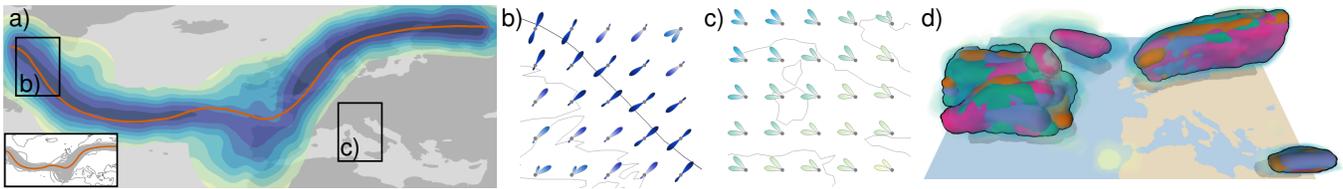


# Visualizing the Central Tendency of Ensembles of Shapes

Ismail Demir\* Mihaela Jarema† Rüdiger Westermann‡  
Computer Graphics and Visualization Group, Technische Universität München, Germany



**Figure 1:** a) Visualization of a 2D ensemble of wind velocity fields. The median iso-contour determined by our method is shown in red, and the local centrality with respect to the ensemble is color coded from dark blue (high centrality) to yellow (low centrality). For the marked regions, (b) and (c) show the directional distributions of vectors to closest points that were used to classify the centrality. d) For a 3D scalar ensemble field, the locally best matching median surfaces are combined in one single shape. Colors indicate different ensemble members.

## Abstract

We propose a new approach for analyzing the central tendency (centrality) of an ensemble of shapes in 2D or 3D space. Our approach provides means to determine the most central shape from a given set of shapes, to quantify the region-wise centrality of the shapes, and to compute a locally most representative shape. Unlike previous approaches, which build upon binary functions or signed distance fields to locate domain points with respect to orientable shapes, we introduce a closest point representation for the analysis of ensembles of shapes. By using this representation, our approach can handle arbitrary non-parametric shapes regardless of dimension and orientability. Shapes are first converted into an implicit representation based on vectors to closest surface points, and the resulting directional distributions are then used to perform region-wise classifications. Shapes are either analyzed separately by evaluating the classifications over the shape, or additional fields are derived from these classifications, in which specific shapes like the locally best mean are given as level-sets. We demonstrate the effectiveness of our approach on synthetic and weather forecast ensembles in 2D and 3D.

**Keywords:** Ensemble visualization, closest point representation, statistical summaries

**Concepts:** •Computing methodologies → Image processing;

## 1 Introduction

Ensemble or multi-run simulations are used to estimate the uncertainty inherent in the prediction of physical quantities, by providing

\*e-mail:ismail.demir@mytum.de

†email:mihaela.jarema@tum.de

‡email:westermann@tum.de

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org). © 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM.

SA '16 Symposium on Visualization, December 05 - 08, 2016, Macao

ISBN: 978-1-4503-4547-7/16/12

DOI: <http://dx.doi.org/10.1145/3002151.3002165>

a representative sample of the possible states that could evolve out of perturbed initial conditions and different models. From the variability of the ensemble members at a particular time step, the uncertainty of the current prediction can be estimated, and, by determining the prediction that best summarizes the ensemble, a central tendency of the ensemble can be conveyed.

Analyzing the variability of a scalar field ensemble is often performed by looking at spaghetti plots of specific features in the data, such as iso-contours. Each spaghetti plot shows the contours in all ensemble members at one single time step simultaneously. Descriptive statistics has been used to describe the main features of contours. Contour boxplots [Whitaker et al. 2013] build upon the concept of statistical data depth to measure the centrality of a contour within the set of contours. Contour variability plots [Ferstl et al. 2016b] model statistically the distribution of contours and generate confidence intervals to emphasize their Euclidean spread. However, these methods require closed and consistently oriented contours to generate indicator fields that allow classifying locations with respect to the contours.

In this work, we aim at developing an alternative approach for analyzing the central tendency of ensembles of curves and surfaces, as illustrated in Fig. 1. Our approach is inspired by the work of Ferstl et al. [2016b], where signed distance fields are used as indicator fields. In these fields, closed shapes are defined implicitly as zero level-sets. In contrast, we adopt a vector-to-closest-point representation. In this representation, every location is characterized by a distance and direction to the closest point, and a shape is defined implicitly as the zero-vector level-set. We use such representations as indicator fields, which allows us to handle arbitrary shapes and obtain an improved classification of domain points with respect to an ensemble of shapes.

Our specific contributions to ensemble visualization are:

- Classification of domain points with respect to the distribution of vectors to closest points: We consider ensembles of vector-to-closest-point fields and classify the point-wise directional distributions using mixtures of von Mises-Fisher distributions.
- Computation of a statistical representative shape: By evaluating the classifications over the shape of each ensemble member, the shape most central to the entire sample in a least-squares sense can be determined.
- Quantification of the region-wise centrality of shapes: We in-

corporate directional distribution information into our process to classify the representativeness of local central tendencies.

- Computation of a locally most representative shape: In addition to the most central ensemble member, we use the vector-to-closest-point distributions to determine the locally most representative shape. The piecewise-defined representatives are then combined to form a new shape.

We believe that the proposed method has the potential to improve the analysis of ensembles of shapes. For instance, the method can be applied to ensembles of stream surfaces or deforming flow surfaces, as well as ensembles of surfaces of material or anatomical structures to generate atlas representations. We demonstrate for closed and consistently oriented contours that our method produces the same or very similar results as existing approaches.

## 2 Related Work

Our method falls into the category of uncertainty visualization. For recent surveys on this topic let us refer to the summary article by Potter et al. [2012] and the book by Bonneau et al. [2014]. In ensemble visualization, it is assumed that uncertainty is represented by a set of possible data occurrences, rather than a stochastic uncertainty model. Obermaier and Joy [2014] classify ensemble visualization techniques into feature-based and location-based approaches. Our method can be seen as a hybrid approach, using location-based summary statistics to analyze the variability of specific shape features.

