

Dynamic Multi-View Exploration of Shape Spaces

Stef Busking, Charl P. Botha and Frits H. Post

Delft University of Technology, the Netherlands

Abstract

Statistical shape modeling is a widely used technique for the representation and analysis of the shapes and shape variations present in a population. A statistical shape model models the distribution in a high dimensional shape space, where each shape is represented by a single point.

We present a design study on the intuitive exploration and visualization of shape spaces and shape models. Our approach focuses on the dual-space nature of these spaces. The high-dimensional shape space represents the population, whereas object space represents the shape of the 3D object associated with a point in shape space.

A 3D object view provides local details for a single shape. The high dimensional points in shape space are visualized using a 2D scatter plot projection, the axes of which can be manipulated interactively. This results in a dynamic scatter plot, with the further extension that each point is visualized as a small version of the object shape that it represents. We further enhance the population-object duality with a new type of view aimed at shape comparison. This new “shape evolution view” visualizes shape variability along a single trajectory in shape space, and serves as a link between the two spaces described above.

Our three-view exploration concept strongly emphasizes linked interaction between all spaces. Moving the cursor over the scatter plot or evolution views, shapes are dynamically interpolated and shown in the object view. Conversely, camera manipulation in the object view affects the object visualizations in the other views. We present a GPU-accelerated implementation, and show the effectiveness of the three-view approach using a number of real-world cases. In these, we demonstrate how this multi-view approach can be used to visually explore important aspects of a statistical shape model, including specificity, compactness and reconstruction error.

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [Computer Graphics]: Three-Dimensional Graphics and Realism—Applications

1. Introduction

Shape spaces are continuous higher dimensional domains where each position represents a complete 2 or 3-dimensional object surface [Ken84]. Statistical shape models (SSMs) represent whole populations of objects by modeling them as distributions in shape space. The most well-known application of this type of modeling is the Active Shape Model (ASM) [CCTG95]. An ASM models the distribution in shape space with its principal components, and is used to locate similar shapes in volume datasets.

Shape spaces and statistical shape models are an important concept in image segmentation, object classification and recognition, the study of anatomical variation and many other areas where the understanding or modeling of the variability in a whole population of shapes is required. Shape models can be used to represent existing (input) shapes, or

they can be used to synthesize new shapes similar to the input shapes.

Currently, visualization of shape spaces is done using straight-forward methods, such as showing object shapes regularly sampled on an axis in shape space, or showing a scatter plot of the input shapes over the first two principal axes. A more comprehensive visualization approach would contribute significantly to the understanding of shape variability and of the behavior and quality of statistical shape models. The need for better visualization is further accentuated by cognitive demands made by the high-dimensional nature of shape space. In spite of these observations, there are currently no examples in literature of visualization applications that focus on comprehensive shape space exploration.

In this paper, we present just such a visualization applica-

tion that enables the visual exploration of shape spaces and shape models. For such a tool, we identify the following requirements, which are further detailed in section 3:

1. The visualizations should provide insight into the high-dimensional structure of the population in shape space.
2. The tool should visualize the statistical shape model and its relation to the population. Important performance aspects of the shape model should be visually verifiable.
3. Information should be presented at both the global (population) and the local (object) level whenever possible.

In the following, we first discuss work related to the visualization of shape models and high-dimensional spaces. We then discuss these requirements in detail and present our contributions, which are:

- A strongly-linked multi-view approach to visual shape model exploration and validation (section 4).
- The shape evolution view: a new type of view in-between shape space and object space allowing direct visual comparison and visualization of a single shape-space direction or trajectory (section 4.3).
- A GPU-accelerated implementation of natural neighbor interpolation, allowing real-time continuous exploration of the population in shape space by smoothly interpolating between individuals (section 5.2).

A prototype implementation of our multi-view approach was created as described in section 5. Using this implementation, we demonstrate the effectiveness of our approach in a number of real-world cases. Finally, we present conclusions and directions for future work.

2. Related work

Our work focuses on the visualization of the variation in a population of shapes by means of shape spaces, where a shape space is defined as the continuous higher dimensional space in which each input shape is represented by a single point [Ken84]. The most well-known use of this type of shape space is probably the Active Shape Model, an SSM used for segmentation, where the main modes of variation are found using principal component analysis [CCTG95].

Shape models have been extensively used for image segmentation both in medicine and other applications, for studying anatomical shape differences between populations [GGSK05, FPO*06], visualizing organ shape variation in 3D anatomical atlases [HH06] and studying evolutionary morphological changes [WAA*05].

In all applications, direct visualization of shape variability coupled with the flexible visual exploration of shape space would contribute to a better understanding of the shape variation, in both the local and global sense, and hence the characteristics of the shape model that is being used. However, examples in literature of comprehensive visualization approaches for shape spaces are scarce. General techniques

for high-dimensional visualization can be used to visualize a population of objects in shape space. An overview of techniques for high-dimensional visualization is given by Wong and Bergeron [WB97], while a more recent overview of multivariate techniques is given by Fuchs and Hauser [FH09]. However, such techniques do not take into account the fact that each high dimensional point also represents a two or three-dimensional shape, the local details of which are often important.

