Indexing Based on Edit-Distance Matching of Shape Graphs

Srikanta Tirthapura[2],Daniel Sharvit[1],Philip Klein[2],Benjamin B. Kimia[1]

[1]LEMS, Division of Engineering, Brown University
Providence RI 02912.

[2] Computer Science Department, Brown University
Providence, RI 02912.

ABSTRACT

We are investigating a graph matching approach for indexing into pictorial databases using *shock graphs*, a symmetry-based representation of shape. Each shape (or a collection of edge elements) is represented by a shock graph. Indexing of a query into a pictorial database is accomplished by comparing the corresponding shock graph to the graphs representing database elements and selecting the best match. This paper introduces a new metric for comparing shock graphs.

The premise underlying the use of symmetry as a cue for indexing is that the correlation of pairs of edge elements in the sketch query and in the image is a more robust measure of indexing than simply correlating edge elements. Each pair of edge elements is represented by a symmetry curve, and can be extracted from real images and represented as singularities resulting from the propagation of each such orientation element. Previous work shows that these singularities, or *shocks*, can be detected, classified, grouped, and organized into a *shock graph* embedded in the plane.

In previous work, we proposed the use of a graph comparison heuristic called the *graduated assignment algorithm* which casts the problem as an integer quadratic program. This heuristic has two drawbacks. First, the quadratic program does not fully capture the topological constraints of graph matching, and so even an optimal solution could be topologically incorrect. Second, since solving the quadratic program is infeasible, the algorithm often finds only locally optimal solutions.

We now present a matching approach based on the notion of *edit distance*. The edit distance between two graphs is defined as the cost of the "least action" path of elementary edit transformations taking one graph to the other. We have defined a set of elementary edit transformations appropriate to shock graphs. These operations are motivated by intuitive and natural deformations of shape and mathematically correspond closely to the set of *shock transitions*. Moreover, we observe that shock graphs have special structure: they are unrooted, planar-embedded trees. By exploiting this structure, we have developed an algorithm to compute the edit distance (and corresponding edit transformations) between shock graphs precisely and quickly (in polynomial time). We have implemented this algorithm for a subset of these edit operations. In this paper, we illustrate these results for a few shapes. We plan to report on a complete implementation and results in a later paper.

Keywords: shape query, shocks, shape-based indexing, deformations, shock hierarchies, shock graphs, tree edit distance, unrooted tree, tree to tree correction, similarity, shock transition

Acknowledgements

We gratefully acknowledge the help of Ilana Frankel in the project.

Correspondence should be sent to kimia@lems.brown.edu. The authors gratefully acknowledge the support of the National Science Foundation. IRI 9700497 and CCR-900146

1. INTRODUCTION

The indexing of image databases by the shape of their embedded objects is complicated by the variety of visual transformations objects undergo in the imaging process (occlusion, articulation, viewpoint variation), and by the fact that recognition of shape outlines must take place hand-in-hand with the segmentation of the outline itself. The challenge lies in the design of representations for shape and of similarity measures between these representations that are robust against both visual transformations and variations in final segmentations.

That segmentations can be based either on extracting of edges or on grouping regions has led to dual approaches to indexing image databases, one relying on shape contours and the other on shape interiors. On the one hand, Hirata and Kato[1,2] used an abstracted edge map as the basic representation of shape and defined a measure of similarity to user-drawn sketching for indexing. Mehothra and Grosky [3] matched a chain code representation of edges for retrieval of images from a database of industrial parts. Lei *et al.*[4] used implicit polynomial models of curve patches as the basic representation of extracted and grouped edges. On the other hand, properties of the shape interior have been used for indexing. These range from global statistics such as shape moments and shape eccentricity used in QBIC,[5] to physically-based models of a shape's modes of vibration,[6] see also.[7] In contrast to approaches which extract features from images and match these against user-drawn sketches, others have compared sketches directly against images. Del Bimbo *et al.*[2,8] deformed contours of a shape query to adhere to object boundaries and used the degree of deformation as a measure of similarity. Kimia *et al.*[9] used a shock-based model of shape represented as a hierarchical graph and inexact graph matching to measure distances between shapes and user-queries. While their use of graduated assignment[10] handles missing/extra information and enforces two-way assignment between two shapes to ensure one-to-one correspondences, these correspondences are only locally consistent. Specifically, the graduated assignment finds a local optimum of a quadratic program, not a global optimum. Furthermore, in this comparison method, the graph topology is not necessarily preserved during the match. In addition, the effect of visual transformations in altering the underlying shape representation is not explicitly modeled.

The current paper describes a novel approach for measuring distances between shock graphs as the cost of the "least action" path consisting of sequences of elementary operations (*edits*). The main idea is that instead of relying on the geometric (*e.g.*, Hausdorff metric for curve representation of shape) distance between two shapes A and B, we generalize the notion of distance to be the minimum cost of deforming shape A to shape B. This is significant to robust recognition since deformations are inherent to visual transformations objects undergo in the imaging process. The shape from deformation approach[11–13] relates a shape to its nearby shapes by means of deformations, and where two canonical deformations define the reaction-diffusion space while the scope of deformations was limilted, the implicit notion of similarity was the extent of deformations necessary to bring one shape in correspondence with the other.[14,15,9] We now define a rich class of edit operations and explicitly consider all paths of deformation in the shape space. If deformation paths are restricted to simplify shapes (as in scale-space methods),[12,16] then similarity is the cumulative extent of deformation of each shape to a common, simplified shape, Figure 1. The shift in emphasis from metrics which measure similarity between representations to those which measure extent of transformation is significant. A rigorous mathematical study along these lines is shown by Younes.[17]

Psychophysical accounts of classification also rely on a notion of similarity. The geometric view assumes that objects are represented as points in a high dimensional space and similarity between any two objects is inversely related to the (Euclidean) distance between their representations (points). Shepard proposed a Universal Law of Generalization[18] where confusability between two objects only depends on the distance between two objects (exponentially decaying function), but not on the nature of the representation. The feature-based theory of Tversky[19] represents objects as a set of features and similarity between two objects is the extent of overlap between there feature sets. Recently a transformation-based view of similarity was proposed,[?] where the similarity between two objects is the extent of distortion required to transform one object to another. It is shown that the Universal Law of Generalization can follow from this transformation-based approach.