The uncertainty present in ensembles of 2D scalar fields can be conveyed via spaghetti plots of iso-contours [Potter et al. 2009; Wilks 2011; Sanyal et al. 2010]. Since the resulting visualizations usually suffer from visual clutter, simplifications and visual abstractions have been proposed. Sanyal et al. [2010] introduced glyphs and graduated ribbons to convey the uncertainty along iso-contours. Different kinds of confidence bands, i.e., regions representing the Euclidean spread of a set of iso-contours, were proposed by Whitaker et al. [2013] and Ferstl et al. [2016b]. Both approaches use indicator fields to classify locations in space with respect to an ensemble of contours. While Whitaker et al. [2013] use binary fields indicating inside and outside locations, Ferstl et al. [2016b] build upon the concept of standard deviation of signed distance functions. Therefore, both approaches require closed and consistently oriented contours.

Contour boxplots were later extended to parametric curves by Mirzargar et al. [2014]. For the same class of curves, Ferstl et al. [2016a] proposed the use of Principal Component Analysis (PCA) to extract the major geometric trends in ensembles of curves.

To visualize the uncertainty in the position and structure of iso-surfaces, previous approaches use confidence envelopes [Pang et al. 1997; Zehner et al. 2010], surface displacement [Grigoryan and Rheingans 2004], as well as animation of the sequence of surfaces [Brown 2004; Lundstrom et al. 2007]. The concept of numerical condition was introduced by Poethkow and Hege [2011] for the visualization of the variability of iso-surfaces implicitly represented in uncertain scalar fields. Level crossing probabilities for uncertain iso-surfaces were introduced by Pfaffelmoser et al. [2011], and used for extracting surfaces from ensembles exhibiting local correlation structures [Pöthkow et al. 2011].

Signed distance function representations, as used for the ensemble-wise classification of domain points in [Ferstl et al. 2016b], were proposed by Gibson [1998] to obtain a higher-order interpolant of 3D surfaces. For a thorough overview of applications of signed distance transforms and algorithms for computing such transforms

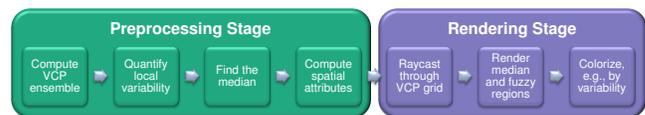


Figure 2: Overview of our method.

from polygon models, see [Jones et al. 2006; Frisken et al. 2000]. Bruckner and Möller [2010] derive a metric for the comparison of iso-contours from signed distance functions. Rathi et al. [2006] and Leventon et al. [2000], use PCA on distance functions for shape analysis. Fofonov et al. [2016] apply dimension reduction techniques to ensembles of distance functions.

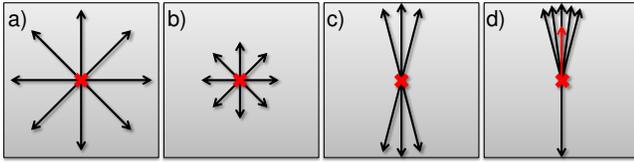
Closest point representations, which comprise a grid structure that encodes a surface implicitly via the vectors to closest surface points, have been used in [Ruuth and Merriman 2008; Macdonald and Ruuth 2008] for solving differential equations on surfaces. In the context of fluid simulation, Auer et al. [2012] proposed a volume rendering technique that samples directly from a uniform closest point grid. Demir and Westermann [2015] improved on this technique by introducing a hierarchical closest point representation and providing means to intersect rays with a zero-vector level-set more accurately.

## 3 Method Overview

Our method starts with an ensemble  $\{s_1, \dots, s_N\}$  of  $N$  2D or 3D shapes, the ensemble members. Without loss of generality, we assume that shapes are given either as polygonal curves or surfaces. Nevertheless, any other representation is possible, as long as for a given location we can compute the point closest to that location on the shape.

Starting with the initial ensemble, our method proceeds in two stages: the preprocessing stage and the rendering stage (see Fig. 2). In the preprocessing stage, we first generate the bounding volume enclosing all shapes, and we discretize this volume using a Cartesian grid structure. If the shapes were extracted from values on a Cartesian grid, e.g., the shapes are level-sets in scalar fields given on such a grid, we use the resolution of the initial grid to discretize the volume. Otherwise, we try to match the resolution of the input shapes, i.e., we adapt the grid resolution to the smallest features represented by the shapes. This is done by iteratively increasing the grid resolution, as long as interpolated closest points differ significantly from their true values on the finer resolved grid points.

For each shape and at each grid vertex, the vector to the closest point on this shape is computed. This results in an ensemble of vectors to closest points at each vertex. We subsequently call this grid the vector-to-closest-point ensemble (VCPE) grid. For the computation of the VCPE grid, we basically follow the GPU implementation by Auer et al. [Auer et al. 2012], with some adaptations to account for the fact that the closest point computations are not restricted to a narrow band around the surface. Therefore, the shape is first partitioned using a regular grid. One GPU thread then computes the closest point of exactly one grid vertex, by sequentially going over the cells of this grid in increasing distance of the cell centers to the vertex position. For each cell, the closest point to the shape contained in this cell is computed, for instance, by iterating over all lines or triangles of the polygonal structure. Processing of cells is stopped once the current closest point is closer than the closest corner of the next cell. Since many grid vertices can be processed in parallel on the GPU, the computation time was always below 5 seconds even for the largest shapes and grid resolutions.



**Figure 3:** *Different cases relevant for centrality quantification. a) With a uniform distribution, placing closest points along the same direction is not possible, resulting in a low degree of centrality. b) Points are lying closer together, which implies that the deviation decreases, and centrality is thus greater. c) Ordering along a single direction and finding a suitable choice as the median is possible. d) Although these closest point vectors can be ordered, the current point represents a poor choice, as it deviates significantly from the mean vector to closest point (colored in red).*

The ensembles of vectors at each grid vertex are now analyzed regarding their local directional variability. Based on a statistical model describing the directional distributions of these vector ensembles, we quantify the local variability at each vertex, and, in particular, we derive a measure quantifying how central the position of the respective grid vertex is to the surrounding closest points. Intuitively, a vertex is most central, i.e., it is suitable as a median point, if its position coincides with the geometric center of the surrounding closest points and it is possible to order these points along a single direction. This gives a measure of centrality that depends on both the distance to the closest points and the modality of the directional distribution. For instance, a bimodal distribution with an equal number of vectors captured by each mode and modes oriented inversely to each other indicates perfect centrality. Fig. 3 shows different cases that are relevant in this regard. We now identify the most central shape by minimizing its overall centrality to the other members. Finally, we generate local statistical attributes to convey the spatial distribution of local centrality to the user.

