

Symmetry-based Indexing of Image Databases

Daniel Sharvit, Jacky Chan, Hüseyin Tek and Benjamin B. Kimia

LEMS, Division of Engineering
Brown University, Providence RI 02912
{des,jcc,tek,kimia}@lems.brown.edu
<http://www.lems.brown.edu/~des>

Abstract

The use of shape as a cue for indexing into pictorial databases has been traditionally based on global invariant statistics and deformable templates, on the one hand, and local edge correlation on the other. This paper proposes an intermediate approach based on a characterization of the symmetry in edge maps. The use of symmetry matching as a joint correlation measure between pairs of edge elements further constrains the comparison of edge maps. In addition, a natural organization of groups of symmetry into a hierarchy leads to a graph-based representation of relational structure of components of shape that allows for deformations by changing attributes of this relational graph. A graduate assignment graph matching algorithm is used to match symmetry structure in images to stored prototypes or sketches. The results of matching sketches and grey-scale images against a small database consisting of a variety of fish, planes, tools, etc., are promising.

Acknowledgments: We gratefully acknowledge the support of the National Science Foundation grant IRI-9700497.

1 Introduction

Shape is a significant cue for queries into pictorial databases, yet its potential in practical systems has not been fully explored. The QBIC systems relied on invariant statistics such as shape moments, eccentricity, *etc.* [24]. In another approach user-drawn sketches are matched against images [11, 2], *e.g.*, in QVE, the image edges are first abstracted through a series of image size reduction, media filtering, coarse edge detection, and global pruning, and then matched against the sketch using a correlation-based similarity score. Gray [10] evaluated this approach and noted its sensitivity with variations of the sketch from the underlying shape and pointed to the need to include deformations in this process. Mehrotra and Grosky matched a chain code representation of edges for retrieval [23]. Del Bimbo [2, 3] in a series of papers advocated the use of deformations and elastic matching both of 1D representations and 2D templates. Lei and Cooper use [20] algebraic invariants of patches of long contours extracted from the image. Sclaroff [28] uses modal matching [29] to measure distance from shape prototypes. In summary, the majority of these approaches have either relied on (i) global methods such as invariant statistics, shape measures, and deformable templates, on the one hand, or (ii) local edge similarity, on the other.

In this paper we propose that a symmetry-based representation is an intermediate representation that retains the advantages of local, edge-based correlation approaches as well as of global, deformable models, and in addition offers two distinct benefits. First, in the computation of a similarity score between two edge maps, the similarity of the symmetry set of edge maps offers further discrimination by encoding relational structures of pairs of edge elements. Second, the hierarchical relations of symmetries captured in a graph structure offers global discrimination power. The use of symmetry in indexing, however, is faced with several difficult challenges. First, the use of traditional skeletons, as loci of symmetries of a shape, requires a difficult pre-segmentation of an image into figure and ground. Recent approaches [36, 25, 39, 42] aim to recover symmetries directly from images. We use the approach of propagating edge elements to recover symmetries directly from images [38, 39, 37]. Second, the traditional view of skeletons does not lead to a natural hierarchical organization of symmetries. In contrast, shocks, or singularities arising from the dynamic evolution of shape, augment the traditional skeleton with a notion of direction, speed, order, and type, as well as a shock grammar which governs the dynamic relationship between shock types and shock groups. We utilize this structure and create an Intrinsic Shock Graph as initially proposed in [13]. An alternative use of symmetry of shape in graphical form is presented in [30], but this requires a segmented shape. The third challenge is the matching of these graphs under *both* similarity transformations (rotation, translation, and scaling), and non-rigid deformations (bending, stretching, *etc.*). We adapt the “graduated assignment” technique [8] to our domain involving both discrete variables representing the relational structure and continuous variables representing arbitrary deformations as graph attributes and similarity parameters. Fourth, the matching should naturally lead to the formation of categories and prototypes. Our proposed notion of similarity naturally leads to multiple levels of structural similarity (graph structure), parametric similarity (rough description of each node and link), and metric similarity (exact distance) which in turn leads to a hierarchical categorization of

shape. The focus of this paper is on the matching of sketches and grey-scale images against a database of shapes including fish, planes, rabbits, tools, *etc.*

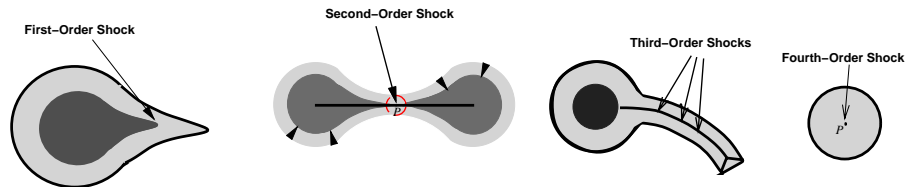


Figure 1: Each of the four type of shocks each is correlated with a perceptual/semantic category, *i.e.*, *protrusion*, *part*, *bends* and *seed*.

2 Representation of Symmetry as Shocks

In this section we review an approach to symmetry based on the notion of *shocks* of a shape [17, 15], and briefly describe methods for detection, classification, and organization of shocks according to a shock grammar [32], and for extraction from realistic grey-scale images [38, 39]. Shape can be completely described as the collection of four types of shocks that arise in the course of deformations of shape, canonically captured in the reaction-diffusion space $\frac{\partial C}{\partial t} = (\beta_0 - \beta_1 \kappa) \vec{N}$ [14, 17, 32]. In other words, shape is evolved and its singularities are recorded. First-order shocks occur when the orientation is discontinuous, whereas in second-order shocks there is curvature discontinuity and a change in the shape topology. Third-order shocks represent higher-order contact as entire pieces of curves collide at the same time (degenerate) and at fourth-order shocks the entire curve disappears (several papers relating to shock theory can be obtained at <http://www.lems.brown.edu/vision/publications>). The four types of shocks, Figure 1, correspond to intuitive elements of shape, namely, parts, protrusions, and bends [16], Figure 2. The set of shocks, implemented to sub-pixel accuracy [32], forms a complete description of the shape, Figure 5.

Shock Type	Orientation	Curvature
First-Order	non- vanishing $\nabla\phi$	high level set curvature
Second-Order	isolated vanishing $\nabla\phi$	$\kappa_1 \kappa_2 < 0$
Third-Order	non-isolated vanishing $\nabla\phi$	$\kappa_1 \kappa_2 = 0$
Fourth-Order	isolated vanishing $\nabla\phi$	$\kappa_1 \kappa_2 > 0$

Table 1: This table depicts the classification of shock types based on the the gradient level set curvature and the principal curvatures of the surface.

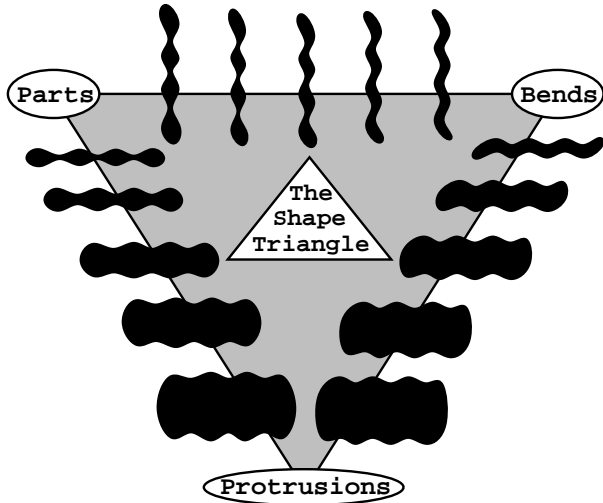


Figure 2: The sides of the shape triangle represent continua of shapes; the extremes correspond to the “parts”, “bends” and “protrusions” nodes [16].