Shape variation can be shown at a local level with color mapping, or by placing ellipsoids at all points on the average or mean shape, representing the Gaussian variation of each point in shape space [FPO*06, FAP*07]. The distribution in an SSM shape space can be shown, for the first two modes at least, with a simple scatter plot, whereas the shape variation over a particular mode can be shown with a number of shapes, regularly sampled over that mode, shown side-by-side [CCTG95], or by animating object shape changes over the main modes of variation. Lamecker et al. extended this basic approach by also animating between the mean shape and the training shapes, although little detail is given on exactly how this is done [LSL*04]. A good overview of other mesh morphing techniques, although not in the context of shape spaces, is given by Alexa [Ale02].

Kilian et al. used shape spaces to interpolate between and extrapolate from input shapes [KMP07]. Their method can also be used to explore the shape space spanned by a number of input models by allowing interaction, primarily curve-drawing, on a specially constructed 2D polygon with the input models at its vertices. Our approach differs from that of Kilian et al. in three ways: 1: We project all shape space points on a 2D plane that can be smoothly and arbitrarily positioned in shape space, inspired by the method of Blaas et al. [BBP07]. 2: Our method supports real-time and smooth visualization of the shape changes represented by interaction on the shape space projection, without any pre-processing. 3: We make use of a coordinated multiple view approach to integrate three different visualization techniques for exploring the shape space.

3. Assumptions and Requirements

Creation of a shape model involves first establishing a point correspondence relation between a population of input shapes. Next, these shapes are parameterized and a statistical model is fitted to the resulting distribution in a high-dimensional shape space. Heimann and Meinzer [HM09] give a good overview of the various techniques that exist for the creation of such shape models. For this paper, we assume a shape model has already been created. In particular, we use active shape models based on landmarks obtained using the GAMEs algorithm [FOP*07] to demonstrate our approach. However, our techniques can be applied to any shape model that satisfies the following assumptions:

- Each shape can be represented as a point in an N-dimensional shape space. Shapes in an active shape model use a fixed set of landmarks; shapes in shape space are defined as the vector of coordinates for all landmarks.
- Shape space points can be interpolated to form new shapes. We assume linear interpolation in shape space is valid. If non-linear spaces are used, the interpolator used should take this non-linearity into account.
- The distribution of points in shape space is modeled using a statistical distribution. As in ASMs, we use PCA to derive the parameters of a multivariate Gaussian distribution. Other statistical distributions could be used, provided that their contours can be plotted in arbitrary linear projections of the shape space.

In the following, we examine the three requirements defined in the introduction in detail, based on these assumptions:

1. *Provide insight into the high-dimensional structure of the population in shape space.* Statistical shape modeling is based on the assumption that the population being modeled follows a certain distribution in shape space, such as the multivariate Gaussian distribution in our models. Visualization of this high dimensional space can help to visually verify such an assumption. Furthermore, it can allow a user to spot clusters or outliers within the population. Dimensional reduction techniques such as PCA can help in selecting appropriate projections for visualizing this space.

2. *Visualize the shape model and its relation to the population.* As noted, SSMs model a statistical distribution in shape space. This means shapes can be described in terms of their likelihood with respect to the model. Additionally, a number of important measures can be computed describing the performance of a given shape model with respect to modeling a given population [DTA*03, FPO*06]:

Compactness describes the accuracy with which a given number of model parameters can be used to model a population. The *reconstruction error* describes the mismatch between the training shapes and their approximations given by the model. The *generalization error* is similar to the reconstruction error, but approximates the error reconstructing unseen shapes in the population. Finally, *specificity* describes the difference between the variation in the population and that in the shape space generated by the model.

These measures were defined originally to yield a single number per shape model [DTA*03]. We have adapted the definitions to show the local behavior of the measures on a per-shape or even per-point basis, in order to explain the global result for a given shape model.

3. *Present information at both a global and a local level.* Shape spaces have a dual nature in that they can be seen both as a high-dimensional space, where shapes are single points, and as a 3D space, in which the 3D objects represented by such points exist. Similarly, various aspects of a shape can

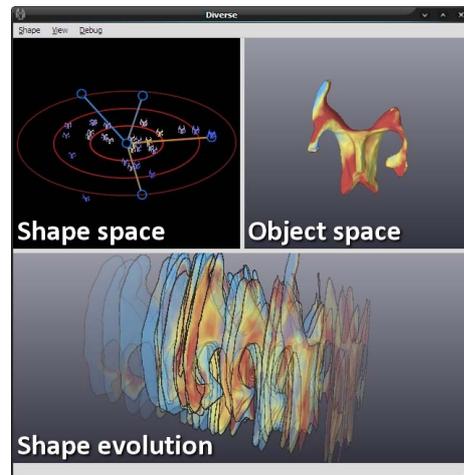


Figure 1: Prototype implementation of our multi-view exploration approach, showing the shape space, object space and evolution views. Due to screen space constraints, the latter can be switched between the configurations shown in figure 4. Multiple evolution views can be added if preferred.

be visualized both on a global (population) or local (object) level. Examples are the deviation from the mean shape or the reconstruction error, both of which can be computed per object or per point on the object.

