International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:7, 2009

Topographic Arrangement of 3D Design Components on 2D Maps by Unsupervised Feature Extraction

Stefan Menzel

***Abstract*—**As a result of the daily workflow in the design development departments of companies, databases containing huge numbers of 3D geometric models are generated. According to the given problem engineers create CAD drawings based on their design ideas and evaluate the performance of the resulting design, e.g. by computational simulations. Usually, new geometries are built either by utilizing and modifying sets of existing components or by adding single newly designed parts to a more complex design.

The present paper addresses the two facets of acquiring components from large design databases automatically and providing a reasonable overview of the parts to the engineer. A unified framework based on the topographic non-negative matrix factorization (TNMF) is proposed which solves both aspects simultaneously. First, on a given database meaningful components are extracted into a parts-based representation in an unsupervised manner. Second, the extracted components are organized and visualized on square-lattice 2D maps. It is shown on the example of turbine-like geometries that these maps efficiently provide a well-structured overview on the database content and, at the same time, define a measure for spatial similarity allowing an easy access and reuse of components in the process of design development.

***Keywords*—**Design decomposition, topographic non-negative matrix factorization, parts-based representation, self-organization, unsupervised feature extraction.

I. Introduction

DURING the daily process of design development, in the industrial area a huge amount of different geometries is compiled for solving a given technical problem. Usually, these designs are created by teams of engineers who store their proposals in large CAD data repositories. Because of the size of the database it is almost impossible for an individual engineer to get an entire overview on the database content just by scanning visually through it. Consequently, reliable computational methods for extracting a structured database overview and processing it automatically to a well-presented visualization of its content are highly required and helpful in many ways. First, the chances of multiple reinventions of similar designs are reduced. Second, a perfect approach defines a measure for similarity, hence, allowing a search algorithm to automatically scan for similar designs and suggest them to the designer. Third, a new team member is capable of acquiring development steps of a design in history and is integrated faster into the team.

Under the headline of content based 3D shape retrieval, classical methods and algorithms are summarized, targeting key issues such as e.g. shape representation, similarity measures, search efficiency [1] which overlap with related topics as shape analysis and shape decomposition [2]. Further details on existing methods and concepts are given in section II. In the present paper, a solution for the analysis and visualization of 3D shape data is suggested which relies on an unsupervised method for extracting features organized in a parts-based representation. The proposed approach is motivated from a biological point of view and refers to methods known from the field of object and pattern recognition in images. Technically, the application of the topographic non-negative matrix factorization (TNMF) [3], a variant of the standard NMF [4], is suggested to extract and organize local features in a parts-based representation in an unsupervised way. A fast application of the TNMF in the area of 3D design development is possible by taking advantage of the analogy of pixels in 2D images and voxels in the 3D world.

With respect to pattern recognition in 2D images, the standard NMF is capable of decomposing the content of large image repositories into salient parts or features with a more local character. These parts or components, as it will be further referred to, are used as an efficient representation of the images and their presence helps to predict the content of a before unseen image. Mathematically, the NMF calculates the decomposition by minimizing the reconstruction error of the image database while iteratively adjusting two matrices. The first one contains the basis vectors which are finally interpretable as the components. The second one contains the coefficients which reflect the contribution of each basis vector to reconstruct one of the images in the database.

In a previous work, the applicability and high quality of the sparse-orthogonal NMF has already been shown in the engineering field of 3D design development [2]. The method robustly generated local 3D design components of general relevance. Because of the non-negativity constraint of the matrices these components comprise a high degree of interpretability which is very important for 3D CAD designs

Stefan Menzel is with the Honda Research Institute Europe GmbH, Carl-Legien-Strasse 30, D-63073 Offenbach/Main, Germany (e-mail: stefan.menzel@honda-ri.de).

in real-world applications. So far, the algorithm suggested in [2] is able to extract salient design components but it drops their global topology context, i.e. the local neighbourhood between different parts is lost. Nevertheless, especially the topology contains valuable information on the arrangement and spatial distance of parts. Therefore, to integrate the local links between components the application of the TNMF is proposed which has been suggested by Hosoda *et al.* [3] for learning objects in the visual cortex. The TNMF extends the standard NMF by a matrix defining neighbourhood connections for local features. More details on the TNMF are given in section III.

In the application field of 3D design development, it is intended to utilize the additional features of the TNMF, as they enable the simultaneous extraction and organization of design components. Based on the spatial distance measure it is possible to arrange the components on 2D design maps. These maps visualize on the one hand an interpretable decomposition of all designs contained in the design repository and on the other hand an overview on the spatial distribution of these components. Similar components are easily detected visually and engineers are able to locate possible further development directions or exclude unsuitable design developments based on their local neighbourhoods.

