Image Categorization Using $N \times M$-grams

Aya Soffer *
Computer Science and EE Department
University of Maryland Baltimore County
5401 Wilkens Avenue
Baltimore, MD 21228-5398 and
CESDIS, Goddard Space Flight Center
E-mail: soffer@cs.umbc.edu

ABSTRACT

A method for categorizing images using $N \times M$-*grams* is presented. The goal of categorization is to find all images from the same category as a given query image. Some example categories are hand-written documents, printed documents, floor plans, satellite images, and fingerprints. The categorization method is based on the N-gram technique that is commonly used for determining similarity of text documents. Intuitively an $N \times M$-gram is a small pattern in an image. The hypothesis that two images that have the same recurring patterns are likely to belong to the same category is examined. The notion of $N \times M$-grams is defined and the process of computing an image profile in terms of its $N \times M$-grams termed an $N \times M$-*gram vector* is explained. Two similarity measures to compare images based on their $N \times M$-gram vectors are proposed. Results of an experiment on images from various categories are presented. The two similarity measures for $N \times M$-gram vectors are compared to each other as well as to results of categorization using a method based on color distribution features. The results show that for our test images $N \times M$-gram based methods were more successful in finding images from the same category as a given query image than color distribution based methods.

Kewords: retrieval by content, image categorization, N-grams , image similarity, image databases, document analysis

1 Introduction

With the proliferation of systems that store electronic images, there has recently been much interest in similarity based image retrieval.[7] The goal is to retrieve all images that are similar to a given query image. The type of similarity varies from system to system. The most common form of similarity is based on color.[15,16] The underlying assumption is that often colors are used to encode functionality, and thus images with similar color distributions have similar themes. Another form of similarity is based on texture.[10] Shape based similarity is also used in some systems .[5] In QBIC,[11] users must outline objects in the image in order to use shape features, thus this method cannot be used non-interactively. In some cases when the domain of images is limited automated indexing based on shape is also possible.[14] One problem with all of these methods is that they are unable to track the structural information that is inherent in the image (e.g., edge information, recurring patterns, etc.).

Testing whether a particular method for image similarity is useful has proven to be an elusive task. One of the

*The support of USRA/CESDIS and NASA Goddard Space Flight Center is gratefully acknowledged.

difficulties is that in many cases it is not clear even to the user whether two images should be considered similar to each other. In our work, we have thus chosen to define the goal of similarity retrieval in a slightly different manner. Rather than trying to find the images that are most similar to a given query image, we assume that the images in our database can be divided into well defined categories, and that the goal of retrieval is to find other images from the same category as the query image. Some example categories are handwritten documents, printed documents, floor plans, satellite images, and fingerprints. With this goal in mind, it is easier to assess how well the system performs. While categorization may not be relevant for all image archives, we believe that for large image archives such as the world wide web, having the ability to find for example pictures of houses is useful. Furthermore, once all images from the same category are retrieved, a domain specific technique tailored for the specific category can be used to refine the results. The method that we suggest is able to implicitly extract structural information from the images, and we hypothesize that images from similar categories have similar structures.

In this paper we present a method for categorizing images using $N \times M$*-grams* . This method is based on the N-gram technique that is used for determining similarity of text documents.[2] Our approach for categorizing images is based on extending the definition of N-grams to 2D. Intuitively an $N \times M$-gram is a small patch or pattern in an image. The hypothesis that we examine in this paper is that two images that have the same recurring patterns are likely to belong to the same category. We define the notion of $N \times M$-grams and show how to compute an image profile in terms of its $N \times M$-grams termed an $N \times M$*-gram vector*. We propose two similarity measures to compare images based on their $N \times M$-gram vectors. We present results of an experiment that we conducted on images from various categories. We compare these similarity measures to each other as well as to results of categorization using a method based on color distribution features.[15] Our results show that for our test images $N \times M$-gram based methods were more successful in finding images from the same category as a given test image than color distribution based methods.

The rest of this paper is organized as follows. In Section 2 we review the definition of N-grams and define $N \times M$-grams. Related work is also described in this section. Our method for generating $N \times M$-gram profiles of images is presented in Section 3. Several similarity measures for comparing two images based on their $N \times M$-gram profiles are given in Section 4. In Section 5 we describe a method for categorization of images based on color distribution features. This method is used as a test case to evaluate categorization using $N \times M$-grams. Our evaluation procedure and experimental results are discussed in Section 6. Section 7 contains concluding remarks.

