Image Ranking

12

All that glitters is not gold.

12.1 Introduction

The image feature extracted is usually an N-dimensional feature vector which can be regarded as a point in R^N space. Once images are indexed into the database using the extracted feature vectors, the retrieval of images is essentially the determination of similarity between a query image and the target images in database, which in turn is the determination of distance between the feature vectors in R^N space. The desirable distance measure should reflect human perception. That is to say, perceptually similar images should have smaller distance between them while perceptually different images should have larger distance between them.

Therefore, given a query, the higher the retrieval accuracy, the better the distance measure. For online retrieval, computation efficiency is also a factor to be considered when choosing a distance measure.

Variety of distance measures have been used in image retrieval; they include city block distance, Euclidean distance, cosine distance, histogram intersection distance, χ^2 statistics distance, quadratic distance, and Mahalanobis distance [1]. In this chapter, commonly used similarity measures will be described and examined. A number of widely used performance measurements will also be discussed.

12.2 Similarity Measures

12.2.1 Distance Metric

A similarity measure $d(\mathbf{x}, \mathbf{y})$ between two feature vectors $\mathbf{x}$ and $\mathbf{y}$ is normally defined as a metric distance. $d(\mathbf{x}, \mathbf{y})$ is a metric distance if for any of two data points $\mathbf{x}$ and $\mathbf{y}$ in space; it satisfies the following properties:

© Springer Nature Switzerland AG 2019

D. Zhang, *Fundamentals of Image Data Mining*, Texts in Computer Science, https://doi.org/10.1007/978-3-030-17989-2_12

(1) $d(\mathbf{x}, \mathbf{y}) \geq 0$ (non-negativity)
(2) $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$ (identity)
(3) $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ (symmetry)
(4) $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$ (triangle inequality).

12.2.2 Minkowski-Form Distance

The Minkowski-form distance is often called the L_p norm or L_p distance. Given a N-dimensional feature vector of a query image $\mathbf{x} = (x_1, x_2, \ldots x_n)$ and a target image in database $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, the L_p distance is defined as

$$L_p(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} (x_i - y_i)^p \right)^{\frac{1}{p}} \tag{12.1}$$

When $p = 1$, L_1 is called the *city block distance* or *Manhattan distance*:

$$L_1(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i| \tag{12.2}$$

When $p = 2$, L_2 is called the *Euclidean distance*:

$$L_2(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} (x_i - y_i)^2 \tag{12.3}$$

When $p \to \infty$, L_∞ is called the *Chebyshev distance*:

$$L_\infty(\mathbf{x}, \mathbf{y}) = \max_{1 \leq i \leq n} \{|x_i - y_i|\} \tag{12.4}$$

By varying the p values, various Minkowski distances can be created. However, among the many Minkowski-form distances, L_2 is the most widely used similarity measures. This is because L_2 is the most consistent with human perception of image similarity. The agreement between distance and perception is demonstrated in Fig. 12.1, where the unit circles of Minkowski distance with different p values are shown. Points on each of the unit circles all have the same distance to the origin under the corresponding p values. As can be seen, the L_2 unit circle agrees most with human perception among the three p values.

L_2 tends to emphasize or amplify the dimensions with high values due to the use of quadratic function. This can cause undesirable results because the distance value is often determined by a few dominant feature dimensions which are often due to local distortion or noise. This in turn can result in rejecting true positives which are perceptually similar images to the query but have local distortion or noise, e.g., a bite out apple would be rejected from the retrieval list using an intact apple as the

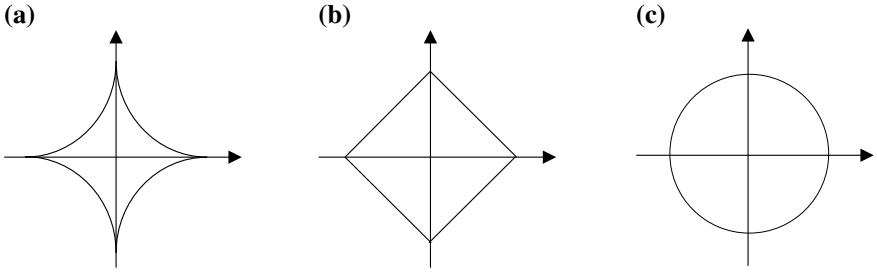

Fig. 12.1 Unit circles of Minkowski distance with different p values. **a** $p = \frac{1}{2}$; **b** $p = 1$; **c** $p = 2$

query. Consequently, L_2 distance can expect lower recall compared with L_1 distance although it can return a top retrieval list with higher precision.

One solution to overcome the lower recall issues of L_2 distance is to apply a logarithm transform to the feature values to suppress the very high feature values and raise the lower feature values, so that all feature values have balanced contributions to the final distance value. Figure 12.2 (top) shows an example histogram from the flower image in Fig. 4.14, notice the histogram feature is dominated by the bins at the end of the histogram. The log-transformed histogram feature vector is shown at the bottom of the figure; it can be seen that while the difference between the feature dimensions has been reduced significantly, the top profile of the histogram has been kept.

