Computing the Earth Mover's Distance under Transformations

Scott Cohen
scohen@cs.stanford.edu
Stanford Vision Laboratory

The ideas and results contained in this document are part of my thesis, which will be published as a Stanford computer science technical report in June 1999. Similar ideas applied to the EMD under translation have already been published in the technical report [1].

Table of Contents

The Earth Mover's Distance (EMD)

We begin some intuition, examples, and informal definitions.

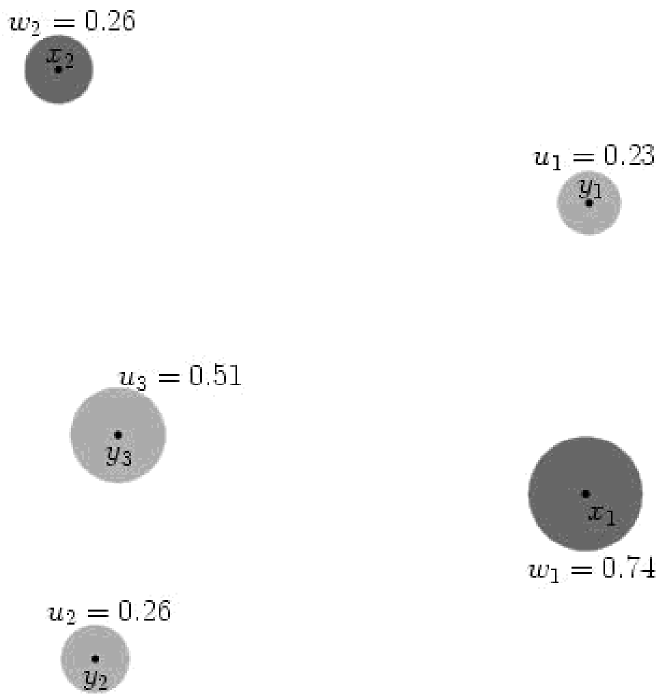
Intuition and Examples

The Earth Mover's Distance (EMD) is a distance measure between discrete, finite distributions

$$\mathbf{x} = \{ (x_1,w_1), (x_2,w_2), ..., (x_m,w_m) \} \quad \text{and}$$

$$\mathbf{y} = \{ (y_1,u_1), (y_2,u_2), ..., (y_n,u_n) \}.$$

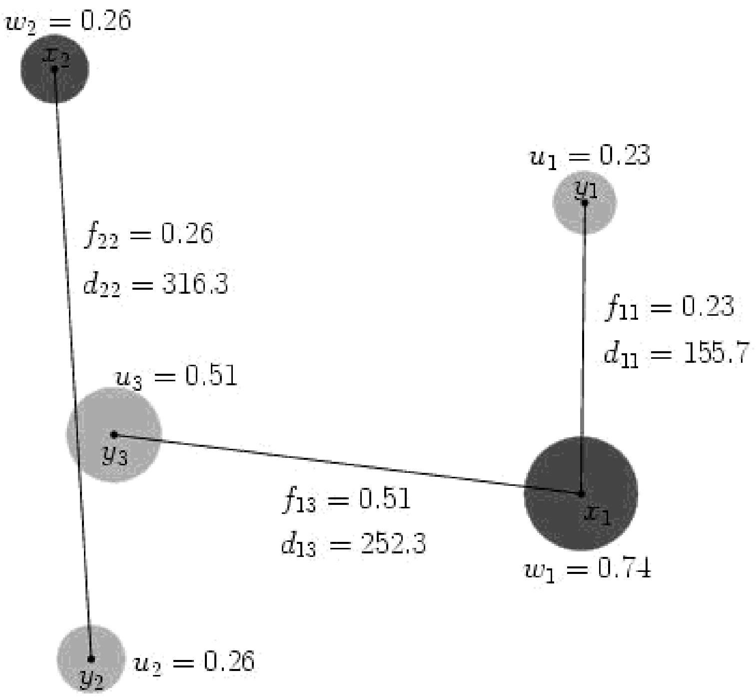
The **x** distribution has an amount of mass or weight w_i at position x_i in $\mathbf{R}^K$, i=1,...,m, while the **y** distribution has weight u_j at position y_j, j=1,...,n. An example pair of distributions in $\mathbf{R}^2$ is shown below.



Here **x** has m=2 masses and **y** has n=3 masses. The circle centers are the points (mass locations) of the distributions. The area of a circle is proportional to the weight at its center point. The total weight of **x** is $w^S=\mathrm{sum}_{i=1..m}\ w_i$, and the total weight of **y** is $u^S=\mathrm{sum}_{j=1..n}\ u_j$. In the example above, the distributions have equal total weight $w^S=u^S=1$.

Equal-Weight Distributions

Although the EMD does not require equal-weight distributions, let us begin our discussion with this assumption. The EMD between two equal-weight distributions is proportional to the amount of work needed to morph one distribution into the other. We morph **x** into **y** by transporting mass from the **x** mass locations to the **y** mass locations until **x** has been rearranged to look exactly like **y**. An example morph is shown below.