2 Background and Definitions

An N*-gram* is a sequence of N consecutive characters in a document. N-grams are generated from a document by sliding a "window" N characters wide across the document's text, moving it 1 character at a time. The word "image", for example, consists of three tri-grams "ima", "mag", and "age". The N*-gram frequency profile* of a document consists of a frequency count of each possible N-gram in the language. For example, for the English case-insensitive alphabet, there are 26^3 possible tri-grams. Thus, the N-gram frequency profile of a document written in English would be the frequency of each of these 26^3 possible tri-grams in the document. The N-gram frequency profile can also be viewed as a vector of dimension 26^3 termed an N*-gram vector*. A similarity measure between the N-gram vector of two documents determines the similarity of these two documents. The assumption is that two documents whose N-gram vectors are "close" are likely to deal with related matter. One measure that is commonly used is the normalized dot product of the two N-gram vectors.

Our approach for categorizing images is based on extending the definition of N-grams to 2D. An $N \times M$*-gram* is a sequence of $N \times M$ pixel values. $N \times M$-grams are generated from an image by sliding a "window" $N \times M$ characters wide across the image moving it 1 pixel at a time in a row-wise manner. For example, the image whose pixel values are shown in Figure 1 consists of four 3×3-grams, "001011110", "011110100", "011110100", and "110100000". The $N \times M$*-gram frequency profile* of an image consists of a frequency count of each possible $N \times M$-gram. For example, for binary images there are $2^{3\times 3} = 512$ possible 3×3-grams. Thus, the 3×3-gram

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

Figure 1: Example image pixel values. The corresponding 3×3-grams are "001011110", "011110100", "011110100", and "110100000".

frequency profile of a binary image would be the frequency of each of these 512 possible 3×3-grams in the image. Note that if we consider the number of occurrences of each $N \times M$-gram rather than the frequency, then the $N \times M$-gram vector is the histogram of $N \times M$-gram values. A similarity measure between the $N \times M$-gram vector of two images determines the similarity of these two images. The assumption is that two images whose $N \times M$-gram vectors are "close" are likely to belong to the same category.

Some other systems that use N-gram techniques for indexing images include CANDID[8] which employs probability density functions in an approach that closely resembles the N-gram work for textual data. The general idea is to first compute several features (local color, texture, and/or shape) at every pixel in the image, and then compute a probability density function that describes the distribution of these features. This approach is analogous to the N-gram approach in that the image is described by a global signature and image similarity is determined by comparing these signatures. However, color and texture features are still used for the signature rather than extending N-grams to 2D. 2D pixel trigrams have been suggested for locating copies of pictures in an image database.[6] A variation of $N \times M$-grams called local binary patterns has been show to be effective for texture classification.[12]

3 Generating $N \times M$-gram vectors

Since the goal of using $N \times M$-grams is to implicitly index on the structure of the image and not on the color or greyscale, and since we wanted to limit the range of possible $N \times M$-grams , we first transform the images into a binary representation. Currently we perform this "binarization" using a global threshold operation on the input images. Other binarization methods such as gradient images and edges will be used in the future. The $N \times M$-grams are calculated for the binary image by sliding a window of size $N \times M$ over the entire image. The pixel values of the N rows of M pixels in this window are concatenated. The number of occurrences of each $N \times M$-gram in the image is counted. This number is divided by the total number of $N \times M$-grams in the image to compute the $N \times M$-gram frequency. We store both the absolute count and the frequency of each $N \times M$-gram for each image. In addition, we compute the average frequency and average count of each $N \times M$-gram for the entire corpus of images.

4 Comparing $N \times M$-gram vectors

The similarity measure commonly used to compare two N-gram vectors of text documents is the normalized dot product of the two vectors. We use a similar measure for images. Let T denote the total number of $N \times M$-grams (e.g., 512 for binary images and 3×3-grams), and let $f(j, I)$ denote the frequency of $N \times M$-gram j in image I. For two images A and B the similarity score $S_{dp}(A, B)$ using the dot product similarity measure is given

in Equation 1.

$$S_{dp}(A,B) = \frac{\sum_{j=1}^{T} f(j,A) * f(j,B)}{\sqrt{\sum_{j=1}^{T} f(j,A)^2 * \sum_{j=1}^{T} f(j,B)^2}} \quad (1)$$

$S_{dp}(A,B)$ is essentially the cosine of the angle between the $N \times M$-gram vector of images A and B, its range of values lie between -1 and 1, with 1 given for an exact match (i.e., identical images). In work on text documents,[2] it was observed that the similarity can be improved by subtracting the average N-gram vector for a corpus of documents from the N-gram vector of the query and database documents. The average N-gram vector is simply the average frequency of each N-gram over all of the documents that are stored in the database. By subtracting this average the weight of N-grams that are very common and hence not very useful in distinguishing between documents is reduced. We have adopted this idea for images. Let $af(j)$ denote the average frequency of $N \times M$-gram j over all images in the database. The similarity score $S_{dpa}(A,B)$ with this modification is given in Equation 2.

