Computer Science and Artificial Intelligence Laboratory

Technical Report

MIT-CSAIL-TR-2003-033
AIM-2003-026

December 5, 2003

Fast Contour Matching Using Approximate Earth Mover's Distance

Kristen Grauman and Trevor Darrell

Abstract

Weighted graph matching is a good way to align a pair of shapes represented by a set of descriptive local features; the set of correspondences produced by the minimum cost of matching features from one shape to the features of the other often reveals how similar the two shapes are. However, due to the complexity of computing the exact minimum cost matching, previous algorithms could only run efficiently when using a limited number of features per shape, and could not scale to perform retrievals from large databases. We present a contour matching algorithm that quickly computes the minimum weight matching between sets of descriptive local features using a recently introduced low-distortion embedding of the Earth Mover's Distance (EMD) into a normed space. Given a novel embedded contour, the nearest neighbors in a database of embedded contours are retrieved in sublinear time via approximate nearest neighbors search. We demonstrate our shape matching method on databases of 10,000 images of human figures and 60,000 images of handwritten digits.

1. Introduction

The minimum cost of matching features from one shape to the features of another often reveals how similar the two shapes are. The cost of matching two features may be defined as how dissimilar they are in spatial location, appearance, curvature, or orientation; the minimal weight matching is the correspondence field between the two sets of features that requires the least summed cost. A number of successful shape matching algorithms and distance measures require the computation of minimal cost correspondences between sets of features on two shapes, e.g., [1, 17, 8, 6, 2, 5].

Unfortunately, computing the optimal matching for a single shape comparison has a complexity that is superpolynomial in the number of features. The complexity is of course magnified when one wishes to search for similar shapes ("neighbors") in a large database: a linear scan of the database would require computing a comparison of superpolynomial complexity for each database member against the query shape. Hierarchical search methods, pruning, or the triangle inequality may be employed, yet query times are still linear in the worst case, and individual comparisons maintain their high complexity regardless.

Recently developed tools allow fast (sublinear time) approximate nearest neighbor (NN) searches, e.g., the randomized algorithm of [7] called Locality-Sensitive Hashing (LSH). LSH indexes a set of training examples living in a normed space by a number of hash tables, such that the probability of collision is high for similar examples and low for dissimilar ones. Additionally, there are recently introduced algorithms that embed un-normed metrics into a normed space, so that they too may exploit fast approximate NN searching techniques. In particular, a low-distortion L_1 embedding of the Earth Mover's Distance (EMD) – the minimum cost that is necessary to transform one weighted point set into another – is given in [11], where it is applied to color-based image retrieval.

To address the computational complexity of current correspondence-based shape matching algorithms, we propose a contour matching algorithm that incorporates these approximation techniques and enables fast shape-based similarity retrieval from large databases. We embed the minimum weight matching of contour features into L_1 via the EMD embedding of [11], and then employ sublinear time approximate NN search to retrieve the shapes that are most similar to a novel query. The embedding step alone reduces the complexity of computing a low-cost correspondence field between two shapes from superpolynomial in the number of features to O($nd \log \Delta$), where n is the number of features, d is their dimension, and Δ is the diameter of the feature space (i.e., the greatest inter-feature distance).

In this work we also introduce the idea of a low-dimensional rich shape descriptor manifold. Using many examples of

high-dimensional local features taken from shapes in an image database, we construct a subspace that captures much of the descriptive power of the rich features, yet allows us to represent them compactly. We build such a subspace over the "shape context" feature of [1], which consists of local histograms of edge points, and successfully use it within the proposed approximate EMD shape matching method.

We demonstrate our fast contour matching method on datasets of 10,000 human figure images (real and synthetic examples) and 60,000 handwritten digits. We report on the relative complexities (query time and space requirements) of approximate versus exact EMD for shape matching. In addition, we study empirically how much retrieval quality for our approximate method differs from its exact-solution counterpart (optimal graph matching); matching quality is quantified based on performance as a k-NN classifier of digits or human poses.

2. Related Work

In this section we review relevant related work on the use of EMD as a similarity measure, current shape matching techniques requiring optimal correspondences between features, and the embedding of EMD into a normed space and fast approximate similarity search. For additional information about various distance metrics for shape matching and their computational complexities, please refer to [18].

The concept of using the Earth Mover's Distance to measure perceptual similarity between images was first explored in [15] for the purpose of measuring distance between gray-scale images. More recently EMD has been utilized for color- or texture-based similarity in [16, 9], and extended to allow unpenalized distribution transformations in [3]. In [6], a pseudo-metric derived from EMD that respects the triangle inequality and positivity property is given and applied to measure shape similarity on edges.