12.2.3 Mass-Based Distance

Minkowski-form distance-based similarity measures are basically a matching of two images feature by feature. However, due to image features usually have very high dimensions and features are imperfect, this kind of detailed feature by feature matching can result in undesirable outcomes in many situations. For example, different images can have the same feature vector as shown in Fig. 12.3, and similar images can also have almost completely different feature vectors as shown in Fig. 12.4. In both cases, the L_p-based similarity measure would give a totally incorrect matching result.

The issue demonstrates L_p-based similarity measures that are not robust. This drawback can be overcome by incorporating neighboring data in the decision-making process.

To address the sensitivity issue of L_p, a mass-based similarity measure m_p has been proposed [2]. The idea of m_p is to use neighborhood data to make a similarity decision collectively instead of making a similarity decision just based on two instances alone. Specifically, m_p uses the neighborhood *data mass* at each subspace of R^d to replace the *difference* at each dimension in the Minkowski-form distance.

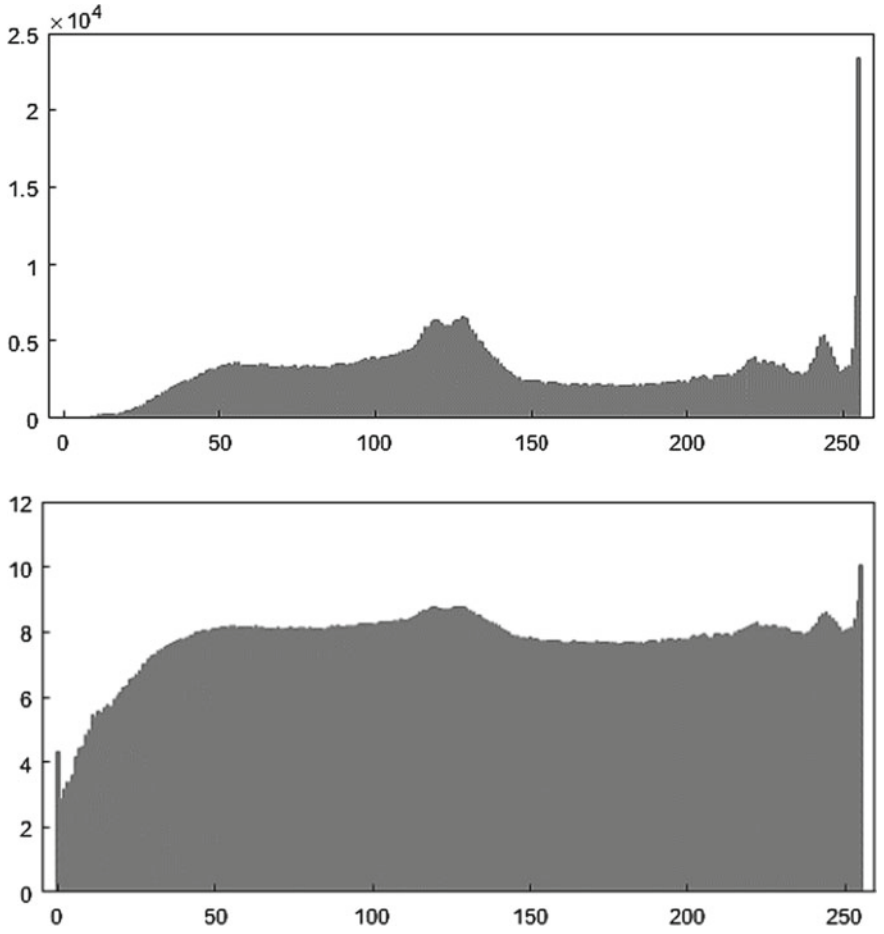

Fig. 12.2 Top: a histogram feature vector; Bottom: the log-transformed histogram feature vector from the top histogram