$$S_{dpa}(A,B) = \frac{\sum_{j=1}^{T} (f(j,A) - af(j)) * (f(j,B) - af(j))}{\sqrt{\sum_{j=1}^{T} (f(j,A) - af(j))^2 * \sum_{j=1}^{T} (f(j,B) - af(j))^2}} \quad (2)$$

The second similarity measure that we use tries only to gauge the similarity between images in terms of the number of common $N \times M$-grams. The motivation for this is the assumption that images from similar categories may have many common $N \times M$-grams while still having other areas of the image that are different. This case is penalized by the dot product similarity measure. For this measure we use the absolute count rather than the frequency of each $N \times M$-gram. The $N \times M$-gram vector is thus the histogram of $N \times M$-grams , and the similarity measure is the histogram intersection.[16] Let $t(j,I)$ denote the total number of occurrences of $N \times M$-gram j in image I. The similarity between two images A and B using this measure $S_{int}(A,B)$ termed *$N \times M$-gram vector intersection* is given by Equation 3.

$$S_{int}(A,B) = \frac{2 * \sum_{j=1}^{T} min(t(j,A), t(j,B))}{\sum_{j=1}^{T} t(j,A) + \sum_{j=1}^{T} t(j,B)} \quad (3)$$

$min(t(j,A), t(j,B))$ is the number of occurrences of $N \times M$-gram j in both images A and B. Thus, $S_{int}(A,B)$ is the number of common $N \times M$-grams in the two images normalized by the total number of $N \times M$-grams in them. In order to reduce the weight of common $N \times M$-grams, we again adjust by the average. Let $at(j)$ denote the average number of occurrences of $N \times M$-gram j in the database. The similarity measure $S_{inta}(A,B)$ with this modification is given in Equation 4.

$$S_{inta}(A,B) = \frac{2 * \sum_{j=1}^{T} \frac{min(t(j,A), t(j,B))}{at(j)}}{\sum_{j=1}^{T} \frac{t(j,A)}{at(j)} + \sum_{j=1}^{T} \frac{t(j,B)}{at(j)}} \quad (4)$$

5 Categorization using color distribution features

In order to test the effectiveness of images categorization using $N \times M$-grams , we have also devised a method for categorization of images using color distribution features. It is based on a similar method for image similarity retrieval.[15] Using this method, we compute five global features that describe the color (or greyscale) distribution throughout the image. Let P denote the number of pixels in the image, and let $I(p)$ denote the value of pixel p in image I. Note that although images are usually treated as 2-dimensional arrays (with K rows and L columns), and a pixel is given by its row and column (k,l), in this case since we are interested in the distribution of intensity values over all of the pixels in the image we view it as a 1-dimensional array with $P = K \times L$ pixels. We are only using grey scale images, thus $I(p)$ is the intensity of pixel p. For color images we would need to compute these features for each band.

We use the following five features, *df(f)* denotes distribution feature f.

$$df(1):\ mean = \frac{1}{P} * \sum_{p=0}^{P-1} I(p)$$

$$df(2):\ stddev = \sqrt{\frac{1}{P-1} * \sum_{p=0}^{P-1} (I(p) - mean)^2}$$

$$df(3):\ RMS = \sqrt{\frac{1}{P} * \sum_{p=0}^{P-1} I(p)^2}$$

$$df(4):\ skew = \frac{1}{P} * \sum_{p=0}^{P-1} (\frac{I(p) - mean}{stddev})^3$$

$$df(5):\ kurtosis = \frac{1}{P} * \sum_{p=0}^{P-1} (\frac{I(p) - mean}{stddev})^4 - 3$$

Each image is thus represented by a five dimensional feature vector. The similarity of two images using this method is determined by the weighted Euclidean distance between their corresponding feature vectors. Let F denote the number of features used to describe the color distribution, and let $df(f, I)$ denote the value of distribution feature f for image I. The similarity score using these statistical distribution features $S_{stat}(A, B)$ between two images A and B is given in Equation 5.

$$S_{stat}(A, B) = \sqrt{\sum_{f=1}^{F} w_f (df(f, A) - df(f, B))^2} \tag{5}$$

w_f is the weighting factor of f^{th} feature of the feature vector. The weighing factor is computed so that features with a smaller variance have a larger weight as described in.[1]

6 Implementation and Experimentation

At this point we have tested our method on 144 images from 16 different categories. These categories are finger prints, aircraft, animal statues, floor plans, topographic maps, road maps, English hand-written documents, Hebrew hand-written documents, English printed documents, Hebrew printed documents, Arabic documents, satellite images, aerial photos, pictures of houses, people, and comics. We computed the $N \times M$-gram profile for each image with $N = 3$ and $M = 3$ (i.e., 3×3-grams). To test the retrieval accuracy, for each image I in our database we run the system with I as the query image, in effect ranking all of the other images with respect to I. We repeat this process using the following similarity measures as described in Section 4 dot product (dp), dot product normalized by averages (dpa), $N \times M$-gram vector intersection (int), $N \times M$-gram -vector intersection normalized by averages ($inta$). In addition we computed the greyscale distribution features for each image as described in Section 5. The same process of ranking images with respect to each image in the database was repeated using these statistical greyscale distribution features (stat).