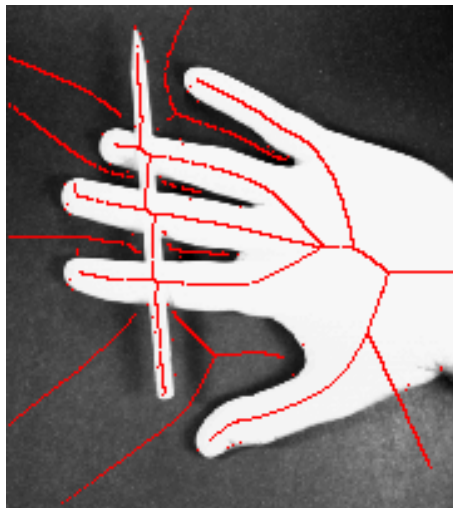


Figure 3: The extraction of symmetries directly from grey-scale images without requiring a pre-segmentation of the image

Figure 3 shows the results of a novel alternative approach [38, 39] for computing shocks that tremendously improves computation time and accuracy, and more significantly allows for the computation of shocks *directly* from images, thus avoiding the pre-segmentation step. This is accomplished by the propagation of orientation elements and a further classification of shocks into *regular*, *semi-degenerate*, and *degenerate* [38, 39], Figure 4.

While the derived shock structure when $\beta_1 = 0$ can be reduced to traditional skeletons, the shock based representations offers distinct and significant advantages, primarily due to notions of shock order, velocity, shock grouping via a grammar, salience, and time-derived shock hierarchy. Specifically, *(i)* certain deformation, *e.g.*, bending, affect selective shock groups, *e.g.*, the third-order shock group of a rectangle; *(ii)* shock order generates an explicit dimension for qualitative approximation; *(iii)* topological and differential properties of shocks, *e.g.*, velocities directly reflect boundary properties; *(iv)* the notion of the time of formation of a shock induces a hierarchy on shocks; *(v)* spurious shocks can be detected via violations of a shock grammar without regularization.

The formation of shocks is not arbitrary and the sequences of shocks in the composition of a group can be concisely described via a shock grammar, SG, which was introduced in [32] and summarized in Table 2.

3 Graph Representation of Shock Structure

The shock hierarchy induced by the formation of shocks in time and the shock grammar naturally leads to a graph-based representation which embeds geometrical and topological constraints. We proceed by first describing the information content stored for each individual shock represented in the extrinsic image coordinates (Extrinsic Shock Set), thus leading to

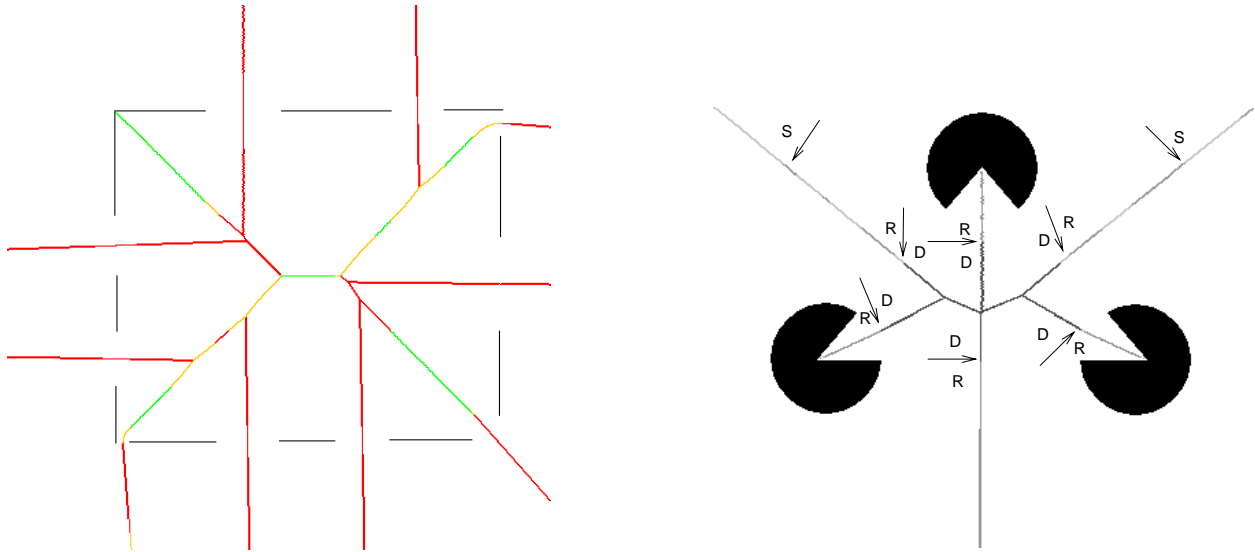


Figure 4: The skeletons of partial contours do not bear much resemblance to those of complete contours, leading to a fundamental instability in computing them from edge maps. Shock labels R=Regular, D=Degenerate, S=Semi-degenerate (shown in green, red, and yellow, respectively), however, allow the recovery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

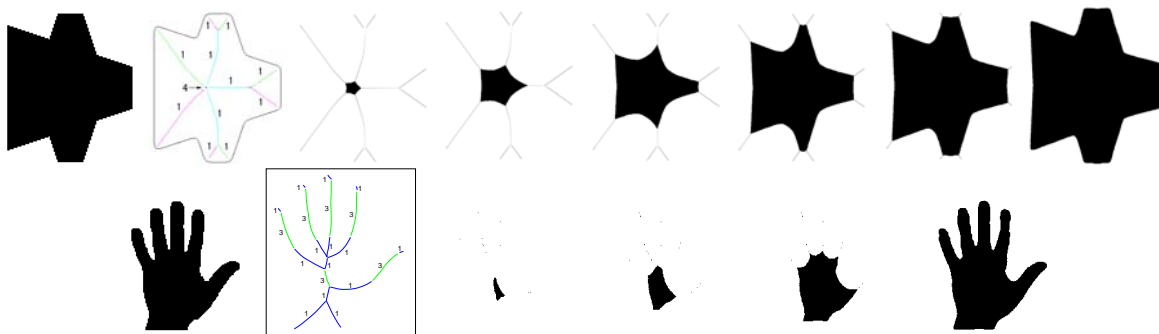


Figure 5: The shock-based description of a shape composed of trapezoids, as a hierarchy of merged protrusions, and its growth from shocks. The shock-based description of a biological shape as five bends (the fingers) attached to protrusions (the latter describing the palm), and its growth from shocks [32].

Shock Grammar

- $V = \{S_1, S_2, S_{\bar{3}}, S_4, S_I, S_T, E\}$. The symbols S_1, S_2, S_4 represent first-, second-, and fourth-order shocks. Since third-order shocks never appear in isolation, a group of third-order shocks is itself an element of the alphabet, denoted by $S_{\bar{3}}$. S_I is a start symbol, S_T is a terminal symbol, and E represents the end of a growing shock sequence.
- $\Sigma = \{S_T\}$.
- $R =$

$$\{S_I \rightarrow S_1E, S_I \rightarrow S_2E, S_I \rightarrow S_{\bar{3}}E, S_I \rightarrow S_4,$$

$$S_1E \rightarrow S_1S_1E, S_1E \rightarrow S_1S_{\bar{3}}E, S_1E \rightarrow S_4,$$

$$S_2E \rightarrow S_2S_1E,$$

$$S_{\bar{3}}E \rightarrow S_{\bar{3}}S_1E, S_{\bar{3}}E \rightarrow S_{\bar{3}}S_T,$$

$$S_4 \rightarrow S_4S_T\}.$$

The symbol E represents the end of a shock group that is being generated, and is used to enforce the requirement that shocks be added only to that end. This reflects the notion of time in the evolution of the shape, making the grammar context dependent.

Table 2: A summary of shock grammar [32].