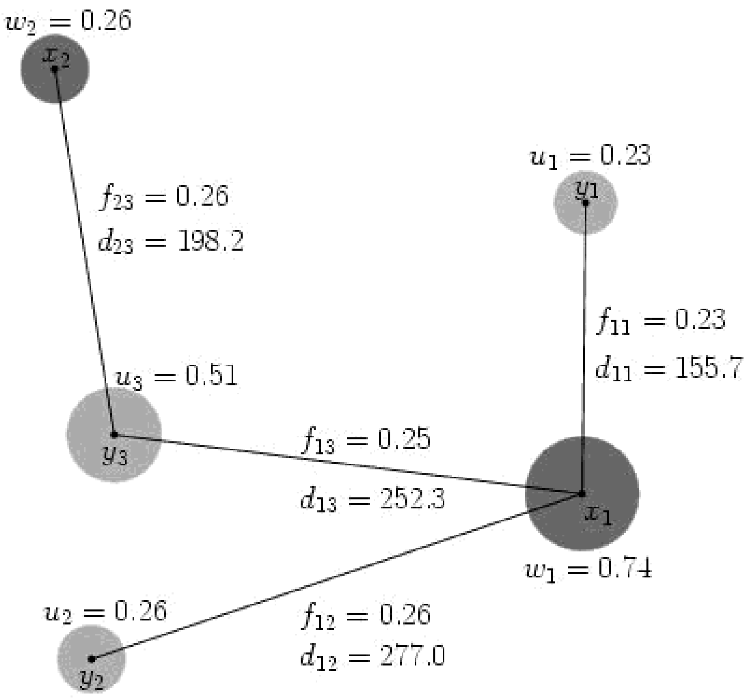


The amount of mass transported from x_i to y_j is denoted f_{ij}, and is called a *flow* between x_i and y_j. The work done to transport an amount of mass f_{ij} from x_i to y_j is the product of the f_{ij} and the distance $d_{ij}=d(x_i,y_j)$ between x_i and y_j. The total amount of work to morph **x** into **y** by the flow $F=(f_{ij})$ is the sum of the individual works:

$$\mathrm{WORK}(F,\mathbf{x},\mathbf{y}) = \mathrm{sum}_{i=1..m,\ j=1..n}\ f_{ij}\ d(x_i,y_j).$$

In the above flow example, the total amount of work done is 0.23*155.7 + 0.51*252.3 + 0.26*316.3 = 246.7.

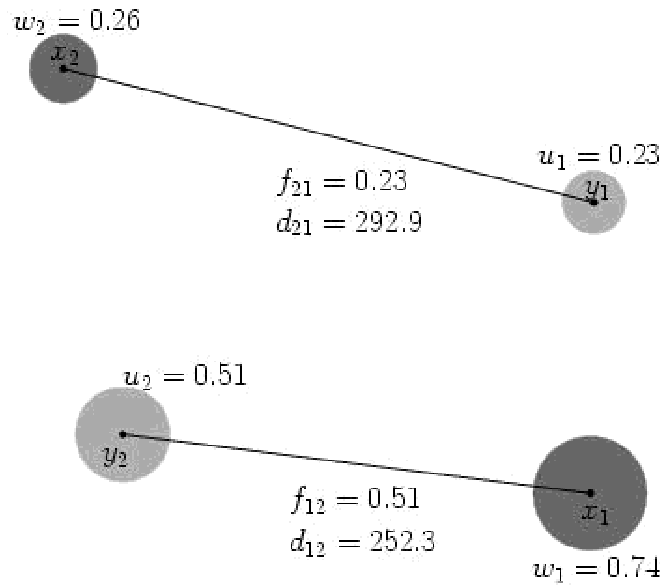
The flow in the previous example is not an optimal flow between the given distributions. A minimum work flow is shown below.



The total amount of work done by this flow is 0.23*155.7 + 0.26*277.0 + 0.25*252.3 + 0.26*198.2 = 222.4. The EMD between equal-weight distributions is the minimum work to morph one into the other, divided by the total weight of the distributions. The normalization by the total weight makes the EMD equal to the average distance travelled by mass during an optimal (i.e. work minimizing) flow. The EMD does not change if all the weights in both distributions are scaled by the same factor. In the previous example, the total weight is 1, so the EMD is equal to the minimum amount of work: EMD(**x**,**y**)=222.4.

Unequal-Weight Distributions

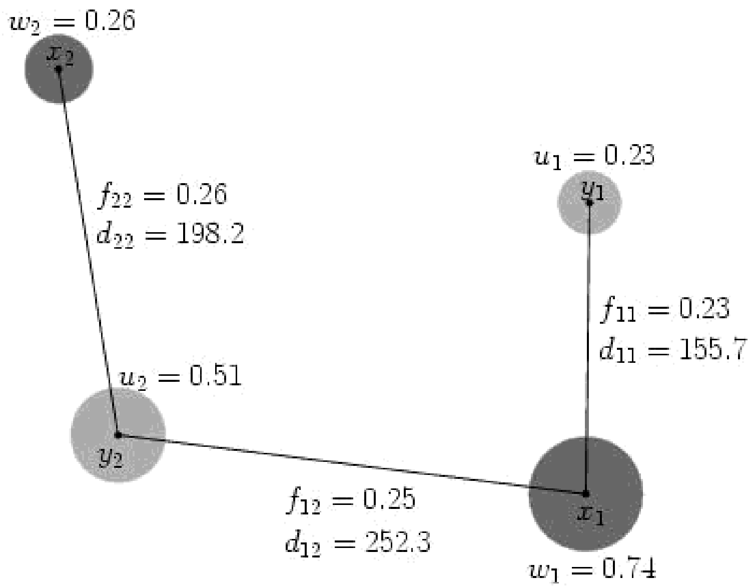
When the distributions do not have equal total weights, it is not possible to rearrange the mass in one so that it exactly matches the other. The EMD between unequal-weight distributions is proportional to the minimum amount of work needed to cover the mass in the lighter distribution by mass from the heavier distribution. An example of a flow between unequal-weight distributions is given below.



Here w^S=1 and u^S=0.74, so **x** is heavier than **y**. The flow covers or matches all the mass in **y** with mass from **x**. The total amount of work to cover **y** by this flow is 0.51*252.3 + .23*292.9 = 196.0. In this example, 0.23 of the mass at x_1 and 0.03 of the mass at x_2 is not used in matching all the **y** mass.