Because of the unsupervised character of the proposed framework, an additional advantage is the extraction of conceptual important components which may not be obviously visible to the engineers if scanning visually through holistic designs. E.g. if a number of designs contain always a part which is shaped in a similar way, the NMF is capable of extracting it as a single component with general character. Furthermore, since the framework as described in section IV relies on voxels, it is possible to perform the algorithm on different hierarchical scales depending on the chosen voxel size. Global maps contain more global components on a high level, and additional maps with a finer granularity, i.e. smaller voxel size, provide more detailed smaller components. In the ideal case, an engineer could zoom in and out of the design maps, by adjusting the size of the underlying voxel representation.

To illustrate the proposed framework, in section V the application to an engineering scenario based on 3D turbine-like structures is presented. This scenario highlights the decomposition performance and illustrates how 2D design maps may look like.

Section VI closes with concluding remarks.

II. Data Structuring, Shape Decomposition and Shape Descriptors

In the area of 3D shape retrieval there exist three topics which are considered as particular important in the context of the framework presented in the present paper. This section points to these aspects, namely data structuring, shape decomposition and shape descriptors providing more explanatory details and related literature.

From a more general point of view, methods for clustering and organizing data are of interest which are possible to be applied either on holistic designs or on design parts. These methods comprise state-of-the-art algorithms like e.g. the *principle component analysis* (PCA) [5], *k-means clustering* [6] or *self-organizing maps* (SOM) [7]. PCA is a well established and widely used statistical technique for discovering main features within data sets. Given a set of *N*-dimensional points the PCA aims to re-express the data by finding a linear transformation of the coordinate system which results in an optimal representation of the data set in eigenvectors. A given data set is then reproducible by using the calculated eigenvectors ranked by their corresponding eigenvalues. K-means clustering attempts to separate given data into *k* distinct clusters. A major drawback of the algorithm is the fact that the number of clusters must be known in advance, thus making it unsuitable for unsupervised problems. Furthermore, the result of the clustering algorithm strongly depends on the choice of the correct metric for calculating the similarity between the shapes. A self-organizing map (SOM) is a very common method for organizing and visualizing data on 2D maps. The method is based on artificial neural networks which utilize neighbourhood functions for structuring the input data. Nevertheless, in contrast to the framework proposed in the present paper for extracting and organizing parts-based representations, SOMs focus on holistic data.

With a closer focus on the application of 3D design data and their decomposition in salient parts, Bozakov *et al.* [2] and Bozakov [8] give a brief overview on relevant methods. The approaches of *convex decomposition* [9] and *approximate convex decomposition schemes* [10], [11] have been suggested to decompose designs into polygons. For the 2D case a number of algorithms for optimal solutions exist and concepts for the 3D case have been developed and implemented. Computational costs restrict these methods mostly to simple shapes. *Geometrical skeletons* have been suggested to represent geometries using a set of line segments. A graph-based representation can then be extracted from the skeleton. On a large number of skeletons methods from graph theory are applicable to find matching sub-graphs, which represent individual components present in multiple geometries [12]. *Fuzzy clustering for object segmentation* and *cuts* to extract regions corresponding to features called patches is presented in [13]. Based on facet distance information, a probability is calculated that two facets belong to a certain patch. Since the method is computationally expensive, a decomposition of large models is accomplished by generating a simplified model on which the decomposition is performed and projecting the resulting patches onto the original model.

A biologically motivated solution to generate a parts-based representation of components has been researched by Lee *et al.* [4]. They suggest the application of the non-negative matrix factorization (NMF) to extract salient parts in images. In Fig. 1, the result of the NMF on a data set of faces is depicted. The NMF decomposes the images given in a database into salient parts like e.g. noses, eyes, eyebrows etc.

contained in the feature matrix. The factorization of feature and coefficient matrix allows the reconstruction of a face image which is contained in the original data set with minimal error. In the context of 3D design decomposition the NMF promises several appealing characteristics. The decomposition is calculated in an unsupervised manner, the resulting feature matrix contains only non-negative components allowing a high degree of interpretability and the coefficient matrix allows an analysis of the occurrence and distribution of the extracted features in the database.

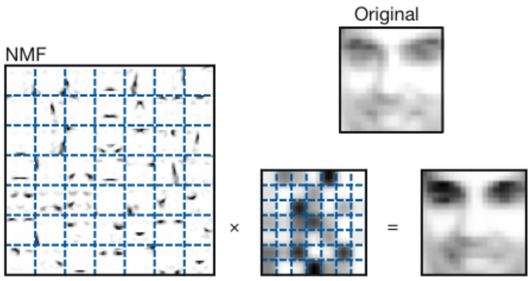

Fig. 1 NMF applied to face image data, taken from [4]

In [8], Bozakov already evaluated the NMF and several variants on their applicability for 3D design decomposition. A sparse-orthogonal NMF framework has been released which offers the advantage of an automatic adaptation of the feature number to the given problem. Instead of defining the number of components in advance which may lead to inappropriate 3D decompositions, the sparse-orthogonal NMF adjusts the feature number during runtime. Additionally, the algorithm contains the possibility to deal with thin design parts by iterative dilation steps on the voxel representation. The good performance of the framework has been illustrated on a design database containing 3D turbine-like shapes. The sparse-orthogonal NMF was capable to decompose a given set of turbines into an optimal set of interpretable components which likewise reconstruct the database with minimal error [2]. The result of the sparse-orthogonal NMF on 3D turbine-like shaped geometries is depicted in Fig. 2. Fig. 2(a) illustrates a horizontal cross-section through the 3D turbine, Fig. 2(b) and (c) depict two extracted components allowing a reconstruction of turbines similar to the type shown in Fig. 2(a).

