

Shape Matching and Anisotropy

Michael Kazhdan*
Princeton University

Thomas Funkhouser†
Princeton University

Szymon Rusinkiewicz‡
Princeton University

Abstract

With recent improvements in methods for the acquisition and rendering of 3D models, the need for retrieval of models has gained prominence in the graphics and vision communities. A variety of methods have been proposed that enable the efficient querying of model repositories for a desired 3D shape. Many of these methods use a 3D model as a query and attempt to retrieve models from the database that have a similar shape.

In this paper we consider the implications of anisotropy on the shape matching paradigm. In particular, we propose a novel method for matching 3D models that factors the shape matching equation as the disjoint outer product of anisotropy and geometric comparisons. We provide a general method for computing the factored similarity metric and show how this approach can be applied to improve the matching performance of many existing shape matching methods.

CR Categories: I.5.3 [Computing Methodologies]: Pattern Recognition—Similarity Measures; I.5.4 [Computing Methodologies]: Applications—Computer Vision

Keywords: shape matching, anisotropy

1 Introduction

With recent improvements in methods for the acquisition and rendering of 3D models, the need for effective retrieval of models has gained prominence in the graphics and vision communities. The ability to retrieve existing models facilitates the tasks of professionals in fields ranging from entertainment to scientific research, by allowing them to obtain desired models quickly without requiring the expenditure of large amounts of time modeling the 3D shape. To address this need, a variety of retrieval methods have been proposed that enable the efficient querying of model repositories for a desired 3D shape [Princeton 3D Model Search Engine ; Protein Data Bank ; CCCC ; ShapeSifter]. Many of these methods use a 3D model as a query and attempt to retrieve models with matching shape from the database.

In this paper we consider the implications of anisotropy on shape matching. In particular, we propose a novel method for matching 3D models that factors the shape matching equation as the disjoint outer product of anisotropy and geometric comparisons. We provide a general method for computing the factored similarity metric and show how this approach can be applied to improve the matching performance of many existing shape matching methods.

*e-mail:mkazhdan@cs.princeton.edu

†e-mail:funk@cs.princeton.edu

‡e-mail:smr@cs.princeton.edu

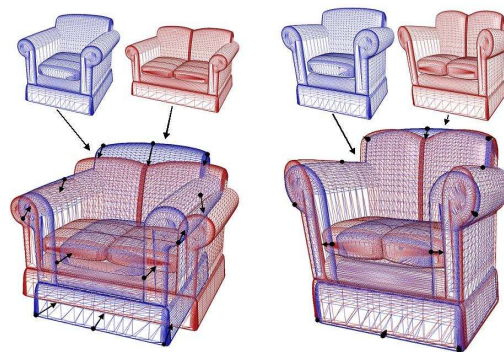


Figure 1: When two models have different anisotropic scales (left), it is harder to establish correct correspondences between the two. Thus, matching methods that depend on correspondences for evaluating similarity will be inaccurate in this case. In contrast, when the models are transformed so that each is isotropic (right), the correspondences are more accurate and the measure of shape similarity is more discriminating.

The key idea of our approach is based on the observation that much of the challenge of shape matching is the establishing of correspondences, and that it is easier to establish correspondences between two models if they are *isotropic* — having constant variance in each direction. Figure 1 demonstrates this for models of an armchair and a sofa. When the models are at their initial anisotropic scales (left), it is difficult to establish correspondences between similar regions. Methods such as associating to a point on one model the nearest point on the other (commonly used in ICP-type approaches [Besl and McKay 1992]) will map points on the corners of the armchair to points in the middle of the sofa, points on the bottom of the arm-rest of the armchair to points on the top of the arm-rest of the sofa, etc. Thus, many poor correspondences will be established, resulting in an inaccurate measure of similarity. If instead both models are rescaled to be isotropic (right), then the correspondences established more accurately reflect corresponding regions in the shape.

While many existing matching methods compare models without explicitly establishing correspondences, the underlying similarity metric is often designed to represent the distance between points on the surfaces of the two models. These observations motivate us to design a shape matching paradigm that compares two models by (1) transforming each of them into isotropic models, (2) comparing the geometric similarity of the isotropic models, and (3) defining the measure of model similarity as a function of both the similarity of the isotropic models, and the difference in their initial anisotropic scales. Figure 2 demonstrates this process for two different models of a table. Each table is represented by its isotropic version and its initial anisotropic scale, represented by the covariance ellipsoid of the original model. The distance between the two tables is then defined as the outer product of the distance between the isotropic tables and the distance between the initial anisotropic scales.

The remainder of the paper is structured as follows. Section 2 provides a general overview of existing shape matching approaches, highlighting some of the central challenges in this area. Section 3 describes our proposed method, and Section 4 provides empirical results demonstrating the efficacy of our approach in improving the matching performance of many existing shape metrics. We conclude in Section 5 by summarizing our results.

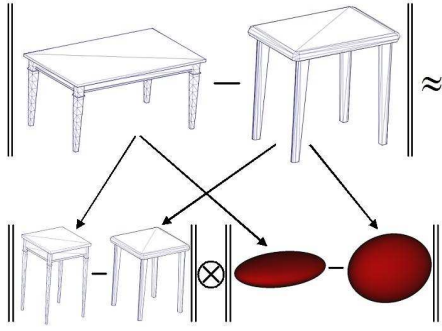


Figure 2: Our approach is to compare two models by rescaling each model so that it is isotropic and then defining the distance between two models as the outer product of the differences between the isotropic models and their initial anisotropic scales.

