

Transfer Function Combinations

Liang Zhou*, Mathias Schott, Charles Hansen

Scientific Computing and Imaging Institute, University of Utah, 72 S Central Campus Drive, Salt Lake City, UT 84112, United States

Abstract

Direct volume rendering has been an active area of research for over two decades. Transfer function design remains a difficult task since current methods, such as traditional 1D and 2D transfer functions are not always effective for all datasets. Various 1D or 2D transfer function spaces have been proposed to improve classification exploiting different aspects, such as using the gradient magnitude for boundary location and statistical, occlusion, or size metrics. In this paper, we present a novel transfer function method which can provide more specificity for data classification by combining different transfer function spaces. In this work, a 2D transfer function can be combined with 1D transfer functions which improve the classification. Specifically, we use the traditional 2D scalar/gradient magnitude, 2D statistical, and 2D occlusion spectrum transfer functions and combine these with occlusion and/or size-based transfer functions to provide better specificity. We demonstrate the usefulness of the new method by comparing to the following previous techniques: 2D gradient magnitude, 2D occlusion spectrum, 2D statistical transfer functions and 2D size based transfer functions.

Keywords:

Transfer function, volume rendering, classification, user interface

1. Introduction

Direct volume rendering has been an active area of research. Mapping of data values to optical properties, known as classification, remains a challenging problem. Transfer functions are most commonly used for classification in volume rendering, yet finding good transfer functions remains a difficult problem. For material boundaries, it has been shown that 2D transfer functions provide greater specificity¹ than 1D transfer functions [1, 2]. In many datasets, separate features may share the same scalar value and gradient magnitudes and as such value, gradient magnitude tuples are not sufficient for separating such features.

Recently, many new 2D transfer function spaces have been proposed to improve the classification from different metrics. The size based transfer function is a transfer function space [3] built upon blob detection techniques using scale space theories to classify objects based on their sizes. The occlusion spectrum is another 2D transfer function space [4] which takes into consideration ambient occlusion within the volume for discriminating between features of similar scalar values. It is also possible to compute statistical measurements such as mean value and standard deviation in a local region around a voxel [5] to form a 2D transfer function space. All of these methods are effective on some datasets. Other datasets however may contain materials which have similar statistical properties but occlude each

other, or have materials share similar statistical properties and occlusion measurements but differ in size.

As such, we propose to combine the best features of these transfer functions to create a transfer function space that provides better specificity. Our contributions in this work are three fold: 1) Combining 2D transfer function space with 1D transfer function spaces with a basic approach for selecting combinations. 2) An user interface supports transfer function design in the combined transfer function space. 3) Experiments and detailed discussions of different transfer function combinations and original 2D transfer functions on various data sets.

We experimented with the combinations of these transfer function spaces and discuss a basic approach for selecting combinations that improve classification and show that this combined transfer function space provides better classification than either just 2D gradient magnitude transfer functions, 2D statistical transfer functions, 2D occlusion based transfer functions or 2D size based transfer functions.

2. Related Work

The most frequently used transfer function for volume rendering is a 1D transfer function that uses scalar values for classification. Realizing the poor classification ability of that transfer function space, Levoy [2] and Kindlmann *et al.* [1] used the gradient magnitude of the volume as another property for better classification. Kniss *et al.* [6] advocated and implemented multi-dimensional transfer functions widgets, making the 2D transfer function a standard method in modern volume renderers. By far, the 1D and 2D transfer functions are the most popular and practical techniques for classification in volume ren-

*Corresponding author.

Email address: zhou1@cs.utah.edu (Liang Zhou)

¹We use the disambiguation definition of specificity rather than the statistical definition which means the proportion of negatives in a binary classification test which are correctly identified.

dering, however, great efforts have been made to define new transfer function spaces to improve the classification ability.

Due to noise and partial volume effects, selecting a boundary in the arches in the gradient magnitude based transfer function is not easy and sometimes even impossible. To resolve this problem, Lundstrom *et al.* [7] employ the local histograms to better discern tissues in medical data sets and propose a 2D transfer function space that uses competitive classification certainty measure in addition to scalar values. Sereda *et al.* [8] use 2D LH histogram based transfer function for easier boundary identification and selection and further use this boundary information for a region growing segmentation schema.

The theory of scale-space, developed originally by the computer vision and image processing communities, can be used to classify objects based on their sizes. A commonly used scale-space representation is the linear Gaussian scale-space, which is essentially a convolution of a volume with differently sized Gaussian filters. Lum *et al.* [9] combine it with an image pyramid representation of different scales to improve classification. Correa and Ma [3] create a continuous scale-space for the volume and use anisotropic diffusion to detect “blobs” in the volume. The size of these defines an additional metric of the volume, which is then used to create size-based transfer functions.

Shape is another important aspect to classify an object, as such Sato *et al.* [10] use eigen value analysis on 3D local intensity structures to classify tissues in medical datasets with 2D transfer function spaces created using shape measurements: sheet, line or blob respectively along with the scalar value. Prassni *et al.* [11] propose shape based transfer functions by computing shape descriptors over pre-segmented volume to provide a manageable set of shape classified volumetric features with an intuitive optical properties assignment interface.

Taking concepts from computer graphics, Kindlmann *et al.* [12] use curvature as a second dimension of their transfer function domain to create non-photorealistic renderings. Correa and Ma [4] use the occlusion of a voxel as an additional dimension of the transfer function domain to classify features of similar scalar value, but different local neighborhoods.

Volumes can also be classified based on their statistical metrics, such as mean value or standard deviation of voxels in a certain neighborhood. Caban *et al.* [13] compute local statistical metrics and use their linear combinations to classify fine structures. Patel *et al.* [14] use a dynamically changing neighborhood to compute mean value and variance for voxels, thus defining a transfer function domain. A user interface then allows to select features based on the mean value, variance and radius of the neighborhood. Haidacher *et al.* [5] further extend this approach by selecting the radius semi-automatically via an adaptive sample selection technique.

Transforming the volume data into frequency domain is another idea for generating transfer function spaces. Vucini *et al.* [15] utilize GPU based Fast Fourier Transformation to support interactive frequency-based transfer function design that enhances conventional volume visualization.

Transfer function spaces that have been proposed are many as can be seen above, however, we argue that as a general approach, it is not necessary to include all of them into our study

although the transfer function spaces can be used are not limited to the ones we choose in this work. We choose each of our element transfer functions from a different category listed above to reduce the dependencies between the transfer function spaces and thus hopefully improve the classification. As a result, the traditional 2D gradient magnitude transfer function, the statistical transfer function [5], the size based transfer function [3] and the occlusion transfer function [4] are chosen as the element transfer functions due to the effectiveness in classification they have exhibited and their relative easiness of implementation.

User interface design, especially to make high dimensional transfer functions practical, is equally important to the specific transfer function domain, and thus has been the focus of recent work. Piringner *et al.* [16] propose to interact with three 2D scatterplot views and get feedback in a 3D scatterplot view to design 3D transfer functions in the SimVis system [17]. They have later added direct volume rendering to the result. This method although provides more generality than ours, there is no systematic evaluation of which scatterplot views should be used. Furthermore, the scatter plots are on the original data, not data transformed by existing transfer function domains. It also introduces more complexity into user interaction. We believe that with the transfer function combination selection algorithm proposed in Section 3.2, our simpler user interface can achieve similar classification precision with an easier and more intuitive user experience. Lum *et al.* [9] use parallel coordinates to select features of different sizes. Patel *et al.* [14] present a 3D transfer function editor that allows the user to select mean value, standard deviation and feature radius respectively. Guo *et al.* [18] develop a system that combines parallel coordinates and dimensional reduction techniques to design high dimensional transfer functions.