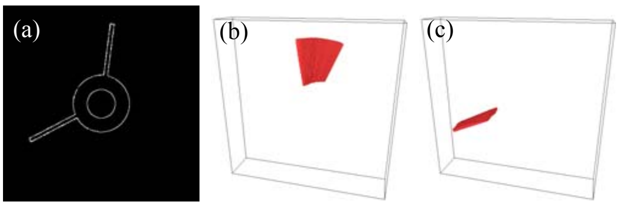

Fig. 2 Sparse-orthogonal NMF applied on 3D turbine-like structures, taken from [2]

So far, more general methods for clustering data and algorithms for decomposing geometries into salient parts have been summarized. From the point of view of handling 3D designs in technical applications, important aspects for organizing geometries and searching for similar designs are shape descriptors and similarity measures. These issues are reflected by methods developed for automatic shape retrieval. Motivated by the high increase of available 3D models in different domain databases, adequate methods for searching similar designs are of high interest and constantly researched. Tangelder *et al.* [1] and Funkhouser *et al.* [14] present well-prepared surveys on the most important issues for shape retrieval systems, such as shape representation, similarity measures, efficiency, discrimination abilities, ability to perform partial matching, robustness and normalization. One main criterion is the way of representing the descriptors characterizing the design as they significantly influence the search process. Three main methods, namely *feature based methods, graph based methods* and *geometry based methods* are broadly categorized [1].

Feature based methods make use of global features, global feature distributions, spatial maps or local features to generate a vector of constant dimension for all designs. Hence, the vector content is dependent on the kind of chosen features comprising e.g. statistical values, ratios or histograms. The similarity is calculated based on these vectors. Graph based methods rely on graph structured descriptions of the design containing the geometric linkage, e.g. for shapes in CAD systems. The similarity is computed by a comparison of these graphs. Geometry based methods are e.g. based on the idea that two 3D models are similar if they look similar from all viewing angles. Practically, the 3D object is mapped to a number of 2D views rotating around the center of the object. For each of the different 2D views descriptors are calculated which are the basis for measuring the similarity [1].

In summary, guided by the objective of extracting meaningful 3D design components from complex objects, a large overlap to the field of NMF algorithms for object recognition is noticeable. It is argued that especially the parts-based approach and the possibilities of an unsupervised application are very promising. As an alternative to the shape descriptors mentioned above, the framework proposed in the present paper favors to map the design to voxel space as suggested in [2]. Therefore, each design is embedded in a 3D voxel cube and the resolution is defined by the number of voxels. As an advantage, the specification of the voxel number is adjustable to the requested size of the components allowing a perfect scaling of the NMF. Recently, the topographic NMF (TNMF) has been proposed in [3] realizing the integration of neighbourhood connections for data structuring in the NMF. Utilizing the TNMF, a unified framework is sketched for an unsupervised extraction and organization of components from 3D designs. To introduce the idea of the TNMF, the following section presents more details on the algorithm in the field of object recognition.

III. Topographic Non-Negative Matrix Factorization for Learning Objects in the Visual Cortex

The topographic NMF (TNMF) has been introduced by Hosoda *et al.* [3] in the field of object recognition. They build an object representation in the inferior temporal cortex (IT) which relies on activated columnar clusters of neurons with two characteristics. First, objects are represented by salient features or parts of objects and second, closely related features are represented continuously along the tangential direction of individual columnar clusters. They utilize the NMF extended by a matrix defining neighbourhood connections between the NMF basis functions. As a result, the authors achieve topographic maps containing the basis functions while keeping the parts-based property of the NMF. Besides presenting the mathematical formulation, Hosoda *et al.* apply the TNMF successfully to a hierarchical model of neural computation by the ventral pathway.

Analogous to the framework presented in section IV, the objective of the TNMF is the following: For N samples of a data set $X \in \mathbb{R}^{N\times I}$, where each data point is given by I voxels in voxel space, the non-negative matrices $W \in \mathbb{R}^{N\times F}$ and $H \in \mathbb{R}^{F\times I}$ have to be found in such a way that the product of W, M and H reconstructs the original data set with minimal square Euclidean distance. $M \in \mathbb{R}^{F\times F}$ is a constant non-negative matrix containing the neighbourhood connections, W denotes the feature matrix and H the coefficient matrix. F is the number of requested components.