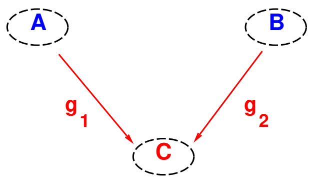

Figure 1. The similarity between A and B is the minimum cumulative cost of g_1 and g_2^{-1}, where g_1 and g_2 are deformations of A and B to a common simpler shape C.

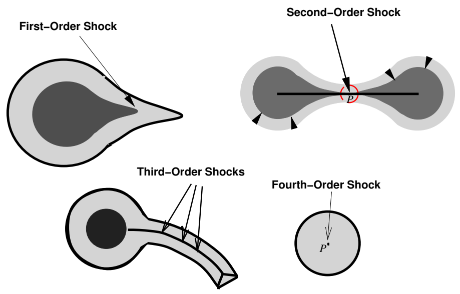

Figure 2. Each of the four type of shocks each is correlated with a perceptual/semantic category, *i.e., protrusion, part, bends* and *seed.*

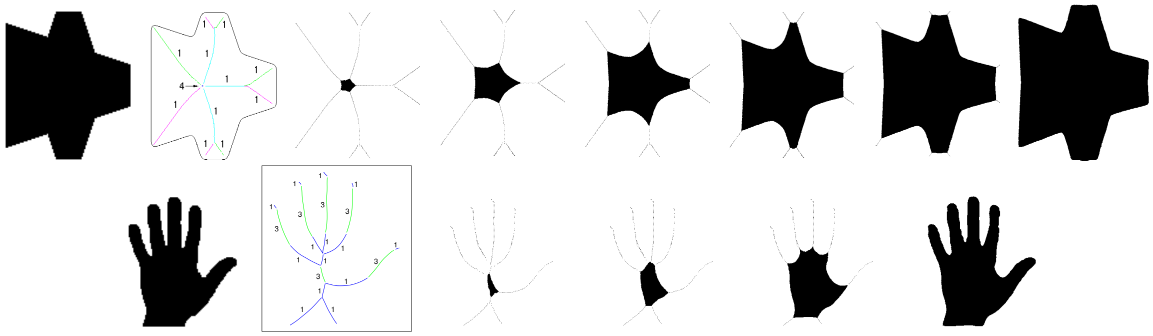

Figure 3. The shock-based description of a shape composed of trapezoids, as a hierarchy of merged protrusions, and its growth from shocks. The shock-based description of a biological shape as five bends (the fingers) attached to protrusions (the latter describing the palm), and its growth from shocks.

2. SHOCKS AND GRAPHS

The notion of shocks, how they represent shapes, their hierarchical organization and their classification, and computations involving them have all been presented in previous work.[20,13,21] For the purpose of this paper, we give a very brief summary of shocks as relevant to this paper.

First we give the mathematical structure of shocks, then we say how the shocks relate to the shape they represent. To each piecewise continuous shape in the plane is associated a set of labeled points. The topology of the set of points is that of a 2-simplicial complex (i.e. that of an embedded graph). The function f mapping a point to its label is piecewise continuous: nearby points have nearby labels. The labeled points are called *shocks.*

Given a binary shape (simple closed curve) in the plane, a point is a shock if there is a circle centered at the point, lying completely inside the shape and touching the shape in at least two propagation places. A label of the shock is the radius of the circle.

If we ignore the labels on the shocks, we get a set of points that comprise a traditional skeleton of a shape; see Figure 3 for examples. However, the set of points alone does not fully capture the shape. The set of shocks does fully capture the shape in that the shape can be uniquely determined from the shocks.

By interpreting the labels as indicating *time* and using continuity, we can interpret the set of shocks as a discrete set of trajectories of particles in time and (two-dimensional) space. A particle originates at each shock having time 0, moves through space (possibly merging with other particles), and terminates at a shock whose time is locally maximal. This interpretation enables us to assign other, secondary parameters to the shocks, such as instantaneous velocity.

The topology of shocks suggests a graph representation. For open curves, as they arise from edge maps of greyscale images, there is a similar notion of shocks based on the propagation of waves, or equivalently, minimal distance from the boundary. Typically, shocks have radii that change monotonically along their path; we call these the first-order shocks. However, these are turning points where the radius is locally a minimum or a maximum; we call these second and fourth order shocks. These are degenerate points whose radii is constant; these are third-order shocks. The third-order shocks can be considered as the limiting point of first-order shocks. In contrast to first and third shocks, second and fourth order shocks are isolated and finite in number, which together with branch points of shock path (J) and initial points of shock paths (I) form the nodes of the shock graph. The remaning first and third order shocks are curve segments augmented by the labels representing shock velocities and serve as *edges* or *links* of the graph. In fact, the knowledge of the curvature of shock paths and the acceleration of shocks is sufficient to reconstruct the shape corresponding to that segment. Thus we obtain an undirected, labeled graph.

