

A Hybrid Data Dependent Dissimilarity Measure for Image Retrieval

Hamid Shojanazeri

Faculty of Science and Technology
School of Engineering and Information Technology
Federation University Australia, Gippsland, Australia
hshojanazeri@federation.edu.au

Shyh Wei Teng, Guojun Lu

Faculty of Science and Technology
School of Engineering and Information Technology
Federation University Australia, Gippsland, Australia

Abstract— In image retrieval, an effective dissimilarity measure is required to retrieve the perceptually similar images. Minkowski-type (l_p) distance is widely used for image retrieval, however it has its limitations. It focuses on distance between image features and ignores the data distribution of the image features, which can play an important role in measuring perceptual similarity of images. l_p also favours the most dominant components in calculating the total dissimilarity.

A data dependent measure, named m_p -dissimilarity, which estimates the dissimilarity using the data distribution, has been proposed recently. Rather than relying on geometric distance, it measures the dissimilarity between two instances in each dimension as a probability mass in a region that encloses the two instances. It considers two instances in a sparse region to be more similar than in a dense region. Using the probability of data mass enables all the dimensions of feature vectors to contribute in the final estimate of dissimilarity, so it does not just heavily bias towards the most dominant components. However, relying only on data distribution and completely ignoring the geometric distance raise another limitation. This can result in finding two instances similar only due to being in a sparse region, however if the geometric distance between them is large then they are not perceptually similar. To address this limitation we proposed a new hybrid data dependent dissimilarity (*HDDD*) measure that considers both data distribution as well as geometric distance. Our experimental results using Corel database and Caltech 101 show that (*HDDD*) leads to higher image retrieval performance than l_p distance (l_pD) and m_p .

Keywords—Image retrieval; Dissimilarity measure; Data dependent dissimilarity measure

I. INTRODUCTION

With development of the Internet and the advancement of image capturing devices along with the cheaper memory devices, the size of digital image collection is increasing rapidly. So, efficient and robust image retrieval system is increasingly in demand by different domains, e.g. fashion, crime prevention, publishing, medicine, and architecture, in order to make use of the images in a large database effectively.

In image retrieval, an image is presented to a database as a query and a set of perceptually similar images are retrieved. An effective image retrieval system requires images to be represented by a robust and discriminative feature vector/descriptor. Images can be described using different low-level features such as colour, shape and texture [1, 2]. In

addition, an effective dissimilarity measure plays an important role in comparison of the query image feature vector and those of the stored images. Most researchers employ the Minkowski-type (l_p -norm) metric, particularly the (l_2 -norm): well known as Euclidean Distance (ED), as the dissimilarity measure [3].

One of the challenging problems in image retrieval is the selection of an effective dissimilarity measure to compare the images. l_pD is the main dissimilarity measure in many applications, such as data mining, clustering, and image retrieval [4]. As we will explain in greater detail in Section III, l_pD has two main limitations: (1) it focuses on distance between features of images and ignores the data distribution of image features, which can play an important role in measuring perceptual similarity of images; and (2) l_pD favours the most dominant components in calculating the total distance.

Psychologists have highlighted the important role of data distribution in humans perceiving similarity between instances in a dataset. They argued that the dissimilarity between two instances is influenced by other instances in the dataset. Two instances in a relatively dense area are perceptually less similar than two instances of the equal distance in less dense area [5]. For example two red apples among green apples perceptually look more similar than the same two red apples among other red apples. Based on this idea, a data dependent dissimilarity measure called “ m_p -dissimilarity” has been proposed to address the problems with l_pD [3].

This data dependent dissimilarity measure [3] calculates the dissimilarity between two instances using data distribution in the dataset instead of geometric distance as used in l_pD . Basically in this method, the value calculated to indicate the dissimilarity between two instances will also take into account their dissimilarity to the other instances in the dataset. A region is defined between two instances. Two instances are less dissimilar if there are not many other similar instances falling in this region and they are more dissimilar if the number of similar instances is large.

The data dependent dissimilarity measure has shown promising results in classification of data such as music, text and digits. In this work we will investigate the performance of m_p -dissimilarity in image retrieval. To evaluate this method, we use two datasets, which are represented with two different

sets of features, colour histograms and local binary patterns (LBP). We use colour histograms as they are more intuitive to explain l_pD limitations and m_p strengths on Corel dataset as is discussed in following sections. Also, we used Caltech 101, represented by LBP features as a real world dataset to evaluate m_p performance.

The rest of the paper is organised as follows, two dissimilarity measures: l_pD and m_p -dissimilarity, are discussed in Section II. The experimental study results are provided in Section III, followed by conclusions in the last section.

II. DISSIMILARITY MEASURE BASED ON GEOMETRIC MODELS

In image retrieval, images are represented using feature vectors. To retrieve a similar set of images, an effective dissimilarity measure must be used to compare the feature vectors. In the following, we will review an existing dissimilarity measure commonly used in image retrieval and a new dissimilarity measure.

A wide range of geometric dissimilarity measures are discussed in [4]. [6, 7] have each provided a comprehensive analysis and comparison of the dissimilarity metrics in image retrieval. The study in [6] has compared the performance of Histogram Intersection, Minkowski-form, Quadratic and Mahalanobis Distance. Its results have shown that ED has achieved the best retrieval results. [7] has compared the performance of sum of squared of absolute differences (SSAD), sum of absolute difference, maximum value, Canberra, city block, Minkowski ($p=3$) and ED on the same version of Corel database [8] which has been used in this work. Its results have also shown that the ED is the most suitable dissimilarity metric for image retrieval.