A number of shape matching techniques require the optimal correspondences between feature sets at some stage. The authors of [1] obtain least cost correspondences with an augmenting path algorithm in order to estimate an aligning transform between two shapes. They achieve impressive shape matching results with their method, but they note that the runtime does not scale well with the representation size due to the cubic complexity of solving correspondences. We utilize the shape context feature (log-polar histograms of edge points) of [1] as a basis for our shape reprentation in this work. The authors of [2] characterize local shape topologies with points and tangent lines and use a combinatorial geometric hashing to compute correspondence between these "order structures" of two shapes. In [17], a polynomial time method is given where the shock

graphs of 2-D contours are compared by performing a series of edit operations, and the optimal alignment of shock edges is found using dynamic programming. In [8], a graduated assignment graph matching method is developed for matching image boundary features that operates in time polynomial in the size of the feature sets.

Our goal is to achieve robust, perceptually meaningful shape matching results as the above methods can, but in a way that scales more reasonably with an arbitrary representation size and allows real-time retrieval from larger databases.

The embedding of EMD into L_1 is given in [11], where it is applied to measure image content similarity based on clustered pixel colors. For fast retrieval of similar shapes in a large database of embedded contours, we employ LSH, the approximate NN algorithm of [7], which enables sublinear search time. While the authors of [14] mention that using approximate NN search algorithms for shape based retrieval is a possibility, their system actually utilizes pruning techniques to speed searches. To our knowledge our work is the first to use the EMD embedding for fast contour matching.

3. Fast Similarity Search with EMD

In this section, for the reader's convenience, we briefly summarize the EMD metric and the randomized algorithms we use in our shape matching method: the approximate similarity search algorithm LSH given in [7], and the embedding of EMD into a normed space given in [11].

EMD is named for a physical analogy that may be drawn between the process of transforming one weighted point set into another and the process of moving piles of dirt spread around one set of locations to another set of holes in the same space. The points are locations, their weights are the size of the dirt piles, and the ground metric between a pile and hole is the amount of work needed to move a unit of dirt. To use this transformation as a distance measure, i.e., a measure of dissimilarity, one seeks the least cost transformation – the movement of dirt that requires the least amount of work. When the total weight in the two point sets is equal, the solution is a complete one-to-one correspondence, and it is equivalent to the bipartite graph matching problem. That is, for a metric space (X,D) and two n-element sets $\mathbf{A}, \mathbf{B} \subset X$, the distance is the minimum cost of a perfect matching between $\mathbf{A}$ and $\mathbf{B}$:

$$EMD(A, B) = \min_{\pi:\mathbf{A}\to\mathbf{B}} \sum_{a\in\mathbf{A}} D(a, \pi(a)). \tag{1}$$

EMD performs partial matching in the case that the two sets have unequal total weights; the distance is then the minimum

work needed to cover the mass in the "lighter" set with the mass in the "heavier" one.

LSH indexes a database of examples living in a normed space by a number of hash tables, such that the probability of collision is high for similar examples and low for dissimilar ones. In particular, LSH guarantees that if for a query point $\mathbf{q}$ there exists a point in the database $\mathbf{p}$ such that $d(\mathbf{p}, \mathbf{q}) \leq r$, then (with high probability) a point $\mathbf{p}'$ is returned such that $d(\mathbf{p}, \mathbf{q}') \leq (1+\epsilon)r$. Otherwise, the absence of such a point is reported. The query time for a database of N d-dimensional examples is bounded by $O(dN^{(1/(1+\epsilon))})$. See [7] for details.

