Fast and Robust Earth Mover’s Distances

Ofir Pele and Michael Werman
School of Computer Science and Engineering
The Hebrew University of Jerusalem
{ofirpele,werman}@cs.huji.ac.il

Abstract

We present a new family of Earth Mover’s Distances. We show that members of this family have many desirable properties. First, they correspond to the way humans perceive distances. Second, they are robust to outlier noise and quantization effects. Third, we prove that they are metrics. We also propose a fast algorithm and a linear time lower bound computation which allows fast filtering of large distances. The algorithm runs in an order of magnitude faster than the original algorithm, which makes it possible to compute the EMD on large histograms and databases. Finally, we show results for image retrieval.

1. Introduction

Histograms are ubiquitous tools in numerous computer vision tasks. It is common practice to use distances such as $L_2$ or $\chi^2$ for comparing histograms. This practice assumes that the histogram domains are aligned. However this assumption is violated through quantization, shape deformation, light changes, etc.

The Earth Mover’s Distance (EMD) [8] is a cross-bin distance that addresses this alignment problem. EMD is defined as the minimal cost that must be paid to transform one histogram into the other, where there is a “ground distance” between the basic features that are aggregated into the histogram. The EMD as defined by Rubner is a metric only for normalized histograms. However, recently Pele and Werman [6] suggested $\hat{EMD}$ and showed that it is a metric for all histograms.

A major issue that arises when using EMD is which ground distance to use for the basic features. This, of course, depends on the histograms, the task and practical considerations. In many cases we would like the distance to correspond to the way humans perceive distances (image retrieval, for example). In other cases we would like the distance to fit the distribution of the noise (keypoint matching, for example). Practical considerations include speed of computation, existence of easy to compute lower bounds (this allows fast filtering of large distances) and the metric property that enables fast algorithms for nearest neighbor searches.

We propose to use thresholded ground distances. i.e. distances that saturate to a constant value. These distances have many desirable properties. First, saturated distances correspond to the way humans perceive distances [11]. Second, many natural noise distributions have a heavy tail, i.e. outlier noise. Thresholded distances give different outliers the same large distance.

The usage of saturated distances with EMD is not new. However, usually the exponent function is used as it is a metric [8, 9]. We show that thresholded metrics are also metrics. In addition, we present a fast algorithm that computes an EMD with a thresholded ground distance.

Fast to compute lower bounds are also of importance as they allow fast filtering of large distances. Rubner et al. [8] have proposed the distance between the centers of mass of two normalized histograms as lower bound for EMD with a norm ground distance. We show a linear time lower bound for an EMD with a thresholded ground distance for any histograms.

The Earth Mover’s Distance was used successfully in many applications such as image retrieval [8], edge and corner detection [9], SIFT matching [4, 6], classification of texture and object categories [17], contour matching [2] and as a melodic similarity measure [13]. The major contribution of this paper is the fast algorithm for the computation of the EMD with thresholded ground distance. We argue that thresholded distances has all the benefits of the exponent function that is usually used as a saturated distance and also a big advantage of its much shorter computation time.

This paper is organized as follows. Section 2 is an overview of previous work. Section 3 describes the Earth Mover’s Distance. Section 4 discusses thresholded distances and proves that they are metrics. Section 5 describes the fast algorithm and the linear time lower bound computation. Section 6 presents the results. Finally, conclusions are drawn in Section 7.
2. Previous Work

Early work using cross-bin distances for histogram comparison can be found in \[10, 16, 15, 7\]. Shen and Wong \[10\] proposed to unfold two integer histograms, sort them and then compute the $L_1$ distance between the unfolded histograms. To compute the modulo matching distance between cyclic histograms they proposed taking the minimum from all cyclic permutations. This distance is equivalent to EMD between two normalized histograms. Werman et al. \[16\] showed that this distance is equal to the $L_1$ distance between the cumulative histograms. They also proved that matching two cyclic histograms by examining only cyclic permutations is in effect optimal. Werman et al. \[15\] proposed an $O(M \log M)$ algorithm for finding a minimal matching between two sets of $M$ points on a circle. The algorithm was adapted by Pele and Werman \[6\] to compute the EMD between two $N$-bin, normalized histograms with time complexity $O(N)$.

