LB-Index: A Multi-Resolution Index Structure for Images

Vebjorn Ljosa Arnab Bhattacharya Ambuj K. Singh
University of California, Santa Barbara
{*ljosa,arnab,ambuj*}*@cs.ucsb.edu*

1 Introduction

In many domains, the similarity between two images depends on the spatial locations of their features. The earth mover's distance (EMD), first proposed by Werman et al. [8], measures such similarity. It yields higher-quality image retrieval results than the L_p-norm, quadratic-form distance, and Jeffrey divergence [6], and has also been used for similarity search on contours [3], melodies [7], and graphs [2].

Computing the EMD is an expensive linear-programming problem: It takes 41 s to compute the EMD between 12-dimensional features extracted from images partitioned into 8×12 tiles, so searching a database of 4,000 images can take 46 h.

In this paper, we redefine the EMD to work with multidimensional feature vectors, and show how the computation can be performed separately for each dimension. We then develop lower bounds that are reasonably tight and can be computed quickly. A multi-resolution indexing scheme based on either sequential scan or the M-tree [1] can answer similarity queries more than 500 times faster by combining the two techniques.

2 The Earth Mover's Distance

The following definition of the EMD between images extends Werman et al.'s [8] formulation for grayscale images, and applies to feature vectors extracted from image tiles. The image feature can be of any dimensionality; we show later that the distance can be computed independently for each dimension of the feature vector and added up to get the total distance. All feature values must be non-negative, but this is not an important restriction, as they can be made positive by adding the same large number to all feature values of all images. This will not affect the value of the EMD.

Suppose that the images A and B are composed of n tiles. For any two tiles $i \in A$ and $j \in B$, the *ground distance* c_{ij} is the spatial distance between them (normally the L_2-distance). Feature vectors are extracted from each tile. The feature vectors of A are $\{\vec{a}_0, \ldots, \vec{a}_{n-1}\}$, and those of B are $\{\vec{b}_0, \ldots, \vec{b}_{n-1}\}$. Each feature vector $\vec{a}_i$ or $\vec{b}_j$ is a column vector of d values. A weight vector $\vec{w} = [w_1 \ldots w_d]^{\mathrm{T}}$ assigns a weight to each dimension. Normally, $\vec{w} = [1 \ldots 1]^{\mathrm{T}}$, but a different $\vec{w}$ may be useful when several image features are concatenated into one vector.

The EMD is computed by finding a minimal-cost $n \times n$ *flow matrix* $\boldsymbol{F} = \{\vec{f}_{ij}\}$, where each $\vec{f}_{ij}$ is a *flow* of mass from tile i to tile j such that image A is transformed into image B. Note that each $\vec{f}_{ij}$ is a column vector of d elements.

The cost of moving mass $\vec{f}_{ij}$ from tile i to tile j is the ground distance from i to j multiplied by the mass to be moved, or $c_{ij}\vec{w}^{\mathrm{T}}\vec{f}_{ij}$. Here, the weight vector $\vec{w}$ is used to combine the d elements of $\vec{f}_{ij}$ into a scalar. The EMD can then be defined as

$$\min_{\boldsymbol{F}} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} c_{ij}\vec{w}^{\mathrm{T}}\vec{f}_{ij}$$

subject to $\vec{f}_{ij} \geq \vec{0}$, $\sum_{j=0}^{n-1} \vec{f}_{ij} = \vec{a}_i$, and $\sum_{i=0}^{n-1} \vec{f}_{ij} = \vec{b}_j$, element-wise and $\forall i, j \in \{0, \ldots, n-1\}$.