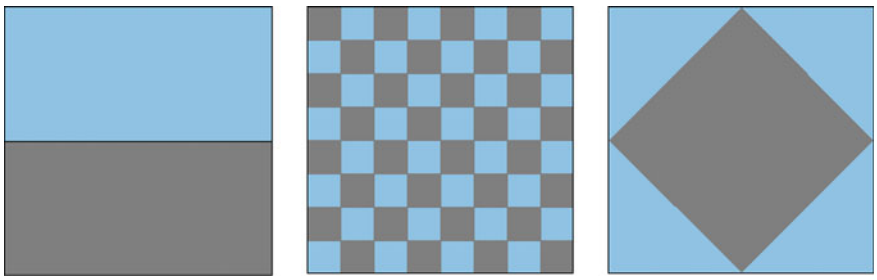

Fig. 12.3 The three images have the same histogram

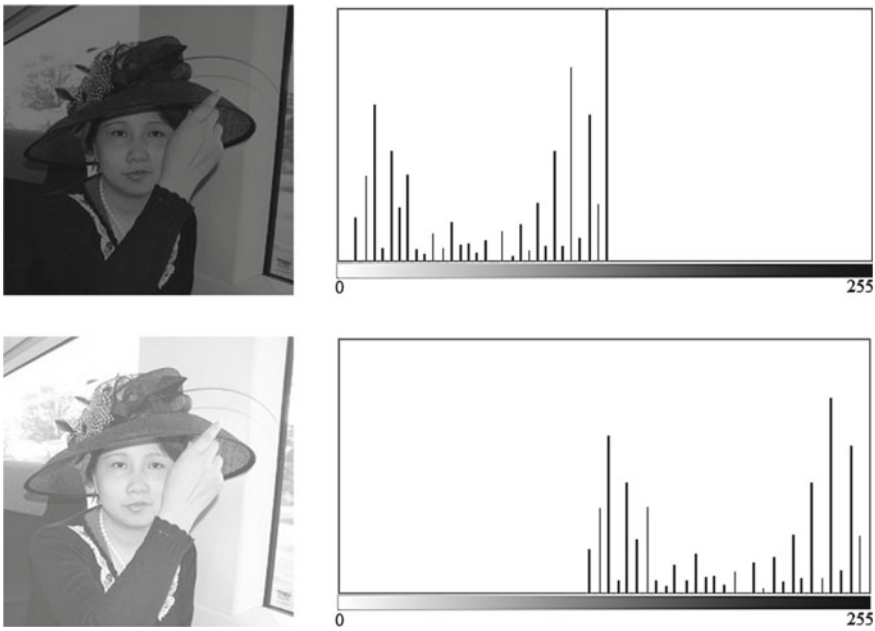

Fig. 12.4 The two images with different brightness have almost completely different histograms

The idea of m_p is based on a distance–density model described by Krumhausl [3] and a psychological discovery that two instances in a sparse region are perceptually more similar than they are in a dense region.

Given two data points in R^n: $\mathbf{x}$ and $\mathbf{y}$, m_p works by defining a region $R(\mathbf{x}, \mathbf{y})$ between the two instances (including the two instances) and finding the data mass of the region. Data mass is the number of data instances from dataset that falls in this region. $R(\mathbf{x}, \mathbf{y})$ is a d-dimensional region, and the ith dimension of $R(\mathbf{x}, \mathbf{y})$ is given as $R_i(\mathbf{x}, \mathbf{y})$, $i = 1, 2, \ldots, n$. The data mass of each $R_i(\mathbf{x}, \mathbf{y})$ depends on the distribution of the data in R^n space.

Specifically, the mass-based similarity measure m_p is defined as (12.5) [4]

$$m_p(\mathbf{x}, \mathbf{y}) = \left(\frac{1}{n} \sum_{i=1}^{n} \left(\frac{|R_i(\mathbf{x}, \mathbf{y})|}{N} \right)^p \right)^{1/p} \tag{12.5}$$

where

- $|R_i(\mathbf{x}, \mathbf{y})|$ is the data mass in region of $R_i(\mathbf{x}, \mathbf{y})$,
- N is the total number of instances in the dataset,
- $R_i(\mathbf{x}, \mathbf{y}) = [\min(x_i, y_i) - \sigma, \max(x_i, y_i) + \sigma]$, and
- σ is a small number and $\sigma \geq 0$.

Figure 12.5 [4] illustrates a data distribution in 2D space and the calculation of data mass between two data points **x** and **y**. For convenience of calculation, σ is set as 0. With this data distribution, the data mass in $R_1(\mathbf{x}, \mathbf{y}) = [x_1, y_1]$ is $|R_1(\mathbf{x}, \mathbf{y})| = 63$ while the data mass in $R_2(\mathbf{x}, \mathbf{y}) = [x_2, y_2]$ is $|R_2(\mathbf{x}, \mathbf{y})| = 40$.

L_p is essentially a *fine similarity measure* between two instances and is sensitive due to the use of feature by feature matching between two instances. It can result in completely incorrect match in cases shown in Figs. 12.2 and 12.3. On the other hand, m_p is essentially a *coarse similarity measure* between two instances, because it is computed using collective info from neighborhood data mass. Therefore, m_p can be *inaccurate* in situations when the features of the two instances are close.

To overcome the limitations of both the L_p and m_p, a hybrid similarity measure called h_p can be used, which is defined in (12.6)

$$h_p(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} (|x_i - y_i| \times |R_i(x, y)|)^p \right)^{\frac{1}{p}} \tag{12.6}$$

h_p is a compromise, it overcomes the sensitivity drawback of L_p while preserves its accuracy. To prevent h_p from being disproportionally determined by a few dominant dimensional features, a log transform on m_p is applied before computing h_p. The modified h_p is given as (12.7)

$$h_p^{'}(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} [|x_i - y_i| \times \log(|R_i(x, y)|)]^p \right)^{\frac{1}{p}} \tag{12.7}$$