Note that each node of the graph has a position in the plane, that each edge follows a simple path in the plane, and that these edge-paths do not cross. Thus, the graph is *embedded* in the plane. Moreover, because the boundary of the shape being represented is a simple closed curve, it follows that the graph of a simple shape (no nodes) is a *tree, i.e.* it is connected and has no cycles. We have been assuming that the shape being represented is a simple closed curve. However, images do not typically contain binary shapes and their edge maps rarely produce closed boundaries. A revision of the basic shocks-from-deformation approach that relies on wave propagation[22–24] makes the approach applicable to realistic edge maps by allowing shocks to be computed from partial contour segments. The notion of interpreting waves and second generation shocks allows a treatment of these shocks, but is not in the scope of this paper. At this stage, we present our approach for shapes with simple closed boundaries.

Figure 4. The skeletons of partial contours do not bear much resemblance to those of complete contours, leading to a fundamental instability in computing them from edge maps. Shock labels R=Regular, D=Degenerate, S=Semi-degenerate (shown in green, red, and yellow, respectively), however, allow the recovery of partial symmetries (regular shocks) as a subset of these. It is this set of regular shocks and the associated transformations of the remaining set (edit operations) that we propose for for indexing into a database of shapes.

3. SHOCK GRAPH MATCHING

Graphs are powerful data structures which describe the relationship among abstracted structure. Shapiro and Haralick[25] were the earlier pioneers in their use in structural description of objects in images via weighted graphs. Fu[26] used attributed relational graphs (ARG's) to describe parametric information as a basis of a general image understanding system based on extracting and matching hierarchical ARG's. Eshera and Fu [27,28] found the best inexact match between two ARG's by minimizing the overall distance between the two graphs, defined as the incremental distance between corresponding nodes and links. Minimization of distance was converted into a

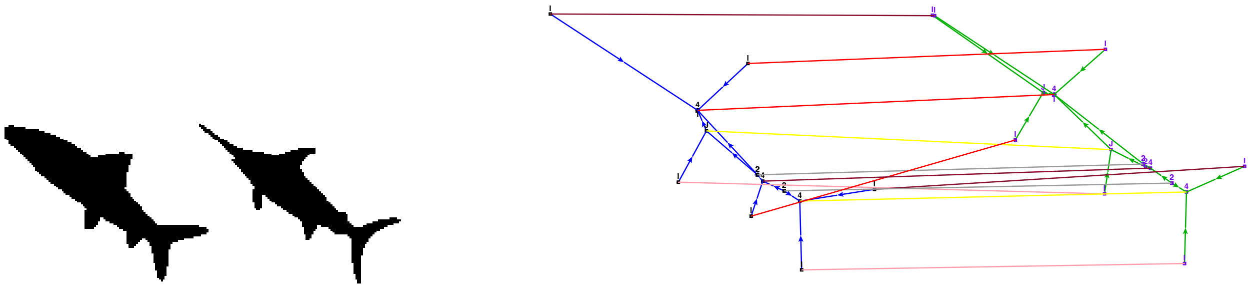

Figure 5. Example of a match between two fish established by the graduated assignment.

shortest path problem over the directed acyclic branch-weighted lattice from the initial state to a state in the set of final states. This is solved by dynamic programming in a time linearly proportional to the number of lattice. Unfortunately, this number is exponential in the number of nodes of the graph, so this approach is only useful for very small graphs. This exponential behavior of computation for graph comparison is unavoidable: most formulations of the problem are computationally intractable (NP-hard). For this reason, researchers have been forced to use heuristics, methods that are not guaranteed to find the correct answer and/or are not guaranteed to terminate quickly for all inputs.[29–33]

A method which we have previously adopted to implement shock graph matching is the *graduated assignment algorithm*.[10] The basic idea underlying graph matching is to associate nodes in two graphs as represented by a match matrix M (a permutation matrix if the numbers of nodes in two graphs are equal) where 1 represents association of two nodes. Slack rows and columns are added to the match matrix to represent missing/extra nodes. An energy functional describes the goodness of the match. Graduated non-convexity is used to turn these discrete (binary) variables into continuous ones. To avoid poor local minima, a control parameter is used to slowly move the matrix towards a $(0, 1)$ discretizaion. At each stage, the best match matrix is estimated and normalized to ensure it remains the continuous analogue of a discrete assignment, a technique discovered by Sinkhorn.[34] Several factors motivate the use of this algorithm. First, it enforces *two-way* assignment via *softassign*[34,35] in contrast to relaxation labeling type algorithms which enforce one-way assignment. Second, it avoids poor local minimum by the use of graduated convexity [36] continuation technique. Third, this algorithm is efficient in comparison to other existing techniques (an order of magnitude better than relaxation labeling), partly due to an explicit encoding of sparsity. Fourth, the algorithm handles missing/extra nodes, which is important in matching shapes, and is superior in this regard to other existing techniques.[10] Finally, the algorithm is stable under noisy conditions.[10] Figures 5 shows the result of correspondences as well as a similarity matrix.

Despite its advantages, however, the graduated assignment approach presents several difficulties. First, due to the quadratic nature of the energy functional only pairwise consistency of nodes is enforced. Thus, not all final correspondences preserve the coherence of shapes. The algorithm relies heavily on the quadric nature of the energy function to transform gradient descent into an assignment problem and thus cannot be easily extended. This weakness of graduated assignment to maintain the topology of the graph is treated by maximum clique approaches, but there it is not clear how to represent link and node attributes, both essential in representing shape. Second, in the graduated assignment, deformations of shapes are either treated as "noise" and implicitly represented as variations in graph attributes or are handled by the slack variables representing missing nodes. However, shape deformations alter graph structure. Given the significant role of visual transformations, we are therefore encouraged to take deformations into account explicitly. Third, graduated assignment produces a final correspondence and a similarity metric, but not a sequence of intervening shapes that optimally morph one shape into another. This paper addresses these issues by considering the optimal match as the least action path consisting of sequences of simple deformations.