The embedding of EMD given in [11] provides a way to map weighted point sets $\mathbf{A}$ and $\mathbf{B}$ from the metric space into the normed space L_1, such that the L_1 distance between the resulting embedded vectors is comparable to the EMD distance between $\mathbf{A}$ and $\mathbf{B}$ themselves. Working in a normed space is desirable since fast approximate NN search techniques such as LSH require it. The general idea of the embedding is to compute and concatenate several weighted histograms of decreasing resolution for a given point set. Formally, given two point sets $\mathbf{A}$ and $\mathbf{B}$, each of cardinality n, and each containing points in $\Re^d$: impose grids G_i, $-1 \leq i \leq \log(\Delta)$, on the space $\Re^d$, with grid G_i having side length 2^i, and Δ equal to the diameter of $\mathbf{A} \cup \mathbf{B}$. Each grid is translated by a vector chosen randomly from $[0, \Delta]^d$. To embed point set $\mathbf{A}$, for each G_i create vector $\mathbf{v}_i$ with one coordinate per grid cell, where each coordinate counts the number of points in the corresponding cell, i.e., each $\mathbf{v}_i$ forms a histogram of $\mathbf{A}$. The embedding of $\mathbf{A}$ is then the concatenated vector of the $\mathbf{v}_i$'s, scaled by the side lengths: $f(\mathbf{A}) = \left[\mathbf{v}_{-1}(\mathbf{A}), \mathbf{v}_0(\mathbf{A}), 2\mathbf{v}_1(\mathbf{A}), \ldots, 2^i\mathbf{v}_i(\mathbf{A}), \ldots\right]$. The distortion of the embedding has an upper bound of $O(\log \Delta))$ [1]. See [11] for details.

4. Approach

The main contributions of this work are a fast contour matching method that exploits the approximate EMD embedding and NN search algorithms described above, and a rich but compact contour feature descriptor manifold that is amenable to approximate EMD.

[1] The distortion C of a metric embedding f describes how much information loss the embedding induces: $\frac{1}{C} EMD(\mathbf{A}, \mathbf{B}) \leq ||f(\mathbf{A}) - f(\mathbf{B})||_{L_1} \leq EMD(\mathbf{A}, \mathbf{B})$.

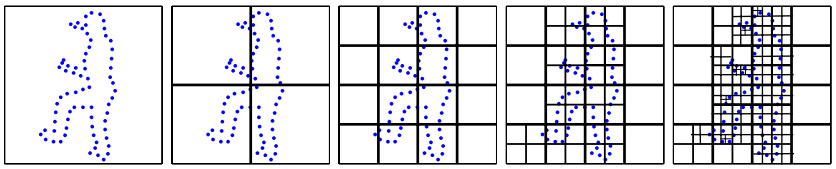


Figure 1: Imposing hierarchy of grids on a set of contour points to get its embedding.

4.1. Matching Contours with Approximate EMD

Recall our motivation to design an efficient means of calculating the least cost correspondences between two shape feature sets: such correspondences are required by a number of effective shape matching algorithms, but typically optimal solutions make large per-object feature set sizes or large database retrieval problems impractical for these algorithms. We will thus embed the problem of correspondence between two sets of local shape features into L_1, and use the approximate solution to match the shapes. The embedded vector resulting from an input point set is high-dimensional, but very sparse; only $O(n \log(\Delta))$ entries are non-zero. The time required to embed one point set is $O(nd \log(\Delta))$. Thus the computational cost of retrieving the near-optimal feature correspondences for our shape matching method will be $O(nd \log(\Delta)) + O(n \log(\Delta)) = O(nd \log(\Delta))$, the cost of embedding two point sets, plus an L_1 distance on the sparse vectors. The exact solutions typically used in shape matching to solve this correspondence (such as dynamic programming, the simplex algorithm for linear programming, or the Hungarian method for bipartite graph matching) require time cubic or exponential in n.

Probably the most direct application of EMD for 2-D contour matching is to compose point sets from the literal points on the two contours (or some subsets of them) and use the Euclidean distance between two contour points' image coordinates as the ground distance D in (1). For this simple positional feature, examples must be translated and scaled to be on par with some reference shape. To embed a set of 2-D contour points, we impose a hierarchy of grids on the image coordinates themselves, starting at a single cell the size of the largest image ($G_{\log \Delta}$), and ending with a grid resolution where each image coordinate receives its own cell (see Figure 1). For efficiency, once a grid cell is found to be empty, its count is no longer polled throughout the following, higher resolution grids. We do an empirical study of the distortion induced on the correspondence solution in Section 5.2.

When two weighted point sets have unequal total weights, EMD does not satisfy the triangle inequality or positivity property, and thus is not a true metric. Yet while the embedding of EMD into a normed space is given for the metric case of

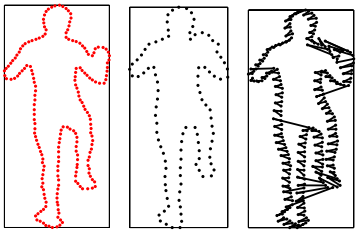


Figure 2: To simulate total unequal weights between two point sets during the EMD embedding process, evenly sampled points from the lighter set are counted multiple times in the embedding grids. Points with two correspondences (right image) were duplicated instances of a single point in the second contour (middle), which has a lower total weight than the first contour (left).