In general, some of the mass (w^S-u^S if **x** is heavier than **y**) in the heavier distribution is not needed to match all the mass in the lighter distribution. For this reason, the case of matching unequal-weight distributions is called the *partial matching* case. The case when the distributions are equal-weight is called the *complete matching* case because all the mass in one distribution is matched to all the mass in the other distribution. Another way to visualize matching distributions is to imagine piles of dirt and holes in the ground. The dirt piles are located at the points in the heavier distribution, and the the holes are located at the points of the lighter distribution. The volume of a dirt pile or the volume of dirt missing from a hole is equal to the weight of its point. Matching distributions means filling all the holes with dirt. In the partial matching case, there will be some dirt leftover after all the holes are filled.

The flow between the unequal-weight distributions in the previous example is not optimal. A minimum work flow is shown below.



The total amount of work for this flow to cover **y** is 0.23*155.7 + 0.25*252.3 + 0.26*198.2 = 150.4. The EMD is the minimum amount of work to cover the mass in the lighter distribution by mass from the heavier distribution, divided by the weight of the lighter distribution (which is the total amount of mass moved). As in the complete matching case, the normalization of the minimum work makes the EMD equal to the average distance mass travels during an optimal flow. Again, scaling the weights in both distributions by a constant factor does not change the EMD. In the above example, the weight of the lighter distribution is u^S=0.74, so EMD(**x**,**y**)= 150.4/0.74 = 203.3.

Formal Definition

Although we have described and informally defined the EMD for the two separate cases of (i) equal-weight distributions, and (ii) unequal-weight distributions, only one linear program is needed to formally define the EMD. Recall the notion for the input distributions:

$$\mathbf{x} = \{ (x_1,w_1), (x_2,w_2), ..., (x_m,w_m) \} \text{ and}$$
$$\mathbf{y} = \{ (y_1,u_1), (y_2,u_2), ..., (y_n,u_n) \},$$

w^S is the total weight of **x**, and u^S is the total weight of **y**. The EMD is defined by the linear program

$$\begin{aligned} \mathrm{EMD}(\mathbf{x},\mathbf{y}) &= \min_{F \text{ in } \mathcal{F}(\mathbf{x},\mathbf{y})} \quad \mathrm{WORK}(F,\mathbf{x},\mathbf{y}) \quad / \ \min(w^S,u^S) \\ &= \min_{F=(f_{ij}) \text{ in } \mathcal{F}(\mathbf{x},\mathbf{y})} \ \mathrm{sum}_{i=1..m,\ j=1..n}\ f_{ij}\ d(x_i,y_j) \ / \ \min(w^S,u^S), \end{aligned}$$

where the minimum is taken over the set of feasible flows $\mathcal{F}(\mathbf{x},\mathbf{y})$ defined by

$$(1) \qquad f_{ij} >= 0 \qquad i=1..m, j=1..n$$

$$
\begin{array}{llll}
(2) & \mathrm{sum}_{j=1..n}\, f_{ij} & <= & w_i \qquad i=1..m \\
(3) & \mathrm{sum}_{i=1..m}\, f_{ij} & <= & u_j \qquad j=1..n \\
(4) & \mathrm{sum}_{i=1..m,\, j=1..n}\, f_{ij} & = & \min(w^S,u^S).
\end{array}
$$

Here the flow variable f_{ij} represents the amount of mass matched between x_i and y_j. Constraint (1) forces this amount to be nonnegative. Constraint (2) requires that the total amount of **y** mass matched to **x** mass at x_i does not exceed the amount w_i of **x** mass at this location. Similarly, constraint (3) requires that the total amount of **x** mass matched to **y** mass at y_j does not exceed the amount u_j of **y** mass at this location. Finally, constraint (4) forces the total amount of mass matched to be the amount of mass in the lighter distribution. Without this constraint, the minimum would be achieved by setting $f_{ij}=0$ for all i,j.

The distance function d between points is called the *ground distance*. In contrast, the EMD is a distance between distributions which is built on top of this point distance function. The total amount of mass moved during any feasible flow (i.e. a flow satisfying (1), (2), (3), and (4)) is equal to the amount of mass $\min(w^S,u^S)$ in the lighter distribution. The EMD is equal to the average distance travelled (in ground distance units) by mass during an optimal flow. A simple units analysis shows that EMD units are the same as ground distance units.

When **x** and **y** are equal weight distributions, then all the mass in **x** must be matched to mass in **y**, and vice-versa. In this case, the previously given definition for the set of feasible flows *F*(**x**,**y**) can be rewritten as

$$
\begin{array}{lllll}
(1) & f_{ij} & >= & 0 & i=1..m,\, j=1..n \\
(5) & \mathrm{sum}_{j=1..n}\, f_{ij} & = & w_i & i=1..m \\
(6) & \mathrm{sum}_{i=1..m}\, f_{ij} & = & u_j & j=1..n.
\end{array}
$$

From (5) and (6), it follows that the total amount of mass matched in the equal-weight case is $\mathrm{sum}_{i=1..m,\, j=1..n}\, f_{ij} = w^S=u^S(=\min(w^S,u^S))$.

The linear program (LP) which defines the EMD is a special type of LP known as the *transportation problem*. The transportation simplex algorithm is a specialization of the simplex algorithm for solving LPs that takes advantage of the structure present in transportation problems ([3]). Y. Rubner has written a C code package to compute the EMD using the transportation simplex algorithm.

Some Notes

An excellent source for applications of the EMD in content-based image retrieval is the work of Y. Rubner. He has successfully applied the EMD to measure global color similarity between images ([5,6,7]) and texture similarity ([4,5,7]). In these cases, the feature space in which the distribution points live and the ground distance used to compare the points are different, but the same EMD framework is applied.

The EMD is a metric between equal-weight distributions whenever the underlying ground distance is a metric between points. The difficult part in the proof of this statement is to show that the triangle inequality holds. Assume that distributions have total weight one, so that the EMD is exactly the minimum of amount of work to change one distribution into another. One way to morph **x** into **z** is to use a minimum work morph of **x** into **y**, and then use a minimum work morph of **y** into **z**. The cost of the morph through **y** is EMD(**x**,**y**)+EMD(**y**,**z**). Since EMD(**x**,**z**) is the minimum amount of work required to change **x** into **z**, it cannot be more than the work for the particular morph through **y**. A formal proof of the metric property can be found in [5].