Mathematically, this is expressed by

$$X \approx WMH\ , \quad \left\| X - WMH \right\|_2^2 \rightarrow \min \qquad (1)$$

M is a constant non-negative FxF matrix and each element M_{ab} is given by

$$M_{ab} = \exp(-\left\| p_a - p_b \right\|^2 / 2\sigma^2) \qquad (2)$$

The vectors p_a and p_b denote the position of the features on the map. The Gaussian radius σ is a user-defined parameter which has to be specified to the problem at hand accordingly.

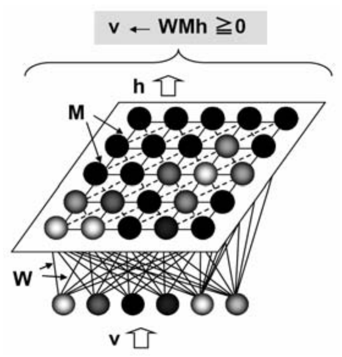

Fig. 3 A diagram of the TNMF, taken from [3]

To apply the TNMF, at first the number of features F, the topographic map and the Gaussian radius σ have to be defined. Based on the chosen topographic map, e.g. a square-lattice grid, the distances between the different position vectors are calculated and inserted into the matrix M according to (2). Fig. 3 illustrates the TNMF schematically. The basis functions are indicated by W, the neighbourhood functions by M, the vector h contains the coefficients and v indicates one data sample. The shading represents the magnitudes of input and coefficient entries.

To illustrate the capability of the TNMF, Hosoda *et al.* [3] learned a parts-based representation of different objects. In Fig. 4 the most relevant result with respect to the application presented in this paper is depicted.

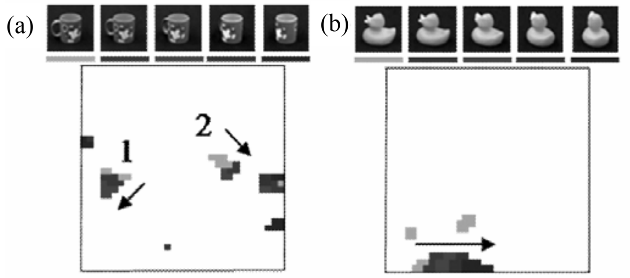

Fig. 4 Object representation calculated by the TNMF, taken from [3]

A data set of 250 grey-level photographs has been presented to the algorithm for training and the activations on the 2D square-lattice map for five images are marked by dots of different grey shadings. As it is visible, the left image containing the first cup view generates clustered activations in the neurons which are continuously shifting when the object view is rotated, as indicated by the arrows. Each peak corresponds to an activated part or component.

With respect to the desired functionality in the area of 3D design decomposition the images show clearly two worthwhile characteristics. First, parts are present which have to be activated simultaneously to reconstruct a given object view, and second, the well-structured spatial distribution of components is visualized on 2D maps. Components which are close in the original image data set are also spatially close on the 2D map. Because of these two characteristics the TNMF is very appealing for the area of 3D design decomposition and component organization and a unified framework is proposed in section IV in more detail.

IV. A Unified Framework for the Extraction and Organization of Design Components from 3D Models

In this section, the transfer of the TNMF from the field of object recognition to the area of 3D design development is described in more detail. The unified framework to generate a parts-based representation and a well-structured visualization of the extracted design components on 2D maps is outlined.

The framework falls into several steps based on two assumptions. First, the 3D design data is stored in a database in a triangulated format, e.g. STL or VRML. Second, all geometries are located in the same coordinate space at the same position to avoid costly computations for repositioning the designs. For real-world environments both assumptions hold true, since first, almost any kind of CAD format is exportable to triangulated mesh formats and second, geometries which are used by teams for design development are usually located at the same offset position in 3D space to allow reusability.

The principal concept of the proposed framework bears high similarities to the one which is described in [2]. The major difference consists in the interpretation of the third matrix which is additionally introduced in the standard NMF algorithm. In [2], this matrix has been defined as a constant smoothing matrix *S* which is used to impose sparseness in *W* and *H*. As an advantage, the feature number adapts during the learning phase to an optimal number and does not need to be defined beforehand correctly. To prepare the algorithm for the topographic NMF, the smoothing matrix *S* is replaced by matrix *M* which contains the neighbourhood connections on a 2D grid. Since *M* is also a constant matrix, the resulting update steps for the NMF algorithm are the same as presented in [2] and the framework is directly transferable.

In a preprocessing step, the given 3D physical bodies are mapped from a triangulated mesh format to the voxel representation. Therefore, all designs are embedded in a virtual 3D cube with a predefined number of voxels. For each voxel it is calculated if or if not the voxel is crossed by the geometry. Each 2D slice of voxels represents a horizontal cross-section through the geometry. To generate the analogy to pixel-based images, all slices are merged into one single 2D image as depicted in Fig. 5.

