefore, we have identified that an MLP with 4 hidden neurons and a single output neuron running as the optimum perceptual relevance classifier for the perceptual classifier task described in this paper.

We feel the approach described is very flexible and attained using actual human perceptual data. It is a universal approach providing a score of perceptual relevance (global goodness) across all shapes regardless of how they are derived. The approach reduces the number of shapes output by the closed shape identification algorithm and is a precursor to the matching phase of image retrieval. Classifying the closed shapes and discarding perceptually irrelevant shapes reduces the search space during image matching and retrieval. Each image is only represented by a sub-section of the candidate shapes output by the closed shape algorithm; the shapes classified as perceptually irrelevant are removed from the search space. Reducing the search space focuses on human-oriented shapes, speeds further processing during image matching and retrieval as fewer shapes need to be

processed and reduces the memory overhead of any further processing.

We intend to use the classifier within the PROFIL project to generate perceptually relevant views of each image. The closed shape identifier produces a set of candidate shapes for each image. The classifier then reduces the set of candidates to the set of perceptually relevant shapes for that image. For all images combined, these reduced sets of shapes represent the database of perceptually relevant shapes for all images. Using shape attributes (such as the 23 attributes detailed in section 3.2, topology attributes such as touching and overlap relations and position attributes such as the centroid coordinates) to represent the shapes as a vector of attributes, the set of shapes for each image may be represented as a similarity graph for the image. In this similarity graph, the nodes are shapes and the arcs in the graph are the relations (similarity) between the shapes calculated using vector distances. Images (trademarks) may then be matched using graph isomorphism matching and attribute matching (vector distance) calculation. The more similar the graphs representing two images, the more similar those two images will be. Thus we can calculate the set of trademarks that are most similar to a query trademark using graph isomorphism and vector distance calculations on the shapes within the images. Graph isomorphism calculations are computationally expensive so by reducing the set of shapes representing each trademark by using our perceptual relevance classifier, we are minimising the graph sizes and minimising the calculation required. We are also eliminating noise (perceptually irrelevant shapes) from the calculation which may adversely affect accuracy.

The methods we have described and the resulting classifier models or regression equation are equally applicable to any underlying shape identifier algorithm such as region growing [11], watershed [12] or closed shape identification [13] providing the result of the algorithm may be represented by a list of boundary points to calculate the attributes used here. Obviously, other attributes could be incorporated or the attribute set changed if, for example fill points were available to allow fill point attributes to be used.

6. ACKNOWLEDGMENTS

This work was supported by E.U. FP6 IST **Project Reference:** 511572 - **PROFIL**.

7. REFERENCES

- [1] Alwis, S. Content-Based Retrieval of Trademark Images, PhD Thesis, Dept. of Computer Science, University of York, UK, 1999
- [2] Beucher, S. Watersheds of Functions and Picture Segmentation, Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'82. pp. 1928-1931, 1982.
- [3] Borgelt, C. Machine Learning Algorithms Implemented in C, 2006, Note: Software available at <http://fuzzy.cs.uni-magdeburg.de/~borgelt/software.html>
- [4] Chan, D. Y.-M. and King, I. Genetic Algorithm for Weights Assignment in Dissimilarity Function for Trademark Retrieval. In 3rd International Conf. on Visual Information and Information Systems (VISUAL'99), LNCS vol 614, Amsterdam, The Netherlands, 1999. Springer Verlag.
- [5] Chang, C.-C. and Lin, C.-J. LIBSVM: a library for support vector machines, 2001, Note: Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [6] Desolneux, A., Moisan, L., and Morel, J.-M. A theory of digital image analysis. 2004. Book in preparation
- [7] Diederich, J. Explanation and artificial neural networks, International Journal of Man-Machine Studies: 37: 335-355, 1992.
- [8] Eakins, J. P. Riley, K. J. and Edwards, J. D. Shape Feature Matching for Trademark Image Retrieval, In, Image and Video Retrieval: Second International Conference, CIVR 2003. LNCS, vol. 2728, Jan 2003, Pages 28 – 38.
- [9] Flickner, M. Sawhney, H. Niblack, et al.. Query by Image and Video Content: The QBIC System, Computer, vol. 28, no. 9, pp. 23-32, Sept., 1995.
- [10] Hodge, V.J., Eakins, J. & Austin, J. Eliciting Perceptual Ground Truth for Image Segmentation. Technical Report YCS 401(2006), Department of Computer Science, University of York.
- [11] Hodge, V.J., Hollier, G., Eakins, J. & Austin, J. Eliciting Perceptual Ground Truth for Image Segmentation. In Proceedings International Conference on Image and Video Retrieval (CIVR2006). Tempe, Arizona, July 13-15, 2006.
- [12] Hu, M.-K. Visual Pattern Recognition by Moment Invariants. IRE Transactions on Information Theory, IT 8:179-187, 1962.
- [13] Jiang, H., Ngo, C.-W. & Tan, H.-K. Gestalt-based feature similarity measure in trademark database, Pattern Recognition, 39: pp. 988 – 1001, 2006.
- [14] John, G.H. and Langley, P. Estimating Continuous Distributions in Bayesian Classifiers. Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338-345. Morgan Kaufmann, San Mateo, 1995.
- [15] Koffka, K.. Principles of Gestalt Psychology. Harcourt Brace. New York, 1963.
- [16] Lowe, D. Three Dimensional Object Recognition from Simple Two Dimensional Images. Artificial Intelligence: 31(3):355-395, 1987.
- [17] Ren, M., Eakins, J. P. and Briggs, P. Human perception of trademark images: implications for retrieval system design. Journal of Electronic Imaging, 9 (4):564-575, 2000.
- [18] Rosin, P. L. Measuring Shape: Ellipticity, Rectangularity, and Triangularity, In Proceedings of 15th International Conference on Pattern Recognition (ICPR'00) - Volume 1, 2000.
- [19] Rumelhart, D. E. and McClelland, J. L. Parallel distributed processing: explorations in the microstructure of cognition (MIT Press, Cambridge, Massachusetts, 1986).
- [20] Sagata Regression v1.0 Copyright © 2002-2003 Sagata, Ltd. www.sagata.com