Pele et al. \[7\] suggested using the EMD for grayscale images and using linear programming to compute it. Rubner et al. \[8\] suggested using EMD for color and texture images and gave a definition of an EMD for non-normalized histograms. They computed the EMD using a specific linear programming algorithm - the transportation simplex. The algorithm worst case time complexity is exponential. Practical run time was shown to be super-cubic. Interior-point algorithms or Orlin’s algorithm \[5\] are both with time complexity of $O(N^3 \log N)$ and can also be used. All of these algorithms have high computational cost.

Indyk and Thaper \[3\] proposed approximating certain EMD by embedding it into an Euclidean space. Embedding time complexity is $O(N d \log \Delta)$, where $N$ is the feature set size, $d$ is the feature space dimension and $\Delta$ is the diameter of the union of the two feature sets. Shirdhonkar and Jacobs \[12\] proposed a linear time algorithm for approximating certain EMD for low dimensional histograms using the sum of absolute values of the weighted wavelet coefficients of the difference histogram.

Ling and Okada proposed EMD-$L_1$\[4\]; i.e. EMD with $L_1$ as the ground distance. To execute the EMD-$L_1$ computation, they propose a tree-based algorithm, Tree-EMD. Tree-EMD exploits the fact that a basic feasible solution of the simplex algorithm-based solver forms a spanning tree when the EMD-$L_1$ is modeled as a network flow optimization problem. The worst case time complexity is exponential. Empirically, they show that this new algorithm has an average time complexity $O(N^2)$.

Pele and Werman \[6\] proposed $\hat{E}MD$ - a new definition of the EMD for non-normalized histograms. They showed that unlike Rubner’s definition $\hat{E}MD$ is a metric also for non-normalized histograms. In addition they proposed a linear time algorithm for an EMD that gives a transportation cost of 0 to corresponding bins, 1 for two nearby bins and 2 for farther bins and for the extra mass. They showed that this distance can be used to improve SIFT matching.

3. The Earth Mover’s Distance

The Earth Mover’s Distance (EMD) \[8\] is defined as the minimal cost that must be paid to transform one histogram into the other, where there is a “ground distance” between the basic features that are aggregated into the histogram.

Given two histograms $P, Q$ the EMD as defined by Rubner et al. \[8\] is:

$$EMD(P, Q) = \min_{\{f_{ij}\}} \sum_{i,j} f_{ij}d_{ij} \quad s.t \quad \sum_{j} f_{ij} \leq P_i \quad \sum_{i} f_{ij} \leq Q_j \quad f_{ij} \geq 0$$

where $\{f_{ij}\}$ denotes the flows. Each $f_{ij}$ represents the amount transported from the $i$th supply to the $j$th demand. We call $d_{ij}$ the ground distance between bin $i$ and bin $j$ in the histograms.

Pele and Werman \[6\] suggested $\hat{E}MD$:

$$\hat{E}MD_{\alpha}(P, Q) = (\min_{\{f_{ij}\}} \sum_{i,j} f_{ij}d_{ij}) + \alpha \max_{i,j} \{d_{ij}\} \quad s.t \quad Eqs. 2, 3$$

Pele and Werman proved that $\hat{E}MD$ is a metric for any two histograms if the ground distance is a metric and $\alpha \geq 0.5$ \[6\].

4. Thresholded Distances

Thresholded distances are distances that saturate to a threshold; i.e. let $d(a,b)$ be a distance measure between two features - $a, b$. The thresholded distance with a threshold of $t$ is defined as:

$$d_t(a,b) = \min(d(a,b), t)$$

We now prove that if $d$ is a metric then $d_t$ is also a metric. Non-negativity and symmetry hold trivially, so we only need to prove that the triangle inequality holds.

$$d_t(a,b) + d_t(b,c) \geq d_t(a,c)$$

if $d$ is a metric.