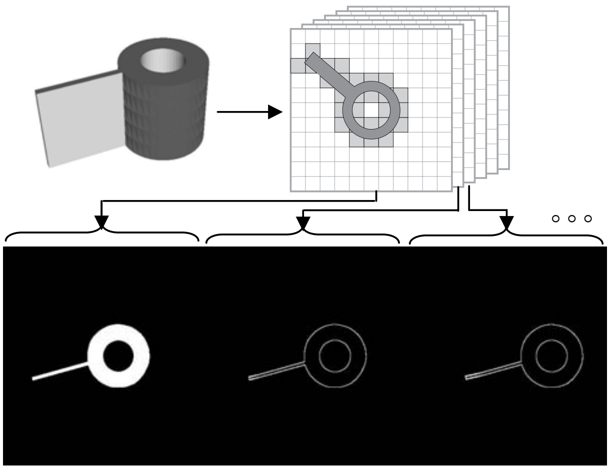

Fig. 5 Transformation of 3D designs into voxel space

The result is one single image per geometry where the black and white colors of each pixel indicate the possible existence of the object in the corresponding voxel. To speed up the computation time of the NMF, for each pixel position the activation in all images is compared. If a pixel is activated (white) or not-activated (black) in all images it is neglected to reduce the input vector dimension. The reduced image vectors containing the remaining pixel activations are afterwards row-wise added to the data matrix *X*.

As stated above, the sparse-orthogonal NMF presented in [2] and the TNMF differ in the interpretation of the matrix *S* but not in the mathematical solution since both matrices are constant. Hence, to calculate the matrix factorization, matrix *S* is replaced by *M* according to (1). A detailed derivation of the update rules for the sparse-orthogonal NMF is presented in [2]. To reflect the changes for the TNMF, the iterative update rules for *H* and *W* are modified by replacing *S* with *M*.

As a consequence, the feature matrix *W* and the coefficient matrix *H* are finally calculated by

$$W_{ij} \leftarrow W_{ij} \sqrt{\frac{(XH^T M)_{ij}}{(W(W^T XH^T M))_{ij}}} \tag{3}$$

and

$$H_{ij} \leftarrow H_{ij} \frac{((WM)^T X)_{ij}}{((WM)^T (WM)H)_{ij}} \tag{4}$$

Before executing the TNMF it is necessary to instantiate the matrix *M* containing the neighbourhood connections. As stated in section III the features will be finally organized on a 2D map where each location of a feature is defined by one of the *F* position vectors. The topographic map can be generated arbitrarily but with respect to the illustrative example of section V a 5x5 square-lattice grid for 25 components is chosen. Fig. 6 depicts the distribution of the features. For a Gaussian radius of 0.8 the matrix entry $M_{1,10}$ is exemplarily calculated. The image depicted in Fig. 8(b) illustrates the matrix containing map topology and neighbourhood connections graphically.

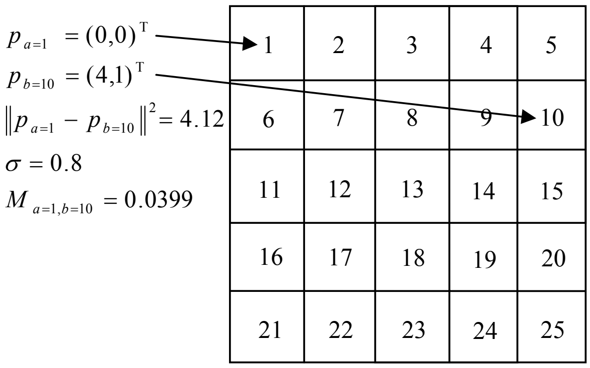

Fig. 6 The 5x5 matrix *M* based on the location of the 25 features

Based on the matrices *X* and *M* the TNMF is executed. The detailed process for calculating *W* and *H* is described in Algorithm 1. Algorithm 1 is similar to the one proposed in [2]. While the iterative updates for *W* and *H* are the same, both algorithms differ in the initialization as well as in the outer loop which is responsible for the dilation of the designs. In the experiments described in section V in detail, the feature and coefficient matrix are initialized with random values.

Additionally, the dilation step is skipped because the design database chosen for the experiments presented in section V does not focus on detecting similarities for geometries with small variations but on conceptionally different designs. Hence, the TNMF algorithm for decomposing 3D designs is as follows. The symbol ⊙ denotes element-wise multiplication; the symbol ⊘ denotes element-wise division.

Algorithm 1: TNMF for 3D Design Decomposition.

$nz \leftarrow 1 \times 10^{-20}$
load image-set X
initialize feature matrix W with random values
initialize coefficient matrix H with random values
initialize matrix M according to p and σ
repeat
$W_s \leftarrow (WM)^T$
$H \leftarrow H \odot (W_s X) \oslash ((W_s W_s^T)H + nz)$
$H_s \leftarrow H^T M$
$W \leftarrow W \odot ((XH_s) \oslash (W(W^T(XH_s)) + nz))^{0.5}$
$L_2 \leftarrow (I \odot (W^T W))^{0.5}$
$H \leftarrow L_2 H$
$W \leftarrow W L_2^{-1}$
until *convergence*

By executing Algorithm 1, matrix W containing the F extracted components and matrix H containing the N coefficients are calculated. The product of W, M and H reconstructs the given data set X with minimal error. In the final step, the components stored in W are arranged on the square-lattice grid. According to their position in the feature matrix, the parts are inserted into the 2D map. For the illustrative scenario described in section V, 25 feature vectors, i.e. 25 salient parts, are calculated by Algorithm 1 and arranged on the 2D map depicted in Fig. 6. The following section provides more detail on the setup and the performance of the proposed method on a data set of 3D turbine-like geometries.