Another interesting property of the EMD is that the ground distance between the centroids of equal-weight distributions is a lower bound on their EMD for any ground distance between points induced by a vector norm (e.g. the Euclidean distance between points is the Euclidean norm of their difference). There are also lower bounds for the equal-weight case which use the EMD between projections of the distributions onto lines through the origin. The EMD between equal-weight 1D distributions can be computed much more efficiently than the EMD between more general distributions. Some information on the EMD in 1D can be found in this document, in the section on the EMD under translation in 1D. The technical report [1] covers the previously mentioned lower bounds, as well as centroid-based and projection-based lower bounds which can be applied in the partial matching case.

The EMD is more general than the way I have described it thus far. There is actually nothing in its formulation that assumes the x_i's and the y_j's in distributions **x** and **y** are points. The defining linear program uses only the distance $d(x_i,y_j)$, so x_i and y_j could be any objects between which a distance can be computed. It is not necessary that these objects have a point representation, or that the ground distance is some standard point distance such as L_2 or L_1. In general, the EMD is a distance measure between two sets of weighted objects which is built upon a distance between individual objects.

The Earth Mover's Distance Under Transformations

We begin with a statement of the problem and the basic motivation for its study.

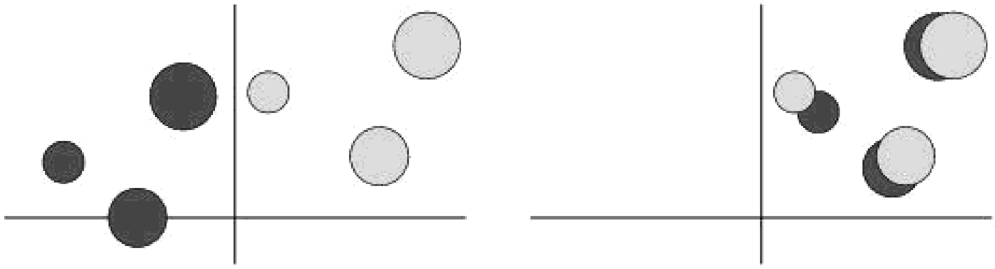
The Problem

The Earth Mover's Distance under a transformation set *G* is defined as

$$\mathrm{EMD}_G(\mathbf{x},\mathbf{y}) = \min_{g \text{ in } G} \mathrm{EMD}(\mathbf{x},g(\mathbf{y})).$$

In words, we want to find the transformation of one distribution that minimizes its EMD to another distribution.

The general motivation for allowing transformations is to have a distance function which measures similarity modulo some factors. Consider the example below in which the two distributions have equal total weight.



The EMD between the distributions on the left is large because mass in the darker distributions must be transported large distances to change the darker distribution into the lighter distribution (lighter in terms of gray level, not total weight). The two distributions, however, appear visually similar. If we allow the darker distribution to be translated before comparing it to the lighter distribution, then the EMD is small. This is clearly shown in the right half of the above figure. We shall give more motivation for allowing transformations in the next section.

One class of distribution transformations contains transformations g that change a distribution's weights, but leave its points fixed:

$$g(\mathbf{y}) = g(Y,u) = (Y,g(u)).$$

Here $\mathbf{y}$=(Y,u) splits the distribution into its points contained in the matrix Y, and its weights contained in the vector u. A useful transformation set of this class arises in the scale estimation phase of the SEDL image retrieval system. In this setting, transformations are parametrized by a scale factor c, and

$$g_c(\mathbf{y}) = g_c(Y,u) = (Y,cu).$$

Here, the distributions **x** and **y** are the color distributions for a database image and a query pattern, respectively, and the goal is to compute an estimate for the scale (size) at which the pattern may occur within the database image. See the description of the SEDL scale estimation phase for more details.

Another class of distribution transformations consists of transformations g that change the points of a distribution, but leave its weights fixed:

$$g(\mathbf{y}) = g(Y,u) = (g(Y),u).$$

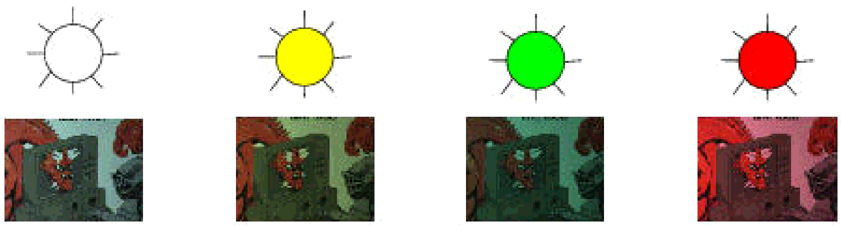
For example, suppose *G*=*E*, the group of Euclidean transformations. Distributions in *E* are defined by a rotation matrix R and a translation vector t, and

$$g_{R,t}(Y) = g_{R,t}([y_1 \ldots y_n]) = [Ry_1+t \ldots Ry_n+t].$$

In words, $g_{R,t}$ rotates each distribution point by R, and then translates the result by t. For the case when transformations change only distribution points, we shall give an iteration that is guaranteed to converge, although it may converge to only a locally optimal transformation. Before we do this, however, let us give some more motivation for allowing transformations.

Motivation from Content-Based Image Retrieval

Suppose we are comparing scenes using the EMD between color distributions. One problem is that the EMD can be large between color distributions for two images of the same scene taken under different illuminants, even if the camera location and orientation is fixed. This is because the pixels colors in the images can be quite different, as illustrated in the images below.