We consider three cases:
1. 
\[
\begin{align*}
& \left( d_t(a, b) < t \right) \land \left( d_t(b, c) < t \right) \land \left( d_t(a, c) < t \right) \Rightarrow \\
& \left( d_t(a, b) = d(a, b) \right) \land \left( d_t(b, c) = d(b, c) \right) \land \left( d_t(a, c) = d(a, c) \right) \Rightarrow \\
& d_t(a, b) + d_t(b, c) \geq d_t(a, c)
\end{align*}
\]

2. 
\[
\begin{align*}
& \left( d_t(a, b) = t \right) \lor \left( d_t(b, c) = t \right) \Rightarrow \\
& d_t(a, b) + d_t(b, c) \geq t \geq d_t(a, c)
\end{align*}
\]

3. 
\[
d_t(a, c) = t \Rightarrow
\]
Assume negatively that: 
\[
\begin{align*}
& d_t(a, b) + d_t(b, c) < d_t(a, c) = t \Rightarrow \\
& \left( d_t(a, b) < t \right) \land \left( d_t(b, c) < t \right) \Rightarrow \\
& \left( d_t(a, b) = d(a, b) \right) \land \left( d_t(b, c) = d(b, c) \right) \Rightarrow \\
& d(a, b) + d(b, c) < t = d_t(a, c) \leq d(a, c) \Rightarrow \\
& d(a, b) + d(b, c) < d(a, c)
\end{align*}
\]

The last statement contradicts the triangle inequality of the metric \(d_t\).

As every case as covered by at least one of the above cases, we proved that \(d_t\) is a metric if \(d\) is a metric.

5. Fast Computation of Thresholded Earth Mover’s Distances

This section describes an algorithm that computes \(\overline{EMD}\) with a thresholded ground distance in a fast way and a linear time lower bound.

\(\overline{EMD}\) can be solved by a min-cost-flow algorithm. Our algorithm does a simple transformation to the flow network that reduces the number of edges. If \(N\) is the number of bins in the histogram, the flow network of \(\overline{EMD}\) has exactly \(N^2 + N\) edges (see (a) in Fig. 1). \(N^2\) edges connect all sources to all sinks. The extra \(N\) edges connects all sources to the sink that gets the difference between the total mass of the two histograms (we assume without the loss of generality that the source histogram total mass is greater or equal than the sink histogram total mass).

The transformation (see Fig. 1) first removes all edges with cost \(t\). Second, it adds a new transhipment vertex. Finally we connect all sources to this vertex with edges of cost \(t\) and connect the vertex to all sinks with edges of cost 0. Let \(K\) be the number of edges going out of each bin that has a cost different than the threshold \(t\). We removed \(N(N - K)\) edges and added \(2N\) edges. Thus the total number of edges is \(N^2 - (K - 2)N\). If \(K = O(N)\) the number of edges is \(O(N)\) as opposed to the original \(O(N^2)\). Note that the new flow network is no longer a transportation problem, but a transhipment problem [1].

Let \(K = O(N)\) the optimization problem can be solved [1, 5] with worst case time complexity of 
\[
O(\min((N^2 \log \log U \log (NC)), (N^2 \log U \sqrt{\log C}, (N^2 \log U \log N), (N^2 \log^2 N)))
\]

Algorithms with \(C\) term assumes integral cost coefficients that are bounded by \(C\). Algorithms with \(U\) assumes integral supply and demands that are bounded by \(U\).

![Figure 1: An example of the transformation on a flow network of an EMD with the ground distance: \(d(a, b) = \min(2, |a - b|)\).](image)

(a)

(b)

We now describe a linear time lower bound computation. If the ground distance is a metric, we start by saturating all zero-cost edges. Second, we greedily flow from each source using the edges which has the minimum cost to all its neighbors. It is easy to see that the constraints \(\sum f_{ij} \leq Q_j\) might be violated. Thus, the resulting cost of this flow is a lower bound of the \(\overline{EMD}\). As each source is connected to \(O(1)\) sinks, the computation time is linear.