2 Related Work

Traditional methods for retrieval of models from large repositories focus on designing a method for defining a measure of similarity between a query model and every target model in the database. The models in the database are then sorted by this measure of similarity, and the nearest models are returned as matches.

In the context of matching 3D shapes, the most common approach is to establish correspondences between the query model and the target model, and then to define the measure of similarity in terms of the distances between corresponding points. Two general classes of methods have been proposed that compute a measure of shape similarity by explicitly establishing such correspondences. The first approach is a local one, seeking to establish correspondences between pairs of points on the two models, and then defining the measure of shape similarity as the sum of the squared distances between pairs of points in correspondence [Besl and McKay 1992]. The second method is more general, decomposing a model into constituent parts, and then representing the model as a graph characterizing the relationship between the different segments [Siddiqi et al. 1998; Hirlaga et al. 2001]. Correspondences between two models can then be established using subgraph isomorphism techniques, which simultaneously define the correspondences between the nodes of the two graph representations, and give the quality of the correspondences.

For both of these approaches, the establishing of correspondences is a difficult and time consuming task that needs to be performed on a per-pair-of-models basis. Thus, much of the necessary computation can only be performed at runtime, once a query is specified. This makes these methods impractical for the retrieval of models from large databases, where efficient comparison is essential.

The computational complexity of establishing correspondences between models has motivated a large body of research in the area of shape descriptors. The general approach of these methods is to define a mapping from the space of models into a fixed-dimensional vector space, and then to define the measure of similarity between two models as the distance between their corresponding descriptors [Horn 1984; Kang and Ikeuchi 1991; Ankerst et al. 1999; Osada et al. 2001; Vranic and Saupé 2001; Funkhouser et al. 2003]. The mapping is often chosen so that the distance between two descriptors measures the proximity of points on the surfaces of the two models, so that a correspondence-based measure of similarity can be obtained without the overhead of explicitly establishing the correspondences. This approach has the advantage of addressing the matching problem on a per-model basis, allowing for the computation of descriptors in an offline process. Then, at runtime, the descriptor of the query is computed and compared against the (pre-

computed) descriptors of all the models in the database, giving rise to methods that can satisfy the efficiency requirements of interactive search.

For a more general survey of shape descriptors, we refer the readers to [Alt and Guibas 1996; Loncaric 1998; Pope 1994; Tangelder and Veltkamp 2004].

A specific challenge that shape descriptor approaches need to address is that in the context of 3D shape matching, a model and its image under a similarity transformation are considered to be the same. In general, this issue is addressed in one of two manners: (1) The mapping is chosen to be *invariant* to similarity transformation, so that the same shape descriptor is defined for every orientation of a single model. (2) Each model is *normalized* by placing it into its own canonical coordinate system, and then the shape descriptor of the alignment-normalized model is computed.

Methods for normalizing a model’s translation and scale are based on [Horn 1987; Horn et al. 1988]. In this work, the authors describe a method for solving for the alignment minimizing the sum of square differences between two ordered point sets. While the solution for the optimal rotation depends on the correspondence between the two point sets, the optimal translation and scale can be computed on a per-model basis, with the optimal translation being the one that transforms a model’s center of mass to the origin, and the optimal scale giving rise to a model whose mean variance from the origin is equal to one.

Methods for addressing rotational similarity have either taken the normalization approach, aligning a model so that its principal axes transform to the x -, y -, and z -axes, or have obtained rotation invariant representations by discarding spherical phase and obtaining a collection of amplitudes that are independent of a model’s alignment [Burel and Henocq 1995; Kazhdan et al. 2003].

3 Methodology

In order to separate anisotropy from the shape matching equation, we propose a method for matching two 3D models that first removes the anisotropy from each of the models, compares the geometry of the isotropic models, and then expresses the measure of similarity of the two models as a function of both geometric and anisotropic similarity. This approach is motivated by earlier work in the area of isotropic scale normalization, which we review in the next subsection. We then show how these results can be generalized to anisotropic scale and describe a method for removing the anisotropy from model. We conclude by describing a method for comparing two models, providing a family of shape metrics parameterized by the importance assigned to anisotropy.

3.1 Isotropic Scale

In [Horn et al. 1988; Horn 1987], the authors address the issue of solving for the optimal scale that minimizes the sum of square distances between two ordered points sets. They pose the problem as follows: Given two ordered point sets, $P = \{p_1, \dots, p_n\}$ and $Q = \{q_1, \dots, q_n\}$, find the value of α that minimizes the sum of squared distances:

$$\sum_{i=1}^n \left\| \alpha p_i - \frac{q_i}{\alpha} \right\|^2$$

This formulation of the optimal scale problem has the property that the measure of similarity at the optimal scale is independent of the

order of P and Q , and is minimized when

$$\alpha = \sqrt[4]{\frac{\sum \|q_i\|^2}{\sum \|p_i\|^2}}.$$

Thus, if each of the point sets is independently scaled so that its mean variance is equal to 1, then the optimal scale value for aligning the two normalized point sets is $\alpha = 1$ and the point sets are in fact optimally pairwise scale-aligned.