Precomputed data is then visualized by either sampling the VCPE grid along the rays of sight as proposed in [Demir and Westermann 2015], or performing a single lookup in case of 2D shapes. We provide options to render the median shape by testing for intersections with the most central shape, or fuzzy structures surrounding the median to convey the local centrality.

## 4 Modeling Vectors to Closest Points

Given a VCPE grid, the vectors to closest points at each grid point are used in two different ways: Firstly, their directional distribution is modeled statistically to enable quantifying the local directional variability and, thus, deriving indicators of the centrality of each grid point with respect to the ensemble of shapes. Secondly, the derived models are used to determine the most central member, i.e., the median. Unlike generating a statistical mean member, our algorithm guarantees that the median is an existing member of the initial ensemble. Thus, our technique conveys information to the user, which truly exists in the original data set.

### 4.1 Quantifying the Local Centrality

At each grid vertex, we compute a value representing the local centrality. This value quantifies how well the respective point in space is suitable as a median point. Intuitively, this happens if the point coincides with the geometric center of the surrounding closest points and it is possible to order these points along a single direction. Fig. 3 shows different cases that are relevant in this regard.

Let  $\{vcp_i(p) : i \in I\}$  denote a set of  $N = |I|$  closest point vectors at a grid vertex  $p$ . Moreover, let  $d_{\text{Cutoff}}^*$  denote a global cutoff-value with respect to the length of all closest point vectors, which is used to scale the vectors to unit length. Now, we consider three criteria indicating whether  $p$  can be deemed as suitable for a point of the most central shape, each giving a value ranging between 0 (least suitable) and 1 (most suitable):

**Bimodality angle.** The point is likely to be representative if the directions of the closest point vectors can be modeled by two clusters having means with approximately the same direction, but contrary orientation. To determine whether this is the case, we model the distributions of vectors to closest points using mixtures of probability density functions (pdfs); this allows us to characterize the directional distributions with relatively few parameters: for each mixture component, its mean, the variation around this mean, and the weight of the component. An appropriate mixture model for spherical data is a mixture of von Mises-Fisher (vMF) components. In 2D, this reduces to the von Mises distribution on the circle. However, due to space considerations, in the following we restrict the discussion to the 3D case.

A unit vector  $v$  follows a mixture of vMFs if its pdf is given by

$$f(v) = \sum_{i=1}^M \alpha_i \frac{\kappa_i}{4\pi \sinh(\kappa_i)} \exp(\kappa_i \mu_i^T v), \quad \alpha_i > 0, \quad \sum_{i=1}^M \alpha_i = 1, \quad (1)$$

where  $M$  is the number of vMF components, the unit vectors  $\mu_i$  are their mean directions,  $\kappa_i$  the concentration parameters, and  $\alpha_i$  the weights of the components. The concentration parameter determines the shape of the distribution, with higher values indicating stronger concentrations around the mean direction. The parameters of a mixture of vMFs are estimated using an implementation of the EM soft-moVMF algorithm introduced by Banerjee et al. [Banerjee et al. 2005].

To qualitatively estimate the bimodality value, we always fit two vMF components. For clearly unimodal distributions, one of the two weights  $\alpha_i$  will be 0; thus, the bimodality value is also 0 and no further computations are performed. Otherwise, we compute the bimodality value as the smallest angle between the confidence cones of the two components, normalized by  $\pi$ . Uniform distributions are also fitted using two components, but the wide confidence cones lead to very small bimodality values. To derive a confidence cone around each mean direction  $\mu$ , we use two approaches [Fisher et al. 1987], depending on the size  $n$  of each component. For  $n < 25$ , we need to apply a bootstrap technique. When  $n \geq 25$ , a simpler method is available. Computing the mean resultant length  $\bar{R}$  and estimated spherical standard error  $\bar{\sigma}$  of the sample mean direction

$$\bar{\sigma}^2 = d / \left( n \bar{R}^2 \right), \text{ where } d = 1 - \frac{1}{n} \sum_{i=1}^n (\mu \cdot vcp_i)^2, \quad (2)$$

a 95% confidence cone for  $\mu$  has a semi-vertical angle equal to  $q = \arcsin(1.7308\bar{\sigma})$ .

To visualize the directional distributions, as can be seen in Fig. 1bc and Fig. 6a, we use oriented glyphs, as proposed by Jarema et al. [2015]. At every grid point, the corresponding mixture of vMFs is represented by at most two lobes (one lobe per component), where the mean direction, weight, and confidence cone of each component are mapped to the orientation, length, and opening angle of the corresponding lobe.

**Length of the mean closest point vector.** For this quantity, we first determine the mean closest point vector

$$\overline{vcP}(p) = \frac{1}{N} \sum_{j=1}^N vcp_j(p).$$

Criterion	Variable	a)	b)	c)	d)
Bimodality	$\varphi$	0.0	0.0	0.9	0.95
Mean length	$1 - \hat{\mu}$	1.0	1.0	1.0	0.2
Maximum length	$1 - \hat{d}_{\text{Max}}$	0.0	0.7	0.0	0.0
Centrality	$\sigma$	0.0	0.7	0.95	0.44

**Table 1:** For the cases in Fig. 3, the respective values of the different criteria are shown. The last row contains the resulting centrality as computed by our method. In all cases, our method coincides with the intuitive understanding of a good median choice.