Kniss *et al.* [19] proposed that separable transfer functions may lead to erroneous classifications; the method discussed in detail in Section 3.1 however does not suffer from similar artifacts. Rezk-Salama [20] briefly discusses a similar idea, called *local transfer functions* where a volume segmentation method is used to create a so called *tag volume*, which is then used to select a transfer function associated with a specific tag. Our proposed combined transfer functions instead use a 2D transfer function to select another 1D transfer function to classify multi variate data sets. Bruckner *et al.* [21], in the context of illustrative volume renderings, use a 1D transfer function to index into a table of *style transfer functions* which then, in conjunction with a screen space normal are used to determine shading of samples. Those are, while similar in implementation, conceptually different from the proposed transfer function combinations, which are used to classify multi-variate data sets.

3. Combining Transfer Functions

3.1. Combining Transfer Functions

In practice, using just one or two metrics during volume classification makes it difficult to robustly classify and separate features in complex volumes. Using more properties in the transfer

function space often can better describe features in the volume, however, user interaction becomes more difficult or even impossible when the number of properties, thus the dimensionality, of the transfer function space increases. Gaussian transfer functions have been proposed by Kniss *et al.* [19] to provide analytical multi-dimensional transfer functions of arbitrary dimensionality, also a procedural high-dimensional transfer function model is proposed in [22]. However, in both works, how to provide an effective user interface remains unclear.

The proposed transfer function combination sequentially applies two transfer functions, a two-dimensional and a one-dimensional one, to all voxels v_X , where X is the 3D position, of the input dataset V that has l properties Y_1, Y_2, \dots, Y_l .

$$\{C_1, C_2, C_3, \dots, C_k\} = \text{TFw}(V) \text{ where} \\ C_j := \{v_X | \text{TFw}_j(Y_p(X), Y_q(X)) > 0, p, q \in [1, l]\} \quad (1)$$

$$\{W_i \subset C_j\} = \text{TF}_r(Y_r(X)) \text{ where} \\ X \in C_j(X), j \in [1, k], r \in \{1, \dots, l\} / \{p, q\} \quad (2)$$

In Equation 1, a number of 2D transfer function widgets, k , are first applied to the volume, resulting in sets of classified voxels $C_1, C_2, C_3 \dots C_k$ respectively. Then one from a set of r , which is typically 1 or 2, 1D transfer functions is applied to the classified region C_j , yielding the final classified volume region W_i . Each 2D transfer function widget has one associated 1D transfer function.

Kniss *et al.* [19] clearly show a 2D example that separating high dimensional transfer functions into lower dimensional ones using multiplication can lead to misclassification, which gets worse when the dimensionality is extended into 3D. Our proposed method however does not suffer from such issue as each 2D transfer function widget has a 1D transfer function that help further separating features within the voxels selected by the 2D transfer function. This dimension reduction method however can cause classification inconsistencies compared to a true 3D transfer function. We believe that this is a reasonable compromise, considering that the losses in classification precision compared to using an equivalent higher dimensional transfer function are typically minor.

Rezk-Salama [20] proposed a similar idea called local transfer functions to set transfer functions for segmented volumes, *i.e.* a transfer function is associated with a tag in the tagged volume; voxels are essentially pre-classified and their tags are stored in a volume. Our method is more flexible as the user essentially interactively labels voxels using the 2D transfer functions and then further classifies the features using the associated 1D transfer function. Bruckner and Gröller [21] similarly use a 1D transfer function to index into a table of style transfer functions for flexible illustrative volume renderings. Their work conceptually differs from ours as our transfer function combination method is utilized to improve the specificity of transfer functions rather than producing illustrative visualizations.

3.2. Selecting Combinations

We propose to separate the transfer function space into a 2D transfer function space with a set of 1D transfer function spaces as a trade off between dimensionality and usability.

A problem naturally arises when more than three properties/attributes are provided, namely which properties contain salient features, which attributes are most effectively used as the 2D transfer function domain, and which are best classified by the associated 1D transfer functions. Thus, we provide a few simple rules to aid the user in selecting appropriate combinations.

For a given set l properties of a data set, the correlation coefficient matrix R of size $l \times l$ is computed, as well as the entropy vector E of size l contains all properties' entropy. The primary property Y_p , is chosen that represents the original information of the dataset (e.g. original intensity dataset or the mean dataset computed from the statistical properties extraction process as shown in section 4.3). A property that is intrinsically associated with Y_p (e.g. gradient magnitude vs. original intensity dataset or standard deviation vs. mean value) is used as the *secondary* property Y_q . The primary and secondary properties define the 2D transfer function space. For all remaining properties Y_i , $i \in [1, l]$ and $i \neq p, q$ a score is computed as a linear interpolation between the correlation coefficient R_{pi} and the normalized entropy $\frac{E(Y_i)}{\max E}$, as shown in Equation 3:

$$s_i = -a|R_{pi}| + (1-a)\frac{E(Y_i)}{\max E}, 0 \leq a < 1 \quad (3)$$

Correlation coefficient depicts the similarity between properties: a lower correlation coefficient value indicates higher independences of properties. By intuition, more independent properties correspond to more interesting features which can be hopefully extracted by combining them together. Therefore, we favor properties that are less correlated with the already chosen properties and as such a negative relationship between the absolute value of correlation coefficient $|R_{pi}|$ and the score s_i is shown in Equation 3. Specifically, the coefficient matrix R of property Y_p and Y_i is computed by Equation 4.

$$R_{pi} = \frac{\text{Cov}(p, i)}{\sqrt{\text{Cov}(p, p)\text{Cov}(i, i)}} \quad (4)$$

where $\text{Cov}(p, i)$ is the covariance matrix of property Y_p and Y_i .

However, using correlation coefficient alone could lead to situations where properties that do not increase classification ability can beat more meaningful properties in the scoring, and to remedy this, the entropy of a property is also considered in Equation 3. The entropy value of a property reflects the amount of information contained in that property, is shown as a normalized form $\frac{E(Y_i)}{\max E}$ in Equation 3. The entropy is defined as

$$E(Y_i) = \sum_{b=1}^n p(y_b) \log_2(p(y_b)) \quad (5)$$

where n is the number of bins in the histogram of property Y_i , b is the current bin, and $p(y_b)$ is the probability of data value y_b at current bin. $E(Y_i)$ describes the homogeneity of property Y_i and is inversely proportional to the homogeneity, *i.e.* higher entropy represents less homogeneity.

Properties that are less homogenous usually contain more features of interest compared to more homogenous ones. Therefore, low homogeneity can be used to rule out less contributing

properties that have a high score from correlation coefficient. As such, high entropy is desired in our scheme: i.e. properties that are less homogenous are favored over more homogenous ones. However, low entropy may also be of interest in some occasions: e.g. it is conceivable that a property may contain large homogenous regions but uniquely highlights some small feature that is missed in all other properties. The classification ability of such a property is difficult to describe mathematically but could be rather easily determined subjectively through interactive exploration. Our system would miss such cases.

The parameter a is data set dependent and allows the user to choose a balance between the correlation of two properties and the amount of information contained in an individual property.

The remaining properties are then ranked based on their scores s_i and used as the *tertiary* attributes for the associated 1D transfer functions. We found that using one or two tertiary attributes provides a good compromise between complexity and effectiveness of the classification. One of the available tertiary attributes is selected as the active one for each widget in the 2D transfer function space.

As an example, the process of combination selection for CT chest scan Artifix discussed in section 5.2 is shown below. Using the rules, we compute the correlation coefficients and the entropies of the five properties of the data set as shown in Table 1.

	x	$ \nabla x $	μ	σ	ρ
x	1.0000	0.1654	0.9973	0.2435	0.7286
$ \nabla x $	0.1654	1.0000	0.1690	0.9569	-0.0583
μ	0.9973	0.1690	1.0000	0.2464	0.7388
σ	0.2435	0.9569	0.2464	1.0000	0.0067
ρ	0.7286	-0.0583	0.7388	0.0067	1.0000
E	5.3828	3.6077	4.8012	3.8392	7.9090

Table 1: The correlation coefficients and the entropies of the properties computed from Artifix CT chest scan dataset with the intensity value x , the gradient magnitude $|\nabla x|$, the mean value μ , the standard deviation σ and the occlusion metric ρ . The bottom row shows the entropy E of each attribute.