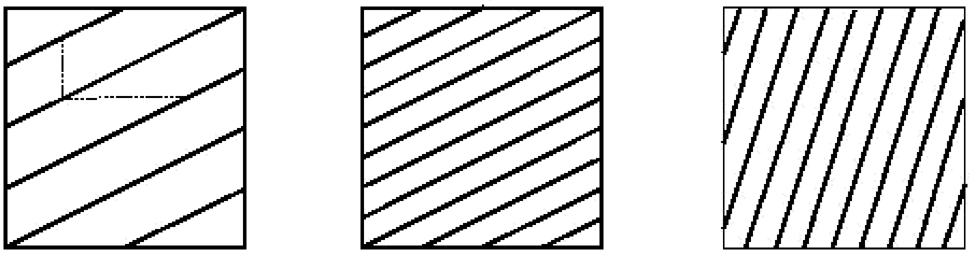


Under certain assumptions on the reflectance functions of scene objects, a change in the spectral power distribution (SPD) from a(w) to b(w) causes a linear transformation $A_{a,b}$ of image pixel colors ([2]):

lighting SPD a(w) --> b(w) ===> $[R\ G\ B]^T \rightarrow A_{a,b}\,[R\ G\ B]^T$.

By allowing for a linear tranformation in the comparison between color distributions, the EMD can show, for example, that the scene under the white illuminant is similar to the scene under the red illuminant.

Texture comparison is another example in which allowing transformations is useful. Excellent results using the EMD to compare textures were obtained by Y. Rubner ([4]). The main idea is to summarize a texture by a distribution of energy in the spatial frequency plane. A distribution point x_i is a point in the spatial frequency plane, and its weight w_i is the fraction of the total energy at that frequency. The textures shown below contain energy at only one spatial frequency, but this will be enough to make our point clear.



$q = (f_x, f_y)$ $\qquad$ $(1/c)(f_x, f_y)$ $\qquad$ $(1/c)R_\theta(f_x, f_y)$

Suppose we want the EMD to be small between the energy distributions for the left and right textures because these differ only by a scaling and rotation. As shown above, let $q=(f_x,f_y)$ denote a point in spatial frequency space. If we work in log-polar spatial frequency space, recording

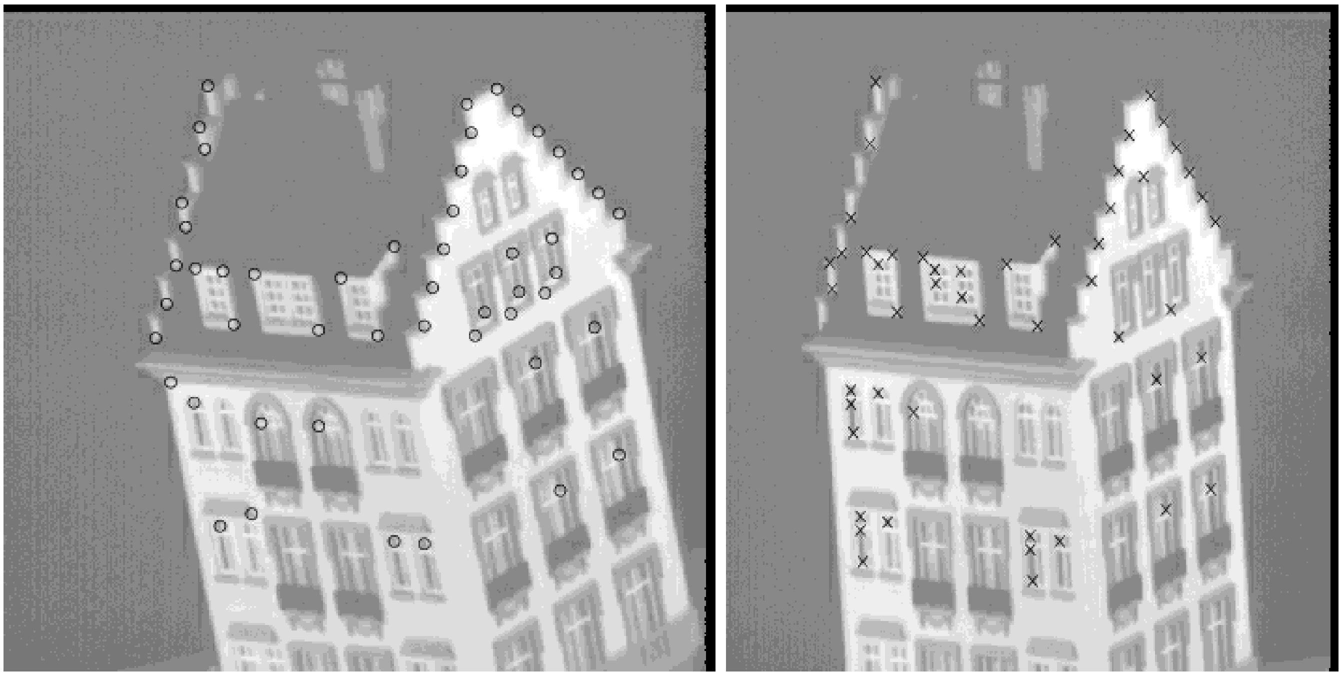
$$p=(\log \|q\|,\mathrm{angle}(q)),$$

then scaling and rotating the texture results in a translation of the point p:

scale texture by c, rotate by theta ===> p --> p + (log(1/c),theta).

By allowing for a translation in log-polar spatial frequency space, the EMD captures the similarity between textures that differ primarily in scale and orientation.

The need for transformations might be more direct than in the previous two applications, in the sense that distribution points may be points in the image plane instead of points in a color space or a spatial frequency space. Suppose for example, that we wish to match features in a stereo pair of images as shown below.



The change in location of a feature point can be modelled (approximately) by an affine transformation

$$p_l = A\, p_r + t$$

if the thickness of the object is small in comparison to its distance from the camera center of projection.