- [21] Sarkar, S. and Boyer, K.L. On optimal infinite impulse response edge detection filters IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI): 13(11): 1154-71 (1991).
- [22] Saund, E. Finding Perceptually Closed Paths in Sketches and Drawings. IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI): 25(4): 475-491, April 2003,
- [23] Tzanetakis, G., Traka, M. and Tziritas, G. Motion estimation based on affine moment invariants. In Proc. European Signal Processing Conference (Eusipico), Rhodes, Greece, 1998
- [24] Vapnik, V.N. The Nature of Statistical Learning Theory. Springer, 1995.
- [25] Wertheimer, M. Laws of Organization in Perceptual Forms (1923). In, Ellis (ed) A Source Book of Gestalt Psychology, Routledge & Kegan Paul, London 1938.
- [26] Witten, I. and Frank, E. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (2000), ISBN 1-55860-552-5
- [27] Wuescher, D.M. and Boyer, K.L. Robust contour decomposition using a constant curvature criterion. IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI): 13(1): 41-51, 1991.
- [28] Zucker, S.W. Region Growing: Childhood and Adolescence, Computer Graphics & Image Processing: 5:382-399, 1976

Table 3 Table listing the recall scores for the four classifiers on each of the six train/test set combinations. The highest recall score for each row is indicated in bold font. The maximum column indicates the number of records, number of perceptually relevant (1) and perceptually irrelevant (0) records in the respective test sets.

| Train/Test | | Max. | Bayes | MLP | SVM | Reg. |
|-------------|------------|------|--------|---------------|--------|------------|
| Set1/set2 | Correct | 306 | 276 | 300 | 285 | 257 |
| | 1s correct | 130 | 112 | 126 | 118 | 122 |
| | 0s correct | 176 | 164 | 174 | 167 | 135 |
| Set2/set1 | Correct | 435 | 356 | 408 | 349 | 359 |
| | 1s correct | 240 | 172 | 218 | 179 | 175 |
| | 0s correct | 195 | 184 | 190 | 170 | 184 |
| Set3/set4 | Correct | 370 | 307 | 362 | 329 | 313 |
| | 1s correct | 169 | 147 | 161 | 145 | 159 |
| | 0s correct | 201 | 160 | 201 | 184 | 154 |
| Set4/set3 | Correct | 371 | 306 | 356 | 314 | 339 |
| | 1s correct | 169 | 148 | 192 | 166 | 145 |
| | 0s correct | 202 | 158 | 164 | 148 | 194 |
| Set5/set6 | Correct | 376 | 308 | 362 | 300 | 321 |
| | 1s correct | 171 | 123 | 161 | 128 | 126 |
| | 0s correct | 205 | 185 | 201 | 172 | 195 |
| Set6/set5 | Correct | 365 | 286 | 355 | 304 | 329 |
| | 1s correct | 199 | 130 | 192 | 155 | 169 |
| | 0s correct | 166 | 156 | 163 | 149 | 160 |
| Overall | Correct | 2223 | 1839 | 2143 | 1881 | 1918 |
| | 1s correct | 1078 | 832 | 1050 | 891 | 896 |
| | 0s correct | 1145 | 1007 | 1093 | 990 | 1022 |
| Overall %ge | Correct | | 82.73% | 96.40% | 84.62% | 86.28% |
| | 1s correct | | 77.18% | 97.40% | 82.65% | 83.12% |
| | 0s correct | | 87.95% | 95.46% | 86.46% | 89.26% |