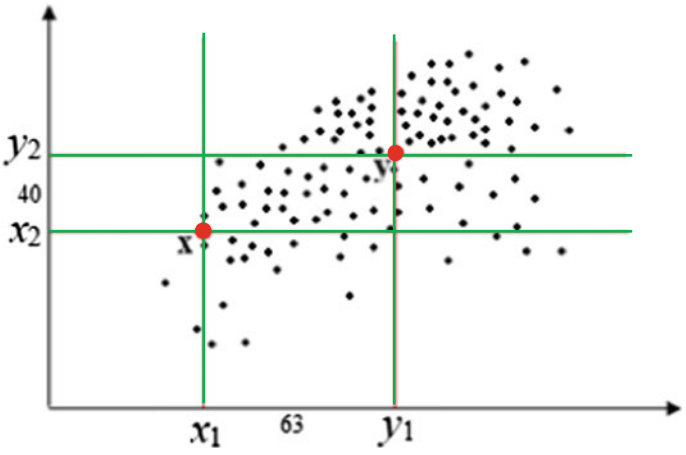

Fig. 12.5 Illustration of m_p dimension calculation between two data points **x** and **y**

12.2.4 Cosine Distance

The *cosine distance* computes the distance between two vectors in terms of direction, irrespective of vector lengths. The distance is computed based on the rule of dot product:

$$\mathbf{x} \times \mathbf{y} = |\mathbf{x}| \times |\mathbf{y}| \times \cos\theta \tag{12.8}$$

where θ is the angle between vector $\mathbf{x}$ and $\mathbf{y}$, and $|\mathbf{x}|$ and $|\mathbf{y}|$ are the magnitudes of $\mathbf{x}$ and $\mathbf{y}$, respectively. The cosine distance is then defined as

$$\cos(\mathbf{x}, \mathbf{y}) = 1 - \frac{\mathbf{x} \cdot \mathbf{y}}{|\mathbf{x}| \cdot |\mathbf{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}} \tag{12.9}$$

If both x_i and y_i have been normalized to probability values between 0 and 1, cos ($\mathbf{x}$, $\mathbf{y}$) becomes

$$\cos(\mathbf{x}, \mathbf{y}) = 1 - \sum_{i=1}^{n} x_i y_i \tag{12.10}$$

The key feature of the cosine distance is that it is invariant to scale change in contrast to Minkowski distance. Figure 12.6 shows the comparison between the cosine distance and the two Minkowski-form distances in two-dimensional space. It can be observed that both L_2 and L_1 respond to scale changes, while cosine distance does not. For example, in Fig. 12.6b, cos($\mathbf{x}$, $\mathbf{y}$) = cos($\mathbf{x}_1$, $\mathbf{y}$), while $L_1(\mathbf{x}, \mathbf{y}) \neq L_1(\mathbf{x}_1, \mathbf{y})$ and $L_2(\mathbf{x}, \mathbf{y}) \neq L_2(\mathbf{x}_1, \mathbf{y})$. The scale invariance can be useful in situations where directional features are more important than magnitudes. For example, if cosine distance is used, two similar colors will keep their similarity after scaling of the color components.

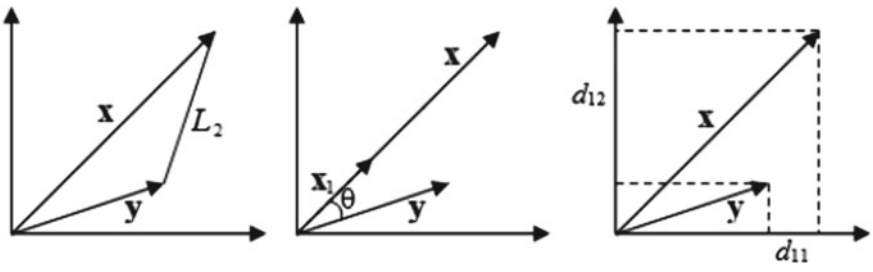

Fig. 12.6 Comparison between the cosine distance and L_p distance. **a** $L_2(\mathbf{x}, \mathbf{y}) = L_2$; **b** cos($\mathbf{x}$, $\mathbf{y}$) = cosθ; **c** $L_1(\mathbf{x}, \mathbf{y}) = d_{11} + d_{12}$

12.2.5 χ^2 Statistics

In χ^2 test, both $\mathbf{x}$ and $\mathbf{y}$ are treated as random variables, the χ^2 statistics is then used to test if the two variables are correlated/independent each other, and how much they are correlated. Formally, χ^2 statistics is defined as (12.10)

$$\chi^2(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \frac{(x_i - m_i)^2}{m_i} \tag{12.11}$$

where $m_i = (x_i + y_i)/2$, which is regarded as the *expected value* for dimension i. A low χ^2 value means that both $\mathbf{x}$ and $\mathbf{y}$ are from the same probability distribution and there is a high correlation between the two feature vectors, which indicates the images represented by the two feature vectors are similar. An advantage of using χ^2 statistics is that it can overcome the mismatch between two histograms from images with very different lighting conditions as shown in Fig. 12.4.