The implication of this result for shape matching is that scale normalization can be done on a per-model basis, independent of correspondence. As with using the center of mass for translation normalization, this method is only provably correct when models are compared by summing distances between corresponding points. In practice however, many existing shape descriptors implicitly define shape similarity in terms of the distance between surfaces and we find that translating models so that their center of mass is at the origin and scaling them so that their mean variance is equal to one provides a robust method for translation and scale normalization.

3.2 Anisotropic Scale

We now show how the results for optimal scale can be generalized to solve for the optimal anisotropic scale. Given two point sets $P = \{p_1, \dots, p_n\}$, with $p_i = (p_x^i, p_y^i, p_z^i)$, and $Q = \{q_1, \dots, q_n\}$, with $q_i = (q_x^i, q_y^i, q_z^i)$, the sum of squared differences between the two point sets is given by the equation:

$$\sum_{i=1}^n (p_x^i - q_x^i)^2 + (p_y^i - q_y^i)^2 + (p_z^i - q_z^i)^2.$$

It follows from the work described above that if we search for the optimal anisotropic scale in any single direction v , then this occurs when the point sets P and Q are normalized so that their variance in the direction v is equal to 1. Consider, for example, the case of solving for the optimal anisotropic scale in the x direction. In this case, we would like to solve for the value of α that minimizes

$$\sum_{i=1}^n (\alpha p_x^i - q_x^i / \alpha)^2 + (p_y^i - q_y^i)^2 + (p_z^i - q_z^i)^2.$$

As in the case of the isotropic scale, the value of α that minimizes the error is:

$$\alpha = \sqrt[4]{\frac{\sum (q_x^i)^2}{\sum (p_x^i)^2}}$$

and the models are optimally scale-aligned in the x direction if the variance of each model, in the x -direction, is equal to 1.

More generally, if both point sets satisfy the property that the variance in any direction is equal to 1, then it follows that any anisotropic scaling of one of the two points sets will only increase the sum of squared differences, and the models are in fact optimally anisotropically aligned.

In order to transform an arbitrary point set into one that has unit variance in any direction, it suffices to compute its covariance matrix C and then apply the transformation $C^{-1/2}$ to the point set. (Since we assume that the points are not all coplanar, the matrix C is positive definite and hence can be inverted, and has a real square root.) To see this, note that the covariance matrix of a point set $P = \{p_1, \dots, p_n\}$ can be defined by the equation:

$$C_P = \sum_{i,j=1}^n (p_i - p_j) \cdot (p_i - p_j)^t,$$

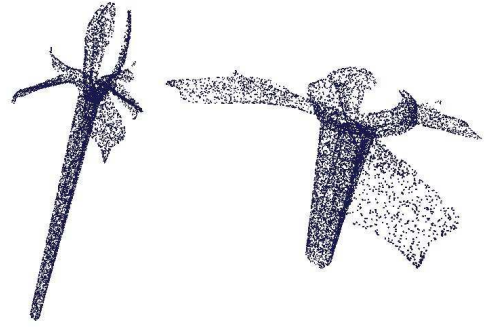


Figure 3: Uniform point samples from the surface of an iris model are shown on the left. The same points after anisotropic rescaling, are shown on the right. Though the point set on the right has constant variance in every direction, it no longer represents a uniform sampling from the surface of the anisotropically rescaled iris.

where the double summation is taken in order to account for the variance with respect to center of mass. If we set Q to be the transformed point set $Q = C_P^{-1/2} p_1, \dots, C_P^{-1/2} p_n$ then the covariance matrix of Q is given by:

$$\begin{aligned} C_Q &= \sum_{i,j=1}^n C_P^{-1/2} (p_i - p_j) \cdot (p_i - p_j)^t C_P^{-1/2} \\ &= C_P^{-1/2} \left(\sum_{i,j=1}^n (p_i - p_j) \cdot (p_i - p_j)^t \right) C_P^{-1/2} = 1 \end{aligned}$$

Thus, the covariance matrix of the transformed point set is equal to the identity, and the variance in any direction is equal to 1. As with isotropic rescaling, this approach has the advantage that it can normalize for anisotropic rescale on a per-model basis, allowing a model to be transformed in a pre-processing stage, independent of the model that it will be matched against.

The difficulty with applying this method directly to triangulated models is illustrated in Figure 3 which shows points uniformly sampled from a model of an iris (left). After an anisotropic transformation is applied to the point set (right), the positions of the points are transformed and they no longer represent a uniform sampling of the surface. Note that points on the stem are tightly clustered, while points on the petal become more spread out. This property of 3D meshes results in the undesired property that often the transformed model is still not isotropic.

In order to address this issue, we propose an iterative approach to transforming the model. At each step of the iteration, the model is first translated so that its center of mass is at the origin, the covariance matrix is computed, and finally the model is rescaled by the inverse square root of the covariance matrix. In our experiments, we find that this approach converges efficiently to an isotropic model and, in practice, no more than five iterations of this process are necessary to obtain a nearly isotropic shape. Figure 4 shows a model of a pen and the transformed model after several steps of the iteration process. The figure also draws the associated covariance ellipsoids, which converge to a sphere as the model becomes isotropic. Note that after the first iteration, the transformed model is still not isotropic, though, as the figure indicates, the iterative process converges quickly to an isotropic model. We provide a proof of the convergence of this approach in the appendix.