So far we have assumed that the images have the same total mass, i.e., $\sum_{i=0}^{n-1} \vec{a}_i = \sum_{j=0}^{n-1} \vec{b}_j$. In general, this is not true. For instance, when the image feature is intensity, a generally dark image will have a lower total mass than a generally light image. The images may be normalized such that the intensities add up to the same value [8], but this causes problems, as the distinction between a dark image and a light image disappears. Instead, we introduce a special "tile" called the *bank* to each image, and allow flows to and from it. The effect of these flows is to allow the total mass of one image to be increased in order to match the total mass of the other, but at a cost proportional to the increase. The bank tile has the same ground distance α (a parameter) to all the other tiles, and, of course, a ground distance of 0 to itself. The banks (n-th tiles) of the images A and B are initialized as $\vec{a}_n = \sum_{j=0}^{n-1} \vec{b}_j$ and $\vec{b}_n = \sum_{i=0}^{n-1} \vec{a}_i$. The EMD can now be restated to include flows to and from the banks:

$$\rho_{AB} = \min_{\boldsymbol{F}} \sum_{i=0}^{n} \sum_{j=0}^{n} c_{ij}\vec{w}^{\mathrm{T}}\vec{f}_{ij} \tag{1}$$

$$\text{subject to} \quad \vec{f}_{ij} \geq \vec{0}, \quad \sum_{j=0}^{n} \vec{f}_{ij} = \vec{a}_i, \quad \text{and} \quad \sum_{i=0}^{n} \vec{f}_{ij} = \vec{b}_j,$$

$$\text{element-wise and } \forall i, j \in \{0, \ldots, n\}.$$

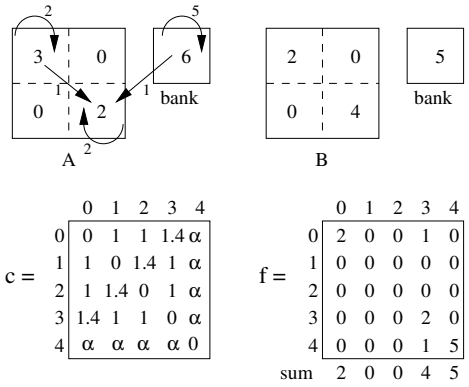

Figure 1. An example of EMD computation between two images A and B. The ground distance matrix $\vec{c}$ is shown on the left while the optimal flow matrix $\vec{F}$ is shown on the right. The flows are shown by arrows with the corresponding mass. The EMD is $1.4+\alpha$.

Notice that when $\alpha < 1/2$, the EMD is the same as the L_1 distance (scaled by 2α) because a flow from tile i to the bank and back to tile j is never more expensive than a flow directly from i to j. Figure 1 shows an example of EMD computation. The example assumes that $\alpha \geq 0.7$.

3 Decomposing the EMD

As defined in Eq. (1), the EMD is a large linear programming problem because the flows are vectors of the same dimensionality as the image features. However, there are no direct flows from one dimension to another ("crosstalk"), so the flows can be decomposed, and only one dimension considered at a time. Consequently, we can solve d smaller linear programming problems (where d is the dimensionality of the feature vector) and then combine their solutions. Theorem 1 states this formally.

Theorem 1 (decomposition). *The minimum cost when all dimensions of the feature vector are considered simultaneously is the same as the sum of minimum costs when each dimension of the feature vector is considered separately, i.e.,*

$$\rho_{AB} = \min_{\boldsymbol{F}} \sum_{i=0}^{n}\sum_{j=0}^{m} c_{ij}\sum_{k=1}^{d} w_k f_{ijk} = \sum_{k=1}^{d} \min_{\boldsymbol{F_k}} \sum_{i=0}^{n}\sum_{j=0}^{m} c_{ij} w_k f_{ijk}$$

Proof sketch. The constraints in Eq. (1) are all element-wise; so, they can be separated and solved independently, and then added up to get the actual solution. □

When the dimensions of the feature vectors are independent, the EMD formulation in Eq. (1) can be applied directly. Otherwise, principal component analysis (PCA) can be used to find their independent bases, and the EMD computed from them. Another approach is to cluster the dimensions so that there is no crosstalk between them, and then compute the EMD separately for each cluster. This is a natural solution, for instance, for protein localization images, where features are extracted for each protein independently.