In a related paper, Siddiqi *et al.*[37] present an alternative use of shock structure for shape matching. Although this approach and ours, presented initially in,[9] are similar in the use of the notions of shocks, shock grammar, and a shock graph, they are distinct in the conversion of the shock structure to a shock graph and in the graph matching approach. First, Siddiqi *et al.* use only the first and third order shock groups as nodes of the shock graph, and the links represent the time-directed relation between them. The modeling of geometrically rich shock segments (curves) by nodes (points) annihilates distinctions between endpoints, forcing an ad-hoc representation of end points of 3rd order shocks via dashed lines, and leading to instabilities with slight deformations. Also, modeling 2nd and 4th order shocks as 3rd order nodes ignores significant distinctions between these types. In contrast, we represent the isolated 2nd and 4th order shocks as nodes, and the continuous stream of first and third order shock *groups* as links. Second, in the matching of nodes Siddiqi *et al.* use an affine transformation that aligns one node to another and the degree of similarity is based on the cost of the transformation. Unfortunately, the transformations at distinct nodes, each consisting of rotation, translation, and scaling of nodes, are not dependent on each other and hence challenge the integrity of the shape and the match, *i.e.* the best local affine match may require one node to be scaled up while requiring another to be scaled down in the *same* shape. Our matching algorithm explicitly defines variables to represent these global transformations. Third, the tree matching algorithm introduced, restricts the search space by heuristic size constraints, and matches using a best-first strategy which may not recover from an erroneous match. Finally, the lack of normalization causes the distance between two graphs to be dependent on the number of nodes, making it impossible to get an accurate absolute similarity between two shapes.

4. EDIT DISTANCE

Edit distance can be used to compare two strings of possibly different lengths. The edit distance between two strings A and B is the minimum cost of a sequence of *edit operations* that transforms A to B. The allowed *edit operations* on a string are : *Insert* a symbol into a string; *Delete* a symbol from a string; *Change* one symbol appearing in a string into another symbol.

Each possible edit operation is assigned a nonnegative cost. The cost can depend on the symbols involved (and on other things as well). An *edit sequence* is a sequence of edit operations. The edit distance between two strings A and B is now the cost of the minimum cost edit sequence that takes A to B. For example, let $A = abaab$ and $B = aabb$. One edit sequence taking A to B is $(abaab \rightarrow aaab \quad \text{(delete b)}, aaab \rightarrow aabb \quad \text{(relabel b to a)})$. Assuming the cost of each operation is 1, the above edit sequence has cost 2.

We can give an equivalent definition of edit distance that makes it somewhat more obviously symmetric and allows us to *simplify* the set of edit operations. Define the edit distance between A and B to be the minimum total cost of two sequences S_A and S_B of edit operations such that applying S_A to A and applying S_B to B yields the same string C. Note that by adopting this definition, we obviate the need to include *insert* as one of our operations. The resulting measure of distance follows the diagram in Figure 1. We use the same idea in the next section when we define the edit distance between trees. By using the above idea, Wagner and Fischer[38] gave an algorithm to compute the edit distance between a string of length m and a string of length n. Their algorithm takes $O(mn)$ time. A slightly different problem, that of cyclic string edit distance is treated in Maes.[39]

We use the notion of *edit distance*, explained above in the context of string comparison, to derive a measure of distance between two shock graphs. We first observe that the shock graphs have significant exploitable structure. We define a formal measure of distance between such graphs, a measure grounded in the semantics of shock graphs, and we apply a polynomial-time algorithm to calculate the distance between two such graphs.

Because of the two-dimensional geometric nature of the shapes shock graphs represent, every shock graph possesses a *combinatorial embedding* (also called a *rotation system*), which is, loosely speaking, a combinatorial representation of a drawing of the graph on the plane. Formally, it is an assignment to each node of a cyclic ordering of the edges incident to that node. (This order corresponds to the order in which one would encounter the edges during a clockwise circumnavigation around the node.) Changing the ordering of edges around any node of a shock graph would violate the semantics of shock graphs.

Another kind of structure must also be preserved. As discussed in Section 3, when the boundary of the shape is a simple closed curve, the underlying topology of a shock graph is that of a *free* (or *unrooted*) tree. That is, it is an undirected, connected graph with no undirected cycles.

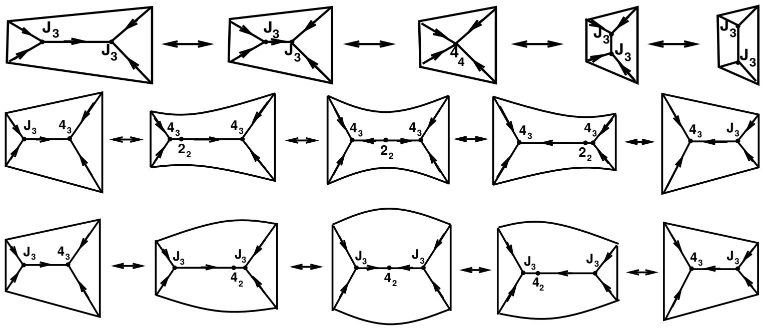

Figure 6. This figure illustrates that sequences of shape deformation consist of both *continuous* changes in graph link and node attributes as well as *discrete* structure changes in the graph itself. The discrete changes, or shock transitions, occur at the third shape of the top sequence and the second and fourth shapes of the middle and bottom sequences.