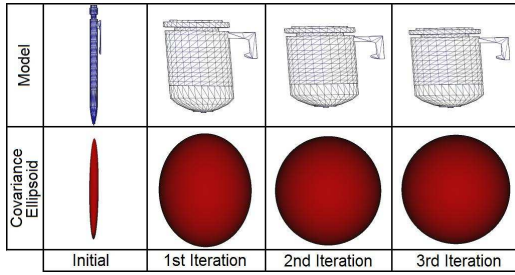


Figure 4: A visualization of a pen model and its covariance ellipsoid is shown on the left. The transformed model and its associated covariance ellipsoid, after one, two, and three iterations are shown on the right. Note that though the model is very anisotropic, after the third iteration of anisotropic rescaling we obtain a model that is nearly isotropic, with the covariance ellipsoid converging to a sphere.

3.3 Anisotropy Factoring

The method that we propose for anisotropy factoring is a general one that can be applied to any of the many methods [Horn 1984; Kang and Ikeuchi 1991; Ankerst et al. 1999; Vranic and Saupe 2001; Osada et al. 2001; Funkhouser et al. 2003] that matches two models by independently representing each one by a feature vector, and then defining the measure of model similarity as the L_2 -difference between the corresponding feature vectors. In particular, we anisotropically rescale a model M to obtain an isotropic model \tilde{M} , storing the sorted triplet of eigenvalues $\lambda_M = (\lambda_1^M, \lambda_2^M, \lambda_3^M)$ of the matrix transforming M into \tilde{M} . The triplet λ_M is a rotation invariant representation of the anisotropy of M and, for simplicity, we normalize the triplet so that $\|\lambda_M\| = 1$. We compute the feature vector $v_{\tilde{M}}$ of the isotropic model and, using the fact that the information contained in $v_{\tilde{M}}$ is orthogonal to the information contained in λ_M , we represent the initial model M by the new feature vector $v_{\tilde{M}} \times \lambda_M$, as shown in Figure 5.

At runtime, when a query model Q is presented to the database, we compute the anisotropy factorization of Q and define the measure of similarity between Q and a database model M to be the value:

$$D_\gamma(M, Q) = \|v_{\tilde{M}}\|^2 + \|v_{\tilde{Q}}\|^2 - 2\langle v_{\tilde{M}}, v_{\tilde{Q}} \rangle \langle \lambda_M, \lambda_Q \rangle^\gamma.$$

If $\gamma = 1$ then $D_\gamma(M, Q)$ is the L_2 -difference between the vectors $v_{\tilde{Q}} \times \lambda_Q$ and $v_{\tilde{M}} \times \lambda_M$. More generally, γ can be treated as a fixed constant representing the importance of anisotropy information in the context of shape matching. Thus, in the case that $\gamma = 0$, anisotropy information plays no role in the matching and the matching method is invariant to anisotropic scale. If additionally the feature vector is itself rotation invariant [Ankerst et al. 1999; Osada et al. 2001; Funkhouser et al. 2003], then we obtain a matching method that is invariant to all *affine* transformations.

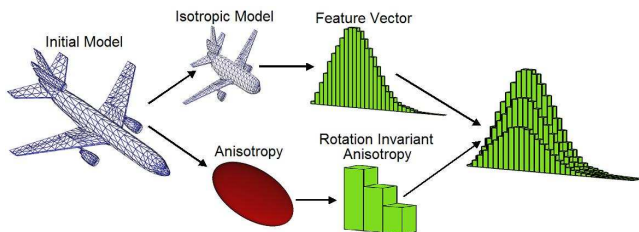


Figure 5: We create a new feature vector for a model by computing the outer product of the anisotropic scales with the feature vector of the isotropic model.

The advantage of this matching approach is that the shape metric defines similarity as the outer product of the similarity of the feature vectors and the similarity of the anisotropy vectors. Thus, the new feature vectors only need to store three additional values, corresponding to the normalized eigenvalues of the symmetric matrix transforming an anisotropic model into an isotropic one. This means that neither the storage nor the comparison time of the anisotropy factorized feature vector is significantly larger than the corresponding storage and comparison time for the original one.

4 Results

To measure the efficacy of the anisotropic rescaling scale approach in tasks of shape retrieval, we computed a number of shape descriptors and compared matching results when the descriptor of the original anisotropic model was used with the results obtained with anisotropy factoring. The descriptors we used were:

- **Shape Histogram (Shells)** [Ankerst et al. 1999]: A representation of a 3D model as a histogram of the distances of surface points from the center of mass.
- **D2** [Osada et al. 2001]: A representation of a 3D model as a histogram of the distances between pairs of surface points.
- **Extended Gaussian Image** [Horn 1984]: A representation of a 3D model as a spherical histogram of the distribution of normal directions over the surface of a model.
- **Shape Histogram (Sectors)** [Ankerst et al. 1999]: A representation of a 3D model as a spherical function associating to each direction from the origin the measure of the surface area in that direction.
- **Spherical Extent Function** [Vranic and Saupe 2001]: A representation of a 3D model as a spherical function associating to each direction from the origin the distance to the last point of intersection of the model with the ray.
- **Gaussian Euclidean Distance Transform** [Funkhouser et al. 2003]: A representation of a 3D model as a voxel grid, where the value at each point is given by the composition of a Gaussian with the Euclidean Distance Transform of the surface.