6.1 Evaluation procedure

Retrievals based on exact match are usually evaluated using precision and recall statistics.[13] However, systems where the result of retrieval is a list of items sorted by similarity, need a more elaborate evaluation metric that

reflects the positions in which the set of relevant items appear in the retrieved set. One such measure is the *normalized recall* metric,[13] which measures how close to the top of the list of retrieved items does the set of relevant items appear compared to the ideal retrieval in which all N relevant objects appear in the top N positions. For the image similarity problem (i.e., find all images similar to the query image) it is necessary to first determine what is the set of all relevant items, a task which is usually not clear to the person evaluating the system. However, for the categorization problem (i.e., find all images from the same category as the query image), it is only necessary to know the list of images from each category for the test set. Since we assume that the images fall into well defined categories, this information is readily available.

In order to calculate the the normalized recall we define two measures[3]:

- $IAVRR$ - ideal average rank of relevant items. Ideally all images from the same category should be ranked at the top. Thus, if C_n is the number of images in category C, then since the ideal rankings are $1 - C_n$ the ideal average ranking of category C is given by

$$IAVRR(C) = \frac{\sum_{n=1}^{C_n} n}{C_n}$$

.

- $AVRR$ - average rank of relevant items for a particular retrieval. For a particular retrieval seeking images from the same category as a query image Q the average rank of relevant items for Q is given by

$$AVRR(Q) = \frac{\sum_{I \in cat(Q)} R(I)}{cat(Q)_n},$$

where $R(I)$ denotes the ranking of image I in the result, *cat(Q)* is the category of query image Q, and $cat(Q)_n$ is the number of images in the category of query image Q.

The ratio of AVRR to IAVRR is a measure of the deviation from the ideal ranking. $\frac{AVRR}{IAVRR} = 1$ for a perfect retrieval (i.e., all images from the same category as the query image were ranked at the top). $\frac{AVRR}{IAVRR} > 1$ for all other cases. In general, a lower ratio implies better results. For each category C we compute the average normalized recall metric for C by taking the average of the results of $\frac{AVRR(Q)}{IAVRR(C)}$ for each image Q in category C. In addition, we compute the overall normalized recall value as the average AVRR to IAVRR ratio for all categories.

6.2 Results

Table 1 summarizes the normalized recall results for each of image category and for each of the five similarity measures that we tested. The total normalized recall for each similarity measure averaged over all categories is also presented. Figures 2– 6 show the first 16 ranked images for one image from five of the sixteen categories. Each figure has three parts (a), (b), and (c) reporting the results using statistical greyscale distribution features (*stat*), 3×3-gram with dot product similarity (*dp*), and 3×3-grams with $N \times M$-gram -vector intersection similarity (*int*), respectively. The first image out of the 16 is the query image. The names of all images that are from the same category as the query image are displayed in reverse video.

From these results it is apparent that overall the $N \times M$-gram based methods perform better than the greyscale distribution method. In addition, between the similarity measures for $N \times M$-gram vectors, the $N \times M$-gram vector intersection score (*int*) is best followed by the dot product between $N \times M$-grams normalized by the average $N \times M$-gram values over all images (*dpa*). For the $N \times M$-gram -vector intersection, normalizing for the average over all images (*inta*), proved to be a hindrance. Note, however, that this performance was not observed for all categories. The greyscale feature distribution method (*stat*) was best in 3 categories, road maps, English

	STAT	DP	DPA	INT	INTA
finger prints	4.164	3.646	3.462	1.841	3.112
aircraft	9.497	11.965	8.892	8.792	8.556
animal statues	2.225	5.050	2.075	1.900	1.025
floor plans	2.990	1.087	2.104	1.726	2.347
topographical maps	2.963	5.864	3.588	1.558	3.412
road maps	3.167	5.198	5.711	5.262	6.215
English print	4.896	5.430	5.249	6.042	6.299
English handwriting	2.526	1.471	1.853	1.894	3.508
Hebrew handwriting	15.865	9.214	10.905	2.587	15.587
Hebrew print	1.444	1.000	1.167	1.222	1.000
Arabic	6.563	3.252	5.484	3.077	5.753
satellite images	4.547	6.543	3.536	4.211	5.572
aerial photos	4.913	6.908	2.592	4.173	10.321
house pictures	7.603	6.381	5.524	1.516	9.270
people	10.444	11.349	11.532	4.802	13.508
comics	2.008	3.285	3.679	2.118	2.954
total	5.364	5.478	4.835	3.295	6.152

Table 1: Normalized recall results for each category (normalized recall= 1 for perfect retrieval, > 1 otherwise).

print, and comics. An example of the superiority of *stat* in the road map category can be seen in Figure 6 where 8 out of 10 of the images from the road map category in our database are among the top 16 using *stat*, as opposed to 6 for *dp*, and only 3 for *int*.