Choosing the scalar value x as the primary attribute suggests using the gradient magnitude $|\nabla x|$ as the secondary attribute. Then scores $s_{\mu,\sigma,\rho}$ for mean, standard deviation and occlusion properties respectively are computed for the remaining attributes by setting a to 0.4, which yields $s_{\mu,\sigma,\rho} = [-0.0347, 0.1938, 0.3086]$. The occlusion property has the highest score meaning it is the best property regarding both the correlation between it and the primary attribute and the information it contains. As such it is used as the tertiary attribute to define a combined 3D gradient magnitude/occlusion transfer function space.

Alternatively, choosing the mean value μ and the standard deviation σ as the primary and secondary attributes, the scores $s_{x,|\nabla x|,\rho}$ are computed for the other attributes, yielding $s_{x,|\nabla x|,\rho} = [0.0094, 0.2061, 0.3045]$ for scalar, gradient magnitude and occlusion properties respectively. The occlusion property has the highest score and is thus used as the tertiary attribute to define a combined 3D statistical/occlusion transfer function space.

3.3. User Interface

In general, true 3D transfer function widgets are relatively difficult to interact with, since robust and effective interaction with a 3D space is still an open research problem [23]. The proposed combined transfer function space however is separable into a 2D transfer function space and a set of 1D transfer function spaces. Haidacher *et al.* [24] proposed a similar separation method for multimodal visualization, in contrast to their simple triangle shaped windowing function our method provides more insights and flexible controls for the 1D transfer function spaces. This separation as stated before can cause decreased classification precision when the 1D transfer function spaces are not independent from the 2D transfer function space compared to a true 3D transfer function space. However, our combination selection rules proposed in section 3.2 help to rule out highly dependant 1D transfer function spaces. Therefore, we believe this separation is a good trade off between interactivity and classification precision.

Each 1D transfer function is attached to every selected region in the 2D transfer function domain based on the usual transfer function widgets or selectors. Thus, features in the volume can be classified by selecting their voxels in the 2D domain defined by the primary and secondary attributes. In cases where those voxels represent multiple separate features, the additional 1D transfer function can be used to further separate such features within the voxels selected in the 2D domain using one of the tertiary attributes. While adding complexity to the manipulation of transfer functions, this technique provides familiar interaction with each of the 2D and 1D transfer functions (TF). We believe this additional interaction (combining familiar 2D TF manipulation with familiar 1D TF manipulation) provides a reasonable method for interacting with the higher dimensionality of transfer function combinations. However, it does require users to be familiar with such interaction techniques.

Figure 1 illustrates the proposed 3D transfer function editor for a 2D gradient magnitude transfer function space with associated 1D occlusion transfer functions. The top part shows the 2D gradient magnitude transfer function domain $x \times \|\nabla x\|$, where user can place and interact with traditional 2D transfer function widgets [6] TFw_{2D} and a more generic lasso tool. The occlusion volume space ρ_c or the size volume space t_c of the region c selected by the currently active 2D transfer function widget TFw_c is represented by a 1D transfer function editor, shown at the bottom, along with a 1D histogram of the occlusion information of all voxels selected by c . That is, the 1D transfer function editor operates strictly on voxels selected by a 2D transfer function widget (the blue one in Figure 1).

The 2D transfer function widgets, such as ellipse, rectangle or triangle widgets as proposed by Kniss *et al.* [6], typically include some default shapes with few degrees of freedom. Users are able to set colors, opacities and different fall-offs for each of these widgets. These tools provide facilities to the user for a general exploration of transfer function spaces using easy to manipulate *high level* widgets.

However, it is difficult for the user to precisely select arbitrary regions. This often prevents a user from exploring the

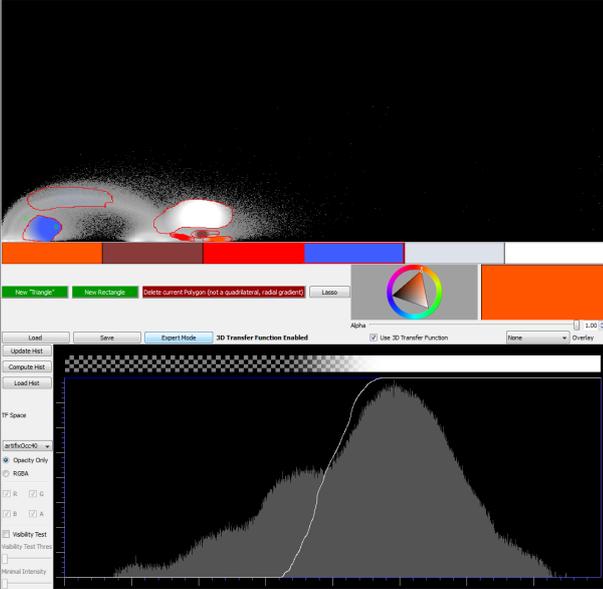


Figure 1: The separable 3D Transfer Function Editor with the 2D gradient magnitude transfer function space $x \times \|\nabla x\|$ shown on top, and the 1D occlusion space ρ_c attached to the currently selected widget c in the 2D space, shown below. In this example, the blue widget is active, and as such the 1D histogram represents the occlusion information of all the voxels with statistical properties selected by the widget in the 2D statistical domain.

subtle structures in the transfer function domain, which may make a significant difference in the final visualization. Thus, similarly to commonly used image processing applications, we also include a *lasso* tool to allow the user to intuitively and easily select arbitrary regions by drawing the region boundaries directly into the transfer function space. In Figure 1, the red curve illustrates the hand drawn boundary path with a spherical fall-off for the color and opacity. A box on the left hand side of the 1D transfer function editor allows the user to select which of the tertiary attributes is used as the 1D transfer function space for each 2D widget.

The proposed user interface allows the user to interact with the 3D transfer function space intuitively. Whenever the user creates a transfer function widget on the 2D transfer function space, the histogram of the voxels selected by that widget is computed and immediately shown in the 1D transfer function editor. Initially, the 1D transfer function maps, as visible, all voxels that are selected by the 2D transfer function widget. With the help of the 1D histogram, one can then design the 1D transfer function intuitively. As such, users are provided with a familiar interface thus providing intuitive interaction. This user interface adds minimal complexity to the standard 1D and 2D transfer function editors in existing volume visualization systems, e.g. Voreen [25] and ImageVis3D [26]. With a 3D transfer function space we are able to leverage the usability of the user interface, however, we are also interested in extending it for higher dimensional transfer function spaces in the future.

4. Specific Transfer Function Spaces Used

In addition to the well known 2D gradient magnitude and scalar value transfer function, we include several recently proposed transfer functions to be used in combinations. Creation of these transfer functions is generally based on the methods described in the respective papers, but with slight modifications, which are discussed in the following subsections: size-based transfer functions [3] in Section 4.1, occlusion-based transfer functions [4] in Section 4.2 and statistical transfer functions [5] in Section 4.3.

4.1. Size Information Computation

Correa and Ma [3] proposed a three step method to create a size volume S from an input volume. The three steps are: scale-space computation, scale detection, and back projection. They use anisotropic diffusion to create the scale space with better localization. The classical normalized Laplacian kernel is used to detect the blobs as local maxima both in spatial and scale domain. A back projection step utilizing Shepard’s interpolation with Wendland polynomials is then conducted for the detected blob tuple (x, y, z, t) .

A single voxel can be part of features with multiple sizes, however only the largest size value is kept at each voxel, thus smaller features get masked out by larger ones, which happens in the brain MRI example shown in Section 5.4. To avoid this situation, we allow the user to specify an intensity range to compute a scale space specifically for that range.

4.2. Occlusion Information Computation

Correa and Ma [4] suggest using an extended ambient occlusion metric to measure the occlusion of the volume. One can view the occlusion information ρ as a weighted sample mean value for a spherical neighborhood with certain radius R centered at each voxel, which results in an isotropic blurring effect that does not preserve the boundaries of the structures.