The first two shape descriptors are rotation-invariant by design, while the other four can either be normalized for rotation by aligning the model with PCA or can be made rotation-invariant using the spherical power spectrum [Burel and Henocq 1995; Kazhdan et al. 2003]; we present matching results for both types of approaches. (Note that since anisotropy factorization makes the covariance matrix of a model a multiple of the identity, PCA alignment needs to be performed prior to the factorization.)

We evaluated the performance of each method by measuring how well it classified models within a test database. The database was provided by the Princeton Shape Benchmark [Princeton Shape Benchmark], and consists of 1814 models decomposed into two groups of roughly 900 models, corresponding to training and test datasets. Each group is provided with a classification, associating each of the models to one of roughly 90 distinct classes. Classification performance was measured using precision/recall plots, which give the percentage of retrieved information that is relevant as a function of the percentage of relevant information retrieved. That is, for each target model in class C and any number K of top matches, “recall” represents the ratio of models in class C returned within the top K matches, while “precision” indicates the ratio of the top K matches that are in class C . Thus, plots that appear shifted up indicate superior retrieval results.

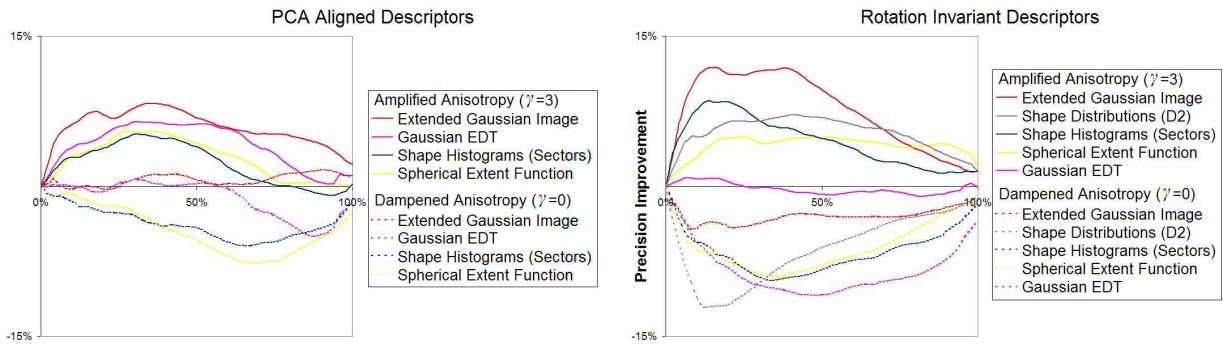


Figure 6: The improvement in precision of anisotropy factorization for four PCA-aligned and five rotation-invariant representations. The plots indicate that when the importance of anisotropy differences is amplified ($\alpha = 3$) retrieval performance is improved, and when anisotropy differences are ignored ($\alpha = 0$) retrieval performance is hampered.

For each shape descriptor, we compared the precision versus recall results obtained using the descriptor applied to the original model with those obtained with anisotropy factorization. For the anisotropy factorization we used the metrics D_γ with $\gamma = 0$ to dampen the importance of anisotropy in retrieval and $\gamma = 3$ to amplify the importance of anisotropy in retrieval. Figure 6 shows the results for the different PCA-aligned and rotation-invariant representations, with precision versus recall plots averaged over the different models in the database. Since we are primarily evaluating the effect of anisotropy factorization on the matching performance of a given descriptor, the plots show the improvement in precision of anisotropy factorization over the results obtained without factorization. The results indicate that the anisotropic scale of a model is an essential classifier of shape and when it is ignored ($\gamma = 0$), retrieval performance deteriorates. On the other hand, the results also indicate that it is easier to match two models when they are anisotropically aligned, so that methods that compare the two models in their anisotropy normalized frames and then penalize for differences in the initial anisotropic scales ($\gamma = 3$) give rise to matching results with improved precision.

Finally, we note that while the results indicate that anisotropy factorization is a technique that works well on average, there are specific types of models for which this method can fail. In particular, we have found that when the initial model is very anisotropic (the difference between the largest and smallest principal eigenvalues is large) anisotropy factorization does not always work well. We believe that the reason for this is due the manner in which anisotropic rescaling acts on a shape. When one of the principal eigenvalues is markedly smaller than the others, anisotropy normalization will scale the model disproportionately in one principal direction. As a result, features that may have been unimportant in the initial model will play a more dominant role in defining models similarity. As an example, Figure 7 demonstrates this for a chess set model. When the model is rescaled to have constant variance in every direction, the side of the board and the chess pieces become more prominent features in the model and play a more dominant role in matching.

5 Conclusion

In this paper we have described a method for factoring the shape matching equation into the product of an anisotropy comparison and a geometric comparison. This factorization provides a family of shape metrics that allows users to specify the importance of anisotropy within the general context of shape similarity. We have shown that this method is sufficiently general to be applied to a wide collection of existing shape matching algorithms, improving

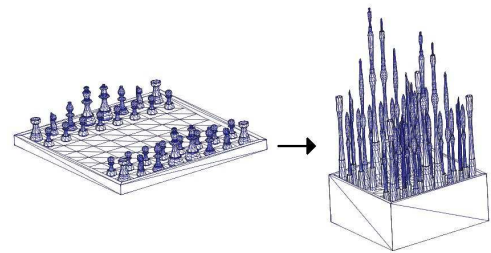


Figure 7: When a model is normalized for anisotropy, parts that may have been unimportant in the initial model may become more pronounced features of the model, adversely affecting shape matching.

the matching performance of most without inducing a large overhead in computation time. Thus, the described factorization is well suited for many of the nascent applications that strive to provide a method for efficiently and effectively retrieving models from large repositories of 3D shapes.

