

3-D FILTERS AND THEIR USE FOR COMPUTER MODELLING

MARTIŠEK Dalibor (ČR)

The paper presents an interesting generalisation of 2-D filters known in image processing. The filters presented in this paper can be used to 3-D objects processing with analogous effects to 2-D ones. Several attributes can be assigned to each voxel in the processing 3-D object (space co-ordinates, colour, transparency, potential e.t.c.). Each of these attributes can be filtered. These filters can be used to surface smoothing, erosive filters delete small 3-D objects. The high-pass 3-D filters detect object surfaces, draw power field isosurfaces e.t.c. Topographical and biological applications are demonstrated.

INTRODUCTION

Data in computer graphics are stored as coordinates of points that, as in the traditional Euclid geometry, are modelled as dimensionless objects. The displaying device is, however, a physical object and, as such, cannot display or read "dimensionless points". For this reason, instead of point, the word "pixel" is used. However, in mathematical modelling, pixels in the "logical" sense have to be considered (that is, the output device is thought of as a set of isolated Euclidean points) as opposed to the "physical" sense (where the output device is taken to represent a "set of elementary small surfaces").

In the nowadays literature, these differences are mostly ignored. Nevertheless, even in cases where the difference is made between these notions, their definitions are very vague and sometimes even false. Exakt mathematical definitions of these terms enable great generalisation and interesting applications. I will demonstrate this fact on so called 3-D filters. This theory is unprecedented. This article introduces new terms, theorems are presented for illustration only and they are cited without proofs. More detailed information is accessible in [1].

1. DIGITAL SPACE

1.1. Definition: Let ${}_k J = \langle a_k; b_k \rangle$, $k = 1, 2, \dots, n$ be intervals, ${}_k D = \{ {}_k x_0, {}_k x_1, \dots, {}_k x_{i_k}, \dots, {}_k x_{m_k} \}$

an equidistant divisions of ${}_k J$. The set $\mathbf{J}^{(n)} = \bigtimes_{k=1}^n {}_k J$ is called a digital space support,

$\mathbf{D}^{(n)} = \bigtimes_{k=1}^n {}_k D$ equidistant $n-D$ division. An ordered pair $\mathbf{D}^{(n)} = (\mathbf{J}^{(n)}; \mathbf{D}^{(n)})$ is called n dimensional digital space.

1.2. Definition: The subset $\mathbf{F}^{(n)} \subset \mathbf{J}^{(n)}$ of $\mathbf{J}^{(n)}$ is called a physical $n-D$ domain only and if only, when

$$\mathbf{F}^{(n)} = \langle x_{i_1}; x_{i_1+1} \rangle \times \langle x_{i_2}; x_{i_2+1} \rangle \times \dots \times \langle x_{i_k}; x_{i_k+1} \rangle \times \dots \times \langle x_{i_n}; x_{i_n+1} \rangle = \bigotimes_{k=1}^n \langle x_{i_k}; x_{i_k+1} \rangle.$$

We denote $\mathbf{F}^{(n)} = \bigotimes_{k=1}^n \langle x_{i_k}; x_{i_k+1} \rangle = \mathbf{F}_{[i_1, i_2, \dots, i_k, \dots, i_n]}^{(n)} = \mathbf{F}_{\mathbf{i}}^{(n)}$. The number ${}_k v_{\mathbf{i}} = x_{i_k+1} - x_{i_k}$; $i_k \in \mathbf{i}$ is called k -th dimension of $\mathbf{F}_{\mathbf{i}}^{(n)}$.

1.3. Theorem: Let $\mathbf{D}^{(n)} = (\mathbf{J}^{(n)}; \mathbf{D}^{(n)})$ be the digital space, $A, B \in \mathbf{J}^{(n)}$ arbitrary points of its support. The relation $\rho \subset \mathbf{J}^{(n)} \times \mathbf{J}^{(n)}$ defined on $\mathbf{J}^{(n)}$ by

$$\rho(A, B) \Leftrightarrow \exists \mathbf{F}_{\mathbf{i}}^{(n)} \in \mathbf{F}^{(n)} : A \in \mathbf{F}_{\mathbf{i}}^{(n)} \wedge B \in \mathbf{F}_{\mathbf{i}}^{(n)}$$

is an equivalence on $\mathbf{J}^{(n)}$.