12.2.6 Histogram Intersection

A histogram is a distribution function with a particular shape of area. The histogram intersection is to test how much area two distributions $\mathbf{x}$ and $\mathbf{y}$ share, the more area they share, the more similar the two distributions are (Fig. 12.7). Mathematically, a histogram intersection is defined as

$$HI(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{n} \min(x_i, y_i)}{\min(|\mathbf{x}|, |\mathbf{y}|)} \tag{12.12}$$

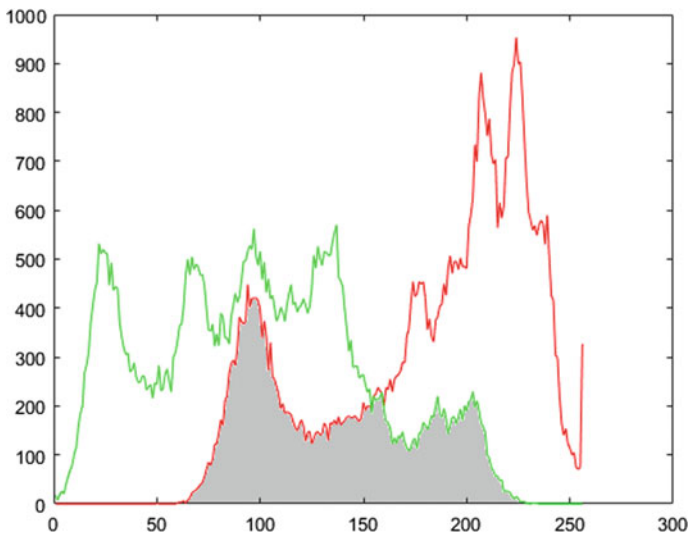

Fig. 12.7 Histogram intersection of two histograms shown as gray area

If both x_i and y_i have been normalized to probability values between 0 and 1, *HI* is simplified as (12.13)

$$HI(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \min(x_i, y_i) \tag{12.13}$$

For two identical histograms, their *HI* value is the maximum 1 and for two similar histograms, their *HI* value is a high. For two different histograms such as the two histograms shown in Fig. 12.4, their *HI* value is close to zero. The *HI* distance is defined as

$$d_{HI}(\mathbf{x}, \mathbf{y}) = 1 - \sum_{i=1}^{n} \min(x_i, y_i) \tag{12.14}$$

d_{HI} also has the same histogram mismatching issue as the L_p distance.

12.2.7 Quadratic Distance

The distances or measures we have introduced so far all make two implicit assumptions: (a) the two feature vectors to be measured **x** and **y** have equal number of dimensions; and (b) the dimensions of **x** and **y** are independent. However, there are applications and situations where these two conditions are not met. For example, the dominant color descriptors described in Chap. 4 typically have different number of dimensions, and colors of neighboring histogram bins are correlated with each other. The quadratic distance measure is one of the methods to address the unequal number of dimensions between two feature vectors and capture the cross dimension information in a feature vector.

The quadratic-form distance between two n-dimensional feature vectors **x** and **y** is given by

$$d_q(\mathbf{x}, \mathbf{y}) = \left[(\mathbf{x} - \mathbf{y})^T \mathbf{A} (\mathbf{x} - \mathbf{y})\right]^{\frac{1}{2}} \tag{12.15}$$

where

- T means transpose,
- $\mathbf{A} = [a_{ij}]$ is an $n \times n$ matrix,
- a_{ij} is the similarity coefficient between dimensions i and j,
- $a_{ij} = 1 - d_{ij}/d_{\max}$,
- $d_{ij} = |x_i - y_j|$, and
- $d_{\max} = \max_{1 \le i,j \le n} d_{ij}$.

For numerical calculations, (12.15) is expanded as (12.16)

$$d_q = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} y_i y_j - 2 \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i y_j \right)^{\frac{1}{2}} \tag{12.16}$$

The a_{ij} is the similarity coefficient between x_i and y_j; it is a *weight* on a cross-dimensional element of the two feature vectors, the higher the correlation between the two cross dimensions, the more the weight is given on that element.

For two feature vectors $\mathbf{x}$ and $\mathbf{y}$ with different dimensions n and m, respectively, the quadratic distance between $\mathbf{x}$ and $\mathbf{y}$ is given as (12.17)

$$d_q = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{m} \sum_{j=1}^{m} a_{ij} y_i y_j - 2 \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} x_i y_j \right)^{\frac{1}{2}} \tag{12.17}$$

If the dimensions of both the two feature vectors $\mathbf{x}$ and $\mathbf{y}$ are independent each other, e.g., after certain decorrelation operations, the quadratic distance between $\mathbf{x}$ and $\mathbf{y}$ is given as (12.18)

$$\begin{aligned} d_q &= \left(\sum_{i=1}^{n} x_i^2 + \sum_{j=1}^{m} y_j^2 - 2 \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} x_i y_j \right)^{\frac{1}{2}} \\ &= \left(\sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} \left(x_i - y_j \right)^2 \right)^{\frac{1}{2}} \end{aligned} \tag{12.18}$$

Equation (12.18) is a *weighted Euclidean distance*; one can expect that d_q is a more desirable similarity measure than both L_2 and d_{HI}; however, the determination of the weights is an issue.