EMD, the partial matching abilities of EMD may still be preserved in the approximate case if the manner of embedding the points is modified. In order to embed two sets of contour features with different cardinalities, we simulate equal cardinalities by adding the appropriate number of duplications of random points from the lower cardinality set (see Figure 2).

Once a database of contours is embedded into a normed space, we do fast (time sublinear in the database size) retrievals for a novel query contour via LSH. In addition to the complexity savings for a single shape match described above, the time required for retrieving similar shapes is reduced to $O(sN^{(1/(1+\epsilon))})$, where N is the number of shapes in the database, ϵ is the LSH parameter relating the amount of approximation of the normed distance, and s is the dimension of the sparse embedded contour vectors, s having a space requirement of $O(n \log(\Delta))$. Results for the quality of the approximate NN contours we retrieve with LSH are reported in Section 5.

4.2. Shape Context Manifolds

There are drawbacks to using the simple positional feature for shape matching with approximate EMD. Though straightforward to embed, it can be a brittle feature, and in order to achieve scale or translation invariance this feature demands a pre-processing stage on all shapes (which itself can be brittle). Richer shape descriptors can overcome this, as several authors have noted [13, 1, 2]. Thus, we have experimented with richer shape descriptors with the approximate EMD distance, as we expect to achieve more robust matchings and more meaningful correspondence fields from feature representations that are highly descriptive. We employ the shape context local descriptor given in [1]. The shape context feature at a single contour point is a log-polar histogram of the coordinates of the rest of the point set, measured using the reference point as the origin. It is inherently translation invariant, and robust scale and rotation invariance may be added (see [1]).

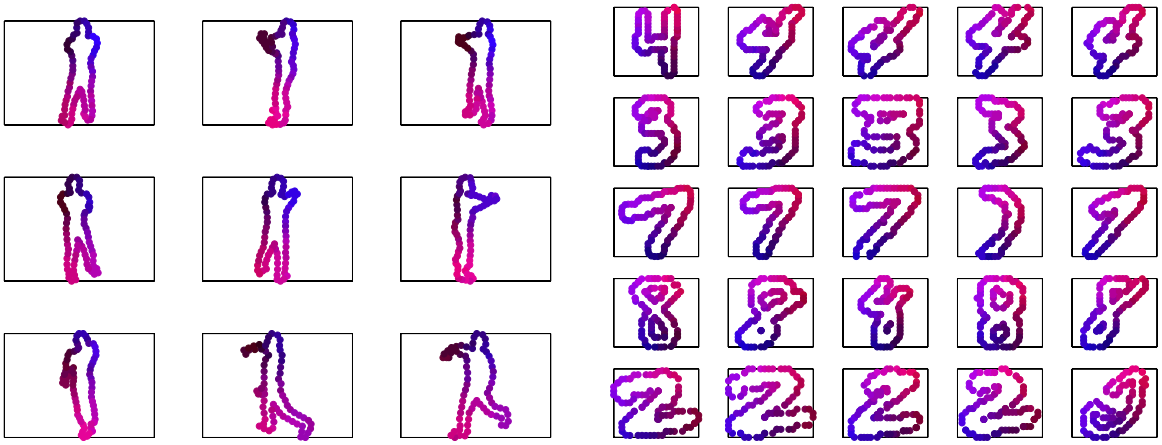


Figure 3: Visualization of feature subspace constructed from shape context histograms. The R-B channels of each point on the contours are colored according to its histogram's 2-D PCA coefficient values. Matching with EMD in this feature space means that contour points of similar color have a low matching cost (require little "work"), while highly contrasting colors incur a high matching cost. (This figure must be viewed in color.)

While matching with the full shape context feature (say, $d = 60$) is possible with our method, a low-dimensional, but rich, feature descriptor is desirable since any constant change in point dimensions changes the constant distortion factor C in the embedding, and also changes the d factor in the complexity of computing the embedding itself [11]. Thus, instead of attaching a raw, full-dimensional shape context feature to each sampled object point, we use principal components analysis (PCA) to find a low-dimensional subspace based on a large sample of the shape context histograms. This sample is drawn from the database of contours on which we wish to apply our method. PCA yields the set of bases that define a low-dimensional "shape context manifold". All contours (database and novel queries alike) are then projected onto the subspace, and the approximate EMD embedding is performed in the domain of a small number of their subspace coordinates.