4. Multi-view exploration

We propose a three-view approach to satisfy these requirements (figure 1). Overview and global-level visualization are provided by a visualization of the high-dimensional shape space. Local details are provided by a 3D object view, showing a single shape.

Variability, however, is hard to visualize on a local scale. For this purpose we introduce a new type of view: the *shape evolution view*. This view shows a combination of object space and shape space dimensions. Object shapes are reduced to two dimensions by extracting their silhouettes, and the third dimension is used to represent a trajectory through shape space. Essentially this creates a visualization of “2D + variability” space.

In this paper we use “global” to mean “per object”. For information at the shape model or population level, it would be straightforward to aggregate performance measures or other statistics over all individuals (each resulting in a single value) and display these, as is the current approach in the field. For example, model-level compactness can be visualized in a standard compactness plot as used by Davies et al. [DTA*03]. The aim of our visualizations is not to replace such statistical methods. Rather, we provide tools that allow results to be examined at all levels, allowing deeper insight

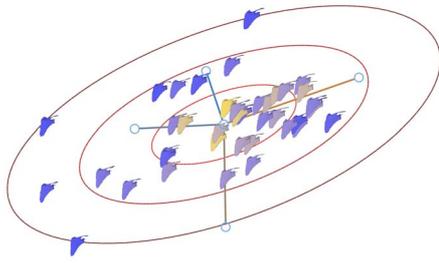


Figure 2: The shape space scatter plot view, visualizing a population of segmentations of the human scapula. Colors indicate the likelihood of each shape with respect to the Gaussian model, which itself is visualized by ellipses representing multiples of the standard deviation.

into the distribution-related and shape-related details of such results.

The visualizations used in these three views are discussed in the following sections. Section 4.4 discusses linking between the views, which enables interactive exploration of the shape space.

4.1. Shape space view

The goal of the shape space view is to give an overview of the high dimensional shape space. Within this view, the user can explore the structure of the population, the statistical distribution modeled by the shape model and global-level information about both.

We visualize shape space using a scatter plot, as shown in figure 2. The scatter plot presents a 2D projection of the high-dimensional points making up the population. We base this projection on the technique presented by Blaas et al. [BBP07], in which the plane of the projection can be defined arbitrarily and interactively. To do this, the user manipulates 2D representations of any number of axes or vectors in the shape space. From these, a 2D coordinate frame is created consisting of linear combinations of these vectors.

Specifically, for a point defined by shape vector s , the 2D projection $P(s)$ is given by $P(s) = AFs$, where F is a projection reducing the high-dimensional shape space to the subspace spanned by the current set of axes, and A is the $2 \times N$ matrix with the user-defined 2D vectors for each axis as its columns. Manipulation of an axis corresponds to rotation (and/or scaling) within the reduced subspace. To enhance this perceptually, we change the drawing order of shapes during manipulation to match the order of their projections along the axis being manipulated.

Central to our approach is that the projections of points are

updated in real-time during interaction. This way, dependencies between points in shape space and the axis being manipulated can be learned intuitively by observing the speed and direction of their movement relative to the axis. A user therefore often does not need to examine all possible projections; manipulation of each axis in turn suffices to locate features like outliers, and projections that highlight such features can be created incrementally.

To further aid the user in locating useful projections, the axes of the scatter plot can be toggled between dimensions from the original shape space, or from the reduced space created by the shape model. As our shape models use PCA, the resulting reduced set of axes is guaranteed to create a projection showing maximum variability. Alternative techniques for locating good viewpoints, such as the Grand Tour [Asi85] and Projection Pursuit [FT74] methods, could easily be integrated in our approach.

In addition to the points in the population, the projection of the shape model's Gaussian distribution is visualized using a set of elliptical contours representing multiples of the distribution's standard deviation. This projection can be computed by projecting the covariance matrix for the population to the scatter plot frame: $P(C) = AFCF^T A^T$. Principal component analysis is then performed using the projected covariance matrix to extract the axes for the ellipses and the 2D standard deviation of the model.

We enhance the basic scatter plot visualization by rendering each high dimensional shape space point as a small representation of the 3D object represented by that point. These small multiples [Tuf90] can be color coded to visualize additional scalar information, such as the likelihood of the shape with respect to the shape model or the shape model performance measures described in section 3.

4.2. Object space view

The object space view visualizes the 3D object corresponding to a single, arbitrary point in shape space. The purpose of this view is to give insight into local details of the shape and/or measures (see figure 3).

Deformation with respect to the mean, for instance, can be visualized for each point on the object by computing the length of the 3D deformation vector (a sub-vector of the shape vector) corresponding to that point. Similarly, we can compute the local distance of each point to the corresponding point on any reference shape as the length of the difference between their deformation vectors. A special instance of this is the local reconstruction error, where the reference shape is defined as the shape reconstructed using a limited number of shape model parameters, computed by projecting the shape to a basis of eigenvectors and then back to the original shape space. For visualization purposes, distances are made signed with respect to the angle between the deformation vector and the local surface normal, and visualized

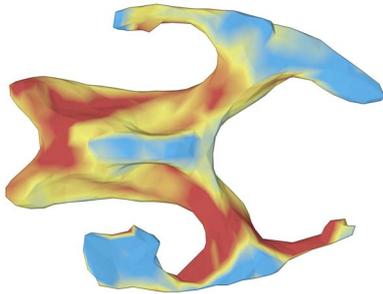


Figure 3: The object space 3D view, here showing per-point reconstruction errors for a brain ventricle shape using a color map. Blue indicates areas where the actual shape is larger than the reconstructed surface, yellow indicates similar size and red indicates areas where it is smaller.

using a isoluminant and perceptually linear color map, as suggested by Borland and Taylor [BT07].