A Convergent Iteration

In this section, we present an iteration to search for an optimal transformation in the case that transformations change the points in a distribution, but leave the weights fixed. Since it is the weights in the distributions which determine the set of feasible flows,

$$F(\mathbf{x},g(\mathbf{y})) = F(\mathbf{x},\mathbf{y}) \quad \text{if g does not change the weights in } \mathbf{x} \text{ and } \mathbf{y}.$$

This fact is clear from examining the transportation problem which defines the EMD. Thus in the description below, the set of feasible flows between **x** and a transformed version of **y** is the same for every transformation.

The idea is to find the best flow for a fixed transformation, the best transformation for the previously computed flow, the best flow for the previously computed transformation, and so on:

$$(1) \qquad F^{(k)} = \arg \min_{F \text{ in } F(\mathbf{x},\mathbf{y})} \text{WORK}(F,\mathbf{x},g^{(k)}(\mathbf{y})), \text{ and}$$

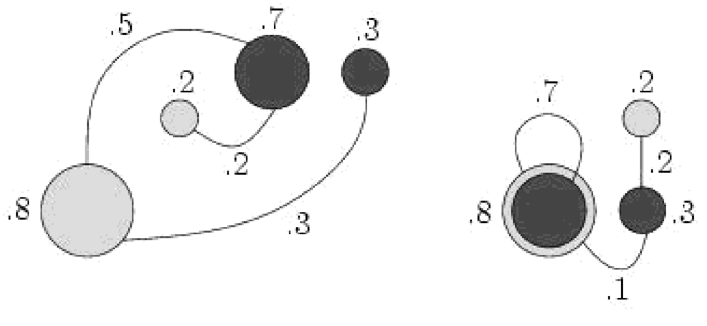
$$(2) \qquad g^{(k+1)} = \arg \min_{g \text{ in } G} \text{WORK}(F^{(k)},\mathbf{x},g(\mathbf{y})).$$

As shown in (1), the kth flow iterate is defined as a flow which minimizes the amount of work needed to match **x** and $g^{(k)}(\mathbf{y})$, where $g^{(k)}$ is the kth transformation iterate. As shown in (2), the (k+1)st transformation iterate is defined as a transformation g which minimizes the amount of work done by $F^{(k)}$ to match **x** and g(**y**). The iteration begins with an initial transformation $g^{(0)}$, and the order of evaluation is

$$g^{(0)} \dashrightarrow F^{(0)} \dashrightarrow g^{(1)} \dashrightarrow F^{(1)} \dashrightarrow \ldots .$$

An example will help clarify the iteration.

The left half of the figure below shows a dark and light distribution that we will match under translation using the previously given iteration.



The initial translation in this example is $g^{(0)}=0$, and we shall translate the dark distribution. The best flow $F^{(0)}$ for $g^{(0)}$ is shown by the labelled arcs connecting dark and light masses in the left half of the diagram. This flow $F^{(0)}$ matches half (.5) the mass over a relatively large distance. When we ask for the best translation for this flow, we should expect that translation to move the .7 weight dark mass closer to the .8 weight light mass in order to decrease the total amount of work done by $F^{(0)}$. Indeed, the optimal translation $g^{(1)}$ aligns these two masses as shown in the right half of the figure. The best flow $F^{(1)}$ for this translation matches all of the .7 weight dark mass to the .8 weight light mass. Since these masses are on top of each other, it costs zero work to match 70% of the distributions. No further translation of the dark distribution improves the work -- $g^{(2)}=g^{(1)}$ and the iteration converges.

The expression $\text{WORK}(F,\mathbf{x},g(\mathbf{y}))$ defines a surface over the space FxG of all possible (flow,transformation)=(F,g) pairs. Our iteration provides a downhill path over this surface. The flow and transformation iterates define WORK iterates

$$\text{WORK}^{(k)} = \text{WORK}(F^{(k)},\mathbf{x},g^{(k)}(\mathbf{y})).$$

From (2), we know that

$$(3) \quad \text{WORK}(F^{(k)},\mathbf{x},g^{(k+1)}(\mathbf{y})) <= \text{WORK}(F^{(k)},\mathbf{x},g^{(k)}(\mathbf{y}))=\text{WORK}^{(k)}$$

because $g^{(k+1)}$ is an optimal transformation for $F^{(k)}$ (and therefore yields a smaller work value than $g^{(k)}$ for $F^{(k)}$). Similarly, (1) implies

$$(4) \quad \text{WORK}^{(k+1)}=\text{WORK}(F^{(k+1)},\mathbf{x},g^{(k+1)}(\mathbf{y})) <= \text{WORK}(F^{(k)},\mathbf{x},g^{(k+1)}(\mathbf{y}))$$

since $F^{(k+1)}$ is an optimal flow for $g^{(k+1)}$. Combining (3) and (4) shows

$$\text{WORK}^{(k+1)} <= \text{WORK}^{(k)}.$$

Since the WORK sequence is decreasing and bounded below (by zero), it must converge. The corresponding transformation sequence may, however, converge to only a locally optimal transformation. Therefore, the iteration should be run with different initial transformations in search of the global minimum.

In just a couple of sections, we shall discuss two specific cases in which the EMD under transformation problem can be solved directly, without the aid of our iteration:

(i) equal-weight distributions under translation with the L_2 distance squared as the ground distance, and
(ii) equal-weight 1D distributions under translation with the absolute value as the ground distance.

Although our iteration is not needed here, it is guaranteed to converge to the global minimum of the WORK function in case (i).

Our iteration is very general, in that it can be applied for a number of different (ground distance d, transformation set G) pairs. The optimal flow step (1) is a transportation problem, which can be solved with the transportation simplex algorithm ([3]). We call the problem in step (2) the *optimal transformation problem*. Our iteration can be applied whenever the optimal transformation problem can be solved.