V. Decomposition of 3D Turbine-like Geometries and Visualization of extracted Parts on 2D Maps

To evaluate the applicability of the framework proposed in section IV, the capability of the TNMF to extract a parts-based representation and organize the components on 2D design maps is studied. At first, an adequate design database has been generated which contains 20 turbine-like structures. Each design consists of a central cylinder and 2 adjacent vertical blades. There exist 20 possible positions for a blade, i.e. the blade positions differ by 18° as shown in Fig. 7(a).

The initial design starts at an angle of 0°. Each follow-up design is a by 18° rotated version of the former one, thus the first blade of the current design is at the position of the second blade of the previous design. Three exemplary consecutive designs are depicted in Fig. 7(b)-(d). By the rotation and blade overlaps the topographic neighbourhood of the designs is defined which has to be visible on the 2D map. Adjacent blades should finally organize spatially close on the design map.

As described in section IV, in the next step each turbine is transferred into voxel space and the cross-sections are merged into one image. Activated pixels are white-colored, not-activated ones are black-colored. To speed up the algorithm the data size is drastically reduced by removing all pixels which are either activated in all images or not-activated in all images. Hence, e.g. the cylinder which is existent in all geometries and images is neglected for the TNMF algorithm. All remaining pixels of one image are added to a vector and all vectors are row-wise added to the matrix X containing the input data set.

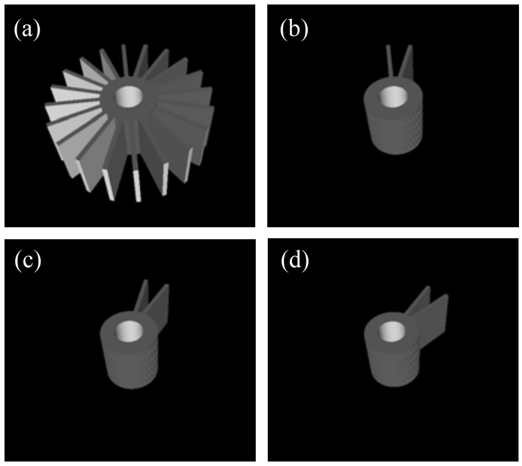

Fig. 7. (a) Turbine prototype showing all possible blade positions; (b)-(d) Consecutive turbine designs, each rotated by +18°

To illustrate the content of X the pixel representation is transferred to a symbolic representation. Analyzing the remaining activated and not-activated pixels for one design, it becomes obvious that there exist 20 different pixel clusters which may be activated or not-activated according to the 20 possible blade positions. Therefore, the activated pixel clusters can be symbolized by a single activated pixel in a 20x20 image. The result is illustrated in Fig. 8(a). Each of the 20 rows of the image depicts one design and each of the 20 columns one possible blade position. Hence, each row contains two white adjacent pixels representing the activated pixel clusters of two blades. Since the follow-up design is rotated by 18°, the two activated pixels move one position to the right generating the stair-like character of the image.

Before executing Algorithm 1, the matrix M defining the neighbourhood connections has to be calculated. For the present scenario 25 features are chosen which are organized on a 5x5 square-lattice grid as shown in Fig. 6. The Gaussian radius σ is defined by 0.8. Based on these assumptions the matrix M is computed and visualized in Fig. 8(b). The white diagonal matrix elements are 1. The brighter a pixel, the closer is the distance of the position vectors on the map and vice versa. This explains why pixel 6 in row 1 has a brighter color than pixel 5 because position vector 6 is spatially closer to position vector 1 then position vector 5.

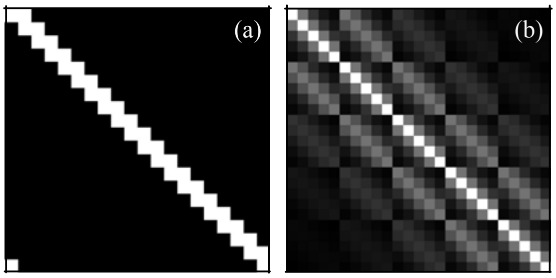

Fig. 8. (a) Symbolic visualization of the turbine data set (b) Matrix M containing neighbourhood connections for 25 features organized on a 5x5 grid

Based on the given data X and matrix M, 25 features are calculated using Algorithm 1. Finally, the extracted parts are stored in matrix W. According to the position vectors p, each component is added to the 2D map. The sequence is visible in Fig. 6, thus the parts are organized on a 5x5 square grid starting with feature 1 in the upper left tile and finishing with feature 25 in the lower right tile. A symbolic visualization is depicted in Fig. 9.