6. Results

In this section we present results for image retrieval. However, please note that we do not claim that this is the right way to do image retrieval. Image retrieval is used here as an example of an application where the EMD was already used to show that thresholded distances give good results. The major contribution is the faster algorithm. We
show that using our algorithm the running time decrease in an order of magnitude.

We use a database that contains 800 landscape images from the COREL database, that were used also in Wang et al. [14]. All images are of size 256 × 384. We resized them to size 24 × 16 and converted them to L*a*b* space. In L*a*b* space small Euclidean distances correspond to perceptual distances. Then we selected 16 images as query (numbers 1, 51, 101, . . . , 751 from the landscapes images) and searched for the 5 most similar images. We used four different distances:

1. EMD-t is EMD where the ground distance for pixels \((x_1, y_1), (x_2, y_2)\) with colors \((L_{1*1}, a_{1*1}, b_{1*1}), (L_{2*2}, a_{2*2}, b_{2*2})\) respectively is:
   \[
   d = \min(||(L_{1*1}, a_{1*1}, b_{1*1}) - (L_{2*2}, a_{2*2}, b_{2*2})||_2 + ||(x_1, y_1) - (x_2, y_2)||_2).
   \]

2. EMD-e is EMD where the ground distance for pixels \((x_1, y_1), (x_2, y_2)\) with colors \((L_{1*1}, a_{1*1}, b_{1*1}), (L_{2*2}, a_{2*2}, b_{2*2})\) respectively is:
   \[
   d = ||(L_{1*1}, a_{1*1}, b_{1*1}) - (L_{2*2}, a_{2*2}, b_{2*2})||_2 + ||(x_1, y_1) - (x_2, y_2)||_2.
   \]

3. \(\sum L_{2-t}\) is \(\sum_{(x_1, y_1)=\{x_2, y_2\}} \min(||(L_{1*1}, a_{1*1}, b_{1*1}) - (L_{2*2}, a_{2*2}, b_{2*2})||_2, 10)\)

4. \(\sum L_{2}\) is \(\sum_{(x_1, y_1)=\{x_2, y_2\}} ||(L_{1*1}, a_{1*1}, b_{1*1}) - (L_{2*2}, a_{2*2}, b_{2*2})||_2\)

The results are given in figures 2, 3, 4, 5, 6, 7, 8, 9 and should be viewed in color. Each figure shows two queries and the 4 most similar images in the database (excluding the query itself) using each of the 4 distances described above. Above each image is the distance of the returned image to the query.

The results show that EMD-t; i.e. with a thresholded distances returns the greatest number of perceptually similar images. Second to it is EMD-e followed by \(\sum L_{2}\) and \(\sum L_{2-t}\).

In Fig. 2 top and bottom, both EMD-t and EMD-e performs good, where \(\sum L_{2}\) and \(\sum L_{2-t}\) performance is poor. In Fig. 3 top all results are poor. In Fig. 3 bottom EMD-t performs best. In Fig. 4 top again EMD-t performs best while in the bottom EMD-e performs best. However, EMD-t performance is also very good. In Fig. 5 top EMD-t performs best. In Fig. 5 bottom and Fig. 6 all performs equally good. In Fig. 6 bottom \(\sum L_{2}\) performs best. In Fig. 7 bottom and top all results are good, although \(\sum L_{2-t}\) and EMD-e each does one mistake. In Fig. 9 top all performance is poor. In the bottom EMD-e performs best, EMD-t also has a good performance with only one mistake, while both \(\sum L_{2}\) and \(\sum L_{2-t}\) performance is very poor.

It should also be noted that as EMD-e ground distance is not thresholded, its computation is in order of magnitude slower. That is, the run time of computing 800 EMD-t computations is a minute, while for EMD-e it is an hour.

The algorithm we use for the computation of the EMD is successive shortest path [1]. This algorithm has a worst time complexity \(O(N^2 \log N)\). A comparison of the practical running time on a Pentium 3GHz of our algorithm and Rubner’s implementation of the simplex algorithm is given in table 1. Our algorithm is in order of magnitude faster.