1.4. Definition: The factor set $\mathbf{F}^{(n)} = \mathbf{J}^{(n)} / \rho$, where ρ is the equivalence from the previous theorem, is called the physical space of $\mathbf{J}^{(n)}$ or $\mathbf{D}^{(n)} = (\mathbf{J}^{(n)}; \mathbf{D}^{(n)})$ respectively.

In many applications we use the so-called logical space and logical domain. Sometimes it is important to which (Euclidean) point of the physical domain we are referring – its centre, vertex, etc. Thus by a logical domain ${}_c L_{\mathbf{i}}$ we mean a representative of the physical domain $\mathbf{F}_{\mathbf{i}}^{(n)}$, the logical space ${}_c \mathbf{L}^{(n)}$ being the set of all the logical domains ${}_c L_{\mathbf{i}}$:

1.5. Definition: Let $\mathbf{F}^{(n)}$ be the physical space of $\mathbf{D}^{(n)}$, ${}_k v$ dimensions of its physical domains, $\mathbf{C} \in \mathbf{J}^{(n)} : \mathbf{C} = [c_1, c_2, \dots, c_k, \dots, c_n] : c_k \in \langle 0; {}_k v \rangle$. The set

$${}_c \mathbf{L}^{(n)} = \bigotimes_{k=1}^n \left\{ {}_k r_{i_k} \in \mathbb{R} \mid \forall k \in \{1, 2, \dots, n\} : {}_k r_{i_k} \in \langle x_{i_k}; x_{i_k+1} \rangle \wedge {}_k r_{i_k} - x_{i_k} = c_k \right\}$$

is called a logical space of $\mathbf{D}^{(n)} = (\mathbf{J}^{(n)}; \mathbf{D}^{(n)})$, its elements ${}_c L_{\mathbf{i}}^{(n)}$; $\mathbf{i} = [i_1, i_2, \dots, i_k, \dots, i_n]$ are logical domains.

The process of assigning logical pixels to physical ones (the choice of representatives) is called mapping.

1.6. Theorem: Let $\mathbf{F}^{(n)} ({}_c \mathbf{L}^{(n)})$ be the physical (logical) space of $\mathbf{D}^{(n)}$. Mapping ${}_c \phi : \mathbf{F}^{(n)} \rightarrow {}_c \mathbf{L}^{(n)}$ such that ${}_c \phi(\mathbf{F}_{\mathbf{i}}^{(n)}) = {}_c L_{\mathbf{i}}^{(n)} \Leftrightarrow {}_c L_{\mathbf{i}}^{(n)} \in \mathbf{F}_{\mathbf{i}}^{(n)}$ is a bijection.

1.7. Definition: The mapping ${}_c \phi : \mathbf{F}^{(n)} \rightarrow {}_c \mathbf{L}^{(n)}$ from previous theorem we call the physical space mapping.

Notice: To model a $2-D$ digital space we can use virtually any output device such as a monitor, printer, etc. The term domain is a generalization of terms pixel ($2-D$ domain) and voxel ($3-D$ domain) which are used in nowadays literature. They are also important for constructing surfaces by interpolating the graph of a function in two or three variables where the function values are known at equidistant points.

2. VALUATION AND FILTER

Classical Euclid synthetic geometry models its objects by study of elements of subsets of space E_n . Defining subset of E_n (Euclid objekt \mathcal{P}) we set a formula which determines wheter $X \in \mathcal{P}$ or $X \notin \mathcal{P}$ fo each $X \in E_n$. We can formaly denote this formula as a mapping $\rho : E_n \rightarrow \{0,1\}$, whereas $\rho(X)=1 \Leftrightarrow X \in \mathcal{P}$. Analytic geometry models its object by mapping $\varphi : E_n \rightarrow \mathbb{R}^n$, that assigns the co-ordinates to points

Performing similar construction in the physical space, it is posible this way to determine subsets of this space - physical objects. Define $\mathcal{P}_F^{(n)} \subseteq F^{(n)}$ and $\rho_F : F^{(n)} \rightarrow \{0,1\}$ such that $\forall \mathbf{F}_i^{(n)} \in F^{(n)} : \rho_F(\mathbf{F}_i^{(n)})=1 \Leftrightarrow \mathbf{F}_i^{(n)} \in \mathcal{P}_F^{(n)}$. It is evident, the set $\mathcal{P}_F^{(n)} \subseteq F^{(n)}$ determine the mapping ρ_F and conversely. The situation is identical for the logical space too.