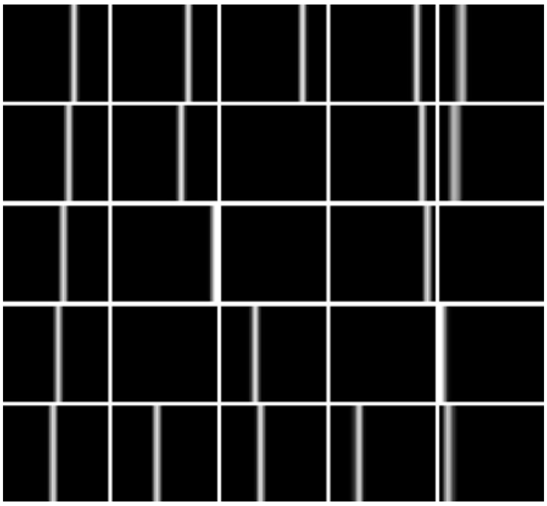

Fig. 9. Symbolic components of the turbine data set extracted by the TNMF. The representation is parts-based and neighbourhood clusters are perfectly visible

Fig. 9 illustrates two important characteristics of the proposed method very well. First, the algorithm extracts exactly 20 components which are required to rebuild the database, so each possible blade position is detected. Overall, 20 features are activated where each one contains exactly one vertical blade. 5 features are not used. Second, the neighbourhood relations between the features are clearly visible. Adjacent components are organized with very small spatial distances on the map, i.e. adjacent single blades are spatially very close on the map.

For the final visualization of the design parts, the symbolic representation is replaced by the geometry components. Since the cylinder has been present at the same position in all designs, it is added to all components for reasons of a better visualization. Hence, e.g. based on the components given by feature 20 and 25 as numbered in Fig. 6, it is possible to reconstruct the turbine types depicted in Fig. 7(b). The two blades are decomposed into two single parts and organized spatially with a very close distance on the map.

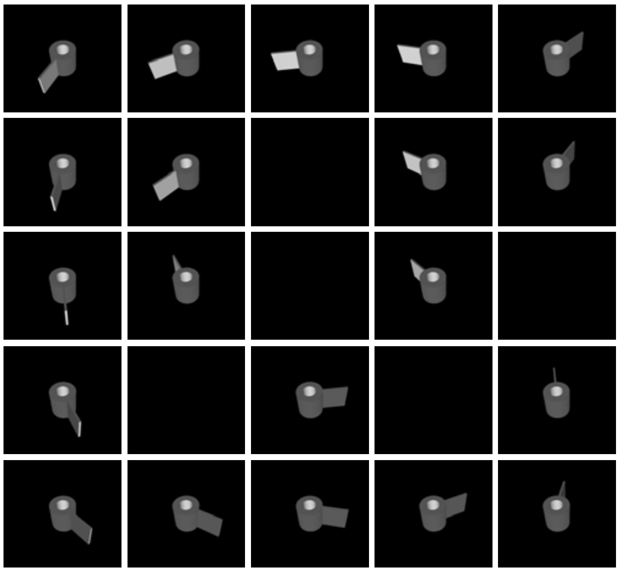

Fig. 10. Extracted 3D components from design database organized on a 2D design map

To analyze statistically the occurrence and spatial distribution of the features on the design map, a second scenario has been performed using 100 experiments based on the given database with different random numbers for initializing the feature and coefficient matrix. For all experiments the coefficient and feature matrix is calculated, matrix M is kept constant. To analyze and visualize the neighbourhood connections of one specific feature, this feature is chosen and placed at the center of the map as reference component. Next, all 100 resulting design maps are scanned for the chosen reference feature and rearranged in such a way that the reference feature is centered. Finally, the sum over all rearranged maps is calculated and visualized on a 5x5 map. As a consequence, each of the 25 tiles reflects the number of occurrences of specific features at a specific spatial position on the design map. An exemplary symbolic visualization for feature 16 as reference which corresponds geometrically to blade position 10 is depicted in Fig. 11. The brighter the color the more occurrences of a feature are counted at the corresponding spatial position on the map. Consequentially, the activation of feature 16 is 1 for the center tile. It can be seen that in the adjacent north, east, south and west tiles of the center tile the activations for feature 11 and 21 which correspond to blade positions 9 and 11 are significantly high, illustrated as grey bars. If one would add these four tiles the activation of each of the features is very close to 1. The result points to a good robustness of the proposed framework because in each of the 100 maps adjacent blades are arranged typically also spatially close on the design maps.