Note that the geometric center of any given set of points  $x_i \in \mathbb{R}^D, i \in I, D \in \mathbb{N}$  in Euclidean space, is obtained as

$$c = \frac{1}{N} \sum_{j=1}^N x_j.$$

That is, the mean closest point vector points to the geometric center with respect to the closest surface points of all members at any given position. Consequently, we can compute the distance from  $p$  to the closest point on a locally optimal median shape as the length of the mean closest point vector. This holds, first, since the geometric center minimizes the mean squared Euclidean distance to each point. Second, because the closest point vector of the potential median member vanishes at all of its surface points. Third, because this vector’s length is the shortest distance to its underlying shape. We now compute the result as the scaled length of the mean closest point vector. The Saturate function is here used to restrict the value to the range  $[0, 1]$ .

$$\hat{\mu}(p) = \text{Saturate} \left( \frac{1}{d_{\text{Cutoff}}^*} \cdot \|\overline{vc}p(p)\| \right)$$

We consider this value in order to decide to which extent the point  $p$  can be regarded as a good representation of the geometric center. Here, smaller values correspond to better degrees. Consequently, we use the value  $1 - \hat{\mu}$  to determine the suitability as median.

**Maximum length of closest point vectors.** This value represents the scaled length of the closest point vector with the greatest Euclidean length, namely,

$$\hat{d}_{\text{Max}}(p) = \text{Saturate} \left( \frac{\max \{ \|\overline{vc}p_i(p)\| : i \in I \}}{d_{\text{Cutoff}}^*} \right).$$

This value indicates how closely together the closest points of all members are located. Again, smaller values stand for better degrees, because this implies that the maximum deviation between all members decreases. In particular, if the maximum length is equal to 0, all members are locally identical, meaning that any member is a perfectly good choice as the median. Hence, we use the value  $1 - \hat{d}_{\text{Max}}$  to determine the suitability as median.

We compute the local centrality based on the aforementioned criteria as

$$\sigma = \max \left\{ 1 - \hat{d}_{\text{Max}}, \sqrt{\hat{\varphi} \cdot (1 - \hat{\mu})} \right\}$$

This formula accounts for all criteria and was verified to produce meaningful results in our test cases. The values corresponding to the cases in Fig. 3, as well as the resulting qualities are shown in Table 1.

## 4.2 Finding the Median

To find the median, we calculate a global deviation wrt. the whole ensemble for each member. The member with the smallest deviation in a least-square sense is then selected as the median. For each

---

### Algorithm 1: Finding the Median

---

**Input:** Ensemble of  $N = |I|$  vector-to-closest-point grids  $vc p_i$

**Result:** Median index  $i \in I$

```

/* Select surface points */
for i in I do
  | P_i ← surface points of member i
end
/* Iterate over all members */
for i in I do
  /* Compute centrality */
  for p_i in P_i do
    | σ(p_i) ← Centrality {vc p_j(p_i) : j in I}
  end
  /* Insert into uniform grid */
  Choose resolution r in N^d
  C_i ← ∅
  for p_i in P_i do
    | Let c_i in C_i, such that c_i contains p_i
    | c_i ← c_i ∪ σ(p_i)
  end
  /* Compute global deviation */
  Δ_i ← 0
  weight(Δ_i) ← 0
  for ∅ ≠ c_i in C_i do
    | Δ_i ← Δ_i + ∑_{σ in c_i} σ / |c_i|
    weight(Δ_i) ← weight(Δ_i) + 1
  end
  Δ_i ← Δ_i / weight(Δ_i)
end
return arg min {Δ_i : i in I}

```

---

ensemble member  $i \in I$ , we first select a sufficiently large set of points,  $P_i$ . For polygonal meshes or line strips, we use the original vertex positions for that purpose. Now, we iterate over all ensemble members  $j \in I$  and over all surface points  $p_i \in P_i$ . For each run, we gather the vectors  $vc p_j(p_i)$ , i.e., the closest point vectors at all given surface points wrt. to all members. Here,  $vc p_i(p)$  denotes the closest point vector at point  $p$  for member  $i$ . Gathering is performed by bi- and tri-linear interpolation in the VCPE grid. Based on the set of interpolated vectors, we quantify the local centrality,  $\sigma(p_i)$ , described in the previous section.

We then obtain the global median deviation  $\Delta_i$  as the expected squared local centrality at a uniformly randomly selected position on the shape’s surface. Note that it is not sufficient to simply average over the centrality values, since the points  $p_i$  are not necessarily distributed uniformly, which would result in assigning unequal weights to regions that are sampled at a denser or less dense resolution than the average. As an example, consider a shape given as an adaptively resolved triangle mesh, where the vertex positions are used as the set of surface points. Even if they were derived from a uniform Cartesian grid, the surface points need not be resolved uniformly, since, depending on the underlying geometry, distant closest point vectors can point to locations close to each other. To overcome this issue efficiently, we proceed in the following way. First, we insert the squared centrality values into a Cartesian grid of lower resolution, such that each cell  $c_i$  contains the average of all values  $\sigma(p_i)^2$ , for all  $p_i$  belonging to  $c_i$ . Second, we compute the average over all cells containing at least one value  $C_i$ , and return the result as the global deviation with respect to the ensemble, i.e.,

$$\Delta_i = \frac{1}{|C_i|} \sum_{c_i \in C_i} \sigma(c_i).$$

Here, the key idea is to choose the grid resolution such that both

the distribution of points within each cell and the distribution of the cells near the initial shape’s surface is roughly uniform. We achieve this by picking the finest resolution such that the largest primitive of the original geometry fits into one cell, with the restriction that it is at most as fine as the underlying VCPE grid. Alternatively, we could utilize a binary search between 1 and the grid resolution in each dimension. Finally, we obtain the median index as the member with the least global deviation, i.e.,

$$\text{Median}(I) = \arg \min \{\Delta_i : i \in I\}.$$

A complete overview is shown in Algorithm 1.

### 4.3 Spatial Attributes

We now discuss the generation of statistical spatial attributes, which are presented to the user as fuzzy structures along with the median shape. In our implementation, this data is stored in the VCPE grid in addition to the index of the median shape and the local centrality data. This allows us to render simultaneously the shape and the fuzzy centrality volume during ray-casting. We begin with computing a fuzziness at each point of the VCPE grid. During rendering this value determines, whether a fuzzy or crisp structure should be drawn at a respective point, as well as its opacity. For this purpose, we compute two different values.

The minimal distance is given as the minimum of the Euclidean distances to all ensemble members from a given point in space. Note that we can easily obtain this distance as the length of the corresponding closest point vector: Thus, we have

$$d_{\text{Min}}(x) = \min \{\|vcpi(x)\| : i \in I\}.$$

Using the minimal distance as the fuzziness allows the user to see all regions where at least one member resides.

