DPF — A Perceptual Distance Function for Image Retrieval

Beitao Li, Edward Chang, Ching-Tung Wu
Electrical & Computer Engineering, U.C. Santa Barbara
beitao@engineering.ucsb.edu, echang@ece.ucsb.edu

Abstract

For almost a decade, Content-Based Image Retrieval has been an active research area, yet one fundamental problem remains largely unsolved: how to measure perceptual similarity. To measure perceptual similarity, most researchers employ the Minkowski-type metric. Our extensive data-mining experiments on visual data show that, unfortunately, the Minkowski metric is not very effective in modeling perceptual similarity. Our experiments also show that the traditional "static" feature weighting approaches are not sufficient for retrieving various similar images. In this paper, we report our discovery of a perceptual distance function through mining a large set of visual data. We call the discovered function *dynamic partial distance function* (DPF). When we empirically compare DPF to Minkowski-type distance functions, DPF performs significantly better in finding similar images. The effectiveness of DPF can be well explained by *similarity theories* in cognitive psychology.

Keywords: content-based image retrieval, data mining, perceptual distance function, similarity search.

1 Introduction

Research in content-based image retrieval has steadily gained momentum in recent years as a result of the dramatic increase in the volume of digital images. To achieve effective retrieval, an image system must be able to accurately characterize and quantify perceptual similarity. However, a fundamental challenge — how to measure perceptual similarity — remains largely unanswered. Various distance functions, such as the *Minkowski metric*, *earth mover distance* [5], and *fuzzy logic*, have been used to measure similarity between feature vectors representing images. Unfortunately, our experiments show that they frequently overlook obviously similar images and hence are not adequate for measuring perceptual similarity.

Quantifying perceptual similarity is a difficult problem. Indeed, we may be decades away from fully understanding how human perception works. In this project, we mine visual data extensively to *reverse-engineer* a good perceptual distance function for measuring image similarity. Our mining hypothesis is this: Suppose most of the similar images can be clustered in a feature space. We can then claim with high confidence that **1)** the feature space can adequately capture visual perception, and **2)** the distance function used for clustering images in that feature space can accurately model perceptual similarity.

We perform our mining operation in two stages. In the first stage, we isolate the distance function factor (we use the Euclidean distance) to find a reasonable feature set. In the second stage, we freeze the features to discover a perceptual distance function that can better cluster similar images in the feature space. In other words, our goal is to find a function that can keep similar images close together in the feature space, and at the same time, keep dissimilar images away. We call the discovered function *dynamic partial distance function* (DPF). We empirically compare DPF to Minkowski-type distance functions and show that DPF performs remarkably better.

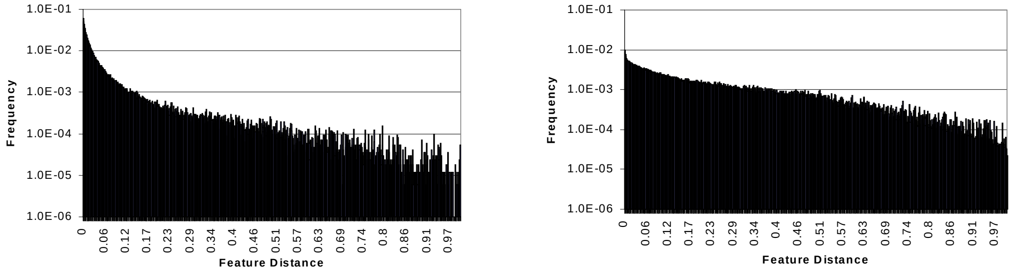
Briefly, the contributions of this paper are as follows:

- We construct a mining dataset to find a feature set that can adequately represent images. In that feature space, we find distinct patterns of similar and dissimilar images, which lead to the discovery of DPF.
- Through empirical study, we demonstrate that DPF is very effective in finding images that have been transformed by rotation, scaling, downsampling, and cropping, as well as images that are perceptually similar to the query image (e.g., images belonging to the same video shot). Our testbed shows that DPF outperforms Minkowski-type functions by 25 percentiles in recall.