We found that a very low-dimensional subspace was able to capture much of the local contour variation in our datasets. Figure 3 gives examples of the shape context subspace for human figures and handwritten digits. In Figure 4 we measure its expressiveness as a function of feature dimension, as compared to a higher-dimensional raw point histogram. In Section 5 we report results using both the positional and shape context representations.

5. Results

In this section we describe the datasets and representations with which we have tested our approximate EMD shape matching algorithm. We present example matches, summaries of retrieval quality on large databases, and a discussion on the empirical

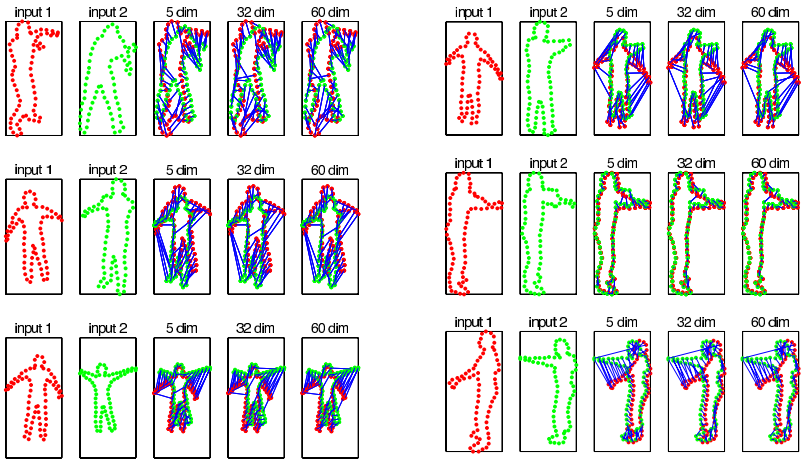


Figure 4: Shape context subspace captures local contour variations in much fewer dimensions. The optimal correspondences for feature sets composed of 5 and 32 shape context PCA coefficients are shown here and compared with the correspondences that result with the raw (60 dim) shape context feature. The optimal correspondence found by the raw high-dimensional feature is generally achieved by projections onto a much lower dimensional subspace.

computational complexity and distortion of our method.

5.1. Datasets and Representation

We have tested our method on two databases of contours: a set of 10,000 images of synthetic human figure contours in random poses that were generated with a computer graphics package called Poser [4], and a set of 60,000 handwritten digits from the benchmark MNIST dataset [12]. The separate test sets with which we query the human figure and digit databases are of sizes 1,050 and 10,000, respectively. A third set of 1,000 real images from a single human subject in various poses was also used to query the synthetic image database.

We have experimented with two feature representations: scale and translation invariant contour points, and projections of shape context histograms onto a low-dimensional subspace. Contour points are extracted from the binary silhouette inputs, then normalized for scale and translation by subtracting the silhouette's center of mass from each contour point coordinate and dividing by the estimated height of the shape. We construct a shape context subspace from 5 x 12 log-polar histograms extracted from the training sets; we used a sample of 816,000 histograms from the human figure data, and 420,000 from the digit data. The representation of a novel contour is determined by projecting its shape context histograms onto the low-dimensional subspace. We found that for our datasets, a 2-D projection adequately captured the descriptive power of the shape context feature and resulted in good contour matches for our datasets (see Figure 3). Since this representation is

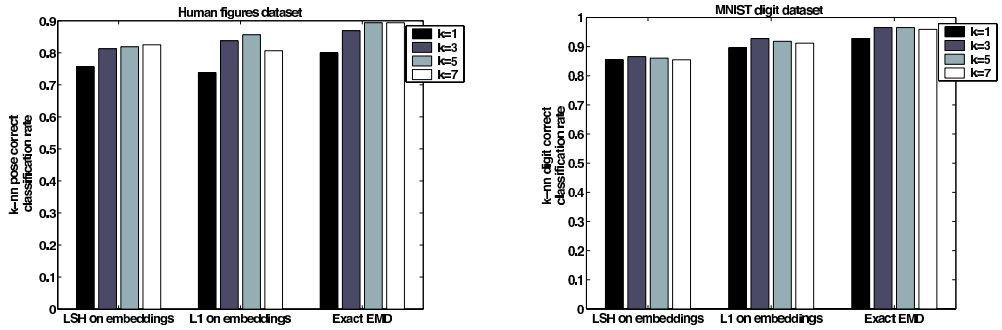


Figure 5: Comparison of the quality of retrievals from the exact versus approximate methods. Quality is measured via k-NN classification task. See text for details.