12.2.8 Mahalanobis Distance

The *Mahalanobis distance* is a special case of the quadratic-form distance (12.15) in which the transform matrix is determined by the *covariance matrix* obtained from a training set of feature vectors, that is, $\mathbf{A} = \Sigma^{-1}$. In order to apply the Mahalanobis distance, a feature vector $\mathbf{x}$ is regarded as a multivariate random variable $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ from certain probability distribution. Then, the correlation matrix is given by $\mathbf{R}$ where

- $\mathbf{R} = [r_{ij}]$
- $r_{ij} = E\{x_i x_j\}$ which is the mean of the random variable $x_i x_j$.
- The covariance matrix Σ is given by $\Sigma = [\sigma_{ij}^2]$.
- where $\sigma_{ij}^2 = r_{ij} - E\{x_i\} E\{x_j\}$.

The Mahalanobis distance between two feature vectors $\mathbf{x}$ and $\mathbf{y}$ is given as

$$d_m(\mathbf{x}, \mathbf{y}) = [(\mathbf{x} - \mathbf{y})\Sigma^{-1}(\mathbf{x} - \mathbf{y})]^{\frac{1}{2}} \tag{12.19}$$

In the special case where x_i are statistically independent but have unequal variances, Σ is a diagonal matrix as follows:

$$\Sigma = \begin{bmatrix} \sigma_1^2 & & & 0 \\ & \sigma_2^2 & & \\ & & \ddots & \\ 0 & & & \sigma_n^2 \end{bmatrix} \tag{12.20}$$

In this case, the Mahalanobis distance is reduced to a simpler form:

$$d_m(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\sigma_i^2} \right)^{\frac{1}{2}} \tag{12.21}$$

Equation (12.21) is another *weighted Euclidean distance*. It gives more weight to dimensions with smaller variance and less weight to dimensions with larger variance. d_m can be regarded as a *standard Euclidean distance*. The Euclidean distance is just a special case of Mahalanobis distance when the covariance matrix Σ is the identity matrix.

12.3 Performance Measures

After image ranking, we need a measure to tell how good is the ranking by a similarity measure we have discussed above. Specifically, we need to assess how many relevant images have been retrieved on the top list and how many relevant images have missed from the top list. The information from the top list of retrieval lets us tell how well a similarity measure performs. A performance measure is usually based on statistics of a *subjective test* which is a test of identifying relevant images to the query and how relevant they are to the query. Different performance measures often use different subjective tests, resulting in different definitions of retrieval performance. In this section, several commonly used performance measures are described and discussed.

12.3.1 Recall and Precision Pair (RPP)

RPP is one of the most widely used retrieval performance measurements in literature. In RPP, for *each query image*, images in a dataset are divided into two categories: *relevant* images (1) and *irrelevant* images (0), based on their similarity

to the query. The similarity is determined by a subjective test on a group of subjects. In the subjective test, each subject selects items relevant to the query from the dataset. An item selected by more than a number of subjects as a relevant image is given a label of "1"; otherwise, it is regarded as an irrelevant image and is given a label of "0".

Now given a query image I and a retrieval list returned by a similarity measure, the *precision* (P) and *recall* (R) statistics are then computed based on the "0" and "1" images presented on the top retrieval list:

$$P = \frac{r}{n_1} = \frac{number\ of\ relevantretrieved\ images}{number\ of\ retrieved\ images} = \frac{|\{relevant\ images\} \cap \{retrieved\ images\}|}{|\{retrieved\ images\}|} \tag{12.22}$$

$$R = \frac{r}{n_2} = \frac{number\ of\ relevant\ retrieved\ images}{number\ of\ relevant\ images\ in\ DB} = \frac{|\{relevant\ images\} \cap \{retrieved\ images\}|}{|\{relevant\ images\}|} \tag{12.23}$$

P can be interpreted as the probability that a retrieved image is relevant, while R can be interpreted as the probability that a relevant image is returned by a retrieval.

The RPP is often given in the following form:

$$P = \frac{t}{t + f_p} \tag{12.24}$$

$$R = \frac{t}{t + f_n} \tag{12.25}$$

where t, f_p, and f_n stand for "True Positive" (a hit), "False Positive" (a mismatch), and "False Negative" (a miss), respectively.

Precision measures the retrieval accuracy while recall measures the retrieval robustness; both are important for a similarity measure. Precision and recall are inversely related, i.e., precision normally degenerates as recall increases. Translating into actual image retrieval result, this inverse relationship means that the shorter the retrieval list (low recall), the higher the accuracy and vice versa.

The RPP based on a single query does not provide a complete picture of the performance of a similarity measure; usually, a number of queries are tested and the P values at each of the R values are averaged. The average (R, P) values are then plotted on a graph to get an approximate performance of a similarity measure.

Figure 12.8 shows an RPP curve from an averaged retrieval result. It can be observed from the figure that, as the *recall* increases (longer retrieval list), the *precision* goes down rather sharply in this case.