Generally, the distance between two d -dimensional vectors x and y based on l_pD is defined as follows [1]:

$$l_p(x, y) = \|x - y\|_p = \left(\sum_{i=1}^d \text{abs}(x_i - y_i)^p \right)^{1/p} \quad (1)$$

where $p > 0$, $\|\cdot\|_p$ is the p order norm of a vector, x_i and y_i are the i^{th} component of a vector and $\text{abs}(\cdot)$ is the absolute value. The limit condition is defined as:

$$l_\infty(x, y) = \|x - y\|_\infty = \max_i \text{abs}(x_i - y_i) \quad (2)$$

l_pD is a popular choice of distance function as it intuitively corresponds to the distance defined in the real three-dimensional world. It has been widely used in many image retrieval systems as the dissimilarity measure to compare the feature vectors derived from images [9-13].

However, l_pD has its limitations, it measures the distance between image features of two images and completely ignores the distribution of other image features in the dataset. However, the distribution of image features considerably impacts the perceptual similarity between images as it is shown in the following example. Consider two red apples among the many green apples; the red apples perceptually are more similar to each other than green apples. However,

considering another data distribution where the two red apples are among other red apples. In this distribution, all the red apples perceptually are similar to each other and the similarity of that two red apples is perceptually more difficult to be spotted. So the data distribution impacts the perpetual similarity of images.

The other limitation of l_pD is that it favours the most dominant components in calculating the total dissimilarity. This characteristic of l_pD results in negligible contribution of feature dimensions that have small values compared with dimensions that have dominant values [14]. This has negative impact when we are looking for the similar objects, e.g. objects in different backgrounds. Using the colour histograms in this work, the image retrieval results based on l_pD will be heavily bias the towards the dominant colours (e.g. ones of the background), so the detailed colours within the objects in the images might not have the satisfactory level of influence in retrieving the closest match. Following the previous example, consider the two red apples among green apples, however this time one of red apples is located on a black background and rest of apples on white background. Although, the red apples are perceptually more similar, l_pD finds the greatest distance between red apples in white (query) and black backgrounds. This is the result of the great difference between the black background, which is the dominant colour located in the first bin and the white colour located at the last bin of histogram.

III. DATA DEPENDENT DISSIMILARITY MEASURE

To address the discussed limitations of l_pD , a data dependent dissimilarity measure have been proposed [3]. This measure is called m_p -dissimilarity and it focuses on the data distribution of the dataset instead of simply measuring the distance. It has been performed equal or better than l_pD in context of information classification and retrieval problems. This method has been evaluated using text, music, digits and artificial datasets.

This idea is based on the distance-density model proposed by Krumhauhl [5] which prescribes that two instances in a sparse region are more similar than two instances in a dense region. To measure the dissimilarity between two instances: x and y , it defines a region between them and search for other instances in the dataset that falls in this region. So, the data distribution plays the main role to determine the number of instances similar to x and y that fall in this region. If this number is large, then m_p considers it as a dense region, and therefore x and y are less similar. Vice versa, if the number is small, then x and y are in sparse region and are considered more similar.

In order to measure the dissimilarity between x and y , m_p considers the relative positions of x and y with respect to the rest of the data distribution in each dimension. The dissimilarity between x and y in dimension i can be estimated as the probability data in the region $R_i(x, y)$ that encloses x and y . If there are many instances in $R_i(x, y)$, then it will be considered as a dense region. Therefore x and y are likely to be more dissimilar in dimension i . Using the same power

mean formulation as in l_p - norm, the data dependent dissimilarity measure based on probability mass can be defined as:

$$m_p(x, y) = \left(\frac{1}{d} \sum_{i=1}^d \text{abs} \left(\frac{|R_i(x, y)|^p}{N} \right) \right)^{1/p} \quad (3)$$

where $|R_i(x, y)|$ is the data mass in which is the number of instances that fall in the region of $R_i(x, y)$, and n is the number of instances in the dataset. The enclosing region is defined as follows. $R_i(x, y) = [\min(x_i, y_i) - \sigma, \max(x_i, y_i) + \sigma]$, and σ is a small number $\sigma \geq 0$. Although m_p employs the same power mean formulation as l_p , the core calculation is based on mass rather than distance. It signifies the degree of dissimilarity: the higher the measure, the more dissimilar the two instances are; just like l_p .

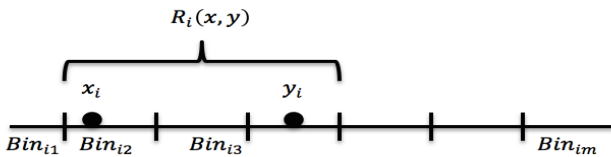


Fig. 1. $R_i(x, y)$ Defined Region between Two Instances using bins

Calculation of m_p is expensive as it requires a range search in each dimension, so to address this problem a new implementation has been propose in [15]. In this new implementation a histogram is used, the real values in each dimension i are divided into m bins. The number of points in each bin is computed as a preprocessing step, and then data mass between two points can be computed using number of bins between them. An illustration of defining $R_i(x, y)$ using bins implementation is shown in Fig 1.

To calculate the data mass between the query and each of the stored images, in each dimension of the colour histograms (96 dimensions representing RGB channels), a region is defined using the values of query and each stored image. The neighbourhood of the region, as shown in Fig.1, is the standard deviation of all values in dataset of that dimension. The number of other images that their values in that dimension of colour histogram falling in the defined region is considered as data mass. As discussed before, the sparser data mass leads to a higher similarity.