The Optimal Transformation Problem

The optimal transformation problem in step (2)

$$(2) \quad g^{(k+1)} = \arg\ \min_{g \text{ in } G} \ \mathrm{WORK}(F^{(k)},\mathbf{x},g(\mathbf{y})).$$

can be written explicitly in terms of the ground distance function d as

$$(5) \quad \arg\min_{g \text{ in } G} \mathrm{sum}_{i=1..m,\ j=1..,n}\ f_{ij}\ d(x_i,g(y_j)),$$

where the flow $F=(f_{ij})$ is fixed. The table below shows some cases in which (5) can be solved.

Transformation Set *G*	**g in *G***	**Ground Distance d**
translation	t	L_2, L_1, $(L_2)^2$
Euclidean	(R,t)	$(L_2)^2$
similarity	(s,R,t)	$(L_2)^2$
linear	A	$(L_2)^2$
affine	(A,t)	$(L_2)^2$

If we let

$$\begin{aligned}
[c_1 \ldots c_N] &= [f_{11}\, f_{12} \ldots f_{1n} \,|\, f_{21}\, f_{22} \ldots f_{2n} \,|\, \ldots \,|\, f_{m1}\, f_{m2} \ldots f_{mn}],\\
[a_1 \ldots a_N] &= [x_1\, x_1 \ldots x_1 \,|\, x_2\, x_2 \ldots x_2 \,|\, \ldots \,|\, x_m\, x_m \ldots x_m], \text{ and}\\
[b_1 \ldots b_N] &= [y_1\, y_2 \ldots y_n \,|\, y_1\, y_2 \ldots y_n \,|\, \ldots \,|\, y_1\, y_2 \ldots y_n],
\end{aligned}$$

where N=mn, then (5) can be rewritten as a single index summation

$$(6) \quad \min_{g \text{ in } G} \mathrm{sum}_{r=1..N}\ c_r\ d(a_r,g(b_r)),$$

In this form, the optimal transformation problem asks for the transformation of one point set which minimizes a sum of weighted distances to corresponding points in another set.

We briefly consider only the easiest case listed in the above table : $d=(L_2)^2$ and $G=T$, the group of translations. In this case, (6) becomes

$$(7) \quad \min_{t \text{ in } T} \mathrm{sum}_{r=1..N}\ c_r\ \|a_r-(b_r+t)\|^2.$$

It is well known (and easily proven using standard calculus) that the unique optimal translation in the least squares problem (7) is the translation that lines up the centroids of the weighted point sets $\{(c_1,a_1), \ldots, (c_N,a_N)\}$ and $\{(c_1,b_1), \ldots, (c_N,b_N)\}$:

$$t^* = \mathrm{centroid}(c,a) - \mathrm{centroid}(c,b)$$

where

$$\text{centroid}(c,a) = \text{sum}_{r=1..N}\, c_r\, a_r \;\; / \;\; c^S,$$
$$\text{centroid}(c,b) = \text{sum}_{r=1..N}\, c_r\, b_r \;\; / \;\; c^S, \quad \text{and}$$
$$c^S = \text{sum}_{r=1..N}\, c_r.$$

In terms of the original flow variables f_{ij} and distributions **x** and **y**, this solution becomes

$$(8) \quad t^* = \text{sum}_{i=1..m,\, j=1..n}\, f_{ij}\, x_i \;\; / \;\; \min(w^S,u^S) \;\; - \;\; \text{sum}_{i=1..m,\, j=1..n}\, f_{ij}\, y_j \;\; / \;\; \min(w^S,u^S),$$

where we have used the fact that $\text{sum}_{i=1..m,\, j=1..n}\, f_{ij} = \min(w^S,u^S)$ for any feasible flow F between **x** and **y**. In the next section, we shall discuss an interesting property of this solution when the distributions **x** and **y** have the same total weight.

Two Specific Cases

In this section, we examine two specific cases in which there is a direct solution to the EMD under transformation problem.

Equal-Weight Distributions under Translation with $d=(L_2)^2$

Recall the unique solution (8)

$$(8) \quad t^* = \text{sum}_{i=1..m,\, j=1..n}\, f_{ij}\, x_i \;\; / \;\; \min(w^S,u^S) \;\; - \;\; \text{sum}_{i=1..m,\, j=1..n}\, f_{ij}\, y_j \;\; / \;\; \min(w^S,u^S)$$

to the optimal transformation problem when the transformation set is translation and the ground distance is the Euclidean distance squared. When **x** and **y** are equal-weight distributions, we must have

$$(9) \quad \text{sum}_{j=1..n}\, f_{ij} = w_i \quad \text{and} \quad \text{sum}_{i=1..m}\, f_{ij} = u_j$$

for any feasible flow between **x** and **y**. This is because all the mass in both distributions must be matched when the two distributions have the same total weight. Substituting (9) into (8), we see that