The dot product of the $N \times M$-gram vectors (*dp*) was best in the floor plan, English hand-writing, and the Hebrew print categories. Figures 2 and 3 illustrate this. The *dp* method found all 8 floor plans and ranked them 1-9 (an airplane is ranked 6th), *int* ranked 7 of the eight in the top 7 positions, however the eighth floor plan was only ranked 16, while *stat* only ranked 5 floor plans in the top 16. In the English handwriting category, 12 out of the top 16 are in fact from this category for *dp* and *int*, while only 9 out the the top 16 are from the English handwriting category using *stat*. The dot product of the $N \times M$-gram vectors normalized for averages (*dpa*) was most effective in the satellite image and aerial photo category. The $N \times M$-gram vector intersection normalized for averages was best in the aircraft and animal statue categories.

Finally the $N \times M$-gram vector intersection method (*int*) was best in 6 categories. However, its overall normalized recall is much better than the other methods. It performed particularly well compared to the other methods in the finger prints, topographic map, Hebrew hand-writing, and house picture categories. Figures 4 and 5 illustrate this. The *int* method ranked the 9 topographic maps in our database 1–9 (a perfect normalized recall), 8 of these 9 topographic maps are ranked in the top 16 using *dp*, while only 5 are in the top 16 using *stat*. In the house picture category, *int* ranked the 6 houses in our database 1–8, *dp* only found 4 in the top 16 and ranked them relatively low, and *stat* only found 3 in the top 16 and also ranked them low.

7 Concluding Remarks

The experiments that we conducted rank all of the images in the database with respect to one particular image. Another interesting experiment is to try to cluster the images using the three similarity measures. This could either be done totally automatically using some form of k-means clustering or by first providing some examples for each category as a training set and then clustering the remaining images according to these initial clusters.