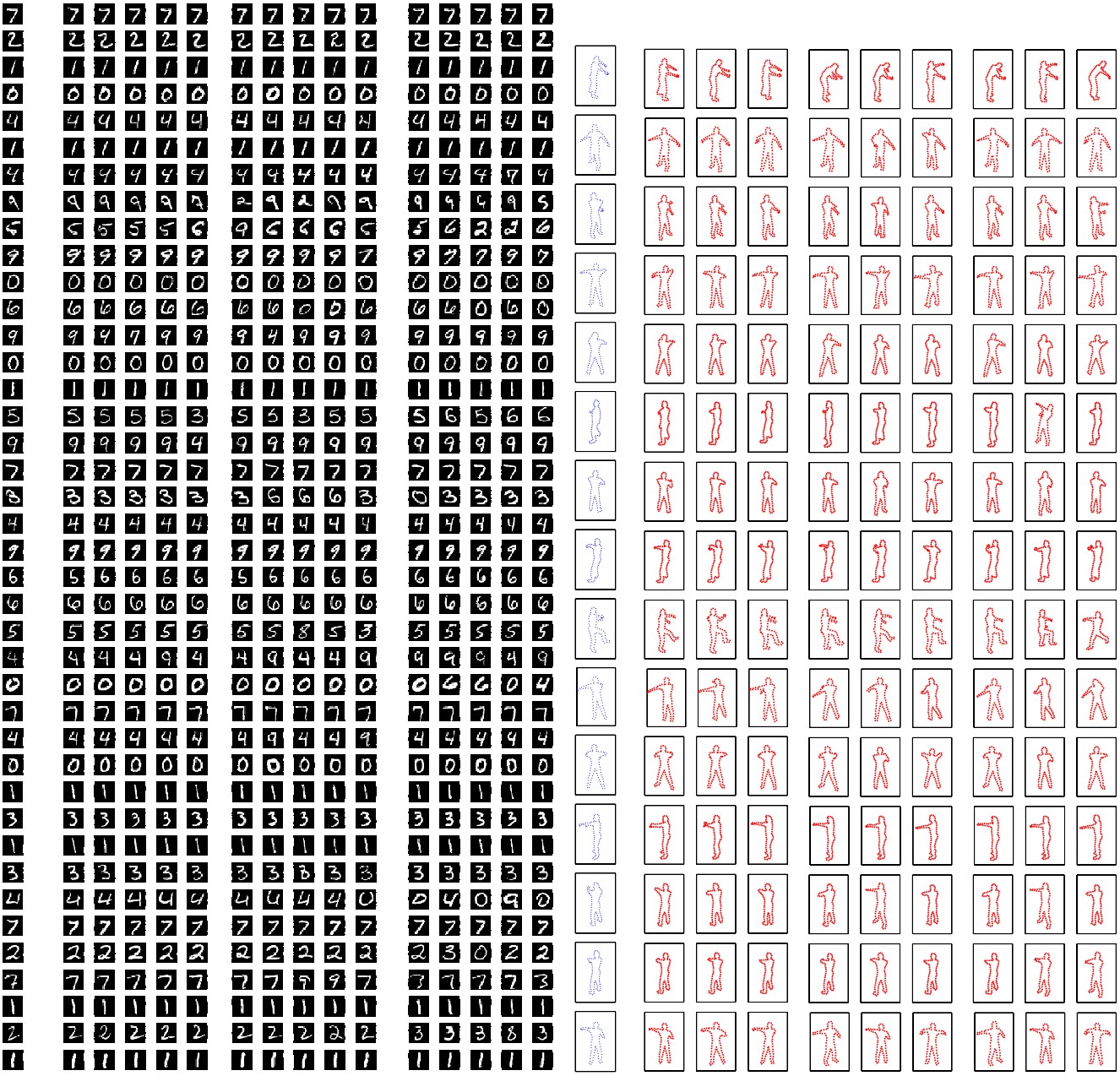
scale and translation invariant, no pre-processing for invariance is necessary as with the raw contour points. Because the coefficients are real-valued, they must be appropriately scaled and discretized before the embedding grids can be imposed. We remap the projection coefficients to positive integers by subtracting the minimum projection value from all examples, then scaling by 100.

5.2. Retrieval Quality

We measure the retrieval quality of our method by comparing the label of each query example with the labels of the k-NN's that are retrieved from the database. For the handwritten digits data, the label is simply the digit's value. For the human figures, the label is pose: when the synthetic human figure database is generated, each example is labelled with the 3-D pose (a set of 3-D joint positions). If the joint positions of a retrieved shape are on average within some threshold of distance, we consider the retrieved shape a good match. We chose a threshold of 10 cm, since this is a distance at which the 3-D poses are perceptually very similar.