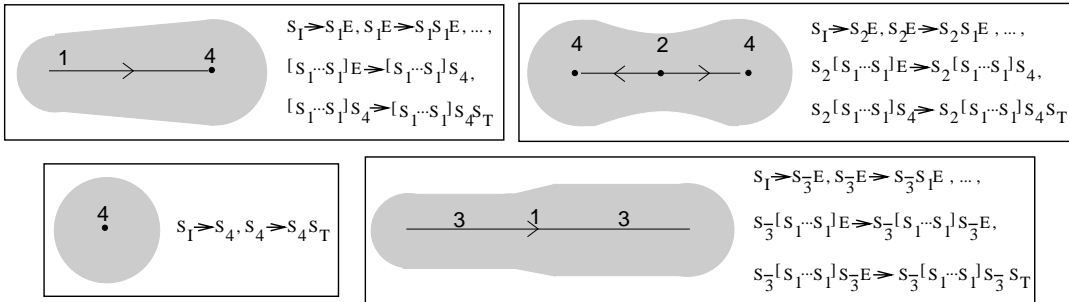


Figure 6: This figure illustrates the application of some of the rules of SG [32]. Note that whereas the grammar suffices to describe the composition of a shock group, it does not reflect the geometric and topological constraints which shocks satisfy. These relations can be represented by embedding the grammar in a graph.

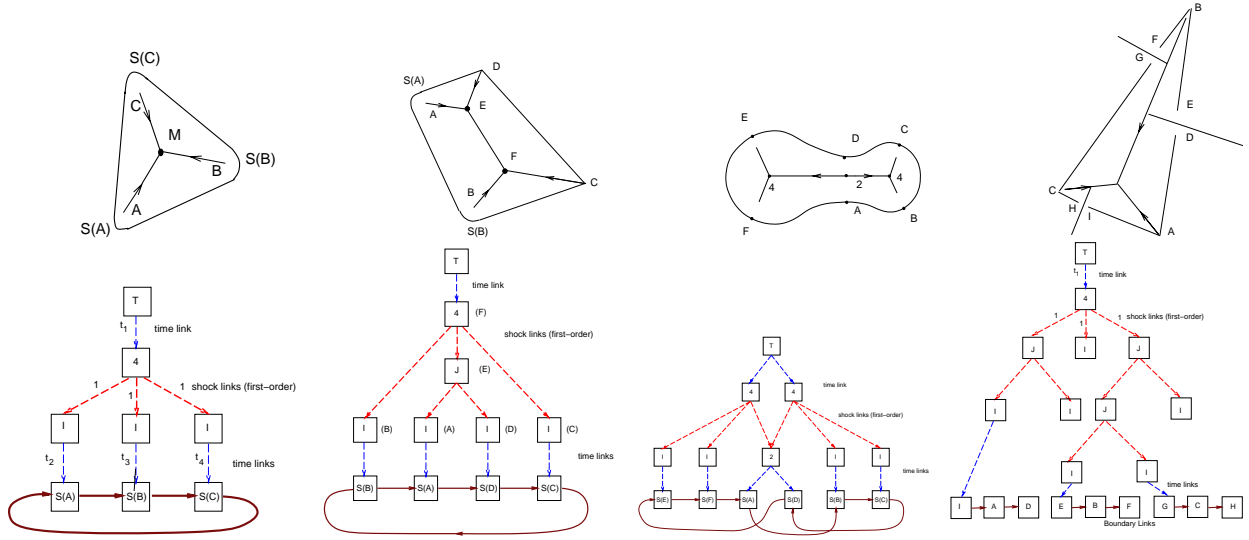


Figure 7: This figure sketches the link and node definitions for some simple shapes. The triangular shape is represented by a terminal node which gives birth to a single fourth order shock (seed) after some delay (t_1) which then grows using information in each of the three 1st-order shock group links. The quadrilateral illustrates a branch point. Note that the hierarchy is established by the flows of shocks in time. The “dumbbell” shape illustrates a 2nd order shock, where shape is decomposed into parts. The triangular shape with gaps in the boundary shows that the approach is application not only to segmentable shapes but also edge maps.

a formal definition of a graph-based representation of shocks represented in shape-based coordinates (Intrinsic Shock Graph). It should be noted that the graph structure does not rely on a priori figure/ground segmentation.

Definition 1 *The Extrinsic Shock Set is the set of individual shocks, each characterized by order $\in \{1, 2, 3, 4\}$, label $*$ $\in \{R=\text{regular}, S=\text{semi-degenerate}, D=\text{degenerate}\}$, location (x, y) , time t , and velocity vector(s) $\{\vec{v}_i\} = \{(\theta_i, v_i)\}$. Define a Joint (J) as a shock point which connects two distinct shock groups not in $\{2, 4\}$. Define an Initial shock (I) as the first shock invoked by the start symbol. Define as a boundary source the boundary point (S) or the set of boundary points (\bar{S}) which form an initial shock (I), an (initial) 2nd order shock, or an initial 4th order shock. Let T denote the terminal node.*

The shock grammar [32] allows us to group individual first and third-order shocks into shock groups, naturally leading to a hierarchical structure which we use as the primary representation for a characteristic view of an object.

Definition 2 (*Intrinsic Shock Graph*). Let $\{T, 4, 2, I, J, S\}$ represent nodes. Let 1st order and 3rd order shock groups represent shock links between nodes $\{4, 2, J, I\}$, represented intrinsically by the curvature function of the link and the acceleration of the shock along this

*label represent additional classification of shocks [39].

Nodes/links	Information stored
Terminal node(T):	$((x_0, y_0), \theta_0, \lambda)$ (translation, rotation, scaling)
4th order node (4) :	$(label, x, y, t, \{\theta_i, v_i\})$
2nd order node (2):	$(label, x, y, t, \theta, \infty)$
Joint node(J):	$(label, x, y, t, \{\theta_i, v_i\})$
Initial node(I):	$(label, x, y, t, \theta, v)$
Boundary source node (S):	$(label, x, y, 0, \{\theta_i, \infty\})$
shock links	$\{\kappa(s), s \in [0, 1], L\}$ and $\{a(t), t \in [0, 1], T\}$
time links	t
boundary links	$\{\kappa(s), s \in [0, 1], L\}$

Table 3: This table summarizes the types of nodes and links of the *intrinsic shock graph* together with the information stored for each. Note that each node’s coordinates are relative to the parent and where the root node represents global shape information, namely, location, orientation, and scale, which are subsequently inherited by children nodes and used in the match process. Observe the intrinsic representation of the graph.

path. Let the connection between node pairs $\{(T, 4), (T, J)\}$, and (I, S) , be represented by time links (t). Let S nodes be connected by boundary links to represent the boundary segment between two S -nodes, each represented by a curvature function. The resulting graph is defined as the Intrinsic Shock Graph.

Figure 7 illustrates the intrinsic shock graph for a few simple shapes. Note that the graph describes the relational/structural aspect of the object as well as an intrinsic representation of metric information for each “component”, via curvature and acceleration functions as attributes to the shock links, which is not shown in figure. The segregation of shape information along shock groups into “geometry” of the link represented by a curvature function and “dynamics” of the shocks traveling along the link as represented by an acceleration function is essential to properly relate a shape as a deformation of another shape: The bending of structure affects curvature, while acceleration “fattens” or “thins” the shape, Figure 9. Each node is also represented intrinsically with respect to parent nodes. Finally, the root node (T) contains global shape information, *i.e.*, location (x_0, y_0) , orientation (θ_0) , and scale (λ) . All other nodes inherit this information from parent nodes to determine the best parameters during the match. Table 3 summarizes the information stored at each node and link. We now describe a method for matching query graphs to those stored in the database.

4 Shock Graph Matching by the Graduated Assignment Algorithm

Graphs are powerful data structures which describe the relationship among abstracted structure. Shapiro and Haralick [31] were among the pioneers in their use in structural description of objects in images via weighted graphs. Fu [7] 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 [5, 6] 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 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's states, which grows very rapidly with the number of nodes.