Sometimes, overly smoothed volumes that lose all their boundary information are not desired, thus we derive a gradient based equation for computing the occlusion information, inspired by work done by Perona and Malik [27].

For a sphere of radius R , we compute the occlusion information of the N voxels x_i surrounding the current voxel x as shown in Equation 6:

$$m_x = \frac{g_x}{N} \sum_{i=1}^N x_i \quad (6)$$

$$g_x = \frac{\eta^2}{\|\nabla I_x\|^2 + \eta^2} \quad (7)$$

In Equation 7, g_x is a term based on the gradient magnitude of the current voxel x .

The data set dependent parameter $\eta \in \mathbb{R}^+$ handles gradients of zero magnitude e.g for $\eta \in [0.001, 0.01]$, essentially helping to preserve boundaries of different structures. If $\eta \geq 1$, the filter behaves similar to a box filter.

Computing m_x is equivalent to convolving the volume with a spherical filter B_R of radius R , and then modulating it with g_x :

$$m_x = g_x \cdot (B_R * I_x) \quad (8)$$

The complexity of this operation is $O(mn)$, where $m = \frac{4}{3}\pi R^3 + 1$, and thus very costly, since the radius should be large enough to maximize the variance of the result [4].

This is infeasible in practice, due to its computational complexity. However, since each sample inside the sphere is treated equally, a box filter of width $2R$ can be used to approximate the sphere. Exploiting the separability of convolving with a box filter and the performance of modern GPUs allows the computation of m_x within seconds. The 3D convolution is then separated into three consecutive convolutions with a 1D box filter b_{2R+1} of width $2R + 1$, as Equation 9 shows:

$$m_x = g_x \cdot \{b_{2R+1} * [b_{2R+1} * (b_{2R+1} * I_x)]\} \quad (9)$$

This separation considerably reduces the computation time. Such an occlusion metric is view-independent and thus can be pre-computed and stored, and therefore does not affect the speed of visualization.

4.3. Statistical Properties

We construct the statistical feature space with a procedure similar to that presented by M. Haidacher *et al.* [5]. They propose to grow a sphere over the neighborhood of each voxel and to compute the following statistical metrics: mean value μ , standard deviation σ , *skewness* as well as *kurtosis*. It is a multi-stage process: first, extract *statistical metrics*, second, conduct the *normality test*. If the test is passed, continue with the *similarity test*. After the similarity test, if the new samples are similar to the old ones, we combine the statistical metrics. If any of the above tests fail or a user defined maximum radius r_{max} is reached, the procedure is terminated, otherwise we increase the neighborhood by one voxel.

Haidacher *et al.* [5] use the Jarque-Bera test [28] for normality since it is easily implementable on a GPU. It however requires a relatively large set of samples in order yield results of sufficient quality. Therefore, various other normality tests have been proposed in the literature; we chose D’Agostino’s K -squared test [29] as a state of the art method. Its robustness with respect to identical values in the dataset makes it a good fit for CT and MRI data sets, which can contain large homogeneous regions.

Utilizing the transformations $Z_1(\sqrt{b_1})$ and $Z_2(b_2)$ of the sample skewness $\sqrt{b_1}$ and the sample kurtosis b_2 , the K -squared test (Equation 10) is then defined as:

$$K^2 = Z_1(\sqrt{b_1})^2 + Z_2(b_2)^2 \quad (10)$$

K^2 is approximately χ^2 -distributed with 2 degrees of freedom; we can test its null hypothesis by looking up the χ^2 -distribution table. The entry for test level $1 - \alpha = 0.999$ with a 2 degrees of freedom in the χ^2 -distribution table is 13.82. Therefore, the normality test will be passed if

$$K^2 < 13.82 \quad (11)$$

If the samples in the spherical neighborhood pass the normality test, it is necessary to further test whether they have the same distribution as that of the samples computed in the previous iteration. As done by Haidacher *et al.* [5], Welch’s T -test [30] is used to compare the similarity of the sample distributions.

5. Results and Discussion

The statistical properties, the occlusion information, and the size information are all pre-computed on the GPU, and those volumes are then used in the interactive visualization stage to define the transfer function space. Users interact with an extended slice-based volume renderer implemented in OpenGL and Qt that supports combined 3D transfer functions to explore and generate final visualizations.

The scoring process is not part of our volume renderer and is conducted in MATLAB only once for a dataset. The input is a matrix where each of its columns is a property volume that is flattened into an 1D array. The correlation coefficient matrix is computed by the MATLAB function `corrcoef` which uses Pearson’s correlation. While the MATLAB function `entropy`, that implements Equation 5, is applied to compute the entropy of each property by taking the histograms of the properties in a column of the input matrix. The number of bins of the histograms are determined by the number of bits of the data, e.g. an unsigned 8 bit volume has 256 bins. Finally, Equation 3 is evaluated for the corresponding row of the major property in matrix R and the normalized entropy vector $\frac{E(Y_i)}{\max E}$. The whole process takes about 10 seconds for each of the examples shown below.

The following discussion compares 2D gradient magnitude, 2D statistical, 2D occlusion, 2D size with 3D combined statistical/occlusion, statistical/size, occlusion spectrum/size or statistical/(occlusion, size) transfer functions applied to a synthetic data set and real world data sets. The combined 3D transfer functions for each data set were typically designed within 15 to 20 minutes, similar to the time required to design the traditional 2D transfer functions. The synthetic data set models a filled shell encompassing varying sized spheres; the “Artifix” data set has been retrieved from the OsiriX DICOM archive [31]. The back pack and the “Artifix” data sets are CT scans of a back pack and chest respectively, “CerebrixCrop” is the T1 channel of an MRI scan of a brain.

The parameters used to create the transfer function spaces are chosen by trial-and-error on each data set. For the synthetic dataset and CT datasets are computed with confidence level 0.1 while the MRI dataset with a confidence level 0.001 when generating the statistical transfer function space. Radius is set to 40 for all datasets when creating the occlusion volumes. The synthetic dataset is processed with a boundary preserving parameter $\eta = 1.0$ in order to overcome the noisiness whereas all other datasets use $\eta = 0.005$ to preserve the boundary details. Size property computed for the MRI dataset is limited to the intensity range [250, 500] in order to classify the tumor.

The transfer function combinations shown below are chosen by applying the algorithm described in section 3.2 with varying

parameter a . An exception to this are the results shown in in Figure 2 where the extremely noisy nature makes Equation 3 ineffective.

5.1. Synthetic Dataset

A synthetic dataset was created, as illustrated in Figure 2(a) in order to mimic a common scenario in real life medical datasets, such as chest CT scans or head MRI scans, where different structures overlap both spatially and in the scalar values. Often, the outer structures occlude the inner ones, but they also can have different sizes. The synthetic dataset contains six different materials: the environment with $\mu_0 = 0.20, \sigma_0 = 0.14$, the middle hull with $\mu_1 = 0.40, \sigma_1 = 0.16$, the outer hull and the upper small inner sphere with $\mu_2 = 0.60, \sigma_2 = 0.11$, and both the remaining larger and smaller inner spheres have $\mu_3 = 0.80, \sigma_3 = 0.13$. In addition, low amplitude noise following a Gaussian distribution has been added across the whole domain to simulate noise introduced by acquiring a volumetric image with a scanner.

Various transfer functions have been applied to the synthetic dataset, as shown in Figure 2. Traditional 2D gradient magnitude based transfer functions, as Figure 2(b) illustrates, suffer severely from the overlapping scalar values in the transfer function domain. There, features are indistinguishable due to noise, which makes it hard to separate features based on their gradient magnitude, as seen in the joint histogram in Figure 2(b).

Occlusion spectrum 2D transfer functions, shown in Figure 2(c) are able to separate the inner and outer structures based on their occlusion property as in the transfer function shown in Figure 2(c). The three inner spheres however cannot be separated clearly due to the similarity in their occlusion information as well as their scalar values. Also, the center of the inner yellow region overlaps with all spheres in the occlusion spectrum, thus causing misclassification.