2 Discovering DPF

To ensure that sound inferences can be drawn from our mining results, we carefully construct the training dataset. First, we prepare for a dataset that is comprehensive enough to cover a diversified set of images. To achieve this goal, we collect $60,000$ JPEG images from Corel CDs and from the Internet. Second, we define "similarity" in a slightly restrictive way so that individuals' subjectivity can be safely excluded. (We address the problem of learning subjective perception in [1, 6].) For each image in the $60,000$-image set, we perform 24 transformations including scaling, downsamping, cropping, rotation, and format transformation. (Details of these transformations are explained in the extended version of this paper [4].) The total number of images in the testbed is 1.5 million.

Our experimental results (see Section 3) show that the perceptual distance function discovered during the mining process on this training dataset, which has a slightly restrictive definition of similarity, can be used effectively to find other perceptually similar images. In other words, our testbed consists of a reasonable representation of similar images, and the mining results (i.e., training results) can be generalized to testing data consisting of perceptually similar images produced by other



(a) Similar Images

(b) Dissimilar Images

Figure 1: The Distributions of Feature Distances.

methods (e.g., changing camera parameters).

From each image, we extract 144 features including color, texture, and shape as its representation. We discuss what these features are, and why they are chosen in [4]. In the remainder of this section, we focus on examining the Minkowski metric and its family. We explain why these functions are ineffective for measuring image similarity, and present our DPF solution.

2.1 Minkowski Metric and Its Limitations

The Minkowski metric is widely used for measuring similarity between objects (e.g., images). Suppose two objects X and Y are represented by two p dimensional vectors $(x_1, x_2, \cdots, x_p)$ and $(y_1, y_2, \cdots, y_p)$, respectively. The Minkowski metric $d(X,Y)$ is defined as

$$d(X,Y) = (\sum_{i=1}^{p} |x_i - y_i|^r)^{\frac{1}{r}}, \qquad (1)$$

where r is the Minkowski factor for the norm. Particularly, when r is set as 2, it is the well known Euclidean distance; when r is 1, it is the Manhattan distance (or L_1 distance). An object located a smaller distance from a query object is deemed more similar to the query object. Measuring similarity by the Minkowski metric is based on one assumption: the similar objects should be close to the query object in all dimensions.

A variant of the Minkowski function, the weighted Minkowski distance function, has also been applied to measure image similarity. The basic idea is to introduce weighting to identify important features. By assigning each feature a weighting coefficient w_i $(i = 1 \cdots p)$, the weighted Minkowski distance function is defined as

$$d_w(X,Y) = (\sum_{i=1}^{p} w_i |x_i - y_i|^r)^{\frac{1}{r}}. \qquad (2)$$

By applying a static weighting vector for measuring similarity, the weighted Minkowski distance function assumes that similar images resemble the query image(s) in the same features. For example, the weighted Minkowski function implicitly assumes that the important features for finding a scaled image are the same as the important features for finding a cropped image.

We can summarize the assumptions of the Minkowski metric as follows:

- Minkowski function: All similar images must be similar in all features.
- Weighted Minkowski function: All similar images are similar in the same way (e.g., in the same set of features) [7].

We questioned the above assumptions upon observing how similar objects are located in the feature space. For this purpose, we carried out extensive data mining work on the 1.5M-image dataset. To better discuss our findings, we introduce a term we have found useful in our data mining work. We define the *feature distance* on the i^{th} feature as $\delta_i = |x_i - y_i|, i = 1, \cdots, p$.