The computational intractability of graph matching as an NP-complete problem has led to the development of several classes of algorithms. *Search-Oriented* methods explore the shortest path in the state space, *e.g.*, via branch and bound methods [19]. These methods require heuristics to reduce exponential time complexity (worst case) to a low-order (2, 3) polynomial in the number of nodes (l) and links (m). Another class of algorithms is based on *nonlinear optimization* which does not explicitly search the state space and its computational complexity is typically linear in the number of nodes/links, *e.g.*, *relaxation labeling* [12]. However, these methods only enforce *one-way* constraints. Eigenvalue decomposition works well when a pair of weighted graphs are nearly isomorphic, and while their combination with hill-climbing improves matching, poor local minima can often result [40]. Other types of techniques for graph matching include the use of neural networks [34], linear programming [1] and Lagrangian Relaxation [26].

We modify the *graduated assignment algorithm* [8] to implement shock graph matching. Several factors motivate the use of this algorithm: First, it enforces *two-way* assignment via *softassign* [35, 18] in contrast to relaxation labeling type algorithms which enforce one-way assignment. Second, it avoids poor local minimum by the use of graduated convexity [4, 41] continuation technique. Third, this algorithm is efficient in comparison to current techniques (an order of magnitude better than relaxation labeling), partly due to an explicit encoding of sparsity. Fourth, the algorithm handles missing/extra nodes/links, which is important in matching shapes, and is superior in this regard to other existing techniques [8]. Fifth, the algorithm is stable under noisy conditions [8]. Finally, the formulation can be adapted to also take into account continuous variables representing similarity transformations and shape deformations.

We now briefly describe the graduated assignment algorithm [9]. 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, Figure 8. Slack rows and columns are added to the match matrix to represent missing/extra nodes. In order to allow for differential movement for one permutation to another, 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) discretization. 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 [35].

Formally, consider two graphs G and g . Refer to nodes of G and g by G_a and g_i , respectively, and links of G and g by G_{ab} and g_{ij} , respectively, where $a, b = 1, \dots, A$, and $i, j = 1, \dots, I$. The match matrix M is defined by $M_{ai} = 1$ if the node $a \in G$ corresponds to

Begin A: (Do A until $\beta > \beta_f$)
Begin B: (Do B until M converges or # of iterations $> I_0$)
 $Q_{ai} \leftarrow -\frac{\partial E_{wg}}{\partial M_{ai}}$
 $M_{ai}^0 \leftarrow \exp \beta Q_{ai}$
Begin C: (Do C until \hat{M} converges or # of iterations $> I_1$)
 Update \hat{M} by normalizing across all rows:
 $\hat{M}_{ai}^1 \leftarrow \frac{\hat{M}_{ai}^0}{\sum_{i=1}^{I+1} \hat{M}_{ai}^0}$
 Update \hat{M} by normalizing across all columns:
 $\hat{M}_{ai}^0 \leftarrow \frac{\hat{M}_{ai}^1}{\sum_{a=1}^{A+1} \hat{M}_{ai}^1}$
End C
End B
 $\beta \leftarrow \beta_r \beta$
End A

Table 4: A summary of the graduated assignment algorithm from [9]

node $i \in g$, and $M_{ai} = 0$ otherwise. In graduated assignment [9] an energy (objective function) $E(M)$ is defined for each possible M . To differentially move from one permutation to another, M_{ai} is allowed to take values between 0 and 1. However, it is necessary to maintain $\sum_a M_{ai} = 1$ and $\sum_i M_{ai} = 1$, i.e. M needs to remain a doubly stochastic matrix [35]. The Taylor Expansion of this energy function

$$E(M) = E(M^0) - \sum_{a=1}^A \sum_{i=1}^I Q_{ai}(M_{ai} - M_{ai}^0) , \quad (1)$$

where $Q = -\frac{\partial E}{\partial M}(M^0)$, turns the energy minimization into maximizing $\sum_{a=1}^A \sum_{i=1}^I Q_{ai}M_{ai}$, an assignment problem [9]. This is, in turn, solved by *softassign* where an initial matrix is moved towards a solution by increasing a parameter β which controls the convexity of the energy landscape. Therefore, the algorithm follows three nested iterations (*i*) for each β and M find $Q(M)$ (*ii*) update M through *softassign*; (*iii*) the procedure is repeated until convergence and β is increased; the latter is repeated until convergence, Table 4.

Having outlined the algorithm, it remains to define E in a meaningful manner. Gold and Rangarajan give a generic definition for attributed relational graphs (ARGs) as

$$E(M) = \sum_{a=1}^A \sum_{i=1}^I \sum_{b=1}^A \sum_{j=1}^I M_{ai}M_{bj}L_{aibj} - \alpha \sum_{a=1}^A \sum_{i=1}^I M_{ai}N_{ai} , \quad (2)$$

where L_{aibj} represents total similarity between links G_{ab} and g_{ij} , and N_{ai} represents total similarity between nodes G_a and g_i ,

$$L_{aibj} = \begin{cases} 1 & \text{if links } G_{ab} \text{ and } g_{ij} \text{ both exist} \\ Null & \text{otherwise} \end{cases} \quad (3)$$

$$N_{ai} = \begin{cases} 1 & \text{if nodes } a \text{ and } i \text{ match} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

This leads to,

$$Q_{ai}(M) = 2 \sum_{b=1}^A \sum_{j=1}^I M_{bj} L_{aibj} + \alpha N_{ai} . \quad (5)$$

This energy function allows robust matching of two arbitrary graphs. The Intrinsic Shock Graph can be matched *structurally*, by simply considering whether a link exists. To motivate this approach, consider that the aim of indexing by shape is to recognize objects from characteristic views. While changes in viewing parameters have been factored out by defining affine (projective) invariants [22], these work well for fairly flat objects viewed at a distance and do not work well in the presence of modest partial occlusion and articulation, which affect the projected boundaries. However, what survives a rather broad range of visual transformations is the *relationship* among the various components of the shape. This is precisely what the above energy minimization task is achieving and we refer to it as *Structural Matching*.

This form of matching, however, does not take into account the various transformations a shape undergoes. While the match is effective in pruning unlikely matches and forming categories, shapes within a category cannot be ranked by a structural match. The matching strategy must therefore take account of shape transformation. We consider two classes of shape deformations, (i) rigid transformations (translation and rotation) which together with scaling form the class of *similarity transformations*; (ii) and non-rigid shape deformations which are intuitively described as bending, stretching, compressing, protruding, indenting, *etc.*

The matching process can take into account similarity transformations by explicitly incorporating it in the energy functional. However, since the structural matching described above is invariant with respect to the similarity transformations, we should first discuss the second class of deformations. As discussed earlier, deformations of shape affect the curvature and acceleration functions of each link. In determining an appropriate energy, the following principle is required:

Principle 1 (*Similarity via Deformation*): *The similarity between two shapes is inversely proportional to the minimum amount of deformation required to bring one shape in register with another.*

In other words, a shape can be deformed to another through sequences of (canonical) transformations, Figure 11. The dissimilarity between two shapes is the length of the *minimum length path* in the space of transformations. We consider the following canonical deformations: (i) stretching or compressing where similarity is effectively translated into length comparison between link length $d_L(L_1, L_2)$; (ii) fattening or thinning a shape, where similarity is measured by comparing acceleration functions $d_a(a_1(t), a_2(t))$; (iii) bending which affects curvature function, where similarity is measured by comparing curvature functions $d_\kappa(\kappa_1(s), \kappa_2(s))$. Link similarities are then functions of these individual similarities considered as separable dimensions of deformations. Node similarities compare the size of each node represented as time $d_t(t_1, t_2)$, and initial velocities $d_v(\{\vec{v}_{1i}\}, \{\vec{v}_{2i}\})$. It should be observed that similarity transformations affect some of these comparisons: if the query shape is scaled up to twice its size, L_2 becomes $2L_2$; thus we are in general comparing the L_1 corresponding to the item in the database and λL_2 , where λ is the scale variable for the query shape, optimized in the match process. The energy can now be formulated as follows:

$$E(M, x_0, y_0, \theta_0, \lambda_0) = \sum_{a=1}^A \sum_{i=1}^I \sum_{b=1}^A \sum_{j=1}^I M_{ai} M_{bj} L_{aibj}(\lambda_0) + \sum_{a=1}^A \sum_{i=1}^I M_{ai} N_{ai}(x_0, y_0, \theta_0, \lambda_0) , \quad (6)$$

Where L_{aibj} and N_{ai} represent the metric similarities described above. The optimization with respect to the match M is carried out by the graduated assignment algorithm, hand in hand with optimization by gradient descent for other variables. Thus, the result of each match is an energy as well as an optimal pose and scale for the query shape. (We have built a visual debugger to examine the correspondence of each match). We refer to this process as *metric matching*.