We specify a small set of elementary *edit operations*, operations that can be performed on a shock graph to alter it in such a way as to simplify the represented shape. The *cost* of an operation depends on how much the shape is altered. The distance between two shock graphs G_1 and G_2 is defined to be the minimum sum of costs of a pair of sequences S_1 and S_2 of operations such that performing S_1 on G_1 and S_2 on G_2 result in the same shock graph. Though applying edit operations to an embedded graph may result in removal of edges and nodes, it must preserve the induced embedding on the remainder of the graph.

We therefore need only define the edit distance between shock graphs that are *ordered* (in the sense of possessing a combinatorial embedding) and are *unrooted trees*. There are known polynomial-time algorithms for precisely calculating the edit distance between ordered trees.* The elementary edit operations considered by these algorithms are not quite appropriate here, but we have determined what we believe to be the correct set of allowed edit operations and have adapted the algorithm of Zhang and Shasha[41,42] to handle precisely this set. Instead, we propose five primary edit operations: *deform*, *contract*, *merge*, *splice*, and *truncate*. There are two other, secondary operations, *rotate* and *magnify*; we have not implemented these operations in the current version of our computer program, but plan to do so for the final version of this article.

We now first motivate the primary operations from the mathematics of shock transition and then describe each for shock graphs.

5. EDIT OPERATIONS FOR SHAPE

We now focus on defining canonical transformations of shape, or shape *edit operations*. Consider a shape A undergoing an arbitrary sequence of deformations to transform to shape B, and consider the intervening continuous family of shapes. The shock graph structure of this family of shapes generally remains the same while the graph attributes change. There exist however, points where the shock graph itself undergoes an abrupt change; we refer to these as *shock transitions*. Thus, the edit operations must represent both *continuous* changes in graph link and node attributes, as well as *discrete* changes in the graph structure itself, Figure 6. For example, in the top sequence where the length of the quadrilateral is decreased, the shock graph structure remains the same, but

*Most of the work in this area concerns *rooted* trees, but it is possible to adapt the algorithms to handle the case of unrooted trees.[40]

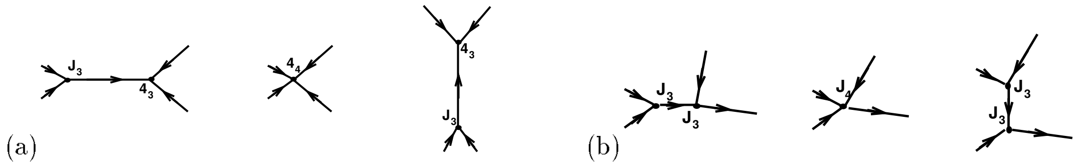

Figure 7. The A_1^4 transition of shocks,[44] where two points with tri-tangent contact come together to form a point with four points of tangency.

the link attributes (lengths, velocities, *etc.*) change. At some point (third shape), however, the graph structure changes abruptly. Thus, shapes slightly to the left or to the right of this point have distinct graph structures. A robust matching procedure in addition to establishing a correct distance when continuous changes are taking place, must recognize the small distance across these shock transitions, despite the apparently large graph distance. Our proposed solution is to consider each shape deformation as an edit operation and assign each an appropriate cost. The cost of this operation must vanish as the shapes on either side of the transition approach the transition in order to re-establish correspondence between these shapes lost due to the shock transition, thus "seaming up" the distinct equivalence classes of shape, created by equivalent shock graph structures, into a contiguous space.

Since each sequence of operations can be reversed, we will take the approach of transforming both A and B into a simpler shape C instead of transforming A into C and C into B, Figure 1. In other words the distance between A and B, $d(A, B)$ will be the minimum cost of $d(A, C) + d(C, B)$ for all shapes C. While this is theoretically not necessary, it is algorithmically more efficient since there are always fewer simpler shapes. Informally, shapes with a high degree of degeneracy like squares serve as "traffic hubs" for minimal paths between shapes, *e.g.*, quadrilaterals. In other words, a rectangle is described as a stretched square and not vice-versa.[43] We now address edits arising from shock transitions.

Edits Arising from Shock Transitions: While edit operations corresponding to discrete changes in the graph structure may be experimentally observed and quantified, Figure 6, or intuitively derived, the question naturally arises as whether the set of all edits can be formally enumerated. Fortunately, a recent effort by Giblin and Kimia[44] has led to the classification of all transitions of 2D shocks as special cases of the transitions of the symmetry set.[45] † The shock transitions themselves are not generic in isolation, but are generic in a family of deformations. Readers not interested in the mathematical motivation can refer to Figure 11 for a summary of graph transformations.

We first consider the A_1^4 transition, where four branches coincide. This transition occurs in two flavors, Figure 7, depending on whether all branches flow in, or all but one flow in, the only two valid possibilities.[44] Thus, the edit operation corresponding to the A_1^4 transition is

contract: "contract a nearby $(J_3, 4_3)$ pair into a 4_3 or a nearby (J_3, J_3) into a J_4".

Second, consider the A_1A_3 shock transition where a continuous segment of a shock group breaks into two pieces and grows a new branch, Figure 8. This occurs frequently with slight variations (protrusions and indentations) in the boundary. The edit operation corresponding to this is again a transformation from complex to simple leading to

splice: "remove the spurious branch and merge the remaining shock groups as one".