As this method works based on the distribution of image feature vectors instead of only considering the distance between each dimension of them, it can address the $l_p D$'s limitation with being in favour of the most dominant components. m_p also considers the dissimilarity of the two instances with the rest of the data in dataset. Generally in a data distribution which has many similar images but are different in their details and these details play a more

important role as compared with the rest of image, m_p is more effective in retrieving perceptually more similar images.

A. Experimental Study of m_p

Since ED, which is a Minkowski-form (l_p) distance where ($p=2$), has been shown to be the most effective existing dissimilarity metrics in image retrieval, we have chosen this metric as the baseline metric to compare with. In this section, we will compare the image retrieval performance of ED and m_p . Colour histograms as an important and useful tool for analyzing colour images that are invariant to rotation and translation are used in this work as image features. They carry the statistical information of the chosen colour space and have been used in several image retrieval research works [16-22].

In this study, we use colour histograms obtained from the RGB colour space. Each colour channel is quantized by steps of eight intensity values and this results in 32 bins for each colour channel (i.e. R, G and B), so a total of 96 dimension feature vectors will represent images.

Also, LBP [23] is a very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. Due to its discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. We extracted the LBP features of Catlech 101 dataset from neighbourhood of 8 pixels and resulted in a feature vector of size 944 dimensions.

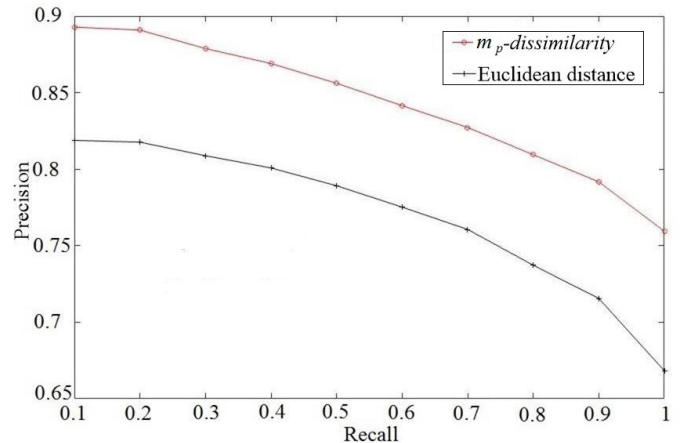


Fig. 2. Precision-recall curves of ED and m_p based on the retrieval results for 200 queries from Corel database

B. Overall Performance Comparison

Image retrieval experiments have been carried out on a version of Corel database [8] that consists of 1000 images from different natural scenes categorized into 10 classes. Images are represented by RGB colour histograms with 32 bins for each colour channel, resulting in a feature vector of 96 dimensions. We randomly selected 200 images from the database as queries. ED and m_p have been used as dissimilarity measures to retrieve set of similar images to the query from the database. The performance of image retrieval using each dissimilarity

measure is evaluated using precision-recall curve. Fig. 2 shows the retrieval results of the 200 query images. It can be seen that m_p has produced better retrieval performance than ED.

Also, the experiment has been performed using LBP features and Caltech 101, which has 101 categories of images from different objects and totally 10k images in the dataset. In this experiment 20 percent of dataset has been used as queries. In this experiment we compared the performance of m_p with ED, Cosine and city block distance metrics. Results been shown in Fig. 3 shows the better retrieval performance achieved by m_p .

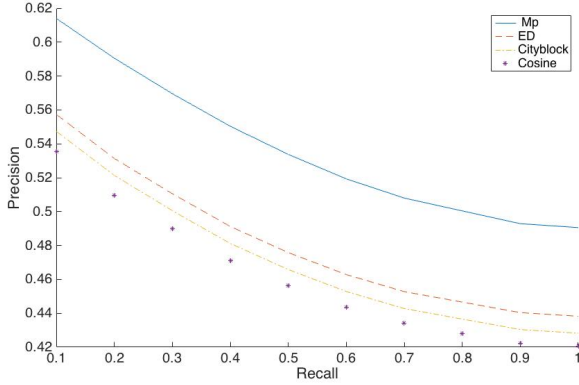


Figure 3. Precision-recall curves of m_p , Ed, cosine and cityblock based on the retrieval results for 2000 queries from Caltech 101 database

C. Visual Examples

To visually investigate the performance of m_p against ED, we present and compare two query images and their top 10 retrieved images as shown in Figs.4-5 from experiment on Corel dataset using color histograms. Visual examples are selection from Corel dataset, which is represented by color histograms to provide better intuition of the results. In each set of images, the top left image is the query and the rest are the 10 highest ranked retrieved images. Generally compared to ED, we can see m_p has retrieved sets of images perceptually more similar to the query.

For example in Fig.4.b we can see the influence of dominant colour (the colour of sands in beach query) in the retrieved images. However in Fig.4.a, by using m_p , we can see the detailed colours (such as red and blue) within the main objects in the query image have played a more prominent role in retrieving perceptually more similar images.

D. Result analysis

To provide further insights into retrieving different images using ED and m_p , we will analyse the colour histograms that are the basis of the calculating ED, as well as the data masses calculated by m_p .