Our initial implementation of eq(6) revealed that energy defined in this way varies with the number of nodes and links in each graph, such that the similarity between pairs of shapes could not be effectively compared. This motivates a normalization of each component of the energy functional by the maximum possible value for each case, so that $0 \leq E \leq 1$,

$$E(M, x_0, y_0, \theta_0, \lambda_0) = \frac{1 - \alpha}{E_{Max}^L(M)} \sum_{a=1}^A \sum_{i=1}^I \sum_{b=1}^A \sum_{j=1}^I M_{ai} M_{bj} L_{aibj}(\lambda_0) + \frac{\alpha}{E_{Max}^N(M)} \sum_{a=1}^A \sum_{i=1}^I M_{ai} N_{ai}(x_0, y_0, \theta_0, \lambda_0) \quad (7)$$

where

$$E_{Max}^L(M) = \sum_{a=1}^A \sum_{i=1}^I \sum_{b=1}^A \sum_{j=1}^I M_{ai} M_{bj} \bar{L}_{aibj} \quad (8)$$

$$\bar{L}_{aibj} = \begin{cases} 1 & \text{if either link } G_{ab} \text{ or } g_{ij} \text{ exists} \\ Null & \text{otherwise} \end{cases} \quad (9)$$

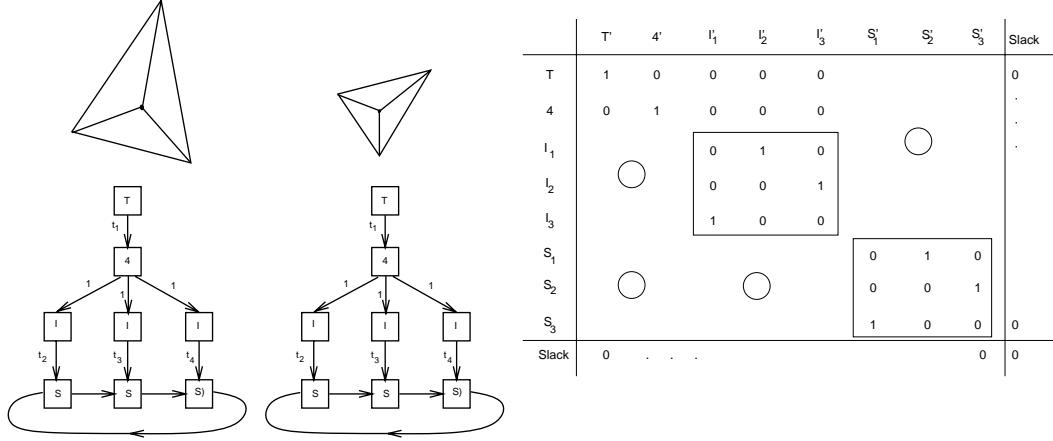


Figure 8: This figure illustrates the match matrix for two simple graphs corresponding to two triangles. Note that an integral aspect of the matching process is the recovery of the similarity transformation parameters (rotation, scale, orientation), as well as deformations.

and,

$$E_{Max}^N(M) = \sum_{a=1}^A \sum_{i=1}^I M_{ai}, \quad (10)$$

The corresponding Q can be derived as,

$$\begin{aligned} \tilde{Q}_{ai}(M) = & (1 - \alpha) \left[\frac{1}{E_{Max}^L(M)} \sum_{b=1}^A \sum_{j=1}^I M_{bj} (L_{aibj} + L_{bjai}) \right. \\ & \left. - \frac{E^L(M)}{E_{Max}^L(M)^2} \sum_{b=1}^A \sum_{j=1}^I M_{bj} (\bar{L}_{aibj} + \bar{L}_{bjai}) \right] + \frac{\alpha}{E_{Max}^N(M)} N_{ai} \end{aligned}, \quad (11)$$

Observe that the metrics defined above for comparing length, size, *etc.* are scale invariant. In addition, we use a *square root distance*, $\sqrt{(L_1 - \lambda L_2)}$, for length comparison, motivated by (i) a re-interpretation of Weber’s law, (ii) to maintain sensitivity to variations when two items are close, and (iii) to reduce sensitivity when two items are very distant. We have found this norm to perform better for our application than others, *e.g.* $\frac{\|L_1 - L_2\|}{\max(L_1, L_2)}$. Finally, metric similarity using a detailed comparison of curvature and acceleration functions is often not necessary to distinguish two shapes. Rather, curvature and acceleration functions can be approximated coarsely via a few parameters, leading to *parametric similarity*. The examples in this paper are generated by parametric matching.

Finally, robust matching must deal with “seams” in the shape space created by the discrete categorization of symmetries into types. The similarity of shapes on either side of the “boundary” of each category is not captured by the graph structure, but is implicit in metric information embedded in it. To illustrate, consider Figure 13 where a rectangle and four slight deformations of it are presented. The similarity can be realized by observing that the limit of a first-order shock group as its velocity is increased is a third-order shock group. Thus, velocity comparisons must give a small distance when comparing $+\infty$ to $-\infty$.

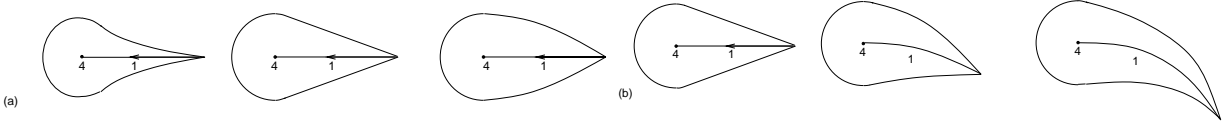


Figure 9: This figure sketches shapes with the same graph structure (a first order shock group terminating in a fourth order shock) generated by altering metric information only, (a) altering shock dynamics (acceleration), and (b) by altering shock geometry (curvature). Observe that (i) the relational knowledge of how various shock groups connect categorizes shapes such that all shapes above are immediately distinguishable, *e.g.*, from rectangles (*structural similarity*). (ii) This category can be further refined by simply observing that acceleration is negative, constant, or positive, respectively, from left to right in (a), and using an approximation using a few parameters [13] leading to *parametric similarity*. Finally, (iii) unique instances can be identified by an exact specification of shock geometry and dynamics. Using such exact descriptions leads to *metric similarity*.

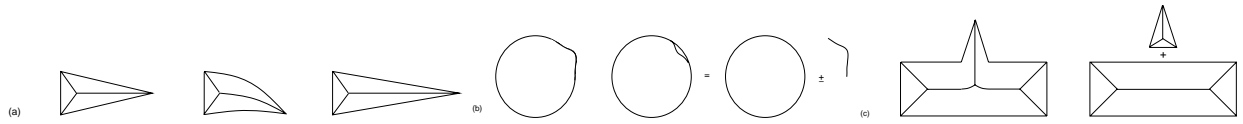


Figure 10: This figure illustrates examples of (a) regional deformations: bending and stretching (b) bending deformations: protrusion and indentation (c) part composition/decomposition

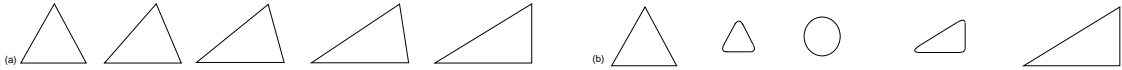


Figure 11: The need for a multistage similarity matching: (a) a linear deformation, and (b) an unlikely method of deforming a triangle to another by deforming it to a blob first. It is this deformation energy this leads to metric similarity.