The size based 2D transfer function applied to the data set (Figure 2(d)) separates the inner spheres from each other and the outer rings, however there are classification artifacts at the top and right part of the green outer ring. The small sphere at the bottom right cannot be properly separated from the purple sphere, since they both overlap in their scalar values.

Statistical 2D transfer functions, as demonstrated in Figure 2(e), are able to separate the overlap in the (μ, σ) transfer function domain. It is thus possible to classify them using different properties. However, both the spheres at the lower center have the same statistical properties, and similarly, the outermost shell shares the statistical properties with the upper central sphere, yet they represent different structures.

Supplementing the statistical information with occlusion information, as shown in Figure 2(f), makes it possible to separate the inner purple sphere, compared to Figure 2(e). The transfer function in Figure 2(f) shows that the 1D occlusion histogram for the highlighted 2D widget can be used to separate the purple sphere with its low amount of highly occluded voxels from the green outer shell, which has a higher amount of less occluded voxels, however the two red spheres at the bottom are not separated from each other.

On the other hand, supplementing the statistical information with size information, as shown in Figure 2(g), makes it possible to separate the two spheres at the bottom into the cyan small one and the larger red one, when compared to Figure 2(e). Noticeable are the purple artifacts in the green outer shell at the right side, since that region has a similar feature size compared to the purple sphere.

Figure 2(h) shows that occlusion and size information together are able to classify all the features of the data set without ambiguity. The 1D transfer function associated to each widget in the 2D statistical transfer function space uses either size information *or* occlusion information to further classify the voxels selected in the statistical 2D transfer function domain, thus allowing the user to exploit benefit of either method, while being able to interact with 1D and 2D transfer functions, instead of 3D or 4D transfer functions.

5.2. CT Scan of a Chest: “Artifix”

In the chest CT scan “Artifix” (Figure 3), both traditional 2D and combined 3D transfer functions were used to classify the lung (blue), bones (shades of gray), blood vessels (red), aorta (dark orange), kidney (brown), the skin (transparent gray).

The gradient magnitude transfer function (Figure 3(a)) fails to correctly separate the blood vessels and the kidneys from the bones. Also noticeable is the relatively high amount of noise distributed across the volume.

The occlusion spectrum (Figure 3(b)) can be used to separate the kidney from the surrounding tissue. However, the aorta is similarly classified, since they are overlapping in the occlusion spectrum. Also details of the lung are lost, since its tissue has similar occlusion values compared to the surrounding tissue, due to the intricacy and delicacy of the alveoli and bronchioles.

A statistical transfer function (Figure 3(c)) removes a noticeable amount of that noise, but still leaves some areas, such as the front part of the ribs, and the kidney misclassified, since they are close with respect to their statistical properties.

Experimentation with the size based transfer function as the associated transfer function space did not measurably improve the classification since the relative similarity of the scalar values in this CT scan mapped them to similar size values.

However combining occlusion information with either a 2D gradient magnitude transfer function (Figure 3(d)) or a statistical transfer function (Figure 3(e)) increases the ability to correctly separate the kidneys from the aorta. The fine structures of the lung’s surface are identifiable, since they have different statistical properties compared to the surrounding tissues. There are only slight differences between the combined transfer functions since they are similar without considering occlusion information.

5.3. CT Scan of a back pack

The scoring with $a = 0.6$ conducted on the back pack dataset with scalar value chosen as the main property, gradient magnitude as the intrinsically associated secondary property results in: $s_{\mu, \sigma, \rho, S} = [-0.5208, -0.3882, 0.2165, -0.1595]$ suggests that the occlusion volume ρ and size volume S should

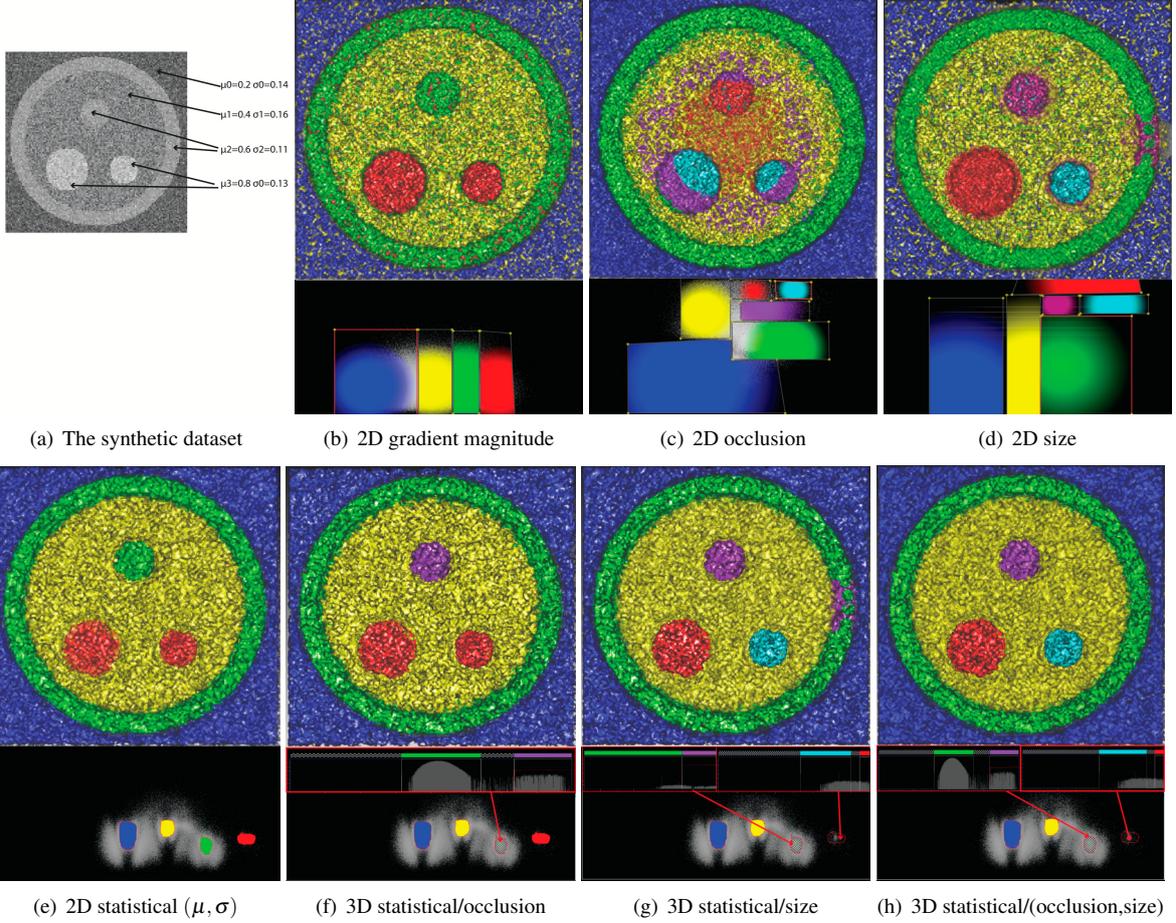


Figure 2: The synthetic dataset was created as mixture of overlapping Gaussian distributions with varying parameters to model a filled shell encompassing varying sized spheres as shown in a). It has been classified, from left to right, using transfer functions (shown right below the rendered images) based on b) 2D gradient magnitude, c) 2D occlusion spectrum and d) 2D size based transfer function, e) 2D statistical, f) combined statistical/occlusion transfer function, g) combined statistical/size transfer function and h) combined statistical/(occlusion, size) transfer function.

be considered for tertiary attributes. Changing the main property to mean volume with standard deviation volume as the secondary attribute gives the scoring for the rest properties: $s_{x,|\nabla x|,\rho,S} = [-0.4847, -0.2140, 0.2153, -0.1679]$ also hints to us that the occlusion volume ρ and size volume S should be used as tertiary attributes.