Third, as in the above sequence the protrusion gets larger, at some point the J may change to a 4 shock by giving birth to a 2 shock Figure 9. Formally, the third transition occurs at infinite velocity shocks with zero acceleration which give rise to a pair of 4th and 2nd order shocks. Thus, the edit operation that simplifies shape

†These transitions occur at $\{A_1^4, A_1^2A_2, A_2^2 - moth, A_2^2 - nib, A_1A_3, A_4\}$ points, but only manifest themselves at $\{A_1^4, A_1A_3\}$ and at infinite velocity shocks with zero acceleration. The A_k^n notation describes the nature of tangency of the maximal circle at each shock point. The number n is the number of tangent points while k describes the order of tangency, *i.e.*, $k = 1$ means regular tangency, $k = 3$ implies a vertex.

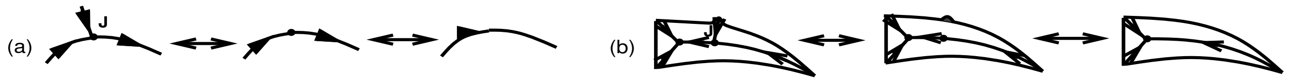

Figure 8. The A_1A_3 shock transition.

merges a pair of 2 and 4, leading to

merge 1: "merge a pair of consequent 2 and 4, and merge all three consequent shock segments into one".

Figure 9. The introduction of a pair of 4 and 2 in a deformation sequence.

Fourth, a transition occurs when a shock meets both the infinite velocity condition and the A_1^3 condition. While branch points generally do not have infinite velocity, this can happen at a shock transition in a family of deformations in three ways, depending on whether the A_1^3 points is a 4 or a J, Figure 10. These define two more

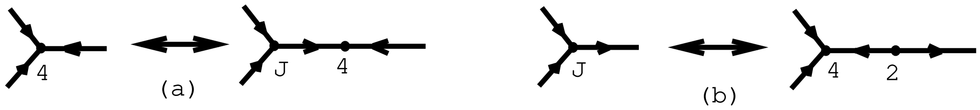

Figure 10. The two possibilities for an A_1^3 plus $v = \infty$ transition.

merge operations.

merge 2: Merge adjacent 4 and J_3 nodes into 4_3
merge 3: Merge adjacent 4_3 and 2 nodes into a J_3

The edit operations can easily be translated as a graph transformation, Figure 11. We also include a truncate operation to deal with non-continuous (abrupt) deformations of shapes, *e.g.*, those that arise from occlusion. Finally, we represent continuous changes by the edit operation. *Deform* which is applied to an edge, and alters the trajectory of shocks lying on the edge without introducing discontinuities. Note that this operation does not change the topology or embedding of the shock graph. We should note that in a related approach, Yuille[46] defined and applied *skeleton operators* which resemble our edit operations. However, these were motivated to treat the unreliability of the skeleton and applied whenever matching residuals were detected.

6. EXPERIMENTAL RESULTS

We have implemented an algorithm to compute the edit distance between two shock graphs using a subset of our edit operations *merge, splice, contract, deform.* Figure 12 shows and describes the behaviour of the algorithm on an example shock-graph pair. Work is progressing on a complete implementation of the algorithm; in a subsequent paper we will present more comprehensive results and a comparison with the graduated-assignment approach.

REFERENCES

1. K. Hirata and T. Kato, "Query by visual example - content based image retrieval," *Advances in Database Techology* , pp. 56–71, 1992.
2. A. D. Bimbo, P. Pala, and S. Santini, "Visual image retrieval by elastic deformation of shapes," in *Proceedings of IEEE VL '94, International Symposium on Viusal Languages*, 1994.

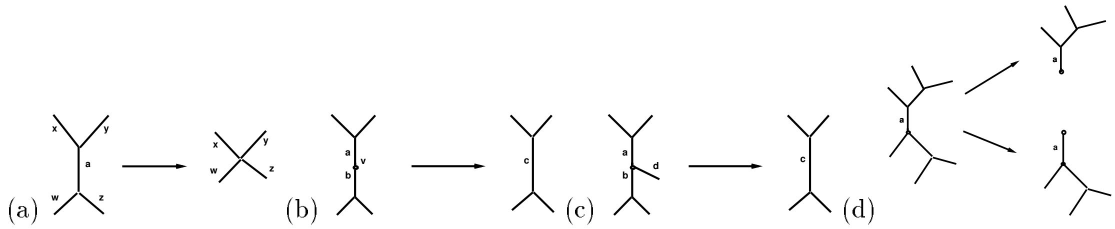

Figure 11. (a) *Contract* is applied to an edge; it shrinks the edge to a node, merging the edge's endpoints as shown in Figure contracted only if both its endpoints have degree 3 (this restriction arises from the semantics of shock graphs). Contraction is a standard graph operation, and one can readily check that the embedding of a graph is preserved under the contraction of an edge. (b) *Merge* is applied to two edges sharing a node v such that no other edges are incident to v. (Note that v represents a discontinuity in the shock trajectory.) The operation merges the two edges, obtaining a single edge and thereby removing the discontinuity. The embedding is preserved. (c) *Splice* resembles *Merge*, but applies to three edges sharing a vertex v. One of the three edges must have a degree-one node as its other endpoint; *splice* removes this edge and merges the other two. (d) *Truncate* is applied at an edge and removes the whole subtree on one side of the edge (not including the edge itself).