In this view, the camera can be manipulated interactively. Alternatively, our system can automatically generate an interesting viewpoint by performing principal component analysis on the 3D deformation vectors. By choosing a viewing direction orthogonal to the plane spanned by the first two principal components, a view can be created that shows maximal deformation in the image plane.

4.3. Shape evolution view

The shape space view provides global information, while the object view provides local details for a single shape. Using only these two views, it is hard to gain insight into the variability of such local details in the population. We introduce the *shape evolution view* as a way to solve this limitation.

The evolution view visualizes local shape variability along a single direction or trajectory through shape space. This way, the high-dimensional space is essentially reduced to a single dimension, leaving two dimensions for visualizing object shapes along the trajectory. Our current implementation supports only straight lines for interpolation between shapes, but extension to other curves, for instance user-defined strokes or shape-space geodesics (as used by Kilian et al. [KMP07]), is relatively straightforward.

The goal of the evolution view is to provide a detail level for trends in shape space or comparisons between sets of shapes. To enable these two applications, we define two types of visualization of shape space trajectories:

- *Trend trajectories* - trends can be explored by visualizing the population itself, as seen from a user defined trajectory through shape space. This means we visualize the actual individuals, with positions along the trajectory determined by their projections.

- *Comparison trajectories* - direct comparisons between shapes are possible by generating (synthetic) objects interpolating between their shapes. This is essentially the direct visualization of a line or curve through the high-dimensional shape space connecting the points being compared.

The combined “2D + variability” space of the evolution view can be shown in a number of visualization configurations, shown in figure 4. Each of these serves a different purpose in analyzing local variability:

- *Side-by-side* - shapes along the trajectory are visualized using a grid of small multiples, in order of appearance along the trajectory. This configuration can be used to get an overview of the population.
- *Overlaid contours* - by directly overlaying all shapes and only visualizing their projected contours, we create a 2D visualization similar to “onion-skinning” techniques used in animation. Contours are color coded using a perceptually linear black-to-white color map to show their ordering with respect to the trajectory. The resulting visualization allows intuitive comparison of object shapes along the trajectory. Due to the sparseness of contours it is not possible to visualize other measures in this configuration.
- *Shape stack* - the most direct interpretation of 2D + variability is the 3D structure created by stacking all object contours along the third dimension. Offsets between shape “slices” are determined by their positions along the trajectory, thereby visualizing the distribution of shapes. Optionally, interpolation can be used to create a continuous shape, representing a continuous morph between all shapes along the trajectory. This configuration provides a level in-between the other two, in that local details can be compared between shapes without losing information about the relation of those shapes to the population.

4.4. Linked interaction

A key part of our multi-view exploration approach is strongly linked interaction between all views (figure 1). Such linkage helps to maintain overview and provides continuous feedback about the relations between the different visualizations displayed in each view. Linked interaction is used in four scenarios:

- *Shape selection* - the object view displays only a single shape. By simply hovering the mouse over the shape space view, the object view smoothly deforms between shapes encountered along the path of the cursor, allowing intuitive exploration of the shape variability in the population. This is achieved by using a GPU-based interpolation technique, detailed in section 5.2. Alternatively, shapes can be selected in the evolution view in order to examine them in detail. When a shape is selected in either view, the projected position of the shape is highlighted in the shape space scatter plot, and the selected shape is integrated and/or highlighted in the evolution view visualization.

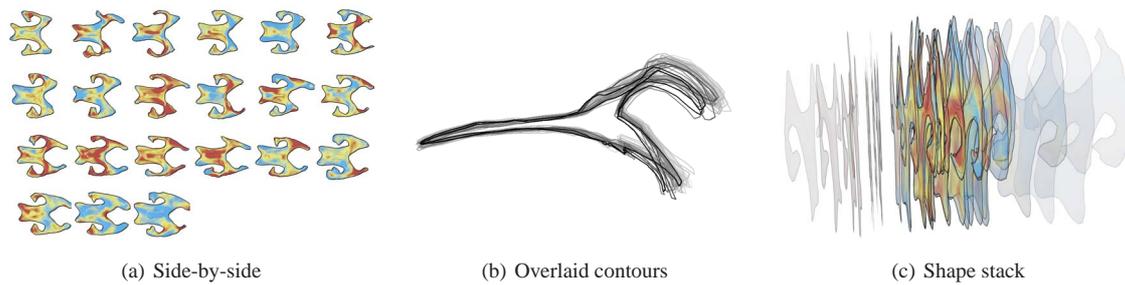


Figure 4: The shape evolution view and its configurations, used to examine variation in the scapula and brain ventricle datasets. (a) Side-by-side display of a subset of the brain ventricle population, ordered by projection along the first principal component. (b) Overlaid contours for a side view of the scapula population, demonstrating that most variation occurs in the acromion and coracoid process. Ordering along the first principal component is visualized using a black-to-white scale. (c) Shape stack visualization of the shapes in (a), showing an increasing trend in the overall size of the shapes along the first eigenvector.