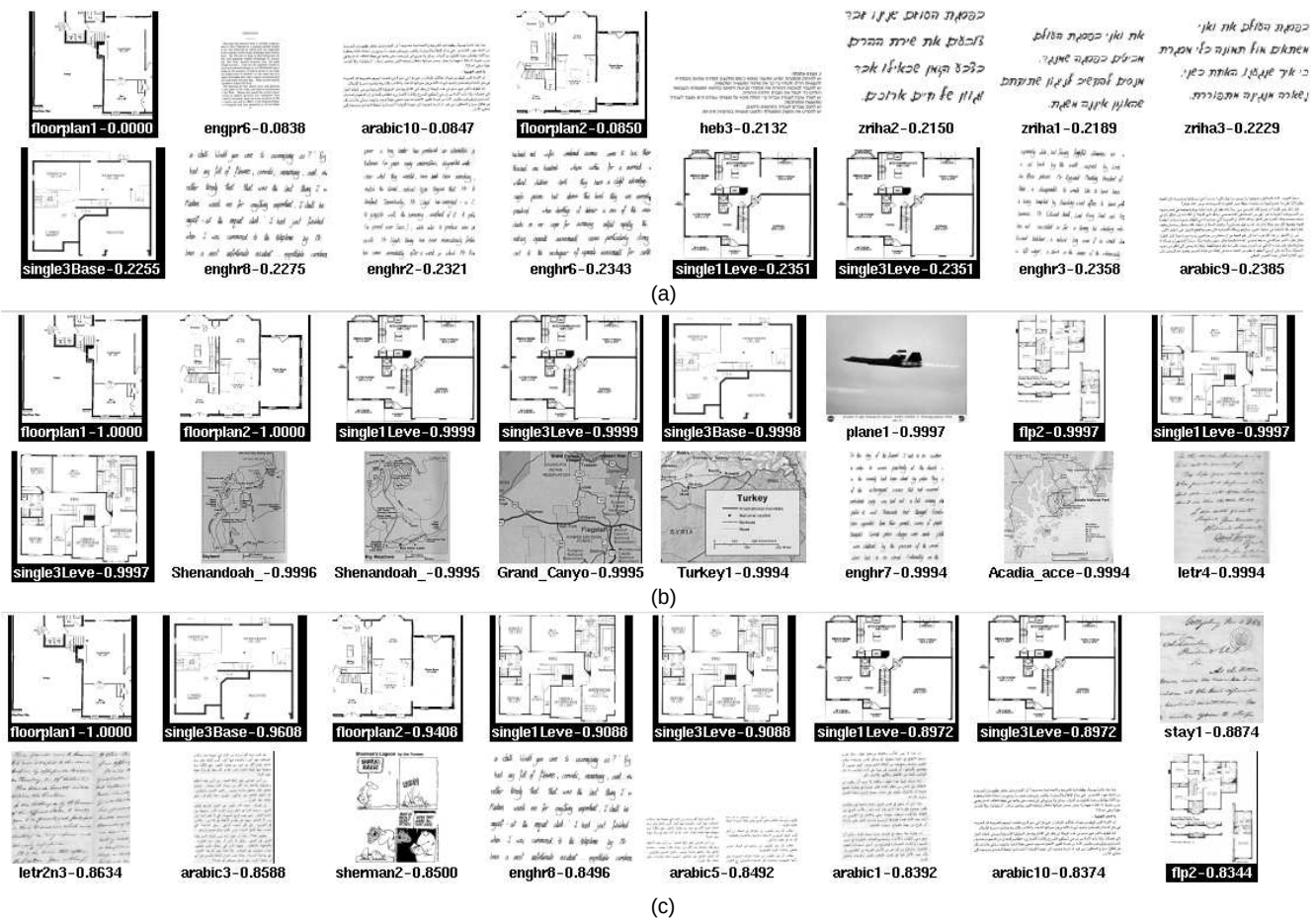

Figure 2: Results for image from the floor plan category (a) greyscale distribution features ($stat$). (b) 3×3-gram with normalized dot product score (dp). (c) 3×3-gram with vector intersection score (int).

While the use of $N \times M$-grams has shown some promise in our experiments, the improvement over using greyscale distribution features was as not as dramatic as we had expected. There are numerous modifications that will most likely improve these results. We computed the $N \times M$-grams on the images following a threshold operation on the original greyscale levels. It may be more useful to compute these on the gradient image or on the images following some edge detection. Furthermore, the choice of 3×3-grams for the experiments was quite arbitrary and may in fact be too small a window. Using a larger window may yield better results, but will increase the storage as the dimensionality of the $N \times M$-gram vectors will be higher. One solution for both problems is to use multiple window sizes by applying the same size window to various resolutions of the image. We intend to use the wavelet transform[9] for this purpose. By computing $N \times M$-grams on both the low pass and the high pass components of the wavelet decomposition of the images we will in effect get both edge information and multi-resolution images.

In our discussion we did not consider the running time and indexing associated with the three methods that we compared. The greyscale distribution measure is significantly faster to compute and is also straightforward to index for fast searches using an appropriate multi-dimensional indexing structure.[4] In contrast, computing $N \times M$-grams is slower and it is not as easy to index $N \times M$-gram vectors since their dimensionality is very high. However, there is much room for improving of the $N \times M$-gram based methods as they are described in this paper

Figure 3: Results for image from the English handwriting category (a) greyscale distribution features ($stat$). (b) 3×3-gram with normalized dot product score (dp). (c) 3×3-gram with vector intersection score (int).

in terms of efficiency. If the results of $N \times M$-gram based methods are significantly better in terms of retrieval accuracy, then the extra computing time will be well worth it.

8 Acknowledgments

I thank Jacqueline Le Moigne for her comments on this manuscript. I have benefited from discussions with Ethan Miller.

9 REFERENCES

[1] G.L Cash and M. Hatamian. Optical character recognition by the method of moments. *Computer Vision, Graphics, and Image Processing*, 39(3):291–310, September 1987.

[2] M. Damashek. Gauging similarity with n-Grams: Language-independent categorization of text. *Science*,

Figure 4: Results for image from the house pictures category (a) greyscale distribution features (*stat*). (b) 3×3-gram with normalized dot product score (*dp*). (c) 3×3-gram with vector intersection score (*int*).

267:843–848, February 1995.

[3] C. Faloutsos, R. Barber, W. Equitz, M. Flickner, W. Niblack, and D. Petkovic. Efficient and effective quering by image content. *Journal of Intelligent Information Systems*, pages 231–62, 1994.

[4] J.H. Friedman, J.L. Bentley, and R.A. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3(3):209–226, September 1977.

[5] W. I. Grosky and R. Mehrotra. Index-based object recognition in pictorial data management. *Computer Vision, Graphics, and Image Processing*, 52(3):416–436, December 1990.

[6] D. P. Huijsmans, S. Poles, and M. Lew. 2D pixel trigrams for content-based image retrieval. In A. W. M. Smeulders and R. Jain, editors, *Proceedings of the 13th International Conference on Pattern Recognition*, pages 139–145, Amsterdam, The Netherlands, August 1996.

[7] R. Jain. NSF workshop on visual information management systems. *SIGMOD RECORD*, 22(3):57–75, September 1993.