Figure 4 shows the CT scan of a back pack filled with liquids (in red, green, blue), a battery (in purple) and a box (in cyan) classified with various transfer functions.

The 2D transfer functions separate the different liquids to varying degrees, but they fail to identify the battery properly. Other features, such as the wires or the small circular shapes are mapped to the same color yellow (Figures 4(a), 4(b)), or the same feature is mapped to different colors (Figures 4(c), 4(d)). Notable is the 2D occlusion transfer function which allows the extraction of the cyan box but classifies the liquids with less specificity.

Adding occlusion as the 3rd axis did not yield meaningful results, since the data set itself has many features which are similarly occluded by the clothing articles (showing in transparent gray) inside the back pack, thus reducing the separability in the occlusion channel.

Utilizing a size transfer function as the third axis allows the clear separation of the battery (purple color), the 3D occlusion spectrum/size transfer function (Figure 4(g)) is additionally able to visualize the cyan box, which is difficult to do using gradient magnitude (Figure 4(e)) and statistical information (Figure 4(f)) as the 2D transfer function domain. However, all of the 3D transfer functions have problems in classifying the wires as features both connected spatially and with respect to their colors, suggesting further investigations of alternative volumetric attributes as the third axis.

5.4. MRI Scan of a Brain: “CerebrixCrop”

MRI datasets, occurring in clinical and research studies where separating the brain from the surrounding tissue is of particular interest, are typically challenging to classify, since they often contain ubiquitous noise [32]. Figure 5 shows such a data set containing a tumor in the center of the brain. Transfer functions are applied to classify the brain tissue (in yellow) and the fluid inside the tumor (in red). Note that although both features can be shown simultaneously by setting transparency of the brain, we set the brain to completely transparent in the second row of images for clear visualizations of the tumor.

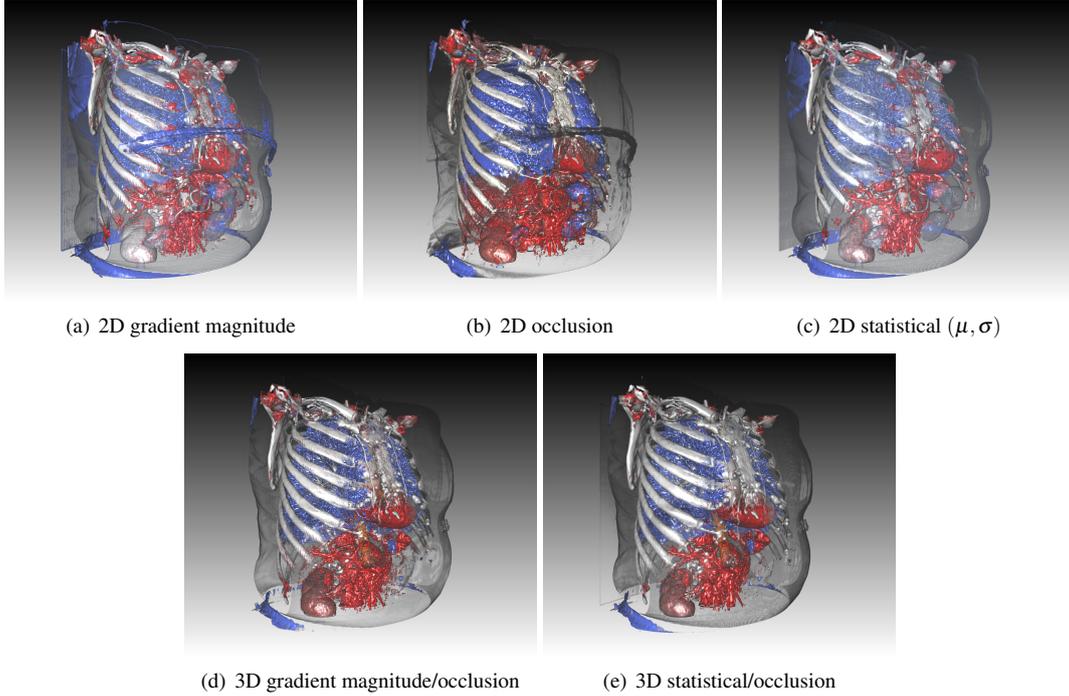


Figure 3: The chest CT scan “Artifex” classified using transfer functions based on a) 2D gradient magnitude, b) 2D occlusion spectrum, c) 2D statistical, d) combined 3D gradient magnitude/occlusion transfer function and e) combined 3D statistical/occlusion transfer function.

We apply the scoring process with $a = 0.6$ to the MRI dataset: set scalar value as primary and gradient magnitude as secondary yields: $s_{\mu, \sigma, \rho, S} = [-0.3188, -0.1247, 0.1674, -0.0224]$ meaning that the occlusion volume ρ and size volume S once again should be considered for tertiary attributes. Substituting the main attribute with mean value while standard deviation as the secondary attribute gives $s_{x, |\nabla x|, \rho, S} = [-0.2841, -0.0591, 0.1505, -0.0298]$ and leads us to the same decision.

Gradient magnitude based 2D transfer functions (Figure 5(a)) fail to properly separate the brain from the skin, since they both share similar ranges of scalar values and gradient magnitudes.

Figure 5(f) demonstrates the inability for the gradient magnitude based 2D transfer functions to clearly pull out the tumor, since similar scalar values and gradient magnitudes appear universally across the data set.

The occlusion spectrum (Figures 5(b) and 5(g)) helps to better separate the brain from its surrounding tissues as well as remove noise with scalar values similar to the tumor. However, the surface of the brain tissue is still incorrectly classified and a large amount of noise still appears around the tumor due to similar occlusion values in these regions.

Statistical transfer functions (Figures 5(c) and 5(h)) significantly smooth the data set, making the creases and recesses of the brain tissue clearly show up, however, noise that heavily affects the visual quality is still seen across the data set, especially in Figure 5(h).

Combining the occlusion information with statistical information, as shown in Figures 5(d) and 5(i) classifies the brain

tissue properly, but fails to clearly extract the tumor.

However a transfer function combination with two tertiary attributes, as shown (Figures 5(e) and 5(j)) clearly separates both the brain tissue and the tumor. The statistical attributes are used as the primary and secondary attributes, and the occlusion and size information are used as the tertiary attributes.

The widget that classifies the yellow brain tissue uses the occlusion attribute to further classify it with the associated 1D transfer function; however the widget classifying the red tumor uses the size attribute instead to further remove the noise via its associated 1D transfer function.

5.5. Multivariate Data set: Hurricane Isabel

One time step (time step 30) of the VisContest 2004 Hurricane Isabel [33] multi-variate data set is used to demonstrate the generality of our method. The original data set contains 12 attributes, many of these attributes however, are redundant or contain little amount of information.

Three most salient attributes are selected, namely pressure, temperature and QVAPOR by evaluating the entropy of each attribute. The 2D transfer function domain is pressure and temperature. We use QVAPOR as the associated 1D transfer function. Each different colored widget in the 2D domain uses a different QVAPOR 1D transfer function. These attributes are then used to classify features, as shown in Figure 6. The eye of the hurricane (shown in red) has a lower pressure but higher temperature than the blue outer bands and lower temperature compared to the yellow and green spiraling bands. The QVAPOR allows us to see the spiraling bands in the data set.