$$t^* = \text{centroid}(\mathbf{x}) - \text{centroid}(\mathbf{y}),$$

where

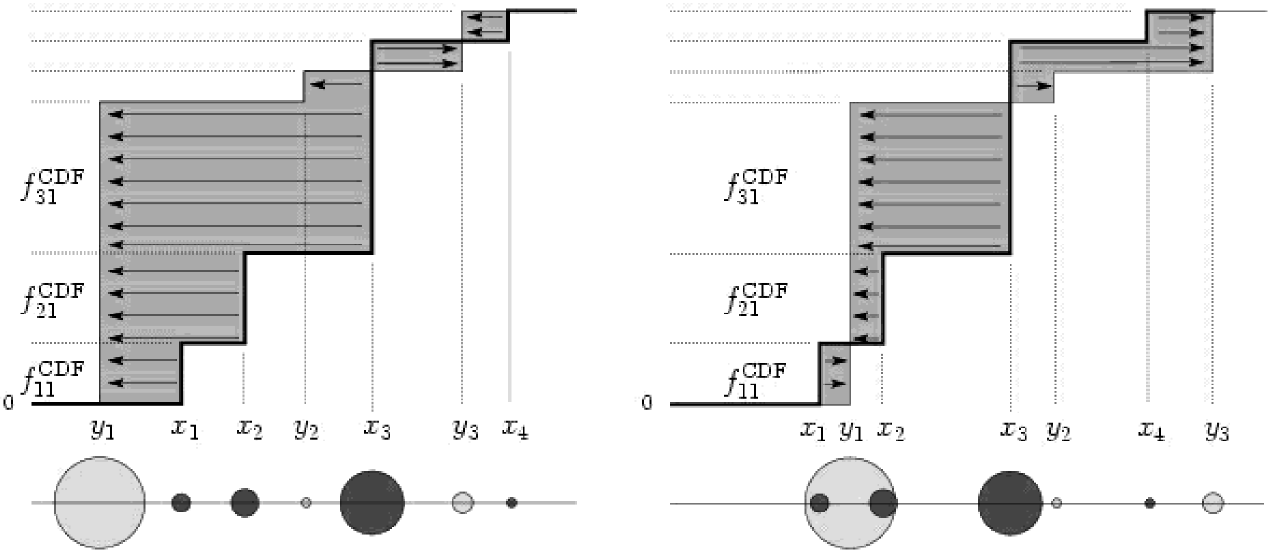
$$\text{centroid}(\mathbf{x}) = \text{sum}_{i=1..m}\, w_i\, x_i \;\; / \;\; w^S, \quad \text{and}$$
$$\text{centroid}(\mathbf{y}) = \text{sum}_{j=1..n}\, u_j\, y_j \;\; / \;\; u^S.$$

Here we have also used the fact that **x** and **y** are equal-weight distributions: $w^S=u^S=\min(w^S,u^S)$.

We have shown that the optimal translation in this case does not depend on the feasible flow used. The translation t^* which lines up the centroids of **x** and **y** is the unique optimal translation for every feasible flow. To compute the EMD under translation between equal-weight distributions with $d=(L_2)^2$, we simply compute EMD(**x**,**y**+centroid(**x**)-centroid(**y**)). Although our iteration is not needed here, it is guaranteed to converge to the global minimum of the WORK function in this case. In fact, it will converge in only a couple of steps since $g^{(2)}=g^{(1)}=t^*$.

Equal-Weight Distributions in 1D under Translation

In this section, we consider matching equal-weight distributions on the real line under translation, with the absolute value ($d=L_1$) as the ground distance. Consider the distributions shown in the left half of the figure below, where **x** is the dark distribution and **y** is the light distribution.



It turns out that there is a simple solution to computing the EMD between equal weight distributions in 1D that involves the cumulative distribution functions (CDFs) of the distributions. The CDF for **x** starts at 0, increases an amount w_i at each point x_i, and eventually becomes constant=w^S at the largest point x_m. The CDF for **x** is the bold staircase graph, while the CDF for **y** is the regular thickness staircase graph. Since **x** and **y** are equal-weight distributions, the two CDFs become constant at the same value $w^S=u^S$. The EMD between **x** and **y** is equal to the area between the CDFs of **x** and **y** divided by the total weight $w^S=u^S$. The area between the CDFs is shaded, and the corresponding optimal flow is indicated with arrows. This optimal flow is called the *CDF flow*, and is denoted F^{CDF}.

The CDF flow is given by

$$f^{CDF}_{ij} = |\ [W_{i-1},W_i] \text{ intersect } [U_{j-1},U_j]\ |, \quad i=1..m,\ j=1..n,$$

where

$$W_i = \mathrm{sum}_{k=1..i}\, w_k \quad \text{and} \quad U_j = \mathrm{sum}_{l=1..j}\, u_j$$

are the partial sums of the distribution weight vectors w and u, $W_0 = U_0 = 0$, and $|X|$ denotes the size of interval X. The partial sums U_0, U_1, ..., U_n are the same for **y** and every translated version of **y**. This means that the CDF flow is an optimal flow between **x** and **y**+t for every translation t. An EMD computation between **x** and a translation of **y** is shown in the right half of the above figure. To compute the EMD under translation between equal-weight distributions in 1D with $d=L_1$, we simply solve the optimal translation problem for the fixed flow $F=F^{CDF}$. The optimal translation problem with $d=L_1$ is covered in [1].

References

1. S. Cohen and L. Guibas. **The Earth Mover's Distance: Lower Bounds and Invariance under Translation**. *Technical Report STAN-CS-TR-97-1597*, November 1997. (ps, pdf)

2. G. Healey and D. Slater. **Global Color Constancy : Recognition of Objects By Use of Illumination-Invariant Properties of Color Distributions**. *Journal of the Optical Society of America A*, 11(11):3003-3010, November 1994.

3. F. Hillier and G. Lieberman. **Introduction to Mathematical Programming**, pages 202-229. McGraw-Hill, 1990.

4. Y. Rubner and C. Tomasi. **Texture Metrics**. *Proceeding of the IEEE International Conference on Systems, Man, and Cybernetics*, October 1998. To appear. (Y. Rubner's publications)

5. Y. Rubner, C. Tomasi, L. Guibas. **The Earth Mover's Distance as a Metric for Image Retrieval**. *Technical Report STAN-CS-TN-98-86*, September 1998. (Y. Rubner's publications)

6. Y. Rubner, C. Tomasi, L. Guibas. **Adaptive Color-Image Embeddings for Database Navigation**. *Proceedings of the 1998 Asian Conference on Computer Vision*, January 1998. (Y. Rubner's publications)

7. Y. Rubner, C. Tomasi, L. Guibas. **A Metric for Distributions with Applications to Image Databases**. *Proceedings of the 1998 IEEE International Conference on Computer Vision*, January 1998. (Y. Rubner's publications)

Email comments to scohen@cs.stanford.edu.