- *Trajectory selection* - in the current implementation, the trend trajectory shown in the evolution view is linked to the x-axis of the scatter plot. It can therefore be defined by manipulating the projection as described in section 4.1. Alternatively, a shape comparison trajectory can be defined by selecting two shapes using the selection methods mentioned above. As mentioned in section 4.3, sketch-based curve selection can be added in a future extension.
- *Object space camera manipulation* - viewing parameters for the object shape visualizations in all views are linked. The camera for these projections can be manipulated in the object space, updating the other views in real-time. This also affects the viewpoint used for contour extraction in the overlay and shape stack configurations of the evolution view.
- *Dimensionality selection* - the number of axes shown in the shape space view directly determines the number of dimensions used for determining shape likelihood and reconstruction errors. The number of axes can be adjusted interactively by the user, with visualizations in all views updating in real-time.

5. Technical details

We implemented a prototype of our multi-view exploration approach (figure 1) using C++ and OpenGL. The OpenGL Shading Language was used to implement all GPU-based algorithms and the IT++ math library (<http://itpp.sf.net/>) was used for CPU-side linear algebra operations.

5.1. Shape visualization

For the purposes of statistical analysis, populations in shape space are often centered on the mean. Therefore, object shapes in a statistical shape model are defined as the sum of the mean shape and a high-dimensional shape vector. In case dimensional reduction has been applied, a shape vector

first has to be transformed back to the original space in order to visualize its corresponding object.

We visualize shapes in object space using a polygonal model of the mean shape, which is deformed on the GPU to match any given shape. Deforming on the GPU allows us to simultaneously compute derived measures to be visualized in object space, such as local reconstruction errors.

The surface normals of the mean shape are no longer valid after deformation. We use a deferred shading pass to compute normals in screen space. When visualizing a large number of shapes, individual objects may be hard to distinguish in the scatter plot and evolution views due to overlap. To alleviate this, the same deferred shading approach used to render the objects can be used to add contours or halos to each of the object miniatures.

5.2. Shape interpolation

As described in section 4.4, we allow a user to select and morph between shapes in the shape space view using interpolation. The interpolation technique should enable interactive performance with minimal pre-computation, as both querying for shapes and updating the (projected) positions of all original points should be real-time operations. To achieve this, we created a GPU-accelerated implementation of natural neighbor interpolation [Sib81]. Natural neighbor interpolation was chosen because it results in smooth interpolation, and can provide results even outside the convex hull of the projected points.

Our approach is image-based rather than geometry-based, as we do not require the potentially higher accuracy of the latter approach and can therefore avoid its higher computational complexity. Our approach differs from that presented by Fan et al. [FEK*05] in that we make use of hardware occlusion queries to avoid slow reads from graphics memory.

Natural neighbor interpolation is based on the Voronoi

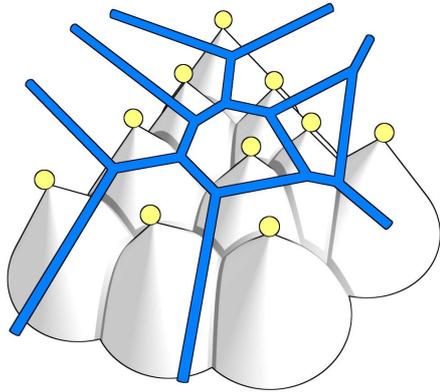


Figure 5: Rendering the Voronoi tessellation of a set of points (shown in yellow) using cones. When viewed from the top, standard depth-buffering ensures the pixels covered by a cone belong to the Voronoi area for the corresponding point.

tessellation of a given set of points. An image of the Voronoi tessellation can be constructed by rendering each point as a cone seen from the top, centered at the point's projected position. This way, the depth values for the cone's pixels correspond to the distance between that pixel and the original point. By making use of a standard depth buffer, parts of the cone which are closer to a different point will be removed, as their pixels will be covered by the cone for the other point (see figure 5). We generate the cone depth values in a full-screen fragment shader pass in order to avoid discretization artifacts that occur when using polygonal approximations.

During rendering of the Voronoi tessellation we use the stencil buffer to record an identifier for each pixel corresponding to the visible cone (i.e. closest point) for that pixel.

In natural neighbor interpolation, we consider insertion of a new point into the Voronoi tessellation. The interpolated value is a weighted summation of the values v_i at the original points p_i . These weights are computed corresponding to the ratio between the Voronoi area $A'(x)$ created by the new point x and the overlap between this area and the areas for the original points in the original tessellation $A(p_i)$ (see figure 6):

$$v(x) = \sum_{i=1}^N \frac{A'(x) \cap A(p_i)}{A'(x)} v_i$$

We use occlusion queries as a fast way to determine these areas. An occlusion query is a fast hardware-accelerated method to query the number of pixels (potentially) modified by a drawing operation. The total area for the new point can be approximated by simply inserting the corresponding cone into the Voronoi tessellation image. The overlap with existing areas can be computed by repeating the process, but