It is obvious, that each physical space is a metric space. Function $C_F^{(n)}(\mathbf{F}_i^{(n)}; \mathbf{F}_j^{(n)}) = \max\{c_k | i_k - j_k\}_{k=1}^n$ is a metric on $F^{(n)}$ for example.

2.1. Definition: Let $F^{(n)}$ be the physical space, A arbitrary set, which contains minimal two elements. Mapping $\beta : F^{(n)} \rightarrow A$ we call a valuation of $F^{(n)}$.

2.2. Definition: Let $({}^*\varepsilon F^{(n)}, \varphi, \mu)$ be the physical space with mapping φ and metric μ , $(F^{(n)}, \varphi, \mu)$ its subspace such that ε -surrounding of each physical domain $\mathbf{F} \in F^{(n)}$ contains the same number of elements in ${}^*\varepsilon F^{(n)}$. The space $\varepsilon F^{(n)} = \bigcap {}^*\varepsilon F^{(n)}$ is called ε -hull of $F^{(n)}$ (considering metrics μ).

2.3. Definition: Let $\beta : F^{(n)} \rightarrow A$ be a numeric valuation of $F^{(n)}$, $\varepsilon\beta : \varepsilon F^{(n)} \rightarrow A$ valuation of its ε -hull such that for every $\mathbf{F} \in F^{(n)}$ is $\beta(\mathbf{F}) = \varepsilon\beta(\mathbf{F})$. Valuation $\varepsilon\beta : \varepsilon F^{(n)} \rightarrow A$ is called ε -hull of valuation $\beta : F^{(n)} \rightarrow A$.

2.4. Definition: Let $\beta : F^{(n)} \rightarrow A$ is a numeric valuation of physical space $F^{(n)}$, $\varepsilon\beta : \varepsilon F^{(n)} \rightarrow A$ its ε -hull such that ε -surrounding $O_\varepsilon(\mathbf{F})$ of each physical domain $\mathbf{F} \in F^{(n)}$ contains r elements. Further let $f : \mathbb{R}^r \rightarrow \mathbb{R}$ a function of r real variables and $O_\varepsilon(\mathbf{F}_i) = \{\mathbf{F}_1; \mathbf{F}_2; \dots; \mathbf{F}_r\} \subset F^{(n)}$ ordered r elements surrounding of $\mathbf{F}_i \in F^{(n)}$. Numeric valuation $\frac{\Phi\beta}{\Phi f} : D^{(n)} \rightarrow A$ such that for every $\mathbf{F}_i \in F^{(n)}$ is $\frac{\Phi\beta}{\Phi f}(\mathbf{F}_i) = f(\beta(\mathbf{F}_1); \beta(\mathbf{F}_2); \dots; \beta(\mathbf{F}_r))$ is called filtered valuation, function f is called $n-D$ filter.

Existing function $g : \mathbb{R}^r \rightarrow \mathbb{R}$ and mapping $C(\mathbf{t}) : \bigotimes_{i=1}^n \{-\varepsilon_i; \dots; 0; \dots; \varepsilon_i\} \rightarrow \mathbb{R}$ such that

$$\frac{\Phi\beta}{\Phi f}(\mathbf{F}_i) = f(\beta(\mathbf{F}_1); \beta(\mathbf{F}_2); \dots; \beta(\mathbf{F}_r)) = \sum_{\mathbf{t} \in O_\varepsilon(\mathbf{F}_i)} C(\mathbf{t}) \beta(\mathbf{F}_i - \mathbf{t})$$

Function f is called a linear filter.

3. SURFACE SMOOTHING AND TERRAIN MODELLING

The image is defined as a function $I: W \times H \rightarrow V$, where $W = \langle 0; w \rangle \subset \mathbb{R}; w \in \mathbb{N}; H = \langle 0; h \rangle \subset \mathbb{R}; h \in \mathbb{N}; V = \langle v_1; v_2 \rangle \subset \mathbb{R}$ in classic image processing. Considering previous theory image is m -arity valuation $\beta: F^{(2)} \rightarrow \{0, 1, \dots, m-1\}$. Value set $V = \langle v_1; v_2 \rangle \subset \mathbb{R}$ of this function is interpreted as a colour set obviously. The value of a pixel, however, can also be thought of as its height, then the graph of the function $z = f(x, y)$ - is modelled as a 3-D surface. Constructing this surface by using computer, it is necessary the definition set $W \times H$ take as physical space and function as real valuation $\beta: F^{(2)} \rightarrow \mathbb{R}$. Filtering this valuation we obtain the valuation $\frac{\Phi\beta}{\Phi f}: F^{(2)} \rightarrow \mathbb{R}$. The filters in this interpretation change the height of points and make possible to model 3-D objects.