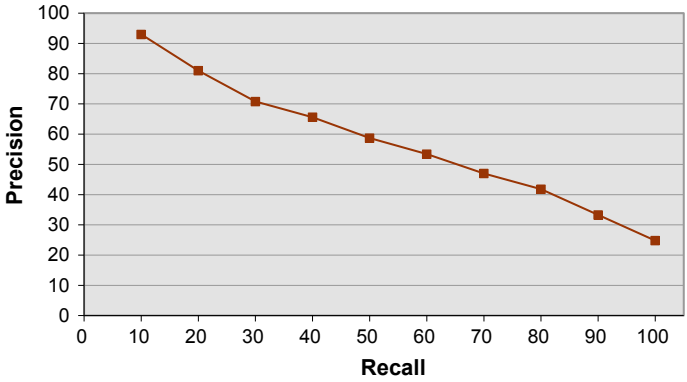

Fig. 12.8 An RPP curve from an actual image retrieval

The RPP curve gives a good picture of a similarity measure's performance. A good similarity measure will have an RPP curve with two characteristics: (a) a high start (how high depends on applications), e.g., 70%+; and (b) a gentle drop. However, it is often difficult to achieve both the two goals; a retrieval method usually either targets high precision on the top list of a retrieval result or targets higher recall depending on applications. Therefore, the P values at the lower recall values are much more important than those at higher recalls. For example, in Fig. 12.8, the P values before the 30% of recall are all above 70%, which indicates a good retrieval result although the full RPP curve does not look good.

Although RPP is intuitive, there are several drawbacks to this performance measure.

- **Need a ground truth**. In order to compute the R value, we need to know the total number of relevant images in a database which is essentially a ground truth. This limits the application of the RPP to databases with small scale.
- **Unrealistic relevance values**. The binary relevance value given to each of the images in the database is not realistic, because image similarity is probabilistic and between 0 and 1.
- **Missing ranking information**. All relevant images on the retrieval list are given the same relevance value; ranking information is not considered in defining the relevance values. However, a similar image at rank 1 is more relevant than a similar image at rank 10.
- **A pair of conflict values**. It is often awkward to tell the performance of a retrieval using *two* values which do not agree with each other. For example, if we have a retrieval which gives $P = 90\%$ and $R = 10\%$, it is difficult to tell how well is the retrieval result. Therefore, we need a measure to reconcile the P and R pair.

A number of other performance measures have been designed to address the above issues.

12.3.2 *F*-Measure

A performance measure which reconciles the *precision* (P) and *recall* (R) into one is called the F-measure. It is defined as *harmonic mean* of P and R, which turns out to be the square of the *geometric mean* of P and R divided by the *arithmetic mean* of P and R.

$$F = \frac{P \cdot R}{\frac{P+R}{2}} = 2 \cdot \frac{P \cdot R}{P+R} \tag{12.26}$$

It can be shown that the following is true:

$$F = \alpha P + (1 - \alpha) R \tag{12.27}$$

where $\alpha = \frac{t+f_p}{2t+f_p+f_n}$. Therefore, F turns out to be a weighted sum of P and R. The weight α can be adjusted to suit a specific data or application.

Figure 12.9 shows the F curve against the same P-R curve from Fig. 12.8. It can be observed that the F score has a low start and reaches the maximum value at the point where the P score is the closest to the R score. Before the peak point, precision is more important; after the peak point, recall is more important. Therefore, the peak point is the optimal tradeoff between P and R. Overall, the higher the F score, the better tradeoff between P and R. Therefore, for two similarity measures or retrieval results, the one gives a higher F score is usually better.

The advantage of using F-measure is that a single value can be used to tell the difference between two similarity measures or two retrieval results. However, it is not as intuitive as the RPP and it is not easy to interpret an F score. It appears we could have used an AUC or *area under* (the RPP) *curve*, as an alternative to F-measure. The AUC would be not only a single value but also as intuitive as the RPP. However, the AUC would not be able to differentiate an RPP with high start but sharp drop (a sliding RPP) and an RPP with low start but relatively flat (a steady