Figs.6-7 show the colour histograms for two query images and their respective top ranked retrieved images using ED and m_p . As we can see, the difference between the largest values in the colour histogram of the query image (Fig.6.d) and that of the top ranked image of using ED (Fig.6.e) is about 5000. However this difference between the histogram of the query

image (Fig.6.d) and that of the top ranked retrieved image using m_p (Fig.6.f) is more than 10000. As discussed in Section II A, ED calculates the distance by only considering the values in corresponding dimensions in the colour histograms, and not the distribution of the values in all the colour histograms. As a result, the dominant dimensions will contribute substantially to the total distance calculated, whereas the contribution from the remaining dimensions might be negligible to impact the final retrieval outcome. For example in Fig.6, the colours of the sand (around histogram bins 28, 59 and 90) on the beach of the query image dominate the distance calculated, whereas the detailed colours, such as the skin colours of people (around histogram bins 21, 43 and 74), though perceptually important when comparing similarity between these images, are not dominant enough to impact the final retrieval outcome. As a result, ED has retrieved Fig.6.b as the top rank rather than Fig.6.c. The colour histograms presented in Fig.9 also follow the same trend.

Next, we will analyse the data mass calculated from the two query images and their top ranked retrieved images using ED and m_p , as shown in Figs.8-9. Data mass is the basis for m_p calculation. m_p takes into account of data distribution by defining a region and by looking for other images in the same bin that have values falling in this region. m_p assigns the maximum dissimilarity (of a dimension) when majority of images having the values falling in the defined region and assigns minimum dissimilarity if less number of images having these values. So, the sparser data mass is considered as the higher similarity.

Figs.8.c and 9.c show the data mass between the feature vectors of each of the two query images and its top ranked retrieved image using m_p . As we can see, the data mass between the query and top ranked retrieved image using m_p is sparser than Figs.8.f and 9.f, which show the data mass of these query images and top ranked retrieved image using ED. Using m_p , this sparser data mass has resulted in Figs.8.b and 9.b being ranked higher than Figs.7.e and 8.e.

IV. LIMITATION OF m_p

In this section we will discuss the limitation that arise when we only rely on data distribution in defining the dissimilarity between two data points. A dissimilarity measure in image retrieval is supposed to retrieve images, which are perceptually similar. As we discussed data distribution has effect on perceptual similarity as considered in m_p . However the geometric distance between two instances should not be ignored, as that measures the dissimilarity in real world. m_p finds two instances similar when the data mass between them is low (they located in a sparse region), in this case we need to find whether they are perceptually similar in real world or not.

Suppose, m_p find two instances similar based on low data mass between them but the geometric distance between them is large, they cannot be considered as similar in real world. We expect that when m_p finds similarity between two instances

their geometric distance is small as well. Similarly, we expect when m_p find two instances dissimilar due to high data mass between them, their geometric distance be large as well to let us consider them dissimilar in real world.

To provide a better insight we refers to the example provided in section II. In that example we discussed that if we have two red apples among many green apples those two red apples are perceptually more similar than if we place those two red apples among many other red apples. So it shows the effect data distribution in perceptual similarity of two images, which is considered by m_p . But in another scenario consider we have one red apple among many green apples and we are looking for apples similar to a green apple (query), m_p will find the red apple as the most similar one due to low data mass between a green and red apple compare to high data mass between green apples. However the red and green apples are very different and have a great geometric distance. In this case we are not considering any of green apples as the most similar because the data mass is high between any two green apples, however in real world they are similar (they have small geometric distance).

We show this limitation through the following example based on how m_p work in each dimension of feature vectors. Table.1 shows a data distribution in one dimension. Suppose we have a dataset $X = \{x(1), \dots, x(10)\}$ in d -dimensional space, to find the similar data to a query $X(Query)$ m_p calculates the data mass in each dimension of feature vectors between query and each of data points in the dataset, then will calculate the total dissimilarity using equation 3. m_p will consider the low data mass between two point as lower dissimilarity and high data mass as higher dissimilarity. We will take a closer look at how data mass works in a dimension of feature vectors x_i . To calculate the data mass, m_p defines a region that enclose $(x_i(Query), x_i(j))$ where $1 < j < 10$ and check how many points in the dataset has the value that fall in this region.

Looking at data mass between query and each data point in the dataset, we will find the lowest data mass ($x_i(Query), x_i(10)$) which is 2. So in this case based on the lowest mass m_p will find $x_i(10)$ as the closet to the query, however, if we look at their geometric distance $ED(x_i(Query), x_i(10))$, they are very different compare to the rest of data points in the dataset. Hence, m_p found the wrong closest match where the data mass between query and a data point was low but they are very different (having large spatial distance). However, there are many other points similar to the query as they have small distance, which are not considered as the closest match because the data mass between query and them was high.

Generally, m_p has the limitation to define the dissimilarity when there is small data mass between two points but they are very different (they have large geometric distance). In this case m_p will find two different data points as similar while it

ignores the similar data points (have small geometric distance) due to high mass between them.

Table 1. Data distribution in one dimension of feature vectors

X	...	x_i
$x(Query)$...	2
$x(1)$...	1
$x(2)$...	1
$x(3)$...	0
$x(4)$...	1
$x(5)$...	1
$x(6)$...	1
$x(7)$...	1
$x(8)$...	0
$x(9)$...	0
$x(10)$...	9

V. HYBRID DATA DEPENDENT DISSIMILARITY (HDDD)

To this end we have discussed about the advantages of considering data distribution as has been proposed in m_p and also the limitation of such a method by completely ignoring the geometric distance in calculation of dissimilarity between two instances. To address the limitation of using m_p as a dissimilarity measure, we propose a hybrid data dependent dissimilarity (HDDD), which take advantage of data distribution and geometric distance at the same time.