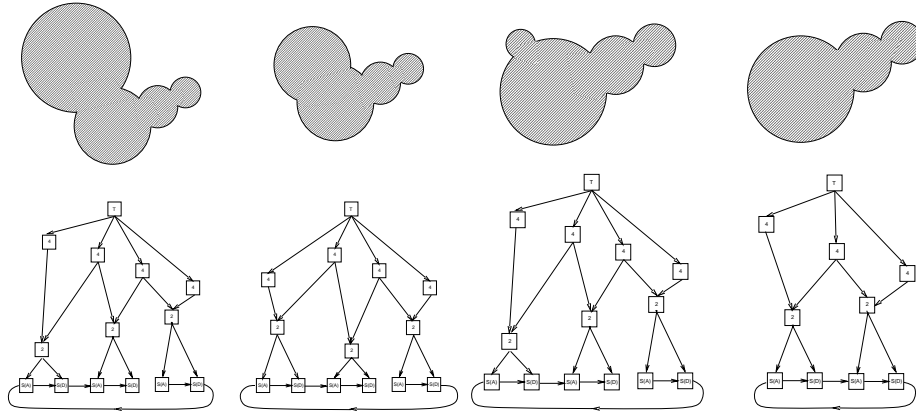


Figure 12: This example illustrates that we cannot use the “oldest” 4th order shock to calibrate the scale of two shapes. Scale (λ) is a global parameter of the shape and should be used to optimize the global fit.

The graduated assignment algorithm handles missing and extra nodes, such that the “bump” in Figure 14 is viewed as an extra node. However, since slight variations cause bumps such as these to appear/disappear, we may build this behavior directly into our representation by creating *virtual links* when warranted. The virtual link complements the existing representation such that the rectangle with a bump can easily match a perfect rectangle. A similar argument holds for creating virtual nodes.

5 Results and Discussion

We now report on the result of applying this approach to a database consisting of binary shapes, and match grey-scale images of isolated objects and user-drawn sketches against this database. Comparisons are made at the structural and metric levels. Structural comparisons quickly rule out a large number of shapes and parametric comparisons are then performed for the remaining. This is not merely an exercise in efficiency, but is required as the metric comparisons combine both relational and metric aspects of the deformation. Table 5 shows the parametric comparison between pairs of shapes along a continuum. Figure 15 shows the resulting correspondence between two pairs of shapes from our shape database. Figure 16 illustrates the parametric similarity scores for this database on a scale of 0-1 and the top four matches are highlighted. Observe the ordering of shapes is intuitive. *i.e.*, the highest scores for the tools images are in the tools category. The use of categorization to organize a database of images, supported by psychophysics [27] as well as the use of hierarchical structure in image indexing [24] is well supported by our approach and is the current focus of ongoing research in our laboratory. Table 6 shows the results of indexing hand-drawn sketches and images containing isolated objects into this database. Note that some of the queries are images of isolated objects, without requiring pre-segmentation of images due to the extraction of symmetries directly from grey-scale images. In fact, this approach allows a potential top-down flow for segmentation to follow matching, *e.g.*, triangle with gaps in Figure 4. The results of our proposed indexing scheme, as evidenced by results in Figure 16

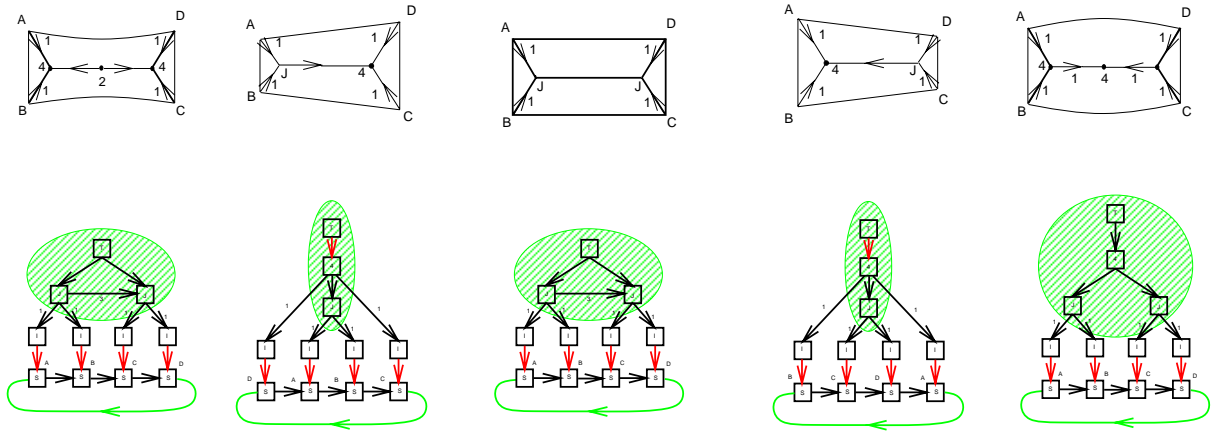


Figure 13: The limit of fast moving 1st order shock group is a third order shock group. A shock transform is needed to “seem up” the shape space between positive and negative fast moving first 1st order shock groups (a), (b) and (c). Links with high speeds invoke the transform, *e.g.*, on (a) left which turns the $(4, I)$ combinations into a (j, j) joined by a third order shock. This asymmetrical reference is not unusual, *e.g.*, as in referring to off-red as red or referring to nearly vertical as vertical [21].

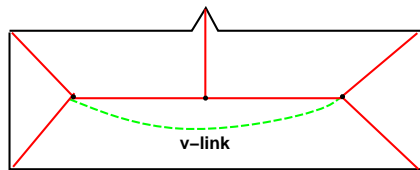


Figure 14: Virtual links (v-links) are introduced across nodes in which one of the shock groups can be easily pruned.

	0.983	0.895	0.722	0.780	0.835	0.821	0.762
		0.983	0.776	0.838	0.790	0.766	0.753
			0.991	0.855	0.795	0.752	0.753
				0.986	0.794	0.756	0.738
					0.993	0.930	0.796
						0.991	0.814
							0.981

Table 5: Metric comparison between pairs of shapes along a continuum. The first row shows that as the quadrilateral moves towards a refuge, the similarity is decreased, but it then increases, because the best match requires a 180° rotation of the shape.

Table 6: Result of matching hand-drawn sketches, as well as grey-scale images against the database of Figure 16.