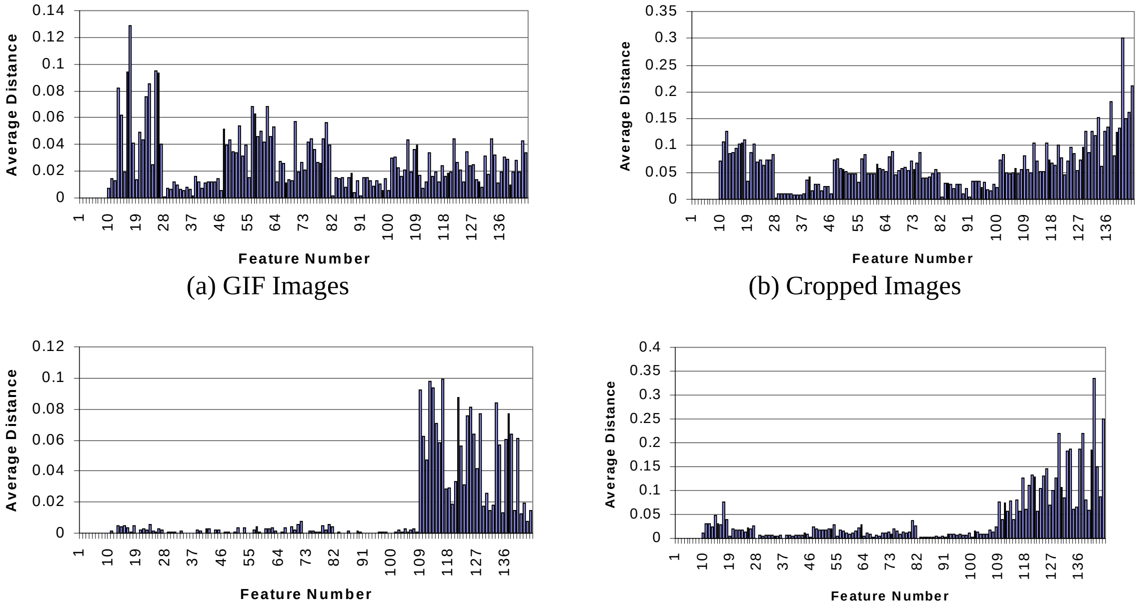
In our mining work, we first tallied the feature distances between similar images (denoted as δ^+), and also those between dissimilar images (denoted as δ^-). Since we normalized feature values to be between zero and one, the range of both δ^+ and δ^- are between zero and one. Figure 1 presents the distributions of δ^+ and δ^-. The x-axis shows the possible value of δ, from zero to one, The y-axis (in logarithmic scale) shows the percentage of the features at different δ values.

The figure shows that δ^+ and δ^- have different distribution patterns. The distribution of δ^+ is much skewed toward small values (Figure 1(a)), whereas the distribution of δ^- is more evenly distributed (Figure 1(b)). We can also see from Figure 1(a) that a moderate portion of δ^+ is in the high value range (≥ 0.5), which indicates that similar images may be quite dissimilar in many features. This observation suggests that the assumption of the Minkowski metric is inaccurate. Similar images are not necessarily similar in all features.

Next, we examined whether similar images resemble the query images in the same way. We tallied the feature distance (δ^+) of the 144 features for different kinds of image transformations. Figure 2 presents four representative transformations: GIF, cropped, rotated, and scaled. The x-axis of the figure depicts the feature numbers, from 1 to 144. The first 108 features are various color features, and the last 36 are texture features. The figure shows that various similar images can resemble the query images in very different ways. GIF images have larger δ^+ in color features (the first 108 features) than in texture features (the last 36 features). In contrast, cropped images have larger δ^+ in texture features than in color features. For rotated images, the δ^+ in colors comes close to zero, although its texture feature distance is much greater. A similar pattern appears in the scaled and the rotated images. However, the magnitude of the δ^+ of scaled images is very different from that of rotated images.

We summarize our observations as follows:

- *Similar feature distance* is distributed differently from *dissimilar feature distance*. *Similar feature distance* skews toward small values, while *dissimilar feature distance* shows more even distribution.
- Similar images do not resemble the query images in all fea-



(a) GIF Images

(b) Cropped Images

(c) Rotated Images

(d) Scaled Images

Figure 2: The Average Feature Distances.

tures.

- Images similar to the query images can be similar in differing features. For example, some images resemble the query image in texture, others in color.