A. Proposed Method for HDDD

We discussed that m_p consider the low data mass between two points as higher similarity (less dissimilarity) and high data mass as less similarity (higher dissimilarity). However, we showed in previous section that if we have a low data mass between two points while the geometric distance between them is large, m_p will not find the best closest match. This happens by ignoring the points, which are more similar (having smaller geometric distance), but the data mass between them is high. To address this limitation we need to give a proper weight to the data mass between two points in each dimension.

We need to set a proper weight when data mass between two points is high but they have small geometric distance (they are similar). The weight needs to lower the data mass proportional to the distance between two points. The reason for choosing a weight proportional to the geometric distance between two points and not using a constant weight is as follows. Using m_p as a dissimilarity measure, we expect that closest matches to query be similar and meaningful in real world as well. It means the closest matches are desirable to have smaller geometric distance to the query compare to the rest of instances in the dataset. So basically m_p should rank its

closest matches proportional to their distance to the query, means data points with smaller distance should be ranked higher compare to the ones with larger distance. The other reason that the weight should be proportional to distance and cannot be a constant number is as follows. Suppose we have equal data mass between the query point and two other points in the dataset while their geometric distance between are different, we expect to find the point with smaller distance to query as the closest match, however m_p based on the equal data mass will find both of them as the closest match. So using a constant weight will result that weighted data mass between them again will be equal which does not help in solving the problem.

We propose to change the equation 3, to:

$$HDDD(x, y) = \left(\frac{1}{d} \sum_{i=1}^d abs \left(W \frac{|R_i(x, y)|}{N} \right)^p \right)^{1/p} \quad (4)$$

Where W is the weight that lower the data mass when it is high between two points while having small geometric distance, which we set that to $\alpha ED(x, y)$, where $0 < \alpha < 1$. In this case data mass will be weighted proportional to geometric distance, so when m_p search for the closest match it also considers their similarity in real world (geometric distance) along with the data distribution.

B. Experimental study of HDDD

To evaluate the proposed dissimilarity measure, we will use the two datasets has been used in this work previously, Corel and Caltech 101, and will represent images using LBP features. As we discussed we proposed to use the weight where data mass between query and a point from dataset is high while their spatial distance is small. So to determine the high data mass in each dimension, we consider data mass above the mid point between minimums and maximum of data masses. Also for determining the small spatial distance we consider distances below the mid point between minimum and maximum of distances of query and all data points in each dimension. The α is set to 0.6 as it showed best performance in our experiments. We used ED as the spatial distance as it showed the best performance among other l_p -norms in Fig 3. The overall retrieval results using HDDD, m_p and ED for Corel and Caltech 101 datasets are shown in Figs 10-11.

As it can be seen the retrieval results has been improved using HDDD over m_p and ED.

C. Visual Examples

In this section we present visual examples to give a better insight about the performance of HDDD against m_p , we present and compare two query images and their top 10 retrieved images as shown in Figs.12-13 images are selected from Corel dataset represented by LBP features. Using visual examples we can see how our proposed dissimilarity measure could address the discussed limitation of m_p . In each set of

images, the top left image is the query and the rest are the 10 highest ranked retrieved images. Generally compared to m_p , HDDD has retrieved sets of images perceptually more similar to the query.

For example in Fig 12.a, retrieved images in ranks 1-2 are not form the same class with query and this has been improved in Fig 12.b using HDDD. Also in Fig 13.a, retrieved images in ranks 4-5 belong to a different class with query while in Fig 13.b all the retrieved images are from the same class with query.

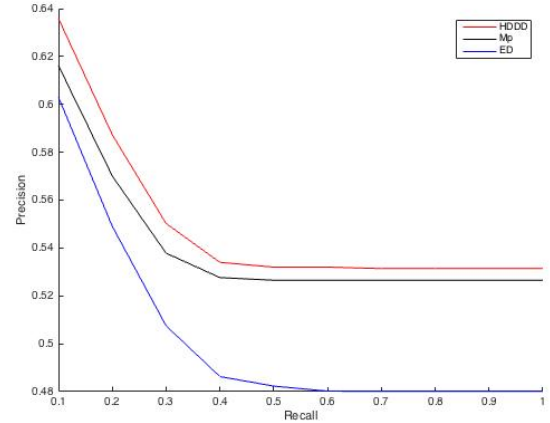


Figure 10. Precision-recall curves of HDDD, m_p and Ed based on the retrieval results from Corel.

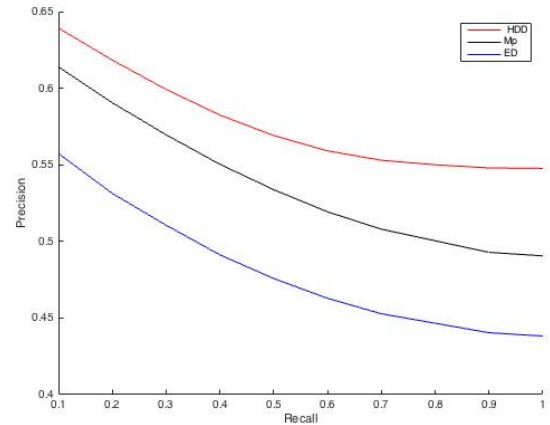


Figure 11. Precision-recall curves of HDDD, m_p and Ed based on the retrieval results from Caltech 101.