Acknowledgements

We would like to thank Viewpoint, Cacheforce, and Jose Maria De Espona, who donated commercial databases of polygonal models for experiments. The National Science Foundation funded this project under grants CCR-0093343, IIS-0121446, and DGE-9972930.

References

- ALT, H., AND GUIBAS, L. J. 1996. Discrete geometric shapes: Matching, interpolation, and approximation: A survey. Tech. Rep. B 96-11, EVL-1996-142, Institute of Computer Science, Freie Universität Berlin.
- ANKERST, M., KASTENMÜLLER, G., KRIEGEL, H., AND SEIDL, T. 1999. 3d shape histograms for similarity search and classification in spatial databases. In *Advances in Spatial Databases, 6th International Symposium*, 207–226.
- BESL, P., AND MCKAY, N. 1992. A method for registration of 3-d shapes. *IEEE PAMI* 14, 239–256.

BUREL, G., AND HENOCQ, H. 1995. Three-dimensional invariants and their application to object recognition. *Signal Processing* 45, 1, 1–22.

CCCC.

<http://merkur01.inf.uni-konstanz.de/>.

FUNKHOUSER, T., MIN, P., KAZHDAN, M., CHEN, J., HALDERMAN, A., DOBKIN, D., AND JACOBS, D. 2003. A search engine for 3d models. *ACM Transactions on Graphics*, 83–105.

HILAGA, M., SHINAGAWA, Y., KOHMURA, T., AND KUNII, T. 2001. Topology matching for fully automatic similarity estimation of 3d shapes. *Computer Graphics (Proceedings of SIGGRAPH 01)*, 203–212.

HORN, B., HILDEN, H., AND NEGAHDARIPOUR, S. 1988. Closed form solutions of absolute orientation using orthonormal matrices. *J. of the Optical Society* 5, 1127–1135.

HORN, B. 1984. Extended Gaussian images. In *Proceedings of the IEEE*, vol. 72, 1656–1678.

HORN, B. 1987. Closed form solutions of absolute orientation using unit quaternions. *J. of the Optical Society* 4, 629–642.

KANG, S., AND IKEUCHI, K. 1991. Determining 3-d object pose using the complex extended Gaussian image. *CVPR*, 580–585.

KAZHDAN, M., FUNKHOUSER, T., AND RUSINKIEWICZ, S. 2003. Rotation invariant spherical harmonic representation of 3d shape descriptors. *SGP*, 167–175.

LONCARIC, S. 1998. A survey of shape analysis techniques. *Pattern Recognition* 31, 8, 983–1001.

OSADA, R., FUNKHOUSER, T., CHAZELLE, B., AND DOBKIN, D. 2001. Matching 3d models with shape distributions. In *Shape Modeling International*, 154–166.

POPE, A. R. 1994. Model-based object recognition: A survey of recent research. Tech. Rep. TR-94-04, University of British Columbia, January.

PRINCETON 3D MODEL SEARCH ENGINE.

<http://shape.cs.princeton.edu>.

PRINCETON SHAPE BENCHMARK.

<http://shape.cs.princeton.edu/benchmark>.

PROTEIN DATA BANK. <http://www.rcsb.org>.

SHAPECIFTER. <http://www.shapesearch.net/>.

SIDDIQI, K., SHOKOUFANDEH, A., DICKINSON, S., AND ZUCKER, S. 1998. Shock graphs and shape matching. *Sixth International Conference on Computer Vision*, 222–229.

TANGELDER, J. W., AND VELTKAMP, R. C. 2004. A survey of content based 3d shape retrieval methods. In *Shape Modeling International*.

VRANIC, D., AND SAUPE, D. 2001. 3d model retrieval with spherical harmonics and moments. *Proceedings of the DAGM*, 392–397.

Appendix

In this appendix, we prove the convergence of the iterative rescaling algorithm for obtaining an isotropic model from an anisotropic one. We assume that the model is not coplanar, so that the variance in any direction is non-zero, and we show that iteratively anisotropically

rescaling the model by the inverse square root of the covariance matrix is a process that converges to a model with constant variance, independent of direction. In particular, the steps of each iteration are:

1. Anisotropically rescale the model by the inverse square root of the covariance matrix
2. Isotropically rescale the model so that the minimum and maximum eigenvalues of the covariance matrix of the new model are reciprocals,

and we show that iterating these steps forces the minimum and maximum eigenvalues of the covariance matrix to converge to 1. To this end, we use the following equation for the covariance matrix of a model M :

$$C_M = \int_M \int_M (p - q) \cdot (p - q)^t dp dq$$

so that the variance of M in a direction v is given by:

$$\text{Var}(M, v) = v^t C_M v = \int_M \int_M \langle p - q, v \rangle^2 dp dq.$$

We will first show that a model with non-zero variance in any direction can always be rescaled so that the minimum and maximum eigenvalues are reciprocals. Next, we prove two lemmas describing the decomposition M into an even partition and the corresponding decomposition of the variance of M across such a partition. Finally, we use the lemmas to show that the extremal eigenvalues must converge to 1.