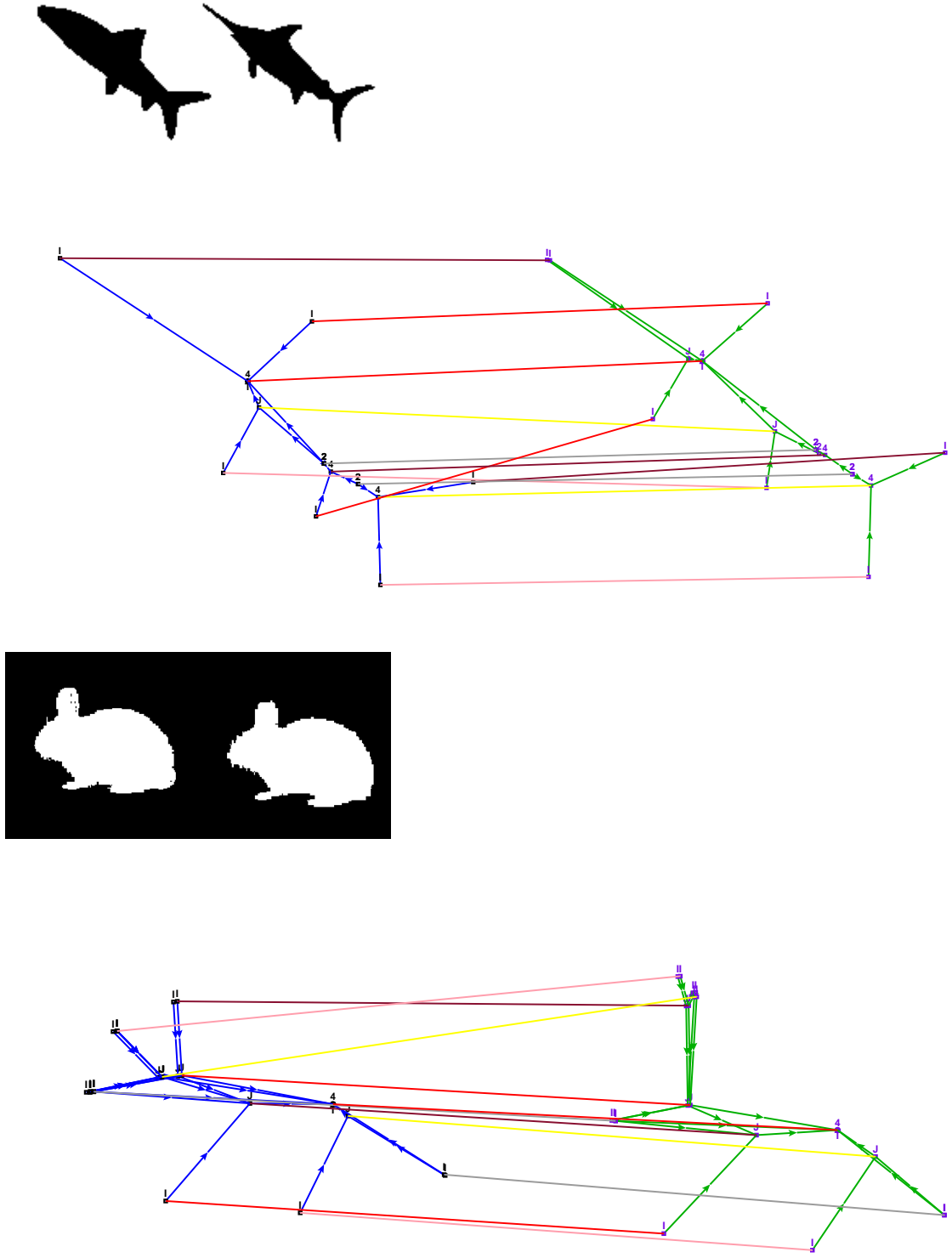


Figure 15: Example of a match between two fish and between two rabbits.

	0.978	0.407	0.618	0.687	0.632	0.458	0.473	0.471	0.563	0.531	0.293	0.27	0.39	0.304	0.316	0.457	0.398	0.355	0.323	0.319	0.544	0.445	0.523	0.507	0.534
	0.527	0.982	0.657	0.505	0.525	0.382	0.426	0.312	0.387	0.462	0.397	0.304	0.248	0.407	0.398	0.341	0.265	0.326	0.336	0.257	0.348	0.407	0.232	0.357	0.384
	0.541	0.656	0.974	0.581	0.603	0.321	0.32	0.272	0.484	0.505	0.276	0.335	0.305	0.42	0.251	0.351	0.295	0.287	0.328	0.345	0.44	0.38	0.277	0.245	0.435
	0.693	0.47	0.585	0.961	0.711	0.524	0.546	0.584	0.591	0.48	0.292	0.474	0.332	0.258	0.482	0.418	0.241	0.367	0.323	0.222	0.465	0.403	0.483	0.451	0.517
	0.629	0.431	0.59	0.716	0.983	0.471	0.493	0.508	0.482	0.222	0.315	0.537	0.353	0.414	0.388	0.476	0.337	0.357	0.377	0.372	0.386	0.449	0.456	0.429	0.537
	0.333	0.375	0.308	0.491	0.295	0.992	0.736	0.528	0.582	0.316	0.267	0.352	0.292	0.347	0.348	0.337	0.248	0.296	0.336	0.273	0.17	0.48	0.338	0.438	0.356
	0.436	0.42	0.308	0.492	0.445	0.734	0.988	0.534	0.592	0.31	0.264	0.18	0.231	0.35	0.227	0.335	0.188	0.278	0.357	0.204	0.24	0.467	0.334	0.414	0.35
	0.345	0.31	0.248	0.544	0.372	0.519	0.532	0.984	0.681	0.176	0.259	0.325	0.299	0.343	0.302	0.334	0.286	0.313	0.27	0.273	0.231	0.396	0.261	0.137	0.312
	0.397	0.372	0.467	0.575	0.471	0.568	0.591	0.68	0.993	0.177	0.239	0.091	0.285	0.158	0.233	0.423	0.236	0.323	0.363	0.277	0.474	0.449	0.445	0.415	0.453
	0.519	0.391	0.35	0.562	0.543	0.386	0.383	0.339	0.344	0.968	0.505	0.686	0.551	0.528	0.388	0.43	0.294	0.505	0.399	0.383	0.339	0.309	0.489	0.518	0.515
	0.306	0.398	0.285	0.3	0.32	0.278	0.395	0.301	0.278	0.523	0.989	0.547	0.476	0.32	0.248	0.416	0.305	0.311	0.37	0.292	0.26	0.228	0.242	0.379	0.429
	0.516	0.347	0.536	0.549	0.55	0.412	0.404	0.225	0.201	0.685	0.54	0.988	0.585	0.358	0.325	0.397	0.354	0.544	0.517	0.555	0.547	0.482	0.528	0.561	0.513
	0.394	0.292	0.276	0.382	0.345	0.516	0.291	0.328	0.335	0.539	0.467	0.568	0.984	0.393	0.49	0.675	0.481	0.573	0.547	0.495	0.533	0.572	0.541	0.485	0.495
	0.468	0.388	0.459	0.554	0.457	0.435	0.404	0.416	0.47	0.344	0.307	0.344	0.397	0.997	0.6	0.606	0.531	0.509	0.483	0.57	0.518	0.472	0.543	0.545	0.553
	0.498	0.506	0.524	0.542	0.52	0.254	0.406	0.325	0.269	0.352	0.239	0.311	0.494	0.602	0.992	0.618	0.475	0.528	0.448	0.519	0.523	0.51	0.506	0.535	0.47
	0.581	0.559	0.485	0.526	0.62	0.405	0.375	0.419	0.472	0.366	0.371	0.316	0.673	0.607	0.608	0.996	0.563	0.53	0.505	0.559	0.546	0.511	0.497	0.476	0.496
	0.41	0.473	0.456	0.398	0.441	0.365	0.352	0.329	0.361	0.438	0.291	0.509	0.463	0.525	0.475	0.56	0.996	0.742	0.661	0.805	0.536	0.514	0.501	0.51	0.481
	0.359	0.528	0.34	0.406	0.35	0.293	0.467	0.368	0.375	0.486	0.299	0.533	0.574	0.508	0.337	0.533	0.742	0.992	0.739	0.713	0.519	0.408	0.473	0.458	0.432
	0.419	0.32	0.479	0.317	0.36	0.528	0.432	0.295	0.424	0.427	0.35	0.448	0.529	0.478	0.444	0.509	0.664	0.739	0.997	0.636	0.552	0.594	0.382	0.442	0.302
	0.518	0.539	0.337	0.315	0.561	0.341	0.394	0.308	0.334	0.364	0.275	0.565	0.519	0.376	0.51	0.537	0.805	0.712	0.636	0.991	0.509	0.503	0.468	0.432	0.446
	0.575	0.391	0.47	0.515	0.482	0.414	0.426	0.427	0.47	0.322	0.415	0.548	0.534	0.515	0.523	0.545	0.536	0.516	0.562	0.502	0.997	0.742	0.702	0.629	0.721
	0.46	0.329	0.413	0.461	0.481	0.576	0.564	0.51	0.523	0.297	0.253	0.471	0.571	0.47	0.504	0.515	0.519	0.465	0.59	0.511	0.737	0.993	0.592	0.643	0.666
	0.561	0.475	0.447	0.504	0.491	0.421	0.432	0.292	0.502	0.261	0.314	0.524	0.544	0.539	0.508	0.498	0.502	0.472	0.381	0.466	0.697	0.586	0.993	0.725	0.703
	0.547	0.384	0.507	0.495	0.104	0.525	0.496	0.452	0.474	0.515	0.26	0.566	0.499	0.537	0.532	0.481	0.513	0.458	0.447	0.439	0.66	0.642	0.724	0.994	0.691
	0.377	0.38	0.507	0.559	0.556	0.434	0.432	0.446	0.495	0.511	0.274	0.515	0.508	0.553	0.47	0.49	0.495	0.435	0.399	0.466	0.732	0.67	0.707	0.694	0.986