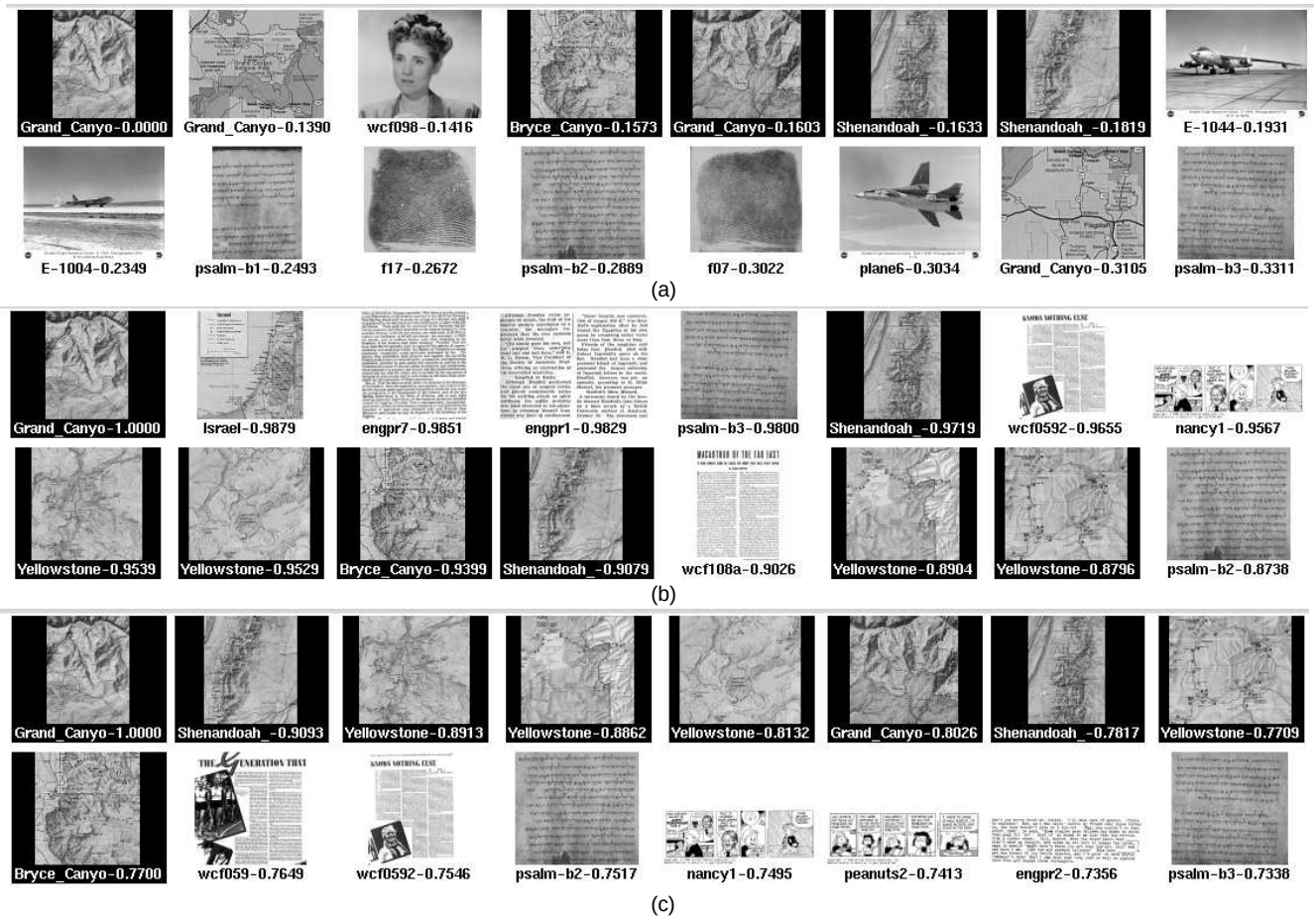

Figure 5: Results for image from the topographical maps category (a) greyscale distribution features (*stat*). (b) 3×3-gram with normalized dot product score (*dp*). (c) 3×3-gram with vector intersection score (*int*).

[8] P.M. Kelly, T.M. Cannon, and D.R. Hush. Query by image example: the CANDID approach. In *Proceeding of the SPIE, Storage and Retrieval of Still Image and Video Databases III*, volume 2420, pages 238–248, San Jose, CA, February 1995.

[9] S. Mallat. A theory for multiresolution signal decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1995.

[10] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of large image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(11):–, November 1996. To appear.

[11] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, and P. Yanker. The QBIC project: Querying images by content using color, texture, and shape. In *Proceeding of the SPIE, Storage and Retrieval of Image and Video Databases*, volume 1908, pages 173–187, San Jose, CA, February 1993.

[12] T. Ojala, M. Pietikäinen, and D. Harwood. A comparitive study of texture measures with clssification based on feature distributions. *Pattern Recognition*, 29(1):51–59, January 1996.

[13] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, New York, 1st edition, 1983.