Isotropic Rescaling: Given a model M and scale factor s , the covariance matrix of sM is defined as:

$$C_{sM} = \int_{sM} \int_{sM} (p - q) \cdot (p - q)^t dp dq = s^6 C_M.$$

Thus, given a model M whose covariance matrix C_M has eigenvalues $0 < \lambda_1 \leq \lambda_2 \leq \lambda_3$, we can rescale the model by $(\sqrt{\lambda_1 \cdot \lambda_3})^{-1/6}$ to obtain a new model whose covariance matrix has as its smallest and largest eigenvalues the reciprocals $\sqrt{\lambda_1/\lambda_3}$ and $\sqrt{\lambda_3/\lambda_1}$.

Lemma 1: Given a continuous function f defined on M , there exists an even partition of M into subsets M^+ and M^- and a value μ such that $|M^+| = |M^-|$ and $f(p^+) \geq \mu \geq f(p^-)$ for all $p^+ \in M^+$ and all $p^- \in M^-$.

Proof: To prove that such a decomposition must exist, we define the function $F(t)$ that gives the area of the subset of M with value less than or equal to t :

$$F(t) = \left| f^{-1} \left((-\infty, t] \right) \right|$$

Then $F(t)$ is a non-decreasing, right-continuous function that starts at 0 and grows to $|M|$, and is discontinuous at points t_0 such that $|f^{-1}(t_0)| > 0$. Set Φ to be the closure of the set of values t for which $F(t) \leq |M|/2$. Since $F(t)$ is monotonic we know that $\Phi = (-\infty, \mu]$, for some value μ . Then for all $t \geq \mu$ we have $F(t) \geq |M|/2$ and for all $t < \mu$ we have $F(t) \leq |M|/2$. If $F(\mu) = |M|/2$ we can set M^- equal to the inverse image of f on the range $(-\infty, \mu]$, and we can set $M^+ = M - M^-$. Otherwise the function $F(t)$ is discontinuous at μ and we must have $|f^{-1}(\mu)| \geq F(\mu) - |M|/2$. Thus we can set M^+ to be the union of the inverse image of f on the range (μ, ∞) and any subset of $f^{-1}(\mu)$ that has area $F(\mu) - |M|/2$.

Lemma 2: Given a partition of M into equal sized subsets M^+ and M^- , the variance across M^+ and M^- is at least as large as half the

variance within M^+ and half the variance within M^- . That is, if:

$$\begin{aligned} I_M^{++}(v) &= \int_{M^+} \int_{M^+} \langle p-q, v \rangle^2 dpdq \\ I_M^{+-}(v) &= \int_{M^+} \int_{M^-} \langle p-q, v \rangle^2 dpdq \\ I_M^{-+}(v) &= \int_{M^-} \int_{M^+} \langle p-q, v \rangle^2 dpdq \\ I_M^{--}(v) &= \int_{M^-} \int_{M^-} \langle p-q, v \rangle^2 dpdq \end{aligned}$$

then we must have:

$$I_M^{++}(v) \leq 2I_M^{+-}(v) \quad \text{and} \quad I_M^{-+}(v) \leq 2I_M^{--}(v).$$

Proof: We show that $I_M^{--}(v) \leq 2I_M^{+-}(v)$, by integrating $I_M^{+-}(v)$ over M^- and using the triangle inequality.

$$\begin{aligned} 2I_M^{+-}(v) &= 2 \int_{M^+} \int_{M^-} \langle p-q, v \rangle^2 dpdq \\ &= \frac{1}{|M^-|} \int_{M^-} \int_{M^-} \int_{M^+} \left(\langle p^- - p^+, v \rangle^2 + \langle q^- - p^+, v \rangle^2 \right) dp^+ dp^- dq^- \end{aligned}$$

By the triangle inequality, we know that:

$$\langle p^- - p^+, v \rangle^2 + \langle q^- - p^+, v \rangle^2 \geq \langle p^- - q^-, v \rangle^2$$

so that:

$$\begin{aligned} 2I_M^{+-}(v) &\geq \frac{1}{|M^-|} \int_{M^-} \int_{M^-} \int_{M^+} \langle p^- - q^-, v \rangle^2 dp^+ dp^- dq^- \\ &= \frac{|M^+|}{|M^-|} \int_{M^-} \int_{M^-} \langle p^- - q^-, v \rangle^2 dp^- dq^- \\ &= I_M^{--}(v) \end{aligned}$$

as desired. The proof for $I_M^{++}(v) \leq 2I_M^{+-}(v)$ is analogous.