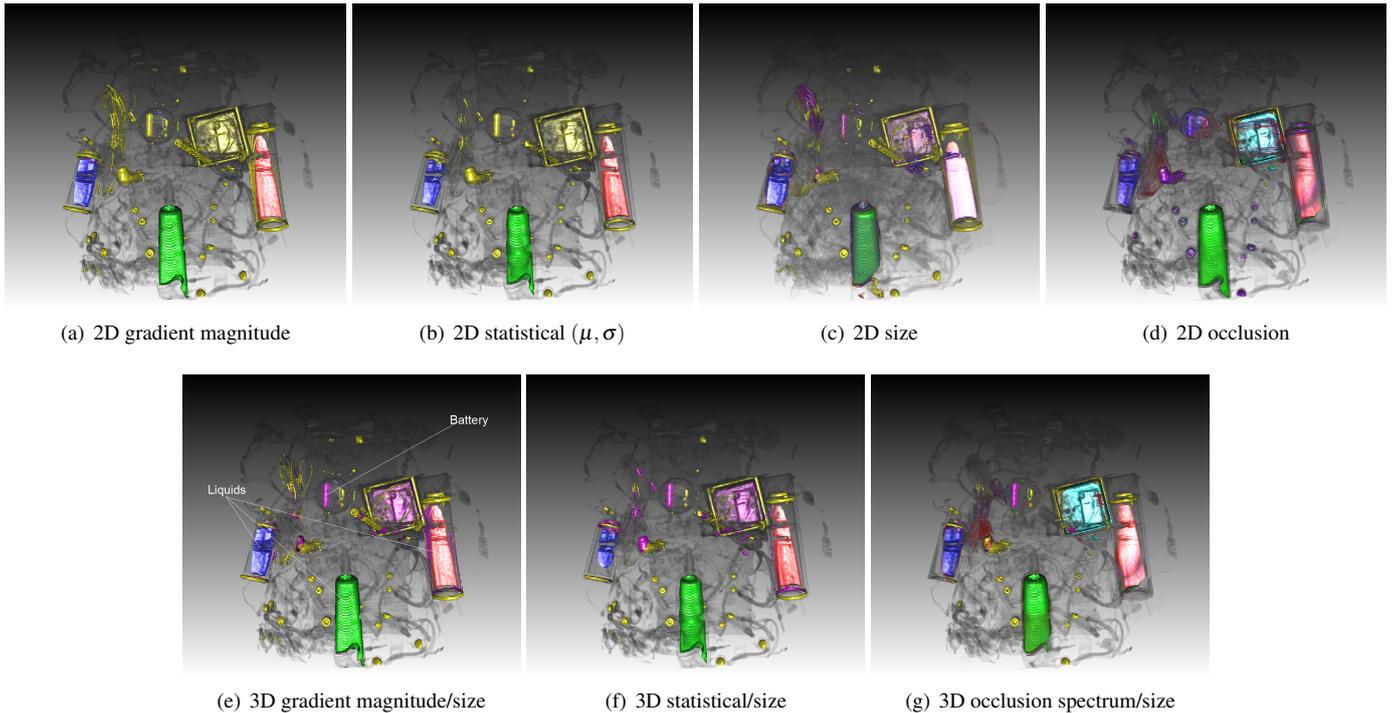


Figure 4: The back pack CT scan classified using transfer functions based on a) 2D gradient magnitude, b) 2D statistical, c) 2D size, d) 2D occlusion, e) combined 3D gradient magnitude /occlusion transfer function, f) combined 3D statistical/size transfer function and g) combined occlusion spectrum and size transfer function.

6. Conclusion and Future Work

In this paper, a transfer function combination method has been introduced to improve classification results using existing transfer function spaces, specifically, the scalar/gradient magnitude transfer function space, the statistical transfer function space, the occlusion transfer function space and the size based transfer function space. Combinations that have better specificity than the element transfer functions are selected. A moderate amount of precomputation which has been accelerated using GPUs and separable convolution filters allows subsequent interactive design and manipulation of the combined transfer functions via an intuitive transfer function editor. Synthetic and scanned data sets were used to demonstrate that combining 2D transfer functions with 1D transfer functions improves the results of classification than that from 2D transfer functions.

Several improvements are of interest in the future. The transfer function combinations are selected using the rules described in section 3.2 to produce the results in the paper, and we hope to develop a more robust method to automatically choose the best transfer function combinations.

We would like to develop new metrics and experiment with transfer function combinations using these metrics. Transfer function combinations are not necessarily restricted to only one 2D transfer function followed by a single selection from a set of 1D transfer functions. A chain of transfer functions (2D + 1D + 1D + ... or 2D + 2D + 1D + 1D + ... or even 2D + 2D + 2D + ...) can be applied to a volume although the user interface becomes difficult due to the high dimensionality imposed by such a chain. In order to improve the classification over a specific

region, metric volumes used for further classification steps may be computed locally from the regions already classified instead of being precomputed globally.

Our proposed method helps to better classify objects in the volumes via low-level user interactions, i.e. the user has to have explicit knowledge about transfer function spaces and has to manually tune the 3D transfer functions. We would like to develop semantics based high level transfer function design mechanisms build upon our frame work: e.g. predefine several transfer functions for objects of interest in a dataset (e.g. bones, skin, and blood vessels etc. in the CT chest scan) as training data and use supervised learning to generate initial transfer functions for similar datasets; or use similar methods proposed in Rezk-Salama *et al.* [34]. Ultimately, more information that guides the user to a quicker and more intuitive classification experience is the overall future goal.

7. Acknowledgements

We would like to thank the reviewers for helpful comments which improved our manuscript. This publication is based on work supported by Award No. KUS-C1-016-04, made by King Abdullah University of Science and Technology (KAUST), DOE SciDAC:VACET, NSF OCI-0906379, NIH-1R01GM098151-01.

References

- [1] G. Kindlmann, J. Durkin, Semi-automatic generation of transfer functions for direct volume rendering, in: IEEE Symposium on Volume Visualization, 1998, pp. 79–86.

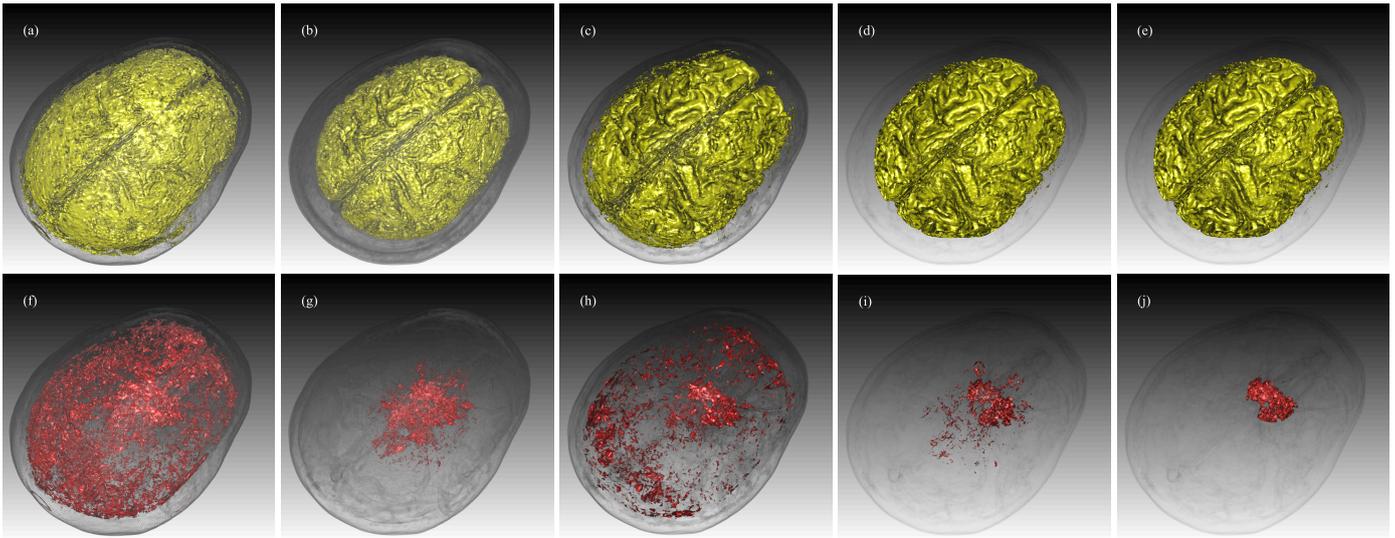
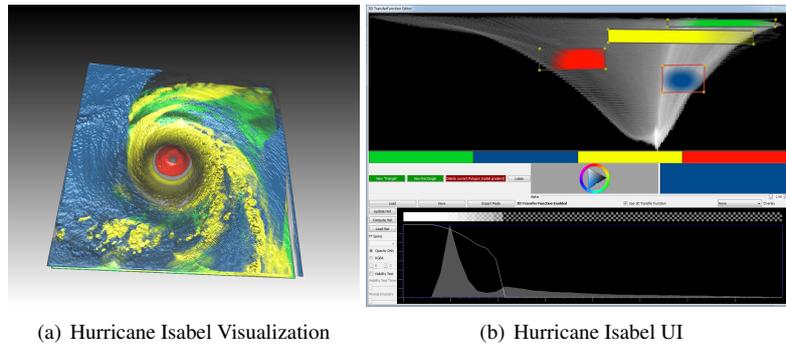


Figure 5: The “CerebrxCrop” MRI data set shown with focus on the brain tissue shown as yellow (top row) and a tumor shown in red (bottom row). The following transfer functions were applied: a,f) 2D gradient magnitude, b,g) 2D occlusion, c,h) 2D statistical, d,i) 3D statistical/occlusion, e,j) 3D statistical/(occlusion,size).