We constructed a separate set of hash functions for each dataset in order to perform LSH approximate-NN retrievals. For each, we determined the parameters (ϵ, the number of hash tables, and the number of bits per hash value) based on the proof in [10] which shows how to select parameters for the case of the Hamming space over a non-binary alphabet, such that the desired level of approximation versus speed tradeoff is achieved. For the digits data, this meant using 7 hash tables and 60-bit hash functions; for the human figures, 4 hash tables and 68-bit functions.

The charts in Figure 5 quantitatively compare the quality of results obtained with our approximate method with those obtained from exact EMD. In this figure the optimal results were obtained from running the simplex algorithm to compute



(a) Queries (b) Exact EMD (c) L1 on embedded EMD (d) LSH on embedded EMD (e) Queries (f) Exact EMD (g) L1 on embedded EMD (h) LSH on embedded EMD

Figure 6: Approximate EMD retrieves shapes very similar to those retrieved by the optimal matching. Figure shows examples of NN (left to right in rank order) with embedded EMD contour matching (c,g) and embedded EMD contour matching with LSH (d,h), compared to NN with exact EMD contour matching (b,f). Five NN are shown for digits, and 3 NN are shown for human figures. Examples shown were chosen randomly from test set results of sizes 10,000 (digits) and 1,050 (humans); NN were retrieved from databases of sizes 60,000 (digits) and 10,000 (humans). Columns (c,d,g,h) use embedded 2-D shape context subspace projections as representation; (b,f) are from exact EMD applied to full 60-D shape context feature. Note that the embedded match results are qualitatively similar, yet several orders of magnitude faster to compute in (d,h), and an order of magnitude faster to compute in (c,g).

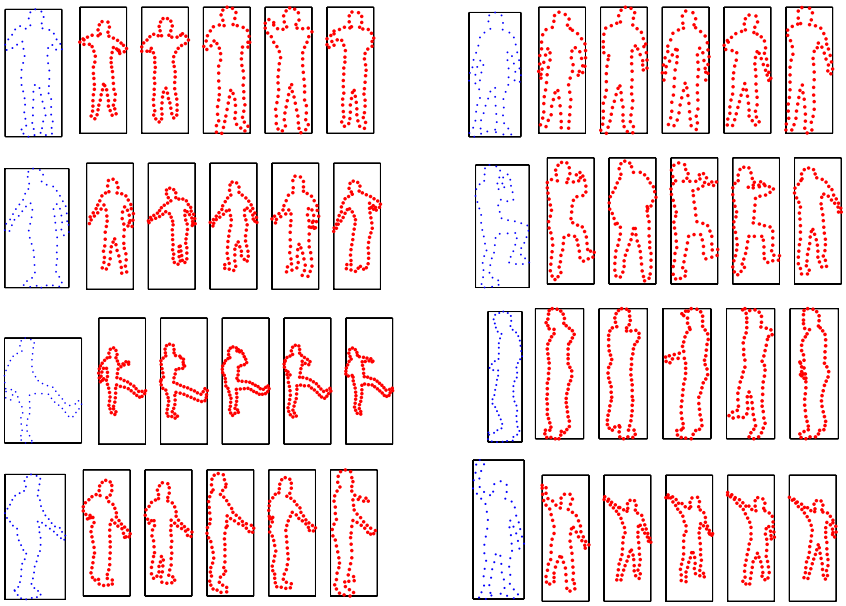


Figure 7: Real image queries: examples of query contours from a real person (left,blue) and 5 NN retrieved from synthetic database of 133,000 images using L_1 on EMD embeddings of shape context subspace features.

EMD on full, 60-D shape context features, whereas results for the two approximations (the embedding and LSH) were obtained using only 2-D shape context subspace features. There is a slight decrease in classification performance at each approximation level; however, we found that the practical bound on the distortion introduced by the EMD embedding is significantly (about one order of magnitude) lower than the upper theoretical bound.

We note that in a practical system classification rates could be improved in the approximate methods if a refining step were implemented – for instance a handful of exact computations on the approximate matches. Figure 6 shows some example retrievals using our approximate EMD method for each dataset. Examples of the synthetic NN that were retrieved for the images from a real person using the EMD embedding are shown in Figure 7.