The above observations not only refute the assumptions of Minkowski-type distance functions, but also provide hints as to how a good distance function would work. The first point is that a distance function does not need to consider all features equally, since similar images may match only some features of the query images. The second point is that a distance function should weight features dynamically, since various similar images may resemble the query image in differing ways. Traditional relevance feedback methods [3] learn a set "optimal" feature weights for a query. For instance, if the user is more interested in color than in texture, color features are weighted higher when similarity is computed. What we have discovered here is that this "static" weighting is insufficient. An effective distance function must weight features differently when comparing the query image to different images. These points lead to the design of the *dynamic partial* distance function.

2.2 Dynamic Partial Distance Function

Based on the observations explained above, we designed a distance function to better represent the perceptual similarity. Let $\delta_i = |x_i - y_i|$, for $i = 1, \cdots, p$. We first define sets Δ_m as

$$\Delta_m = \{The \ \ smallest \ \ m \, \delta's \ \ of \ \ (\delta_1, ..., \delta_p)\}.$$

Then we define the *Dynamic Partial Distance Function* (DPF) as

$$d(m, r) = (\sum_{\delta_i \in \Delta_m} {\delta_i}^r)^{\frac{1}{r}}. \qquad (3)$$

DPF has two adjustable parameters: m and r. Parameter m can range from 1 to p. When $m = p$, it degenerates to the Minkowski metric. When $m < p$, it counts only the smallest m feature distances between two objects, and the influence of the $(p - m)$ largest feature distances is eliminated. DPF dynamically selects features to be considered for different pairs of objects. This is achieved by the introduction of Δ_m, which changes dynamically for different pairs of objects. In Section 3, we will show that DPF makes similar images aggregate more compactly and locate closer to the query images, simultaneously keeping the dissimilar images away from the query images. In other words, similar and dissimilar images are better separated by DPF.

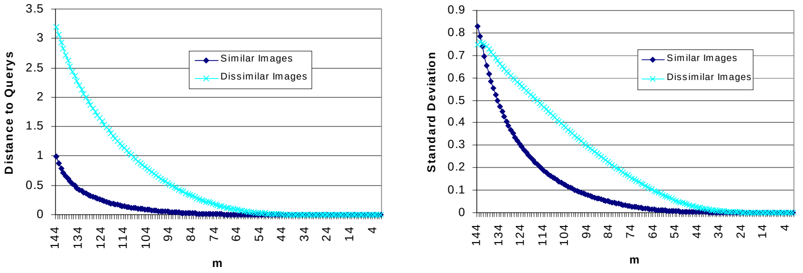
3 Empirical Study

Our empirical study consists of two parts: training and testing. In the training part, we used the same 1.5M-image dataset to predict the optimal m value. In the testing part, we used a 50K-image dataset to examine the effectiveness of DPF.

3.1 Predicting m Through Training

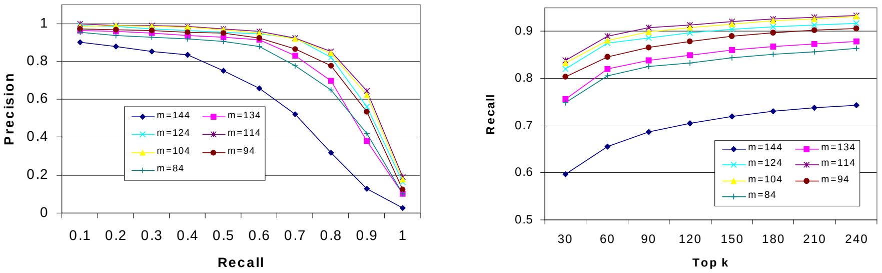
We used the $60,000$ original images to perform queries. Applying DPF of different m values to the 1.5M-image dataset, we tallied the distances from these $60,000$ queries to their similar images, and their dissimilar images, respectively. We then computed the average and the standard deviation of these distances. We denote the average distance of the similar images to their queries as μ_d^+, of the dissimilar images as μ_d^-. We denote the standard deviation of the similar images' distances as σ_d^+, of the dissimilar images as σ_d^-.