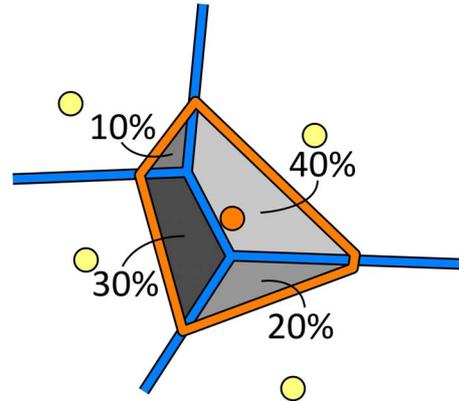


Figure 6: Weights for natural neighbor interpolation are computed as the ratio between the area of the Voronoi cell corresponding to a given point if it were inserted into the tessellation (orange), and the overlap between this cell and each of the existing cells in the diagram (gray).

using the stencil buffer to restrict drawing to the area covered by each existing cone. After running all queries, results are collected, weights are computed and the interpolation is performed on the shape vectors in the population.

6. Results and validation

In this section, we demonstrate how our approach enables exploration of shape space using two real-world datasets. The first consists of two sets of segmentations of the ventricles in the human brain. One set consists of 28 control individuals, the other of 58 patients with Alzheimer's disease. For each individual, positions are given for 949 points on the ventricle surface. The second dataset consists of 39 segmentations of the human scapula, with 379 points per segmentation surface.

We will use these as example use cases to demonstrate how our approach satisfies each of the requirements defined in section 3.

1. *Provide insight into the high-dimensional structure of the population in shape space.* The interactive scatter plot projection of the shape space view can be used to explore the distribution of the population in the high dimensional shape space. By choosing an appropriate projection, clusters, trends and outliers can be identified. Figure 7 shows an example of an outlier, while figure 2 shows a skewed distribution of shapes that may not satisfy the Gaussian distribution assumption.

2. *Visualize the statistical shape model and its relation to the population.* For any population, the shape space view displays the projected Gaussian distribution of the shape model, allowing visual inspection of the model's statistical properties. Shapes in the population with low likelihood can be

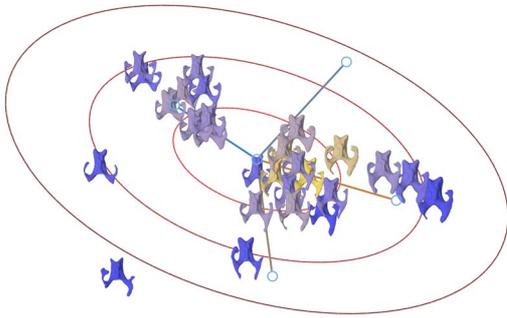


Figure 7: Outliers in the population can be highlighted in the shape space view, both by selecting an appropriate projection and by coloring shapes according to their likelihood. High-valued shapes are yellow, low values are blue.

found by changing the projection, and are highlighted in all projections when coloring shapes by their likelihood (see figure 7).

Alternatively, shapes can be colored by *reconstruction error*. As the number of shape model parameters is directly linked to the number of axes in the shape space view, varying this number while visualizing reconstruction errors allows for visual inspection of the model's *compactness*, and helps in determining a good cut-off for the shape model's dimensionality (see figure 8).

Specificity is defined as the average distance of random shapes to the nearest shape in the population [DTA*03]. The nearest shape can be determined during interaction in the shape space view, and can be used as the reference shape. This way, the dissimilarity of arbitrary shapes to the population can be visualized on a local level.

For computing the *generalization error*, Ferrarini et al. [FPO*06] use a leave-one-out approach, where the generalization error for any object is essentially the reconstruction error of the object using a shape model based on the population with that object removed. Such models could be created in a pre-processing step, allowing the generalization error to be visualized in the shape space view in a way similar to the reconstruction error, but using a different model for each object in the computation.

3. Present information at both a global and a local level. The strong linking between all views in our exploration concept enables findings in the shape space view to be easily explored at a local level. By simply hovering the cursor over clusters or along trends in the shape space view, the local details of these features can be explored interactively in the object view. If a problematic shape is identified, such as the outlier in figure 7, the local views can be used to determine which parts of the object cause it to be an outlier. Similarly,

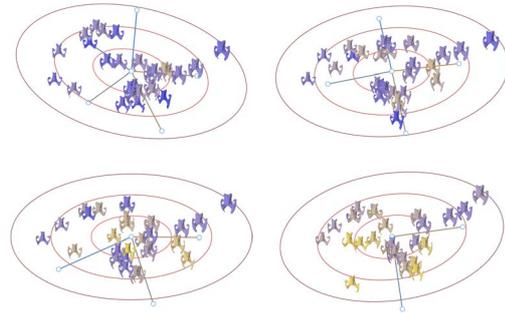


Figure 8: Inspecting compactness of the shape model by varying the number of axes. Shapes are colored by their reconstruction error, based on a reconstruction using the axes shown. When reducing the number of parameters of the model from 5 (top left) to 2 (bottom right), the number of shapes with high reconstruction errors (yellow) visibly increases.

the reconstruction error can be examined on a per-point basis, as seen in figure 3.