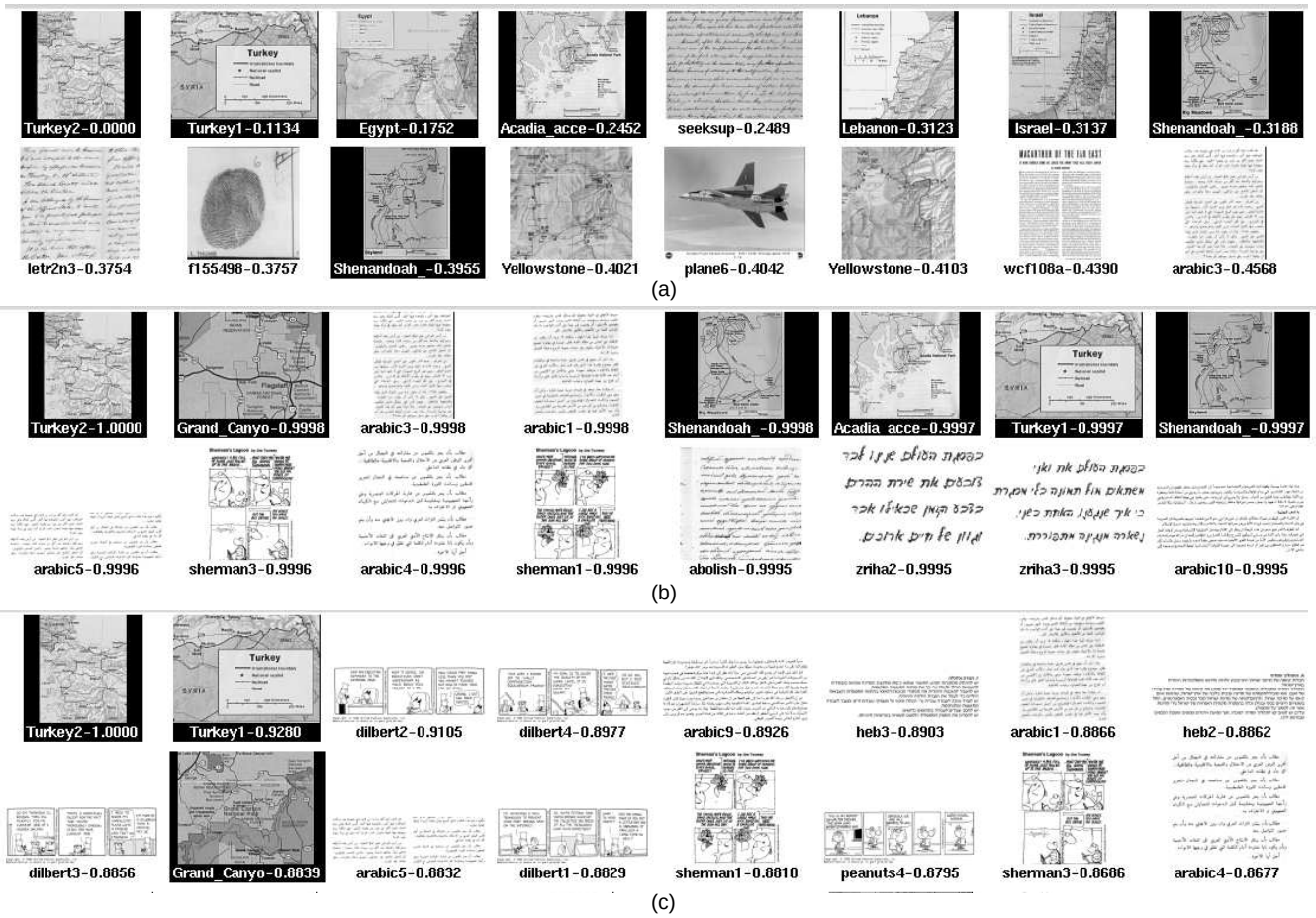

Figure 6: Results for image from the road maps category (a) greyscale distribution features (*stat*). (b) 3×3-gram with normalized dot product score (*dp*). (c) 3×3-gram with vector intersection score (*int*).

[14] A. Soffer and H. Samet. Retrieval by content in symbolic-image databases. In *Proceedings of the SPIE, Storage and Retrieval of Still Image and Video Databases IV*, volume 2670, pages 144–155, San Jose, CA, February 1996.

[15] M. Stricker and M. Orengo. Similarity of color images. In *Proceeding of the SPIE, Storage and Retrieval of Still Image and Video Databases III*, volume 2420, pages 381–392, San Jose, CA, February 1995.

[16] M. Swain. Interactive indexing into image databases. In *Proceeding of the SPIE, Storage and Retrieval for Image and Video Databases*, volume 1908, pages 95–103, San Jose, CA, February 1993.