C. Result Analysis

In this section we show the limitation of m_p in completely ignoring the spatial distance between two images and how HDDD could improve it. m_p is calculated based on the average data mass in all dimensions between query and each dataset image, so smaller data mass in each dimension will result in smaller m_p . In Fig 12.a m_p ranked retrieved images based on data mass between query and each image in the dataset, the smaller data mass made the elephant ($m_p = 6.43$) and the other irrelevant image from food class ($m_p = 6.53$) come up in the first and second rank compare to the

relevant image in the third rank where ($m_p = 6.59$). However, the Euclidean distance between query, elephant ($ED = 3.8$) and food dish ($ED = 2.9$) are larger than distance between query and third rank which is relevant image ($ED = 1.01$). So ignoring the distance and only relying on data mass caused that smaller data mass retrieved images, which are very different with query in highest ranks.

The same scenario is in Fig 13.a where m_p retrieved images based on lower data mass in high ranks while they are different and having larger distance to the query compare to the relevant images that are ranked lower due to the higher data mass (while having relatively smaller distance to the query). Data masses between query, rank 4 and 5 are ($m_p = 6.72, 7.03$) which are lower than data mass between query and rank 6 ($m_p = 7.8$) however ED for the formers are 2.6, 2.3 Which is much larger than ED for latter, 1.3.

As we showed relying only on data mass between two instances, may result in following situation: m_p ranks instances with lower data mass but different with query (large distance) higher than instances similar to the query (small distance) which has higher data mass. In *HDDD*, we used the ED as the weight in each dimension where data mass is high between query and dataset image but the distance is small. This improved the result by ranking those images higher than the ones with small data mass but large distance. The results in Figs 12.13.b show this effect visually.

VI. CONCLUSIONS

In this work, we studied m_p strengths and limitations as a data dependent dissimilarity measure for image retrieval. Our experimental results show that m_p outperforms ED, Cosine and cityblock distance in retrieving perceptually more similar images. We also showed the limitation of m_p by completely ignoring the spatial distance and only relying on data distribution. This could result in retrieving irrelevant images in high ranks, which has large distance to the query by only considering low data mass between them. We proposed a new hybrid data dependent dissimilarity measure by considering both data distribution and spatial distance. The proposed dissimilarity measure could perform better than m_p and yield perceptually more similar retrieved images.

REFERENCES

- Zhang, D., M.M. Islam, and G. Lu, *A review on automatic image annotation techniques*. Pattern Recognition, 2012. **45**(1): p. 346-362.
- Zhang, D. and G. Lu, *Review of shape representation and description techniques*. Pattern Recognition, 2004. **37**(1): p. 1-19.
- Aryal, S., et al. *Mp-Dissimilarity: A Data Dependent Dissimilarity Measure*. in *Data Mining (ICDM), 2014 IEEE International Conference on*. IEEE.
- Rogers, P.L., *Encyclopedia of distance learning*. 2009: IGI Global.
- Krumhansl, C.L., *Concerning the applicability of geometric models to similarity data: The interrelationship between similarity and spatial density*. 1978.
- Zhang, D. and G. Lu. *Evaluation of similarity measurement for image retrieval*. in *Neural Networks and Signal Processing, 2003. Proceedings of the 2003 International Conference on*. 2003. IEEE.
- Malik, F. and B. Baharudin, *Analysis of distance metrics in content-based image retrieval using statistical quantized histogram texture features in the DCT domain*. Journal of King Saud University-Computer and Information Sciences, 2013. **25**(2): p. 207-218.
- Wang, J.Z., *Wang dataset*.
- Bosch, A., A. Zisserman, and X. Muñoz, *Scene classification via pLSA*, in *Computer Vision-ECCV 2006*. 2006, Springer. p. 517-530.
- Jain, A.K. and A. Vailaya, *Image retrieval using color and shape*. Pattern Recognition, 1996. **29**(8): p. 1233-1244.
- Khotanzad, A. and Y.H. Hong, *Invariant image recognition by Zernike moments*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1990. **12**(5): p. 489-497.
- Kim, H.-K., et al. *A modified Zernike moment shape descriptor invariant to translation, rotation and scale for similarity-based image retrieval*. in *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*. IEEE.
- Singh, C., *Local and global features based image retrieval system using orthogonal radial moments*. Optics and Lasers in Engineering, 2012. **50**(5): p. 655-667.
- Kruskal, J.B., *Nonmetric multidimensional scaling: a numerical method*. Psychometrika, 1964. **29**(2): p. 115-129.
- Aryal, S., et al., *Data-dependent dissimilarity measure: an effective alternative to geometric distance measures*. Knowledge and Information Systems, 2017: p. 1-28.
- Saad, M.H., et al., *Image Retrieval Based on Integration Between YCbCr Color Histogram and Texture Feature*. International Journal of Computer Theory and Engineering, 2011. **3**(5): p. 701.
- Vailaya, A., et al., *Image classification for content-based indexing*. Image Processing, IEEE Transactions on, 2001. **10**(1): p. 117-130.
- Fan, J., et al. *Automatic image annotation by using concept-sensitive salient objects for image content representation*. in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM.
- Feng, S.L., R. Manmatha, and V. Lavrenko. *Multiple bernoulli relevance models for image and video annotation*. in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. IEEE.
- Yang, C., M. Dong, and F. Fotouhi. *Image content annotation using Bayesian framework and complement components analysis*. in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*. IEEE.
- Goh, K.-S., E.Y. Chang, and B. Li, *Using one-class and two-class SVMs for multiclass image annotation*. Knowledge and Data Engineering, IEEE Transactions on, 2005. **17**(10): p. 1333-1346.
- Jeong, S., C.S. Won, and R.M. Gray, *Image retrieval using color histograms generated by Gauss mixture vector*

quantization. *Computer Vision and Image Understanding*, 2004. **94**(1): p. 44-66.

local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 2002. **24**(7): p. 971-987.