The evolution view allows trends to be explored at a local level (see figure 4). The three configurations serve different purposes. The side-by-side configuration is suitable for giving a quick overview of the entire population. Overlaid contours give insight into the local variability in the shape of parts of the object. Shape stacks give simultaneous insight into such trends and into the distribution of shapes along a user-defined direction in shape space.

The evolution view can also be used to determine if a certain local error occurs more than once in the population, or if, for instance, high reconstruction errors have various causes when examined locally. In order to facilitate exploration of such trends in local details, we allow the user to define a threshold on the reconstruction error. Parts of the object with values above this threshold will be highlighted in the visualization. While contours remain, the opacity of other parts of the objects is reduced to increase visibility of the areas with large errors.

An example of this can be seen in figure 9. As in figure 3, red indicates parts of the actual individuals that are larger than the reconstructed shapes while blue indicates smaller parts. Although many errors are unique to specific objects, several shapes show a red region in the area around the glenoid cup as well as red and blue areas around the acromion, indicating that the current shape model has trouble modeling these parts of the scapula.

7. Conclusions and future work

We presented a new method for visual shape space exploration and validation, based on multiple strongly-linked



Figure 9: The evolution view can be used to explore trends on a local level. Here, reconstruction errors for the scapula dataset are explored on a local level and compared over the population by showing a shape stack highlighting errors above a user-defined threshold.

views showing both global and local aspects simultaneously. Additionally, we make the following technical contributions:

- The shape evolution view: a new type of view visualizing “2D + variability” space, a combination of shape space and object space. This view allows for direct visual comparison of local details over a single shape-space direction or trajectory.
- A GPU-accelerated implementation of natural neighbor interpolation, allowing real-time continuous exploration of the population in shape space by smoothly interpolating between individuals (section 5.2).

The continuity provided by interpolation helps in initial exploration of shape changes and trends, understanding of which is further aided by visualizations in the evolution view. Continuity may also benefit other aspects of the visualizations, for instance, animations may help maintain overview when switching between evolution view configurations or scatter plot projections.

Although not considered for this work, our shape exploration approach could further benefit from the inclusion of filtering techniques. For example, in addition to highlighting single shapes, it would be helpful to allow subsets of shapes to be extracted and explored. Selections could be made manually, or based on clustering techniques applied to the high-dimensional population.

The complete implementation of our shape space exploration prototype is to be released as open source.

7.1. Future extensions

Although the shape modeling approach used in this paper is the most common in shape modeling literature, alternative

techniques have been proposed. Our framework should be extended to allow for the visual comparison of shape models using different parameterizations, landmark sets or statistical distributions. In particular, non-linear shape modeling techniques have been presented in recent literature, allowing for more flexibility in modeling non-Gaussian distributions of shape in a population [CKS01, COS06]. Most of these non-linear techniques consist of first non-linearly mapping shapes to a different space, in which normal PCA or other density estimators can be applied. This means we can either use our current techniques on the space resulting from the mapping. Alternatively, contours of the non-linear distributions can be shown in the original shape space, similar to the figures in the papers by Cremers et al., but using our interactive techniques to select the projection.

Different dimensional reduction techniques such as independent component analysis could also be integrated, to aid in selecting good projections of the high-dimensional shape space. Additionally, automated approaches could be integrated, such as the Grand Tour [Asi85] and Projection Pursuit [FT74] algorithms, or even intelligent methods aimed at locating specific patterns.

The current techniques for shape space interpolation remain close to the original points when extrapolating. This could be extended with more accurate extrapolation techniques to allow a user to magnify trends in the population by displaying shapes with “exaggerated” deformations. Boback et al. [BFHU09] enhanced extrapolation in the natural neighbor algorithm by dynamically adding extra “ghost points” to the population. Such a technique would be straightforward to implement on our GPU accelerated implementation, as the basic interpolation scheme remains identical. Alternatively, geodesic-based interpolation and extrapolation techniques could be used, such as those presented by Kilian et al. [KMP07].

Finally, we plan to apply our techniques to interactive shape-model-based segmentation by integrating surface fitting techniques. This way, possible (locally optimal) locations for a new object in shape space can be explored interactively, with both global and local feedback on, for instance, the reconstruction error or likelihood of a given reconstruction of the surface.

Acknowledgments

We are grateful to Dr. L. Ferrarini and Dr. J. Milles of the Department of Radiology, Leiden University Medical Center, for providing the brain ventricle shape models and population data, and to the Department of Orthopaedics, Leiden University Medical Center (Ir. E. van IJsseldijk, Dr. B.L. Kaptein, Dr. E.R. Valstar, Prof. P.M. Rozing) for providing the scapula dataset. This research is supported by the Netherlands Organization for Scientific Research (NWO), project number 643.100.503 “Multi-Field Medical Visualization”.