(a) Hurricane Isabel Visualization

(b) Hurricane Isabel UI

Figure 6: Visualization of the multi-variate Hurricane Isabel data set using pressure and temperature in the 2D transfer function with different 1D transfer functions using QVAPOR for each 2D transfer function widget shown in different colors.

- [2] M. Levoy, Display of surfaces from volume data, *IEEE Computer Graphics and Applications* 8 (3) (1988) 29–37.
- [3] C. Correa, K.-L. Ma, Size-based transfer function: A new volume exploration technique, *IEEE Transactions on Visualization and Computer Graphics* 14 (6) (2008) 1380–1387.
- [4] C. Correa, K.-L. Ma, The occlusion spectrum for volume classification and visualization, *IEEE Transactions on Visualization and Computer Graphics* 15 (6) (2009) 1465–1472.
- [5] M. Haidacher, D. Patel, S. Bruckner, A. Kanitsar, M. E. Gröller, Volume visualization based on statistical transfer-function spaces, in: *Pacific Visualization Symposium*, 2010, pp. 17–24.
- [6] J. Kniss, G. Kindlmann, C. Hansen, Multidimensional transfer functions for direct volume rendering, *IEEE Transactions on Visualization and Computer Graphics* 8 (3) (2002) 270–285.
- [7] C. Lundstrom, P. Ljung, A. Ynnerman, Local histograms for design of transfer functions in direct volume rendering, *Visualization and Computer Graphics*, *IEEE Transactions on* 12 (6) (2006) 1570–1579.
- [8] P. Sereda, A. Bartroli, I. Serlie, F. Gerritsen, Visualization of boundaries in volumetric data sets using lh histograms, *Visualization and Computer Graphics*, *IEEE Transactions on* 12 (2) (2006) 208–218.
- [9] E. B. Lum, J. Shearer, K.-L. Ma, Interactive multi-scale exploration for volume classification, *The Visual Computer* 22 (9-11) (2006) 622–630.
- [10] Y. Sato, C. Westin, A. Bhalerao, S. Nakajima, N. Shiraga, S. Tamura, R. Kikinis, Tissue classification based on 3d local intensity structures for volume rendering, *Visualization and Computer Graphics*, *IEEE Transactions on* 6 (2) (2000) 160–180.
- [11] J.-S. Prassni, T. Ropinski, J. Mensmann, K. Hinrichs, Shape-based transfer functions for volume visualization, in: *Pacific Visualization Symposium (PacificVis)*, 2010 IEEE, 2010, pp. 9–16.
- [12] G. Kindlmann, R. Whitaker, T. Tasdizen, T. Moller, Curvature-based transfer functions for direct volume rendering: Methods and applications, in: *IEEE Visualization*, 2003, pp. 513–520.
- [13] J. J. Caban, P. Rheingans, Texture-based transfer functions for direct volume rendering, *IEEE Transactions on Visualization and Computer Graphics* 14 (6) (2008) 1364–1371.
- [14] D. Patel, M. Haidacher, J.-P. Balabanian, M. E. Gröller, Moment curves, in: *Pacific Visualization Symposium*, 2009, pp. 201–208.
- [15] E. Vučini, D. Patel, M. E. Gröller, Enhancing visualization with real-time frequency-based transfer functions, in: *Proceedings of IS&T/SPIE Conference on Visualization and Data Analysis*, 7868, IS&T/SPIE, 2011, pp. 78680L–1–78680L–12.
- [16] H. Piringer, R. Kosara, H. Hauser, Interactive focus+context visualization with linked 2d/3d scatterplots, in: *Coordinated and Multiple Views in Exploratory Visualization*, 2004. *Proceedings. Second International Conference on*, 2004, pp. 49–60.
- [17] Simvis, <http://www.simvis.at/>.
- [18] H. Guo, H. Xiao, X. Yuan, Multi-dimensional transfer function design based on flexible dimension projection embedded in parallel coordinates, in: *Pacific Visualization Symposium*, 2011, pp. 163–170.
- [19] J. Kniss, S. Premoze, M. Ikits, A. Lefohn, C. Hansen, E. Praun, Gaussian transfer functions for multi-field volume visualization, *Visualization Conference*, IEEE (2003) 497–504.

- [20] C. Rezk-Salama, Volume Rendering Techniques for General Purpose Graphics Hardware, Ph.D. thesis (2002).
- [21] S. Bruckner, M. E. Gröller, Style transfer functions for illustrative volume rendering, *Computer Graphics Forum* 26 (3) (2007) 715–724.
- [22] M. Hadwiger, J. M. Kniss, C. Rezk-salama, D. Weiskopf, K. Engel, *Real-time Volume Graphics*, A. K. Peters, Ltd., Natick, MA, USA, 2006.
- [23] D. A. Bowman, J. Chen, C. A. Wingrave, J. Lucas, A. Ray, N. F. Q. Li, Y. Haciahetoglu, J. sun Kim, S. Kim, R. Boehringer, T. Ni, New directions in 3d user interfaces, *International Journal of Virtual Reality* 5 (2) (2006) 3–14.
- [24] M. Haidacher, S. Bruckner, A. Kanitsar, M. E. Gröller, Information-based transfer functions for multimodal visualization, in: W. N. C.P Botha, G. Kindlmann, B. Preim (Eds.), *VCBM, Eurographics Association*, 2008, pp. 101–108.
- [25] Volume rendering engine, <http://www.voreen.org/>.
- [26] ImageVis3D: A Real-time Volume Rendering Tool for Large Data. Scientific Computing and Imaging Institute (SCI). [link].
URL <http://www.imagevis3d.org>
- [27] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12 (7) (1990) 1364–1371.
- [28] C. M. Jarque, A. K. Bera, Efficient tests for normality, homoscedasticity and serial independence of regression residuals, *Economics Letters* 6 (3) (1980) 255 – 259.
- [29] R. B. D’Agostino, A. Belanger, J. D’Agostino, Ralph B., A suggestion for using powerful and informative tests of normality, *The American Statistician* 44 (4) (1990) 316–321.
- [30] B. L. Welch, The generalization of ‘student’s’ problem when several different population variances are involved, *Biometrika* 34 (1/2) (1947) 28–35.
- [31] Osirix dicom sample image sets, <http://pubimage.hcuge.ch:8080/>.
- [32] G. Gerig, O. Kubler, R. Kikinis, F. Jolesz, Nonlinear anisotropic filtering of mri data, *IEEE Transactions on Medical Imaging* 11 (2) (1992) 221–232.
- [33] Viscontest2004 datasets, <http://vis.computer.org/vis2004contest/>.
- [34] C. Rezk-Salama, M. Keller, P. Kohlmann, High-level user interfaces for transfer function design with semantics, *IEEE Transactions on Visualization and Computer Graphics* 12 (5) (2006) 1021–1028.