Using the minimal distance enables the user to identify all regions potentially of interest. However, it does not reveal how many members cross a certain region, i.e., the density of shapes at a given point in space. To compensate for this, we propose a weighted distance value, given as an adjusted mean of all distances,

$$d_{\text{Weighted}}(x) = \frac{\sum_{i=1}^N (\|vcpi(x)\| \cdot \exp(-\alpha \cdot \|vcpi(x)\|))}{\sum_{i=1}^N \exp(-\alpha \cdot \|vcpi(x)\|)},$$

where  $\alpha \in \mathbb{R}^+$  denotes an importance parameter giving an additional weight to smaller distances. This allows us to retain regions containing outliers, whereas choosing  $\alpha$  too small would result in smoothing out fine structures distant to most other members. On the other hand, choosing  $\alpha$  too large would produce results similar to the minimal distance measure. In our test cases,  $\alpha \approx 3$  yielded good results. Using the weighted distance criterion enables the user to quickly identify outliers, as these are regions characterized by a relatively low fuzziness.

## 5 Ensemble Visualization

To visualize the median shape in combination with the local centrality information, we use a volume ray-caster sampling the extended VCPE grid along the view rays. Since both the closest point vectors and centrality values are stored in the same grid, visualizing both structures can be performed simultaneously in one rendering pass.

Ray-casting the VCPE grid is performed in essentially the same way as proposed by Demir and Westermann [2015]. Thus, we give only a brief overview and explain the differences to our implementation. To render the median shape via ray-casting, we march along

the view rays through the VCPE grid and interpolate the closest point vectors corresponding to the shape at the sampling points. Now, if the length of the closest point vector is less than a given threshold  $\varepsilon > 0$ , we assume the shape was hit and compute the color by a local illumination model. At this point, the approach in [Demir and Westermann 2015] terminates. However, as we include transparency effects to visualize the centrality information, we continue the process until the accumulated opacity reaches a prescribed threshold or the ray position lies outside of the VCPE grid. In addition, these visualization options are provided:

**Local best median.** While traversing the VCPE grid along the rays, we test for the minimum centrality value of all ensemble shapes. If this value falls below a user-selected threshold, the shape for which the value is determined is visualized, i.e., a unique color assigned to this shape is rendered and the ray is terminated. In this way, we obtain a shape composed of many different shapes, depending on which of these shapes represents the locally best median. This combined shape is one that does not exist in the initial ensemble, but allows determining local trends and conveying the ensemble member that best follows the local trend.

**Fuzzy regions.** To visualize the ensemble centrality around the median, we render a semi-transparent fuzzy structure, similar to the approach by Grigoryan and Rheingans [2004]. The fuzziness, mapped to opacity, is computed according to a user-selected criterion: Either the minimal distance,  $d_{\text{Min}}$ , or the weighted distance,  $d_{\text{Weighted}}$ , can be picked by the user. In the first case, we choose a threshold  $d_{\text{Min}}^*$  such that the structure is rendered only if the minimal distance is less than the diameter of a grid cell. For this, we set the fuzziness to 1, if  $d_{\text{Min}} < d_{\text{Min}}^*$ , and to 0 otherwise. In doing so, we ensure that fuzzy structures only occur where initial members exist. By determining the fuzziness according to  $d_{\text{Weighted}}$ , the user is able to explore the density of shapes in each region. Again, to avoid visualizing non-existent data, we combine both the weighted and the minimal distance in this case, i.e, we set the fuzziness to  $d_{\text{Weighted}}$ , if  $d_{\text{Min}} < d_{\text{Min}}^*$ , and to 0 otherwise. For coloring the fuzzy structures, we use the local centrality, mapped from dark blue (high centrality) to yellow (low centrality).

**2D ensembles.** We provide the option to visualize 2D VCPE grids. Therefore, a grid of depth 1 is generated, and one single lookup per ray retrieves all the information necessary to determine whether the median shape is hit and what the local centrality at the sampling points is. This information is then color-coded.

**Silhouettes.** To enhance the user’s recognition of the shapes, especially if fuzzy structures are visualized in front of them, we extend our rendering technique by adding silhouettes to the shape. They are blended over the shapes, such that they are never obscured by other structures. To detect silhouettes, we extend the raycasting algorithm by a second threshold  $\varepsilon^*$ , given by  $\varepsilon^* = \varepsilon \cdot (1 + \tau \cdot \lambda)$ , where  $\tau$  denotes the silhouette thickness and  $\lambda$  the distance traveled along the ray. If, while marching along the ray, no silhouette was hit so far, and the current closest point length is less than  $\varepsilon^*$ , we consider this a hit and store the distance to the camera, i.e.,  $\lambda_0$ . If a hit occurs in the following steps, we check, if the current distance to the camera is close enough to the silhouette hit position, i.e.,  $\lambda < \lambda_0 + \varepsilon$ . In this case, the silhouette is discarded, since it does not lie on the boundary of the shape. By letting the silhouette threshold depend on  $\lambda$ , we ensure that all silhouettes are of roughly equal thickness.

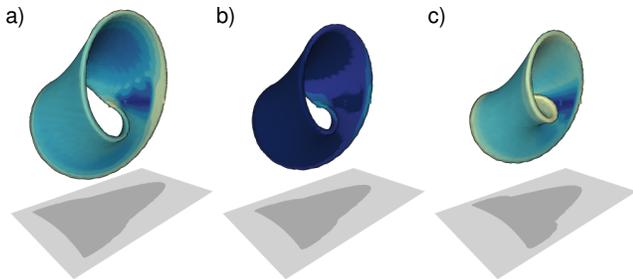
## 6 Results

To demonstrate the practical application of the proposed method, we discuss two synthetic data sets, two real-world weather forecasting cases that occurred during an atmospheric research campaign,

and a synthetic ensemble of 3D fluid simulations. The first synthetic data set is comprised of 50 Möbius bands, each represented by a 5K triangle mesh, which were slightly perturbed to simulate an ensemble with changing geometry. The second synthetic ensemble comprises 50 polygonal spheres of 4K triangles each, which have exactly the same geometry in most regions, but turn inward and outward with varying strength and shape in one of the hemispheres. Both synthetic data sets were discretized on a VCPE grid of resolution  $128 \times 128 \times 128$ . The forecast data are obtained from the ECMWF Ensemble Prediction System (ENS). The ensemble comprises an unperturbed control run (i.e., started from the “best” initial conditions) and 50 perturbed members. Our example region in this ensemble covers the North Atlantic and Europe, encompasses  $335 \times 135 \times 62$ , and comprises geopotential height, a typical measurement variable in weather forecasting. 2D and 3D polygonal iso-contours were first extracted from the physical fields, resulting in up to 16K triangles per surface, and these contours were then used as input to our approach. Our last ensemble features a Navier-Stokes fluid simulation over an ellipsoid obstacle. Here, 56 simulation runs with different viscosities were performed at a grid resolution of  $145 \times 49 \times 49$ . The vorticity magnitude of each run was written out as a scalar field, and polygonal iso-surfaces were extracted from each ensemble member.