Anisotropic Rescaling: Given a model M , we set C_M to be the covariance matrix of M , $0 < \lambda_1 \leq \lambda_2 \leq \lambda_3 = 1/\lambda_1$ to be the eigenvalues of C_M , and $B_M = C_M^{-1/2}$ to be the inverse square root of C_M . Applying B_M to the model M , we obtain a model whose variance in direction v is given by:

$$\begin{aligned} \text{Var}(B_M(M), v) &= \int_{B_M(M)} \int_{B_M(M)} \langle p-q, v \rangle^2 dpdq \quad (1) \\ &= \int_M \int_M \langle p-q, B_M(v) \rangle^2 b(p)b(q) dpdq \end{aligned}$$

where $b(p)$ is the differential change of area at the point M and must satisfy:

$$\frac{\sqrt{\lambda_1}}{\sqrt{\lambda_2}} \leq b(p) \leq \frac{1}{\sqrt{\lambda_1 \cdot \lambda_2}}.$$

Using the fact that each summand in Equation 1 is positive, we can apply the above inequalities to get:

$$\begin{aligned} \int_M \int_M \langle p-q, B_M(v) \rangle^2 dpdq \frac{\lambda_1}{\lambda_2} &\leq \text{Var}(B_M(M), v) \\ \text{Var}(B_M(M), v) &\leq \int_M \int_M \langle p-q, B_M(v) \rangle^2 dpdq \frac{1}{\lambda_1 \cdot \lambda_2} \end{aligned}$$

so that

$$\frac{\lambda_1}{\lambda_2} \leq \text{Var}(B_M(M), v) \frac{1}{\lambda_1 \cdot \lambda_2}.$$

We observe that when we rescale the model so that minimal and maximal eigenvalues of the covariance matrix are reciprocals, the minimal eigenvalue is no smaller than λ_1 so that transforming M by B_M cannot make the minimal variance smaller, nor can it make the maximal variance larger.

To show that the minimal and maximal eigenvalues must actually converge to 1, we use the lemmas above. To do this, we use the function $b(p)$ and Lemma 1 to evenly partition M into M^+ and M^- and obtain a value μ satisfying:

$$\sqrt{\frac{\lambda_1}{\lambda_2}} \leq b(p^-) \leq \mu \leq b(p^+) \leq \sqrt{\frac{1}{\lambda_1 \cdot \lambda_2}}$$

for all $p^- \in M^-$ and $p^+ \in M^+$. (Though $b(p)$ is not continuous on M , it is only discontinuous on a closed subset with 0 area, so Lemma 1 still holds.) Expressing the variance of $B_M(M)$ in the direction v in terms of this partition we get:

$$\begin{aligned} \text{Var}(B_M(M), v) &= \int_{M^-} \int_{M^-} \langle p^- - q^-, B_M(v) \rangle^2 b(p^-)b(q^-) dp^- dq^- \\ &\quad + 2 \int_{M^-} \int_{M^+} \langle p^- - p^+, B_M(v) \rangle^2 b(p^-)b(q^+) dp^- dp^+ \\ &\quad + \int_{M^+} \int_{M^+} \langle p^+ - q^+, B_M(v) \rangle^2 b(p^+)b(q^+) dp^+ dq^+ \end{aligned}$$

This allows us to bound the minimal variance by:

$$\begin{aligned} \text{Var}(B_M(M), v) &\geq I_M^{--}(B_M(v)) \frac{\lambda_1}{\lambda_2} + 2I_M^{+-}(B_M(v)) \mu \sqrt{\frac{\lambda_1}{\lambda_2}} + I_M^{++}(B_M(v)) \mu^2. \end{aligned}$$

Since $I_M^{--}(B_M(v)) + 2I_M^{+-}(B_M(v)) + I_M^{++}(B_M(v)) = 1$, since $\sqrt{\lambda_1/\lambda_2} \leq \mu$, and since $2I_M^{+-}(B_M(v)) \geq I_M^{--}(B_M(v))$, it follows that the minimum variance is bounded by:

$$\frac{\lambda_1/\lambda_2 + \mu \sqrt{\lambda_1/\lambda_2}}{2} \leq \text{Var}(B_M(M), v).$$

In a similar manner we can get an upper bound for the variance:

$$\frac{\lambda_1}{\lambda_2} + \mu \sqrt{\frac{\lambda_1}{\lambda_2}} \leq \text{Var}(B_M(M), v) \leq \frac{1}{\lambda_1 \lambda_2} + \mu \sqrt{\frac{1}{\lambda_1 \lambda_2}}$$

Isotropically rescaling $B_M(M)$ to get a model \tilde{M} with minimal and maximal variances that are reciprocals, we get:

$$\lambda_1 \sqrt{f(\mu)} \leq \text{Var}(\tilde{M}, v) \quad \text{with} \quad f(t) = \frac{1+t\sqrt{\lambda_2/\lambda_1}}{1+t\sqrt{\lambda_2 \cdot \lambda_1}}.$$

In order to find the minimum of the variance, we compute the derivative:

$$f'(t) = \frac{\sqrt{\lambda_2/\lambda_1} - \sqrt{\lambda_2 \cdot \lambda_1}}{(1+t\sqrt{\lambda_2 \cdot \lambda_1})^2}.$$

Since $\lambda_1 \leq 1$, the derivative is never negative, and hence the variance of \tilde{M} is minimized when μ is as small as can be, which is to say $\mu = \sqrt{\lambda_1/\lambda_2}$. In this case we get:

$$\lambda_1 \sqrt{\frac{2}{1+\lambda_1}} \leq \text{Var}(\tilde{M}, v) \leq \frac{1}{\lambda_1} \sqrt{\frac{1+\lambda_1}{2}}.$$

Thus, the minimal and maximal variances of the model are guaranteed to converge to 1, and the iterative method described in Section 3 is guaranteed to converge to a model with variance 1 in every direction.