4 Multi-Resolution Lower Bounds

Theorem 1 makes the EMD-computation's running time proportional to the dimensionality of the feature vector. The number of variables in the LP problem increases quadratically with the number of tiles, however, so the running time is still high when the number of tiles is large—which seems necessary for capturing the characteristics of some classes of images. For instance, we were able to increase the classification accuracies of confocal images of retinas from 90 % to 96 % by raising the number of tiles from 6 to 24. This, however, also increased the running time from 4 ms to 62 ms per computation. With 96 tiles, the accuracy was 98 %, but each computation took 2.9 s.

In this section, we show how, using a small number of tiles, we can compute a lower bound for the distance that would be computed using a large number of tiles. This allows us to combine the high speed of few tiles with the high accuracy of many tiles. This is crucial to indexing the EMD.

From any image A with $n = n_x \times n_y$ tiles (not including the bank), we can construct a coarser-grained image A' with $n' = n'_x \times n'_y$ tiles ($n'_x = n_x/2, n'_y = n_y/2$ and $n' = n/4$). Each tile i' of A' corresponds to the 4 tiles of A whose indices i satisfy the constraints

$$i_x = 2i'_x + p \qquad \text{and} \qquad i_y = 2i'_y + q \tag{2}$$

where $i' = i'_y n'_x + i'_x$, $i = i_y n_x + i_x$, and $p,q \in \{0,1\}$. The feature value $\bar{a}'_i$ of a tile i in A' is computed as the sum of the feature values of the 4 corresponding tiles in A:

$$a'_{i'} = \begin{cases} \sum_{p=0}^{1}\sum_{q=0}^{1} a_{\langle 2i'_x+p, 2i'_y+q\rangle} & \text{if } i' \neq n' \\ a_n & \text{if } i' = n' \end{cases} \tag{3}$$

Our lower bound for the EMD between two images A and B is the EMD between their summaries A' and B', but with one crucial modification: The ground distance c is replaced by another ground distance c'.

$$c'_{i'j'} = \begin{cases} \left[\max\{0, 2|i'_x - j'_x| - 1\}^2 + \max\{0, 2|i'_y - j'_y| - 1\}^2\right]^{\frac{1}{2}} & \\ \qquad \text{if } i', j' \neq n' & \\ 0 \qquad \text{if } i' = j' = n' & \\ \alpha \qquad \text{otherwise} & \end{cases} \tag{4}$$

The c'-distance between two coarse tiles i' and j' is never more than the c-distance between any fine tile corresponding to i' and any fine tile corresponding to j'. The following lemma states this formally. (Proof omitted.)

Lemma 1. *If i, j, i', and j' satisfy the constraints of Eq. (2), then $c'_{i'j'} \leq c_{ij}$.*

We can now solve the linear programming problem

$$\rho'_{AB} = \min_{\boldsymbol{F}'} \sum_{i'=0}^{n/2} \sum_{j'=0}^{n/2} c'_{i'j'} \vec{w}^{\mathrm{T}} \vec{f'_{i'j'}} \qquad (5)$$

$$\text{subject to} \quad \vec{f'_{i'j'}} \geq \vec{0}, \quad \sum_{j'=0}^{n/2} \vec{f'_{i'j'}} = \vec{a'_{i'}}, \text{ and } \sum_{i'=0}^{n/2} \vec{f'_{i'j'}} = \vec{b'_{j'}},$$

$$\text{element-wise and } \forall i', j' \in \{0, \ldots, n/2\}.$$

This is less computationally demanding because the number of variables is reduced by a factor of 16. The following theorem claims that ρ'_{AB} is a lower bound for ρ_{AB}. (Proof omitted because of space limitations.)

Theorem 2 (lower bound). *The distance ρ'_{AB}, defined in Eq.* (5)*, computed from the coarse images A' and B' using the modified ground distance c', is a lower bound for the EMD ρ_{AB}, defined in Eq.* (1)*.*

Although the lower bound is presented in terms of regular tiles, it can be formulated using arbitrary regions [4]. It can also be adapted to work with an alternative definition of EMD, used by Rubner et al. [6], where the image is clustered into regions of similar feature values and the mass is the number of pixels in each region [4]. Finally, the lower bound can be generalized to apply even when there is crosstalk, i.e., when there are flows directly from one dimension of one region to another dimension in another region. (The definition in Eq. (1) allows for such flows only indirectly, through the bank.)