All presented results were generated on a standard desktop PC (Intel Xeon X5675 processor with  $6 \times 3.0$  GHz, 8 GB RAM and an NVIDIA Geforce GTX 680). The time required to compute the VCPE grids, including the statistical modeling of the directional distributions per vertex, took less than 15 seconds on the GPU in all of our experiments. This time scales linearly in the number of grid points and the number of lines or triangles to which closest points have to be computed.

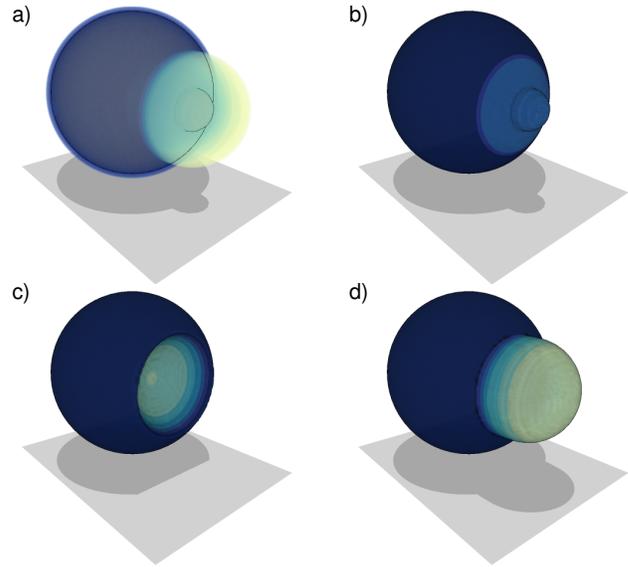
In Fig. 4, we demonstrate the use of our method to find a median surface in an ensemble of non-orientable 3D surfaces, and to visualize the local centrality over the given surfaces with respect to the median. As can be deduced from the surface coloring, the computed median has very high centrality across the entire shape, while the other shapes show locally low centrality, which is indicated by the color shift towards yellow.



**Figure 4:** From an ensemble of 50 slightly perturbed Möbius bands, the median surface (middle) is computed using our method. The left and right surfaces show the two ensemble member with the highest overall deviation from the median. Color encodes centrality, ranging from dark blue (high) to yellow (low).

In Fig. 5, we use our method to analyze the central tendency in the ensemble of spheres. Note that in this ensemble, roughly  $3/4$  of all spheres were perturbed outwards, and the remaining spheres were perturbed inwards. Firstly, we show a rendering of the fuzzy centrality volume to indicate the different shapes of the ensemble members. It can be seen that, in one hemisphere, all members have exactly the same geometry, while a high variability in the shapes can be observed in the frontal region. Secondly, the median shape is

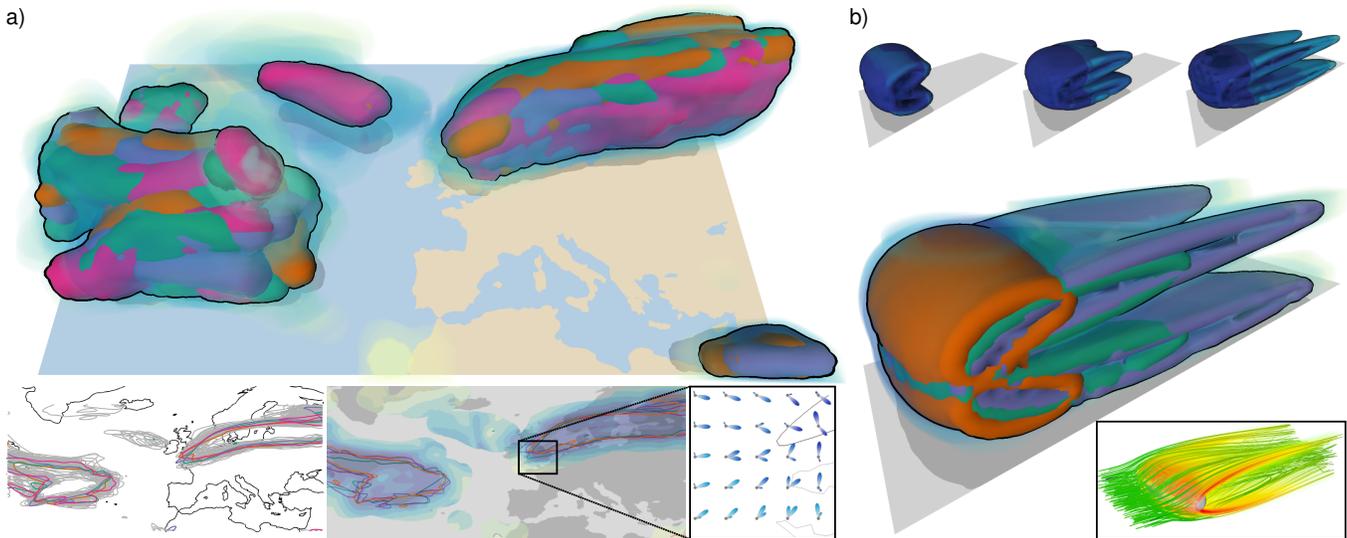
shown, color coded with local centrality. In the last two images, two members exhibiting low centrality are shown, which is indicated by the color shift towards yellow.