Figure 3 depicts the effect of m (in the x-axis) on μ_d^+, μ_d^-, σ_d^+, and σ_d^-. Figure 3(a) shows that as m becomes smaller, both μ_d^+ and μ_d^- decrease. The average distance of similar images (μ_d^+), however, decreases at a faster pace than that of dissimilar images (μ_d^-). For instance, when we decrease m from 144 to 130, μ_d^+ decreases from 1.0 to about 0.3, a 70%



(a) Average of Distances (b) Standard Deviation of Distances

Figure 3: The Effect of DPF.



(a) Precision vs. Recall (b) Recall vs. k

Figure 4: Search Performance of Different m at $r = 3$.

decrease, whereas μ_d^- decreases from 3.2 to about 2.0, a 38% decrease. This gap indicates μ_d^+ is more sensitive to the m value than μ_d^-. Figure 3(b) shows that the standard deviations σ_d^+ and σ_d^- observe the same trend as the average distances. When m decreases, similar images become more compact in the feature space at a faster pace than dissimilar images do. Our training result indicates that when m is set as 114, similar images are best clustered.

3.2 Testing New Distance Functions

The test dataset consists of 100 similar-image sets, each set is composed of 30 images. Of these 30 images, we have the original image, 24 transformed images (using the same transformation methods described in Section 2), and five images that are visually identified as similar. We then added 50K randomly crawled Web images to these 100×30 images to form our testset.

We conducted 100 queries using the 100 original images. For each query, we recorded the ranks of its similar images. We experimented with m values from 84 to 144, with r fixed at three. Figure 4 depicts the experimental results.

The precision-recall curves of selected m values are plotted in Figure 4(a). The peak search performance is achieved when $m = 114$, and it does significantly better than the Minkowski distance ($m = 144$). Figure 4(b) plots the recall at selected m values for top-k retrievals. As we decrease the value of m from 144, the recall improves steadily until m reaches 114, where the peak performance is achieved. Our DPF outperforms the Minkowski distance function by 25 percentiles in recall.

Because of space limitation, we present extensive experimental results and our comparison between the weighted version of DPF and the weighted Minkowski metric in [4]. DPF consistently outperforms Minkowski-type function significantly for finding similar images.

4 Conclusion

In this work we tackled one fundamental problem in image retrieval—how to measure perceptual similarity between images—using data mining techniques. We discovered the *dynamic partial distance function* (DPF) through mining a large set of visual data, and showed that DPF outperformed the traditional functions by significant margins. The effectivenese of DPF can be explained by *similarity theories* in cognitive psychology [2, 4].

References

[1] E. Chang and B. Li. Mega — the maximizing expected generalization algorithm for learning complex query concepts (extended version). *Technical Report http://www-db.stanford.edu/∼echang/mega.ps*, November 2000.

[2] R. L. Goldstone. Similarity, interactive activation, and mapping. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 20:3–28, 1994.

[3] Y. Ishikawa, R. Subramanya, and C. Faloutsos. Mindreader: Querying databases through multiple examples. *VLDB*, 1998.

[4] B. Li, E. Chang, and Y. Wu. Dynamic partial function — a perceptual distance function for measuring similarity (ext. version). *http://www-db.stanford.edu/∼echang/dpf-ext.pdf*, February 2002.

[5] Y. Rubner, C. Tomasi, and L. Guibas. Adaptive color-image embedding for database navigation. *Proceedings of the the Asian Conference on Computer Vision*, January 1998.

[6] S. Tong and E. Chang. Support vector machine active learning for image retrieval. *Proceedings of ACM International Conference on Multimedia*, pages 107–118, October 2001.

[7] X. S. Zhou and T. S. Huang. Comparing discriminating transformations and svm for learning during multimedia retrieval. *Proc. of ACM Conf. on Multimedia*, pages 137–146, 2001.