23. Ojala, T., M. Pietikainen, and T. Maenpaa, *Multiresolution gray-scale and rotation invariant texture classification with*








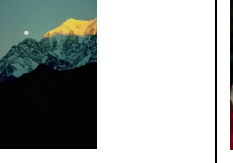
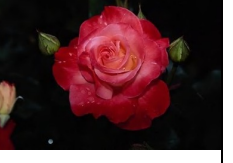




(a) using m_p














(b) using ED

Figure 4. Top 10 retrieval for Query 1.

					
Query	1	2	3	4	5
					
6	7	8	9	10	

(a) using m_p

					
Query	1	2	3	4	5
					
6	7	8	9	10	

(b) using ED

Figure 5. Top 10 retrieval for Query 2.

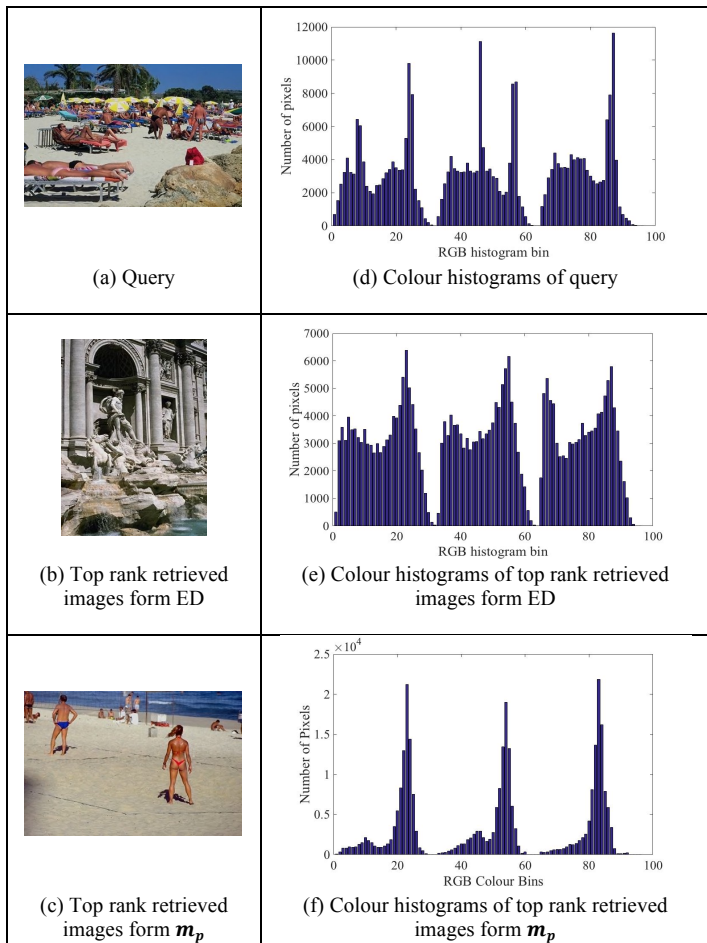


Figure 6. Colour histogram comparison of query 1