References

- [Ale02] ALEXA M.: Recent advances in mesh morphing. *Computer graphics forum* (2002).
- [Asi85] ASIMOV D.: The grand tour: a tool for viewing multidimensional data. *SIAM Journal of Scientific and Statistical Computing* 6 (1985), 128–143.
- [BBP07] BLAAS J., BOTHA C., POST F.: Interactive visualization of multi-field medical data using linked physical and feature-space views. *Proceedings EuroVis 2007* (2007), 123–130.
- [BFHU09] BOBACH T., FARIN G., HANSFORD D., UMLAUF G.: Natural neighbor extrapolation using ghost points. *Computer-Aided Design* 41, 5 (2009), 350–365.
- [BT07] BORLAND D., TAYLOR II R.: Rainbow color map (still) considered harmful. *IEEE Computer Graphics and Applications*, April (2007), 14–17.
- [CCTG95] COOTES T. F., COOPER D. H., TAYLOR C. J., GRAHAM J.: Active Shape Models - Their Training and Application. *Comp. Vision & Image Understanding* 61, 1 (1995), 38–59.
- [CKS01] CREMERS D., KOHLBERGER T., SCHNORR C.: Non-linear shape statistics via kernel spaces. *Lecture Notes in Computer Science* (2001), 269–276.
- [COS06] CREMERS D., OSHER S. J., SOATTO S.: Kernel Density Estimation and Intrinsic Alignment for Shape Priors in Level Set Segmentation. *International Journal of Computer Vision* 69, 3 (2006), 335–351.
- [DTA*03] DAVIES R. H., TWINING C. J., ALLEN P. D., COOTES T. F., TAYLOR C. J.: Shape discrimination in the hippocampus using an MDL model. *Information processing in medical imaging, proceedings 18* (juli 2003), 38–50.
- [FAP*07] FILLARD P., ARSIGNY V., PENNEC X., HAYASHI K. M., THOMPSON P. M., AYACHE N.: Measuring brain variability by extrapolating sparse tensor fields measured on sulcal lines. *NeuroImage* 34, 2 (januari 2007), 639–50.
- [FEK*05] FAN Q., EFRAT A., KOLTUN V., KRISHNAN S., VENKATASUBRAMANIAN S.: Hardware-Assisted Natural Neighbor Interpolation. In *Seventh Workshop Algorithm Engineering and Experiments (ALENEX), Proceedings* (2005).
- [FH09] FUCHS R., HAUSER H.: Visualization of Multi-Variate Scientific Data. *Computer Graphics Forum* 28, 6 (2009), 1670–1690.
- [FOP*07] FERRARINI L., OLOFSEN H., PALM W. M., VAN BUCHEM M. A., REIBER J. H. C., ADMIRAAL-BEHLLOUL F.: GAMES: growing and adaptive meshes for fully automatic shape modeling and analysis. *Medical Image Analysis* 11 (juni 2007), 302–314.
- [FPO*06] FERRARINI L., PALM W. M., OLOFSEN H., VAN BUCHEM M. A., REIBER J. H. C., ADMIRAAL-BEHLLOUL F.: Shape differences of the brain ventricles in Alzheimer's disease. *NeuroImage* 32 (september 2006), 1060–1069.
- [FT74] FRIEDMAN J. H., TUKEY J. W.: A Projection Pursuit Algorithm for Exploratory Data Analysis. *IEEE Transactions on Computers* 23, 9 (1974).
- [GGSK05] GOLLAND P., GRIMSON W. E. L., SHENTON M. E., KIKINIS R.: Detection and analysis of statistical differences in anatomical shape. *Medical image analysis* 9, 1 (2005), 69–86.
- [HH06] HACKER S., HANDELS H.: Representation and visualization of variability in a 3D anatomical atlas using the kidney as an example. In *SPIE Medical Imaging 2006, Visualization, Image-Guided Procedures, and Display* (San Diego, 2006), Cleary K., Galloway R., (Eds.), vol. 7, SPIE, pp. 61410B–1 – 61410B–7.
- [HM09] HEIMANN T., MEINZER H.-P.: Statistical shape models for 3D medical image segmentation: a review. *Medical image analysis* 13, 4 (2009), 543–63.
- [Ken84] KENDALL D. G.: Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces. *Bulletin of the London Mathematical Society* 16, 2 (maart 1984), 81–121.
- [KMP07] KILIAN M., MITRA N. J., POTTMANN H.: Geometric modeling in shape space. *ACM Transactions on Graphics (TOG)* 26, 3 (2007).
- [LSL*04] LAMECKER H., SEEBASS M., LANGE T., HEGE H.-C., DEUFLHARD P.: Visualization of the variability of 3D statistical shape models by animation. *Studies in health technology and informatics* 98 (januari 2004), 190–196.
- [Sib81] SIBSON R.: *A brief description of natural neighbor interpolation*. John Wiley and Sons, New York, 1981, ch. 2, pp. 21–36.
- [Tuf90] TUFTE E. R.: *Envisioning Information*. Graphics Press, 1990.
- [WAA*05] WILEY D., AMENTA N., ALCANTARA D., GHOSH D., KIL Y., DELSON E., HARCOURT-SMITH W., ROHLF F., ST. JOHN K., HAMANN B.: Evolutionary Morphing. In *IEEE Visualization, Proceedings* (2005), IEEE, pp. 431–438.
- [WB97] WONG P. C., BERGERON R. D.: 30 Years of Multi-dimensional Multivariate Visualization. In *Scientific Visualization, Overviews, Methodologies, and Techniques* (Washington, DC, USA, 1997), IEEE Computer Society, pp. 3–33.