### 7. Conclusions

We presented a new family of Earth Mover’s Distances. Members of this family correspond to the way humans perceive distances, robust to outlier noise and quantization effects. We proved that they are metrics. We also propose a fast algorithm and a linear time lower bound computation which allows fast filtering of large distances. The algorithm runs in an order of magnitude faster than the original algorithm, which makes it possible to compute the EMD on large histograms and databases.

We are working on a faster implementation of the algorithm and testing the proposed lower bound. In addition, we want to test the new Earth Mover’s Distance in applications where it was not possible to use it before because of the long running time.

### References


<table>
<thead>
<tr>
<th>Image size</th>
<th>10 × 10</th>
<th>20 × 20</th>
<th>30 × 30</th>
<th>40 × 40</th>
<th>50 × 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our’s algorithm</td>
<td>0.005</td>
<td>0.09</td>
<td>0.43</td>
<td>1.6</td>
<td>4</td>
</tr>
<tr>
<td>Rubner’s algorithm</td>
<td>0.023</td>
<td>2.23</td>
<td>30.4</td>
<td>239.7</td>
<td>1084</td>
</tr>
</tbody>
</table>

Table 1. Run time results in seconds of a computation of EMD on greyscale images of various sizes.
Figure 2.
Figure 3.
<table>
<thead>
<tr>
<th>EMD-t</th>
<th>EMD-e</th>
<th>( \sum L_2-t )</th>
<th>( \sum L_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image 1" /></td>
<td><img src="image1.png" alt="Image 1" /></td>
<td><img src="image1.png" alt="Image 1" /></td>
<td><img src="image1.png" alt="Image 1" /></td>
</tr>
<tr>
<td><img src="image2.png" alt="Image 2" /></td>
<td><img src="image2.png" alt="Image 2" /></td>
<td><img src="image2.png" alt="Image 2" /></td>
<td><img src="image2.png" alt="Image 2" /></td>
</tr>
<tr>
<td><img src="image3.png" alt="Image 3" /></td>
<td><img src="image3.png" alt="Image 3" /></td>
<td><img src="image3.png" alt="Image 3" /></td>
<td><img src="image3.png" alt="Image 3" /></td>
</tr>
<tr>
<td><img src="image4.png" alt="Image 4" /></td>
<td><img src="image4.png" alt="Image 4" /></td>
<td><img src="image4.png" alt="Image 4" /></td>
<td><img src="image4.png" alt="Image 4" /></td>
</tr>
</tbody>
</table>

Figure 4.
Figure 5.
Figure 6.
Figure 7.
Figure 8.
<table>
<thead>
<tr>
<th>Query</th>
<th>EMD-t</th>
<th>EMD-e</th>
<th>$\sum L_2$-t</th>
<th>$\sum L_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Query Image 1" /></td>
<td><img src="image1" alt="EMD-t Image 1" /></td>
<td><img src="image1" alt="EMD-e Image 1" /></td>
<td><img src="image1" alt="Partial Image 1" /></td>
<td><img src="image1" alt="Partial Image 1" /></td>
</tr>
<tr>
<td><img src="image2" alt="EMD-t Image 2" /></td>
<td><img src="image2" alt="EMD-e Image 2" /></td>
<td><img src="image2" alt="Partial Image 2" /></td>
<td><img src="image2" alt="Partial Image 2" /></td>
<td></td>
</tr>
<tr>
<td><img src="image3" alt="EMD-t Image 3" /></td>
<td><img src="image3" alt="EMD-e Image 3" /></td>
<td><img src="image3" alt="Partial Image 3" /></td>
<td><img src="image3" alt="Partial Image 3" /></td>
<td></td>
</tr>
<tr>
<td><img src="image4" alt="EMD-t Image 4" /></td>
<td><img src="image4" alt="EMD-e Image 4" /></td>
<td><img src="image4" alt="Partial Image 4" /></td>
<td><img src="image4" alt="Partial Image 4" /></td>
<td></td>
</tr>
</tbody>
</table>

Figure 9.