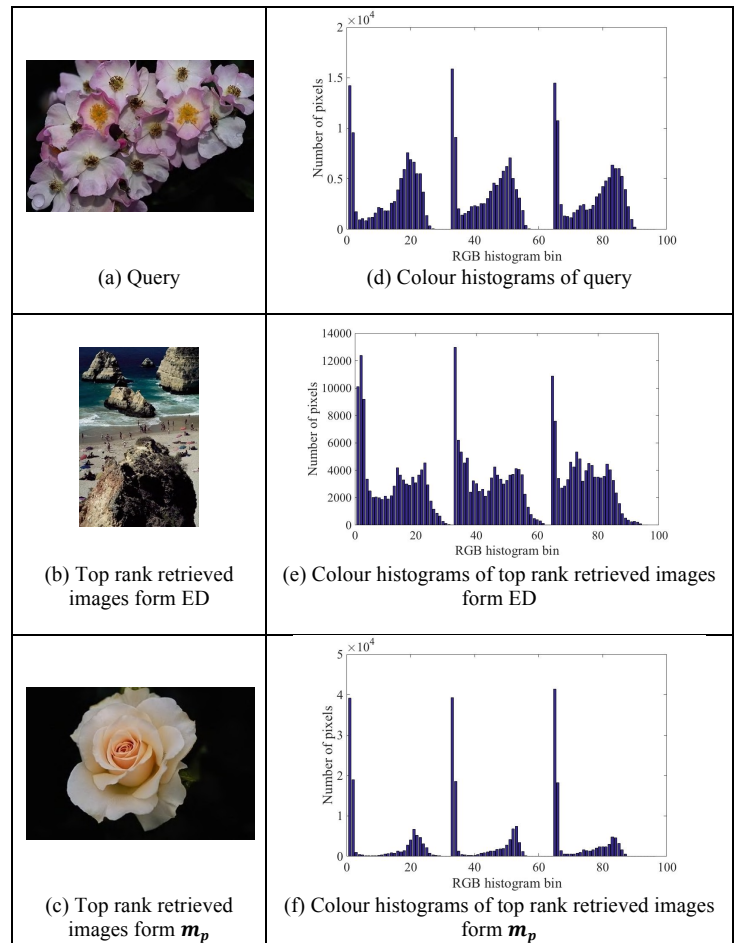


Figure 7. Colour histogram comparison of query 2

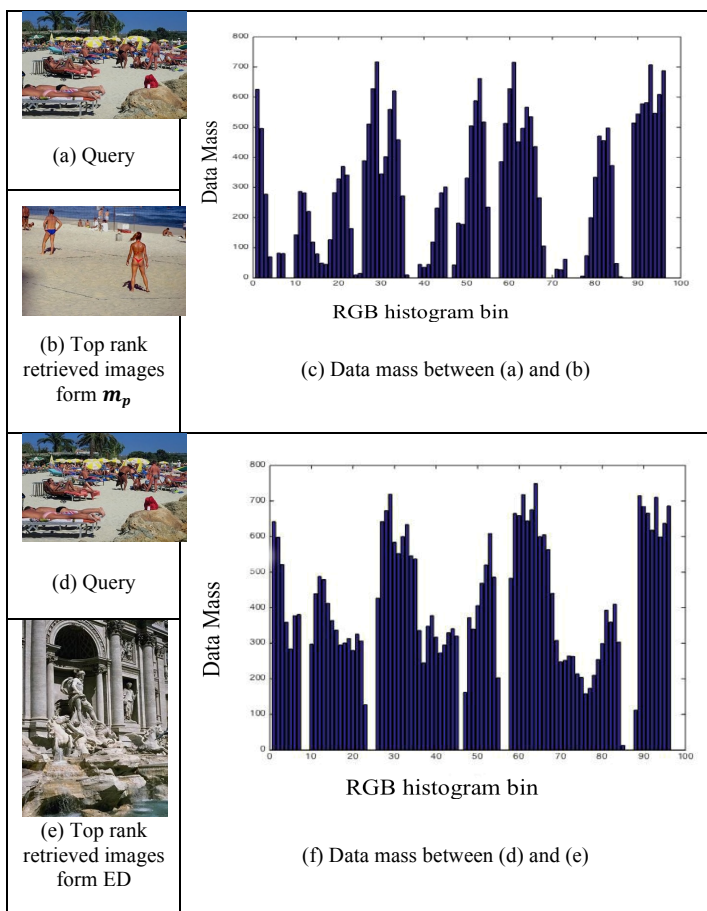


Figure 8. Data mass comparison of query 1

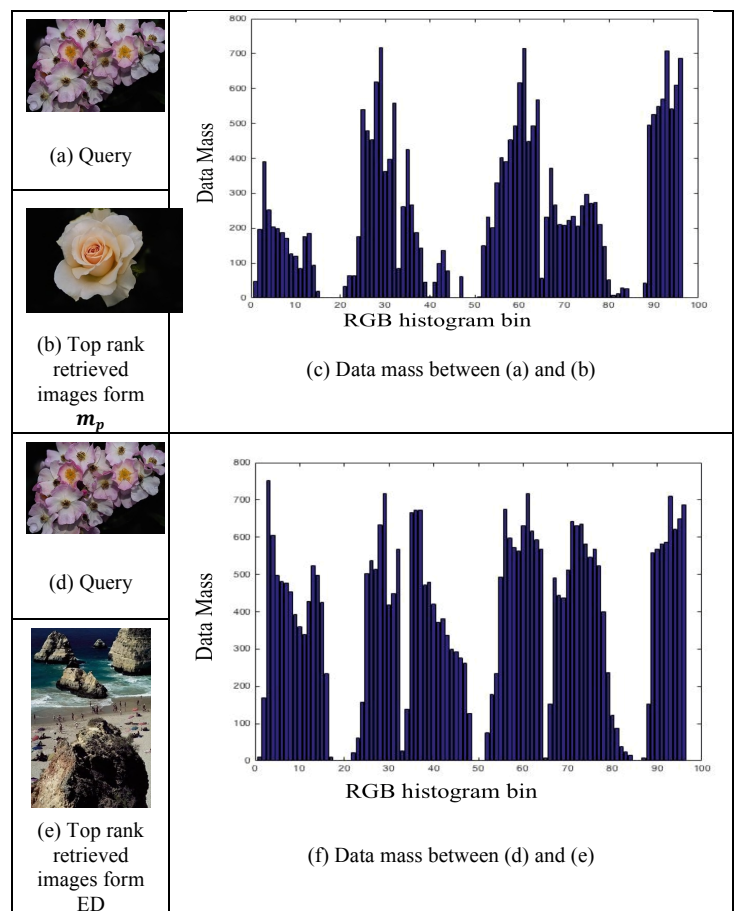







Figure 9. Data mass comparison of query 2

					
Query	1	2	3	4	5
					
6	7	8	9	10	

(a) using m_p

					
Query	1	2	3	4	5
					
6	7	8	9	10	

(b) using HDDD

Figure 12. Top 10 retrieval for Query 1.

				
Query	1	2	3	4
				
6	7	8	9	10

(a) using m_p

					
Query	1	2	3	4	5
					
6	7	8	9	10	

(b) using HDDD

Figure 13. Top 10 retrieval for Query 2