Figure 16: The similarity metric for the database. Observe that most of the shapes are categorized well: The first choice is always correct, as expected. The second, third, and fourth choices are correct 23/25, 21/25, and 20/25 times, respectively. It should be noted that, in general, this parametric match should also be considered in conjunction with a first-stage structural match.

and Table 6, are highly promising. We are in the process of compiling formal performance measures on a much larger database.

In a related paper, Siddiqi et. al [33] present an alternative use of shock structure for shape matching. Although this approach and ours, presented initially in [13], 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 shape 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 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.* one node may be scaled up and another may 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.

6 Conclusion

We have demonstrated an approach based on *(i)* extraction of symmetries from grey-scale images, *(ii)* organization of symmetries into a hierarchical groups, *(iii)* segregation of relational structures of groups of symmetry into graphs and continuous variables representing deformation and similarity deformations as attributes of graphs, *(iv)* matching of two intrinsic shock graph via a multistage graph matching algorithm. We have not discussed the role of salience of shape components, shock transforms to deal with spurious edge elements, gaps and occlusions [38, 39]. These will be presented in forthcoming papers. We have, however, presented an approach for the use of symmetry as a cue for indexing with very promising results. We are in the process of compiling an extensive evaluation of the match results for a much larger database. We expect that given the generic formulation of this approach which inherently includes arbitrary deformations, the performance of this approach in conjunction with a level of categorization will be excellent with enlarging the size of the pictorial database to practical numbers.

References

- [1] H. A. Almohamad and S. O. Duffuaa. A linear programming approach for the weighted graph matching problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15:522–525, May 1993.
- [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 Visual Languages*, 1994.
- [3] A. D. Bimbo, P. Pala, and S. Santini. Image retrieval by elastic matching of shapes and image patterns. In *Proceedings of MULTIMEDIA '96*, pages 215–218, 1996.
- [4] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, Cambridge, MA, 1987.
- [5] M. A. Eshera and K. S. Fu. A graph distance measure for image analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, 14:398–408, 1984.
- [6] 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, pages 75–77. IEEE, IEEE, 1984.
- [7] M. A. Eshera and K. S. Fu. An image understanding system using Attributed Symbolic Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(5):604–618, Sept. 1986.
- [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] S. Gold, A. Rangarajan, and E. Mjolsness. Learning with preknowledge: clustering with point and graph matching distance measures. *Neural Computation*, 1996. (in press).
- [10] R. S. Gray. Content based image retrieval: Color and edges. In *Proceedings of 1995 Dartmouth Institute for Advanced Graduate Studies: Electroning Publishing and the Information Superhighway*, pages 77–90, 1995.
- [11] K. Hirata and T. Kato. Query by visual example - content based image retrieval. *Advances in Database Technology*, pages 56–71, 1992.
- [12] R. Hummel and S. W. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 6:267–287, 1983.
- [13] 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*, volume 3229, pages 288–302, Dallas, Texas, November 1997.
- [14] 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*, pages 402–407, Antibes, France, 1990. Springer Verlag.
- [15] B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker. On the evolution of curves via a function of curvature, I: The classical case. *JMAA*, 163(2):438–458, January 1992.

- [16] B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker. On the shape triangle. In C. Arcelli, editor, *Proceedings of the International Workshop on Visual Form*, pages 307–323, Capri, Italy, May 1994. World Scientific.
- [17] 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*, 15:189–224, 1995.
- [18] J. J. Kosowsky and A. L. Yuille. The invisible hand algorithm: Solving the assignment problem with statistical physics. *Neural Networks*, 7:477–490, 1994.
- [19] E. Lawler and D. Wood. Branch and bound methods: A survey. *Operations Research*, 14:699–719, 1966.
- [20] 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 Barbara, CA, October 1997.
- [21] M. Leyton. *Symmetry, Causality, Mind*. MIT press, April 1992.
- [22] A. L. VanGool, P. Kempenaers. Recognition and semi-differential invariants. *CVPR*, pages 55–60, 1991.
- [23] R. Mehrotra, F. Kung, and W. Grosky. Industrial part recognition using a component index. *Image and Vision Computing*, 3:225–231, 1990.
- [24] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project: Querying images by content using color texture and shape. Technical Report 9203, IBM Research Division, 1993.
- [25] S. M. Pizer and C. A. Burbeck. Object representation by cores: Identifying and representing primitive spatial regions. *Vision Research*, 35(13):1917–1930, 1995.
- [26] A. Rangarajan and E. Mjolsness. A Lagrangian relaxation network for graph matching. In *IEEE International Conference on Neural Networks (ICNN)*, volume 7, pages 4629–4634. IEEE Press, 1994.
- [27] E. Rosch. *On the Internal Structure of Perceptual and Semantic categories: Cognitive Development and the acquisition of language*. Academic Press, 1973.
- [28] S. Sclaroff. Deformable prototypes for encoding shape categories in image databases. *Pattern Recognition*, page submitted, 1997.
- [29] S. Sclaroff and A. Pentland. Modal matching for correspondence and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):545–561, 1995.
- [30] S.C. Zhu and A.L. Yuille. Forms: A flexible object recognition and modeling system. *Int'l Journal of Computer Vision*, 20(3), 1996.
- [31] L. G. Shapiro and R. M. Haralick. Structural descriptions and inexact matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:504–519, Sept. 1981.

- [32] K. Siddiqi and B. B. Kimia. A shock grammar for recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 507–513, San Francisco, California, June 1996. IEEE Computer Society Press.
- [33] K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker. Shock graphs and shape matching. *ICCV*, pages 222–229, 1998.
- [34] P. D. Simic. Constrained nets for graph matching and other quadratic assignment problems. *Neural Computation*, 3:268–281, 1991.
- [35] R. Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. Math. Statist.*, 35:876–879, 1964.
- [36] S. Tari and J. Shah. Extraction of shape skeletons from grayscale images. *Computer Vision Image Understanding*, 66(2):133–146, 1997.
- [37] H. Tek and B. B. Kimia. Extracting shocks of shape from grey-scale images. Technical report, LEMS, Brown University, November 1997.
- [38] H. Tek, F. Leymarie, and B. B. Kimia. Multiple generation shock detection and labeling using CEDT. In *Proceedings of the International Workshop on Visual Form*, Capri, Italy, May 1997. World Scientific.
- [39] 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*, Puerto Rico, June 15-16 1997. IEEE Computer Society Press.
- [40] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:695–703, September 1988.
- [41] G. V. Wilson and G. S. Pawley. On the stability of the traveling salesman problem algorithm of hopfield and tank. *Biological Cybernetics*, 58:63–70, 1988.
- [42] M. W. Wright, R. Cipolla, and P. J. Giblin. Skeletonization via the realization of the fire fronts' propagation and extinction in digital binary shapes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(5):367–375, 1995.