5.3. Empirical Measure of Complexity

As discussed in Section 4, the theoretical computational complexity of retrieving the approximate minimum cost feature correspondences with our method for feature sets of cardinality n and dimension d living in a space of diameter Δ is $O(nd\log(\Delta))$. The diameters of the spaces in which our point sets reside are on the order of 10^3 up to 10^5, depending on the representation; with n on the order of 10^2, $d = 2$, theoretically this means that a single embedding and L_1 distance

cost requires on the order of 10^3 operations. This is the cost of embedding two point sets, plus performing an L_1 distance on the very sparse vectors. In practice, for $n = 100$ an unoptimized Matlab implementation of our method takes about 0.1 seconds to perform a single matching with exact L_1 this way (approximately 0.1 seconds to compute the two embeddings, plus $7.9 \times 10^{-}5$ seconds to compute the L_1 distance). To compute the exact EMD using a C implementation of the simplex algorithm required on average 0.9 seconds for data of the same dimensions. In accordance with the upper bound on the embedding's space requirements, the number of non-zero entries in the sparse embedded vectors was on average 350 for the histogram representation and 150 for the contour point representation.

The larger payoff for using the approximate embedding, however, comes when we use LSH to query a large database with an embedded point set. Using our Matlab implementation, the average runtime for an approximate-NN query in a database of 60,000 (previously embedded) images is 1.75 seconds. On average, LSH needed to compute only 624 L_1 distances per query, (about 1% of the database) for the digits dataset; for the human figure data, an average of 477 distances per query were searched (about 5% of the database). Since the exact EMD computation requires $O(n^3)$ operations, we could only collect a limited number of comparison results using exact EMD with full-dimensional features; we were able to collect a far greater number of trials with our approximate method because of its speed.

6. Conclusions and Future Work

We have presented a new fast contour matching algorithm that utilizes an approximation to EMD to judge similarity between sets of rich local shape descriptors. Our technique enables fast shape-based similarity retrieval from large databases, and its runtime is only linearly dependent on the number of feature points used to represent a shape. We have also constructed a rich but compact contour feature descriptor manifold based on shape contexts for approximate EMD.

In the future we intend to experiment with approximate EMD and different shape representations. We will also explore alternative means of compactly representing inherently continuous features within the discrete embedding framework, such as vector quantization or multi-dimensional scaling. We are also interested in investigating ways to improve the efficacy of the NN hashing process in this context.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24):509–522, 2002.

[2] S. Carlsson. Order structure, correspondence and shape based categories. In *International Wkshp on Shape Contour and Grouping*, Sicily, May 1998.

[3] S. Cohen and L. Guibas. The earth mover's distance under transformation sets. In *International Conference on Computer Vision*, 1999.

[4] Egisys Co. Curious Labs. Poser 5 : The Ultimate 3D Character Solution. 2002.

[5] Y. Gdalyahu and D. Weinshall. Flexible syntactic matching of curves and its application to automatic hierarchical classification of silhouettes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12), 1998.

[6] P. Giannopoulos and R. Veltkamp. A pseudo-metric for weighted point sets. In *Proceedings of European Conference on Computer Vision*, pages 715–730, 2002.

[7] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB)*, 1999.

[8] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):377–388, 1996.

[9] H. Greenspan, G. Dvir, and Y. Rubner. Region correspondence for image matching via emd flow. In *IEEE workshop on Content-based Access of Image and Video Libraries*, 2000.

[10] P. Indyk. *High-Dimensional Computational Geometry*. PhD thesis, Stanford University, 2000.

[11] P. Indyk and N. Thaper. Fast image retrieval via embeddings. In *Third International workshop on Statistical and Computational Theories of Vision, ICCV*, Nice, France, 2003.

[12] MNIST handwritten digits data set. *http://yann.lecun.com/exdb/mnist*.

[13] F. Mokhtarian, S. Abbasi, and J. Kittler. Robust and efficient shape indexing through curvature scale space. pages 53–62, Edinburgh, UK, 1996.

[14] G. Mori, S. Belongie, and J. Malik. Shape contexts enable efficient retrieval of similar shapes. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, Lihue, HI, 2001.

[15] S. Peleg, M. Werman, and H. Rom. A unified approach to the change of resolution: space and gray-level. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:739–742, 1989.

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

[17] T. Sebastian, P. Klein, and B. Kimia. Recognition of Shapes by Editing Shock Graphs. In *International Conference on Computer Vision*, pages 755–762, Vancouver, Canada, December 2001.

[18] R. Veltkamp and M. Hagedoorn. State-of-the-Art in Shape Matching. In *Technical Report UU-CS-1999-27*, Utrecht University, the Netherlands, 1999.