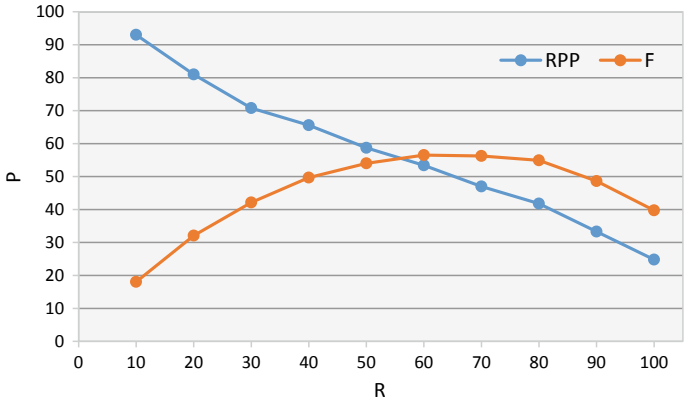

Fig. 12.9 The *RPP* curve and F curve from an actual image retrieval

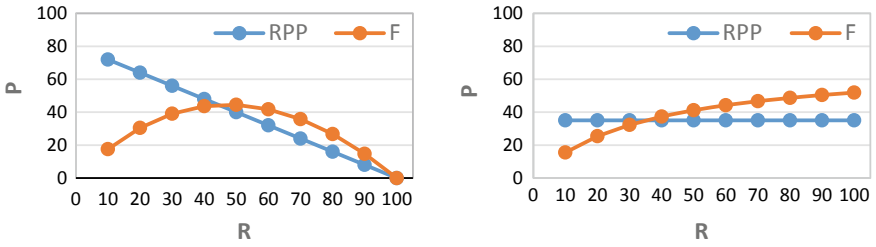

Fig. 12.10 The F curves for two different RPP curves with the same AUC

RPP). The former RPP is usually more preferable than the later one, even though the former is not a good RPP either. Therefore, the AUC would not be as effective as the F-measure. For example, Fig. 12.10 shows a *sliding* RPP and a *steady* RPP with the same AUC. However, the maximum/optimal F score of the slide RPP is at middle of the RPP curve, while the maximum/optimal F score of the steady RPP is at the very end of the RPP curve, which is undesirable, because it is unlikely that a user would wait until all the relevant images are shown up.

12.3.3 Percentage of Weighted Hits (PWH)

The PWH can be regarded as a weighted recall. The subjective test is the same as in RPP, that is, each subject select items relevant to the query from the dataset. However, instead of measuring recall based on *binary relevance value* as in RPP, PWH assigns a *weighted relevance value* w_i to each item in the dataset. The sum of the weights w_i is equivalent to the number of subjects selecting item i as relevant or similar to the query. Therefore, PWH is defined as

$$PWH = \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{N} w_i} \tag{12.26}$$

where n is the number of items retrieved and N is the total number of items in the database. It is easy to see that the R measure in RPP is a special case of PWH when w_i takes the value of 0 and 1. Similar to the R measure, PWH needs to identify every item in the database as relevant or not relevant to the query, and this need for ground truth of the database limits its usage.

12.3.4 Percentage of Similarity Ranking (PSR)

PSR is a performance measure of detecting the agreement between an algorithm ranking and a human ranking [5]. In this method, each subject assigns a similarity rank to each item i in the dataset based on the item's similarity to the query j. For each query j, the final result of the subject test is a matrix $\{Q_j(i, k)\}$, where

- $Q_j(i, k)$ is the number of subjects ranking item i at kth position.
- $\bar{p}_j(i)$ and $\bar{\sigma}_j(i)$ are the mean and variance of each row of $\{Q_j(i, k)\}$.
- $\bar{p}_j(i)$ represents the average ranking of the ith image to query j.
- $\bar{\sigma}_j(i)$ represents the degree of agreement among the subjects on ranking item i.

Given a query j, if a *retrieval algorithm* returns an item i at rank $P_j(i)$, the *percentage similarity ranking* $S_j(i)$ is defined as

$$S_j(i) = \sum_{k=P_j(i)-\frac{\sigma_j(i)}{2}}^{P_j(i)+\frac{\sigma_j(i)}{2}} Q_j(i,k) \tag{12.27}$$

A plot of $S_j(i)$ as a function of item i shows the retrieval performance of the retrieval algorithm. A high $S_j(i)$ curve indicates a high retrieval accuracy of the algorithm. An average PSR value can also be computed as the overall performance of the retrieval algorithm.

The PSR takes into account the number and agreement of human ranking. However, if for a query, the percentage of humans giving a particular item at particular ranking is high (high degree of agreement on ranking the item), then the variance for the ranking would be small. This would result in unusually low PSR if the retrieval algorithm's ranking differs from the subject mean ranking. On the other hand, if the variance of a ranking is large, then the PSR would be unusually high even if the ranking by the retrieval algorithm differs substantially from the subject mean ranking.

12.3.5 Bullseye Accuracy

A simple Bullseye performance measure (BEP) is called *Precision at K* or P@K, which is defined as the ratio of the "number of relevant images on the top K retrieval" to K. This ratio is called a *Bullseye score*. The higher the Bullseye score, the better the retrieval. P@K is a very convenient and useful performance measure because it does not need the ground truth of the database. P@K measure is widely used in applications where the data is massive and the accuracy of the top retrievals is the most important. For example, in online web search, users only care about the relevance of the top pages returned by a search engine.

The K can be determined based on the actual data or application. If the total number of relevant images in the database is known to be N, K is typically determined as N or $2N$. In practice, Bullseye scores are obtained from a number of queries and an average Bullseye score is obtained as the overall performance value for a retrieval algorithm.

The *Bullseye score* can also be defined as the *Average rank* (AR). In AR, instead of precisions, the ranks of relevant images on the top K retrieval list are averaged. The lower the AR, the better the retrieval result.