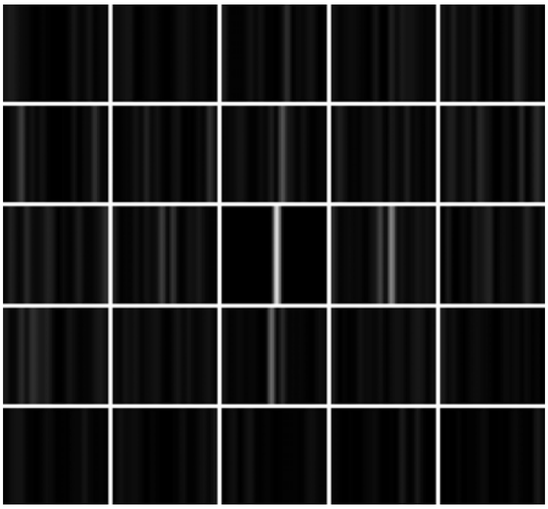

Fig. 11 Spatial distribution of parts with feature 16 as reference

VI. Conclusion

In the present paper, a unified framework is proposed for an unsupervised extraction of components from a data set of 3D designs which simultaneously allows an arrangement of the components on 2D design maps. The visualization of the components bears the advantages of having a well-structured overview on the content of the database and at the same time providing a simple distance measure between design parts. In contrast to a self-organizing map, the presented framework focuses on the parts-based aspect of the TNMF as it is targeted to prepare a method which allows small local changes instead of holistic ones. Technically, the insertion of the matrix M into the standard NMF algorithm invokes the self-organization process of the components in the feature matrix which finally allows the arrangement of the components on 2D maps.

Based on the generated design maps the search for similar design parts to a given one is easily achievable allowing the designer to locate his current state in a design development process. Furthermore, computer driven algorithms like optimization methods can rely on these distant measures to automatically exchange parts if needed.

As further extensions, from a practical perspective, the aspect of varying the voxel resolution is very appealing. It would be very attractive for an engineer to zoom smoothly through different scales of large geometries being able to have design decompositions on each scale available. If thinking of cars, decompositions on a top level would contain doors, wheels, engine etc. while on a low level single components of an engine, like valves or pistons would be visible. Additionally, an extension of the proposed framework to higher-dimensional maps is worth for further research since it would allow more possibilities to visualize the resulting components focusing on different design aspects.

Acknowledgment

The author would like to thank Heiko Wersing from the Honda Research Institute Europe GmbH for his valuable support during the project.

References

[1] J. W. H. Tangelder, and R. C. Veltkamp, “A survey of content based 3D shape retrieval methods,” *Multimedia Tools and Applications*, vol. 39, Springer Netherlands, pp. 441-471, Sep. 2008.

[2] Z. Bozakov, L. Graening, S. Hasler, H. Wersing, and S. Menzel, “Unsupervised extraction of design components for a 3D parts-based representation,” *IJCNN*, pp. 2009-2016, IEEE, Hongkong, 2008.

[3] K. Hosoda, M. Watanabe, H. Wersing, E. Koerner, H. Tsujino, H. Tamura, and I. Fujita, “A model for learning topographically organized parts-based representations of objects in visual cortex: topographic non-negative matrix factorization, “ *Neural Computation*, 2009, in press.

[4] D. D. Lee and S. H. Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, vol. 401, no. 6755, pp. 788-791, 1999.

[5] I.T. Jolliffe, “Principle Component Analysis,” *Series: Springer Series in Statistics*, 2nd ed., Springer, New York, 2002.

[6] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297, 1967.

[7] T. Kohonen, “Self-Organizing Maps,” Springer-Verlag, Berlin 1995, ISBN 3-540-58600-8.

[8] Z. Bozakov, “Unsupervised Component Extraction for Design Optimisation using Feature Analysis,” Master Thesis, Control Theory and Robotics Lab, Institute for Automatic Control, TU Darmstadt, 2007.

[9] B. M. Chazelle, “Convex decomposition of polyhedra,” in *STOC ’81: Proceedings of the thirteenth annual ACM symposium on Theory of computing*. New York, NY, USA: ACM Press, pp. 70-79, 1981.

[10] J.-M. Lien and N. M. Amato, “Approximate convex decomposition of polygons,” in *SCG ’04: Proceedings of the 20th annual symposium on Computational geometry*, NY, USA: ACM Press, pp. 17-26, 2004.

[11] J.-M. Lien and N. M. Amato, “Approximate convex decomposition,” in *SCG ’04: Proceedings of the 20th annual symposium on Computational geometry*. NY, USA: ACM Press, pp. 457-458, 2004.

[12] H. Sundar, D. Silver, N. Gagvani, and S. Dickenson. Skeleton based shape matching and retrieval. In SMI 2003, pages 130–139, 2003.

[13] S. Katz and A. Tal, “Hierarchical mesh decomposition using fuzzy clustering and cuts,” in *SIGGRAPH ’03: ACM SIGGRAPH 2003 Papers*. NY, USA: ACM Press, pp. 954-961, 2003.

[14] T. Funkhouser, M. Kazhdan, P. Min and P. Shilane, “Shape-based Retrieval and Analysis of 3D Models,” in *Communications of the ACM*, Vol. 48, No.6, pp. 58-64, June 2005.