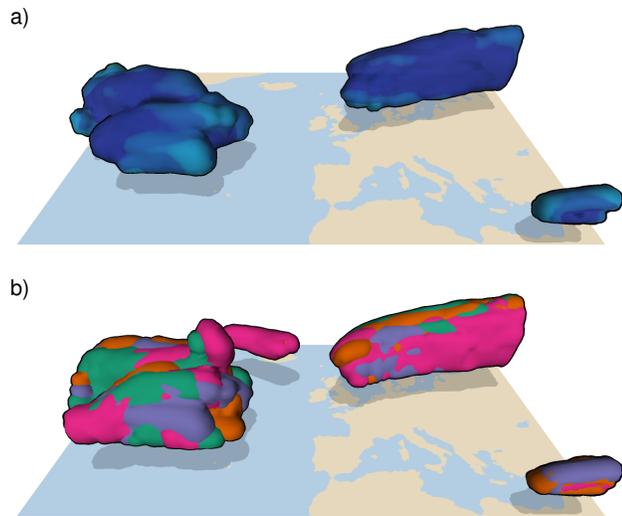
**Figure 5:** Visualization of an ensemble of locally perturbed 3D spheres. a) The fuzzy volume indicating local variability of the ensemble. b) The median surface. Color encodes centrality, ranging from dark blue (high) to yellow (low). c) and d) Two members of the ensemble being less central than the median.

In Fig. 7a, we show the computed median surface for one of the 3D weather forecast ensembles. Below, for the same ensemble, the locally best matching median surface is visualized, by using color to indicate which of the ensemble members provides the best median surface locally. It can be seen that the surface in the bottom image shows details not present in the median shape, i.e., meaning that the median is not always the best representative locally. The same experiment is shown in the top image in Fig. 6a, and in Fig. 6b for the fluid ensemble. The visualization of the ECMWF data set reveals certain features that become immediately apparent to the user. For instance, one can discover the major regions where higher wind velocities are predicted, namely above the Baltic Sea and to the south of Greenland. Note that by analyzing other iso-values, we can verify that higher velocities occur within these region, although this is not shown here. By studying the fuzzy structures, we can spot a region located northwest of Africa exhibiting a lower degree of centrality. This indicates the existence of outliers at that region. In the fluid ensemble, three representative shapes extracted from disjoint subsets of members are shown on the top. The visualization of a piecewise median is depicted at the bottom. By volume rendering the fuzzy structure, the user gets an overview of the distribution regarding the remaining ensemble members.

The first three images in the teaser, and the bottom images in Fig. 6a show 2D iso-contours in the ECMWF ensembles. In both figures, spaghetti plots of all considered contours are shown, and the computed median contours are highlighted in red. Here it is worth noting that our approach yields exactly the same median contour as the contour boxplots [Whitaker et al. 2013] do. In addition, we show for selected regions the directional variability of vectors to closest points, which was used to classify the local centrality. It can be seen that the directional variability is a good indicator for classifying the domain locally. While in central regions we see a clear bimodality with opposing directions, the distributions are tending towards unimodality or uniformity in the more de-central regions.



**Figure 6:** a) Top: 3D weather forecast ensemble. The coloring indicates the locally best matching median surface. Bottom: 2D slice from the same ensemble. First, a spaghetti plot of all iso-contours for a selected threshold is shown. The contours contributing to the median are highlighted. Next, the domain is colored according to local centrality, and the distribution of vectors to closest points is shown for a selected region. b) Median and fuzzy regions are visualized in the same view for an ensemble featuring vorticity magnitude of a fluid simulation. On the bottom right, is a single member rendered using streamlines.



**Figure 7:** Visualization of an ensemble of 3D iso-surfaces in wind velocity fields. a) The median surface. Color encodes centrality, ranging from dark blue (high) to yellow (low). b) The locally best matching medians. Colors indicate different ensemble members.

## 7 Conclusion and Future Work

In this work, we have presented a new approach for visualizing the spatial uncertainty that is represented by ensembles of shapes in 2D or 3D space. We introduced a vector-to-closest-point representation in combination with techniques to statistically model the resulting directional distributions, and use them to classify regions and shapes with respect to centrality and uncertainty. For ensembles of 2D curves, a comparison has shown results similar to those achieved by alternative approaches. However, our approach does not make any assumption about the input shapes and can be ex-

tended to 3D in a straightforward way. In combination with GPU-based ray-casting of closest point grids, we provided an interactive visual exploration tool for ensembles of multi-dimensional shapes.

In the future, we plan to extend and analyze our approach with respect to the following aspects: Firstly, we will elaborate on the use of closest point fields for clustering ensembles of shapes. In previous works, clustering has been performed on signed distance fields, and it will be interesting to compare the resulting clusters quantitatively and qualitatively. Secondly, we will analyze in more detail the shape medians that are computed by our and other approaches, especially with respect to the local representativeness. We have observed that computing medians is sensitive to the VCPE grid resolution and the weighting of variability values in the mean square integration over the shapes. Experiments using ground truth medians have to be pursued to shed light on this aspect. Finally, a more elaborate discussion of the locally best matching median has to be performed, in the context of the application in which it is used. This particular median includes sub-shapes from many different ensemble members, and we need to analyze how well the resulting shape can convey application-specific information concerning the local trends in the data.

## Acknowledgments

Access to ECMWF prediction data has been kindly provided in the context of the ECMWF special project "Support Tool for HALO Missions". We are grateful to the special project members Marc Rautenhaus and Andreas Dörnbrack for providing the ECMWF ENS dataset of 17 October 2012. This work was supported by the European Union under the ERC Advanced Grant 291372 - SaferVis - Uncertainty Visualization for Reliable Data Discovery.