Multi-resolution lower bounds. So far, a coarser summary of an image has been obtained by combining *n level-0* tiles into $n' = n/4$ *level-1* tiles. An even coarser summary can be obtained by repeating this process, combining the n' level-1 tiles into even fewer $n'' = n'/4$ *level-2* tiles, and so on. The ground distance between tiles at level i ($i > 0$) is the minimum pairwise distance between the corresponding tiles at level $(i-1)$. Multiple levels of lower bounds are key to building efficient index structures for computationally costly distances such as the EMD: Most objects can be pruned based on lower bounds computed from the higher-level summaries, and the time-consuming lower-level distances need only be computed for the remaining ones.

5 Experimental Results

We measured the effect our lower bounds (Theorem 2) had on range queries and *k*-nearest-neighbor queries on a database of 12-dimensional Color Layout Descriptors [5] extracted from 3932 retinal images, both using sequential scan and an M-tree [1]. The sequential-scan algorithms compute lower bounds for the distances to all objects. The M-tree algorithms are similar to Ciaccia et al.'s original search algorithms for the M-tree [1], but make conservative choices during the search using the lower bounds. Two levels of lower bounds were used in the experiments.

For range queries (with range equal to 3.7 % of the largest distance in the database, which returns 25 objects on average), the lower bounds resulted in a speedup of 36 compared to sequential scan without lower bounds. The lower bounds made *k*-NN queries ($k = 25$) 7 times faster.

Range search on the M-tree without using lower bounds is slower than sequential scan, except for small ranges, because exact distances are computed for each internal node searched. With lower bounds, the M-tree performs well, answering range queries 36 times faster than sequential scan *without* lower bounds, but not significantly faster than sequential scan *with* lower bounds.

The M-tree without lower bounds speeds up *k*-NN queries ($k = 25$) 2.2 times compared to sequential scan without lower bounds. Adding lower bounds increases this speedup to 5.4.

Theorem 1 (decomposition) can reduce the running time of EMD computations by factors of up to 14. This effect is orthogonal to the speedup from using lower bounds; combined with the speedup from Theorem 2, this can result in speedups of over 500.

A more thorough experimental evaluation of our proposed techniques is available [4].

Acknowledgements. We would like to thank Geoffrey P. Lewis from the laboratory of Steven K. Fisher at UCSB for providing the retinal micrographs used in the experiments and shown in the paper. This work was supported in part by grant no. ITR-0331697 from the National Science Foundation.

References

[1] P. Ciaccia, M. Patella, and P. Zezula. M-tree: An efficient access method for similarity search in metric spaces. In *Proc. VLDB*, pages 426–435, 1997.

[2] M. F. Demirci, A. Shokoufandeh, S. Dickinson, Y. Keselman, and L. Bretzner. Many-to-many feature matching using spherical coding of directed graphs. In *Proc. ECCV*, 2004.

[3] K. Grauman and T. Darrell. Fast contour matching using approximate earth mover's distance. In *Proc. CVPR*, 2004.

[4] V. Ljosa, A. Bhattacharya, and A. K. Singh. Indexing spatially sensitive distance measures using multi-resolution lower bounds. Technical Report 2005-16, Dept. of Computer Science, University of California, Santa Barbara, 2005.

[5] B. S. Manjunath, P. Salembier, and T. Sikora, editors. *Introduction to MPEG 7: Multimedia Content Description Language*. Wiley, 2002.

[6] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[7] R. Typke, R. Veltkamp, and F. Wiering. Searching notated polyphonic music using transportation distances. In *Proc. multimedia*, pages 128–135, 2004.

[8] M. Werman, S. Peleg, and A. Rosenfeld. A distance metric for multi-dimensional histograms. *Computer, Vision, Graphics, and Image Processing*, 32(3):328–336, Dec. 1985.