3. R. Mehrotra, F. Kung, and W. Grosky, "Industrial part recognition using a component index," *Image and Vision Computing* **3**, pp. 225–231, 1990.
4. Z. Lei, Y. Chan, and D. Lopresti, "Curvelet feature extraction and matching for content-based image retrieval," in *1997 IEEE International Conference on Image Processing*, (Santa Barbar, CA), October 1997.
5. W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutos, and G. Taubin, "The QBIC project: Querying images by content using color texture and shape," Tech. Rep. 9203, IBM Research Division, 1993.
6. S. Sclaroff, "Deformable prototypes for encoding shape categories in image databases," *Pattern Recognition* , 1997. submitted.
7. P. Gamba and L. Lombardi, "Shape analysis by means of the bounadry integral resonant mode expansion method," pp. 227–236, May 1997.
8. A. D. Bimbo, P. Pala, and S. Santini, "Image retrieval by elastic matching of shapes and image patterns," in *Proceedings of MULTIMEDIA '96*, pp. 215–218, 1996.
9. B. B. Kimia, J. Chan, D. Bertrand, S. Coe, Z. Roadhouse, and H. Tek, "A shock-based approach for indexing of image databases using shape," in *Proceedings of the SPIE's Multimedia Storage and Archiving Systems II*, vol. 3229, pp. 288–302, (Dallas, Texas), November 1997.
10. S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18(4)**, pp. 377–388, 1996.
11. B. B. Kimia, "Conservation laws and a theory of shape," Ph.D. dissertation, McGill Center for Intelligent Machines, McGill University, Montreal, Canada, 1990.
12. B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker, "Nonlinear shape approximation via the entropy scale space," in *Proceedings of the SPIE's Geometric Methods in Computer Vision II*, vol. 2031, pp. 218–233, (San Diego, California), July 1993.
13. B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker, "Shapes, shocks, and deformations, I: The components of shape and the reaction-diffusion space," *IJCV* **15**, pp. 189–224, 1995.
14. D. Sharvit, J. Chan, and B. B. Kimia, "Symmetry-based indexing of image databases," in *Workshop on Content-Based Access of Image and Video Libraries, CVPR98*, Submitted, January 1998.
15. D. Sharvit, J. Chan, and B. B. Kimia, "Symmetry-based indexing of image databases," *Journal of Visual Communication and Image Representation* , To be Submitted, February 1998.
16. B. B. Kimia and K. Siddiqi, "Geometric heat equation and nonlinear diffusion of shapes and images," *Computer Vision Graphics and Image Processing: Image Understanding* **64**, pp. 305–322, November 1996.

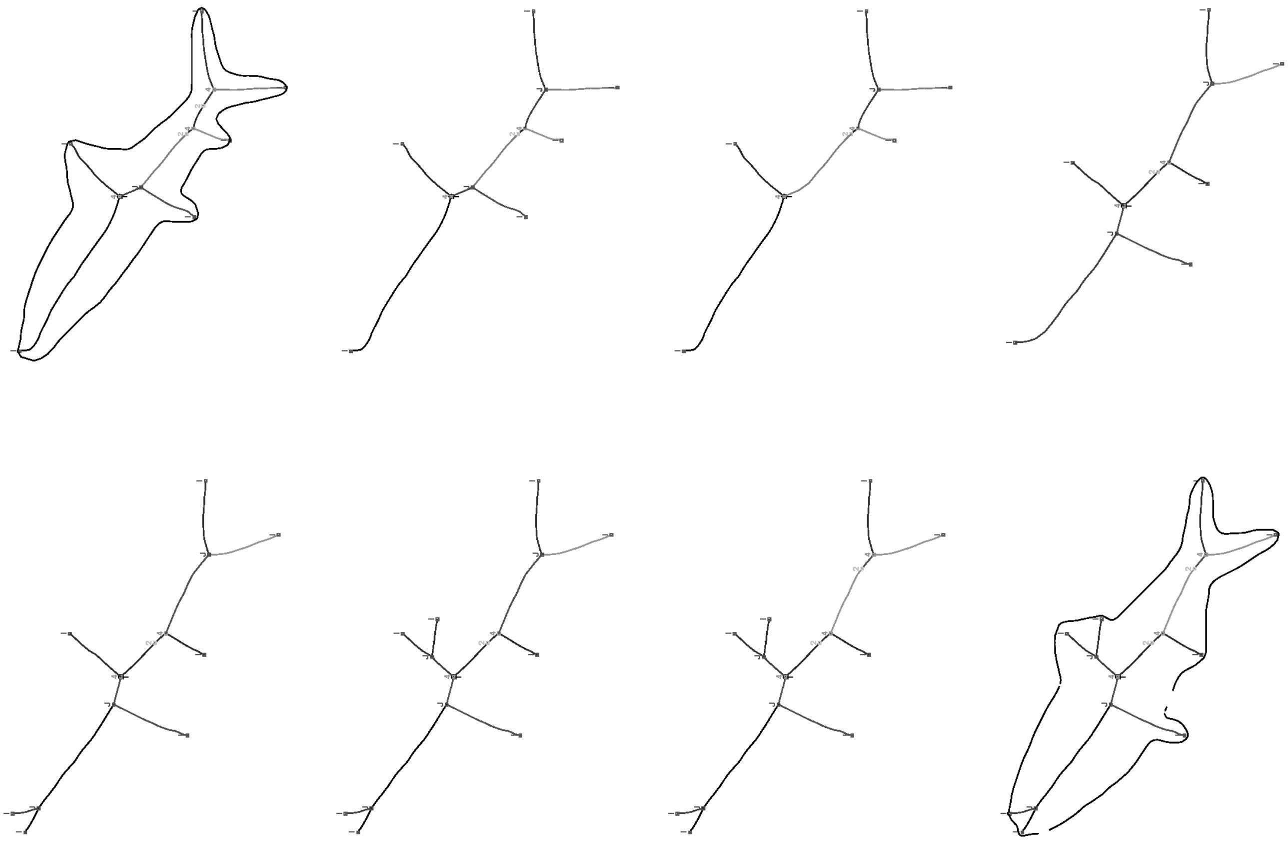

Figure 12. A sequence of edit operations transforming one fish into another. The first and last shapes are the inputs to the algorithm; the intermediate shapes form the edit sequence.