## References

- AUER, S., MACDONALD, C. B., TREIB, M., SCHNEIDER, J., AND WESTERMANN, R. 2012. Real-time fluid effects on surfaces using the closest point method. *Computer Graphics Forum* 31, 6.
- BANERJEE, A., DHILLON, I. S., GHOSH, J., AND SRA, S. 2005. Clustering on the unit hypersphere using von mises-fisher distributions. *J. Mach. Learn. Res.* 6 (Dec.), 1345–1382.
- BONNEAU, G.-P., HEGE, H.-C., JOHNSON, C., OLIVEIRA, M., POTTER, K., RHEINGANS, P., AND SCHULTZ, T. 2014. Overview and state-of-the-art of uncertainty visualization. In *Scientific Visualization*. Springer, 3–27.
- BROWN, R. 2004. Animated visual vibrations as an uncertainty visualisation technique. In *GRAPHITE*, 84–89.
- BRUCKNER, S., AND MÖLLER, T. 2010. Isosurface similarity maps. *Computer Graphics Forum* 29, 3, 773–782.
- DEMIR, I., AND WESTERMANN, R. 2015. Vector-to-Closest-Point Octree for Surface Ray-Casting. In *Vision, Modeling & Visualization*, The Eurographics Association, D. Bommes, T. Ritschel, and T. Schultz, Eds.
- FERSTL, F., BÜRGER, K., AND WESTERMANN, R. 2016. Streamline variability plots for characterizing the uncertainty in vector field ensembles. *IEEE Transactions on Visualization and Computer Graphics* 22, 1, 767–776.
- FERSTL, F., KANZLER, M., RAUTENHAUS, M., AND WESTERMANN, R. 2016. Visual analysis of spatial variability and global correlations in ensembles of iso-contours. In *Computer Graphics Forum (Proc. EuroVis)*. to appear.
- FISHER, N., LEWIS, T., AND EMBLETON, B. 1987. *Statistical analysis of spherical data*. Cambridge Univ. Press.
- FOFONOV, A., MOLCHANOV, V., AND LINSEN, L. 2016. Visual analysis of multi-run spatio-temporal simulations using iso-contour similarity for projected views. *IEEE Trans. Visual. and Comp. Graphics, to appear*.
- FRISKEN, S. F., PERRY, R. N., ROCKWOOD, A. P., AND JONES, T. R. 2000. Adaptively sampled distance fields: A general representation of shape for computer graphics. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '00*, 249–254.
- GIBSON, S. F. F. 1998. Using distance maps for accurate surface representation in sampled volumes. In *Proceedings of the 1998 IEEE Symposium on Volume Visualization, VVS '98*, 23–30.
- GRIGORYAN, G., AND RHEINGANS, P. 2004. Point-based probabilistic surfaces to show surface uncertainty. *IEEE Transactions on Visualization and Computer Graphics* 10, 5, 564–573.
- JAREMA, M., DEMIR, I., KEHRER, J., AND WESTERMANN, R. 2015. Comparative visual analysis of vector field ensembles. In *Visual Analytics Science and Technology (VAST), 2015 IEEE Conference on*, 81–88.
- JONES, M. W., BAERENTZEN, J. A., AND SRAMEK, M. 2006. 3d distance fields: A survey of techniques and applications. *IEEE Transactions on Visualization and Computer Graphics* 12, 4 (July), 581–599.
- LEVENTON, M., GRIMSON, W., AND FAUGERAS, O. 2000. Statistical shape influence in geodesic active contours. In *Proc. IEEE Computer Vision and Pattern Recognition*, 316–323.
- LUNDSTROM, C., LJUNG, P., PERSSON, A., AND YNNERMAN, A. 2007. Uncertainty visualization in medical volume rendering using probabilistic animation. *IEEE Transactions on Visualization and Computer Graphics* 13, 1648–1655.
- MACDONALD, C. B., AND RUUTH, S. J. 2008. Level set equations on surfaces via the Closest Point Method. *J. Sci. Comput.* 35, 2–3 (June), 219–240.
- MIRZARGAR, M., WHITAKER, R., AND KIRBY, R. M. 2014. Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE Trans. on Visualization and Computer Graphics* 20, 12, 2654–2663.
- OBERMAIER, H., AND JOY, K. 2014. Future challenges for ensemble visualization. *IEEE Comp. Graphics and Applications* 34, 8–11.
- PANG, A. T., WITTENBRINK, C. M., AND LODHA, S. K. 1997. Approaches to uncertainty visualization. *The Visual Computer* 13, 8, 370–390.
- PFAFFELMOSER, T., REITINGER, M., AND WESTERMANN, R. 2011. Visualizing the Positional and Geometrical Variability of Isosurfaces in Uncertain Scalar Fields. *Computer Graphics Forum* 30, 3, 951–960.
- PÖTHKOW, K., AND HEGE, H.-C. 2011. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *IEEE Transactions on Visualization and Computer Graphics* 17, 10, 1393–1406.
- PÖTHKOW, K., WEBER, B., AND HEGE, H.-C. 2011. Probabilistic marching cubes. *Computer Graphics Forum* 30, 931–940.
- POTTER, K., WILSON, A., BREMER, P.-T., WILLIAMS, D., DOUTRIAUX, C., PASCUCCI, V., AND JOHNSON, C. R. 2009. Ensemble-Vis: A framework for the statistical visualization of ensemble data. In *Proc. IEEE Int. Conf. on Data Mining Workshops*, 233–240.
- POTTER, K., ROSEN, P., AND JOHNSON, C. R. 2012. From quantification to visualization: A taxonomy of uncertainty visualization approaches. In *Uncertainty Quantification in Scientific Computing*. 226–249.
- RATHI, Y., DAMBREVILLE, S., AND TANNENBAUM, A. 2006. Statistical shape analysis using kernel PCA. *Proc. SPIE* 6064, 60641B.
- RUUTH, S. J., AND MERRIMAN, B. 2008. A simple embedding method for solving partial differential equations on surfaces. *J. Comput. Phys.* 227, 3 (Jan.), 1943–1961.
- SANYAL, J., ZHANG, S., DYER, J., MERCER, A., AMBURN, P., AND MOORHEAD, R. J. 2010. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE Transactions on Visualization and Computer Graphics* 16, 1421–1430.
- WHITAKER, R. T., MIRZARGAR, M., AND KIRBY, R. M. 2013. Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE Transactions on Visualization and Computer Graphics* 19, 12, 2713–2722.
- WILKS, D. S. 2011. *Statistical Methods in the Atmospheric Sciences*. Academic Press.
- ZEHNER, B., WATANABE, N., AND KOLDITZ, O. 2010. Visualization of gridded scalar data with uncertainty in geosciences. *Computers & Geosciences* 36, 10, 1268–1275.