17. L. Younes, "Computable elastic distance between shapes," *SIAM Journal of Applied Mathematics* , 1996.
18. R. N. Shepard, "Toward a universal law of generalization for psychological science," *Science* , pp. 1317–1323, 1987.
19. R. Beals, D. H. Krantz, and A. Tversky, "The foundations of multidimensional scaling," *Psychological Review* **75**, pp. 127–142, 1968.
20. B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker, "Toward a computational theory of shape: An overview," in *Proceedings of the First European Conference on Computer Vision*, pp. 402–407, Springer Verlag, (Antibes, France), 1990.
21. K. Siddiqi and B. B. Kimia, "A shock grammar for recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 507–513, IEEE Computer Society Press, (San Francisco, California), June 1996.
22. B. B. Kimia, "Curve evolution, wave propagation, and mathematical morphology," in *Fourth International Symposium on Mathematical Morphology (ISMM'98)*, June 1998.
23. H. Tek, P. Stoll, and B. B. Kimia, "Shocks from images: Propagation of orientation elements," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society Press, (Puerto Rico), June 15-16 1997.
24. K. Siddiqi, B. Kimia, A. Tannenbaum, and S. Zucker, "Shocks, shapes, and wiggles," in *Proceedings of the International Workshop on Visual Form*, World Scientific, (Capri, Italy), May 1997.

25. L. G. Shapiro and R. M. Haralick, "Structual decriptions and inexact matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **3**, pp. 504–519, Sept. 1981.
26. M. A. Eshera and K. S. Fu, "An image understanding system using Attributed Symbolic Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**, pp. 604–618, Sept. 1986.
27. M. A. Eshera and K. S. Fu, "A graph distance measure for image analysis," *IEEE Transactions on Systems, Man, and Cybernetics* **14**, pp. 398–408, 1984.
28. M. A. Eshera and K. S. Fu, "A measure of similarity between attributed relational graphs for image analysis," in *Seventh International Conference on Pattern Recognition (Montreal, Canada, July 30-August 2, 1984)*, IEEE Publ. 84CH2046-1, pp. 75–77, IEEE, IEEE, 1984.
29. R. Hummel and S. W. Zucker, "On the foundations of relaxation labeling processes," *IEEE Trans. Pattern Analysis and Machine Intelligence* **6**, pp. 267–287, 1983.
30. S. Umeyama, "An eigendecomposition aproach to weighted graph matching problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **10**, pp. 695–703, September 1988.
31. P. D. Simic, "Constrained nets for graph matching and other quadratic assignment problems," *Neural Computation* **3**, pp. 268–281, 1991.
32. H. A. Almohamad and S. O. Duffuaa, "A linear programming approach for the weighted grapd matching problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**, pp. 522–525, May 1993.
33. A. Rangarajan and E. Mjolsness, "A Lagrangian relaxation network for graph matching," in *IEEE International Conference on Neural Networks(ICNN)*, vol. 7, pp. 4629–4634, IEEE Press, 1994.
34. R. Sinkhorn, "A relationship between arbitrary positive matrices and doubly stochastic matrices," *Ann. Math. Statist.* **35**, pp. 876–879, 1964.
35. J. J. Kosowsky and A. L. Yuille, "The invisible hand algorithm: Solving the assignment problem with statistical physics," *Neural Networks* **7**, pp. 477–490, 1994.
36. A. Blake and A. Zisserman, *Visual Reconstruction*, MIT PRess, Cambridge, MA, 1987.
37. K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker, "Shock graphs and shape matching," *ICCV* , pp. 222–229, 1998.
38. R.A.Wagner and M.J.Fischer, "The string-to-string correction problem," *J. Association Computing Machinery* **21**(1), pp. 168–173, 1974.
39. M. Maes, "On a cyclic string-to-string correction problem," *Information Processing Letters* **35**, pp. 73–78, June 1990.
40. P. Klein, "Computing the edit distance between unrooted ordered trees," in *Proceedings, 6th European Symposium on Algorithms*, pp. 91–102, 1998.
41. K. Zhang and D. Shasha, "Simple fast algorithms for the editing distance between trees and related problems," *SIAM Journal On Computing* **18**, pp. 1245–1262, December 1989.
42. D. Shasha, *Pattern Matching Algorithms*, ch. Approximate Tree Pattern Matching. Oxford University Press, June 1997.
43. M. Leyton, *Symmetry, Causality, Mind*, MIT press, April 1992.
44. P. J. Giblin and B. B. Kimia, "On the local form and transitions of symmetry sets, and medial axes, and shocks in 2D," Tech. Rep. LEMS-170, LEMS, Brown University, April 1998.
45. P. J. Giblin and S. A. Brassett, "Local symmetry of plane curves," *The American Mathematical Monthly* **92**, pp. 689–707, December 1985.
46. S. C. Zhu and A. L. Yuille, "Forms: A flexible object recognition and modeling system," *Int'l Journal of Computer Vision* **20**(3), 1996.