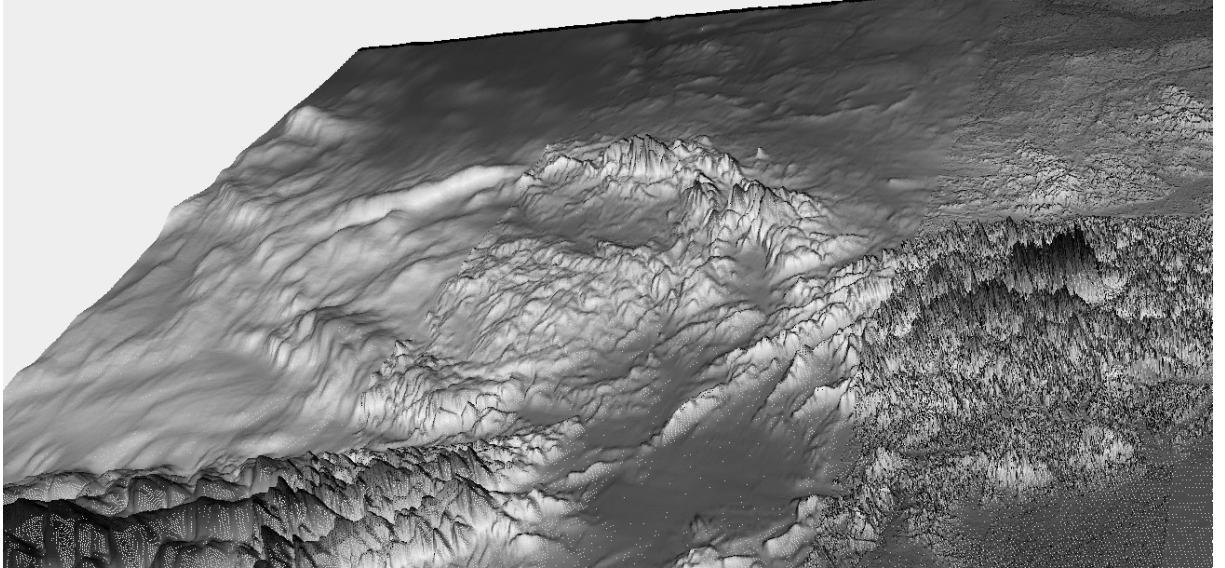


Fig. 1: Smoothing of the Czech republic topographic terrain

The Fig.1. shows the topographic terrain of the Czech Republic. In the right hand part there are the input data. These data was filtered by linear filter on def. 2.4., where

$$C(t) = \left((2\varepsilon + 1)^{-n} \right)_{t=[-\varepsilon_1; -\varepsilon_2; -\varepsilon_3]}^{[\varepsilon_1; \varepsilon_2; \varepsilon_3]} \quad (3.1.)$$

($\varepsilon = 3$, $\varepsilon = 6$ is used in the middle or on the left respectively).

4. OBJECT FILTER AND BORDER SURFACE DETECTION

Border surface detection is very important problem in computer graphics and data visualization. Today common used methods are based on vector principle and are very complicated. They calculate intersections of "viewing rays" with processed objects. They solve particular events only and are not very consistent. Very problematical terminology is used ("manifold solid", "ring edges" "crease angle", "winged-edge") inclusive of quite incorrect terms ("solid normal", "vector normal to solid", e.t.c). Filter apparatus is very efficient for this problem.

4.1. Definition: Let $F^{(n)} ({}_C L^{(n)})$ be a physical (logical) space. Its arbitrary subset ${}_F \mathcal{P}^{(n)} ({}_L \mathcal{P}^{(n)})$ we call $n-D$ physical (logical) object.

4.2. Definition: Let $\beta: F^{(n)} \rightarrow A$ be a valuation of physical space $F^{(n)}$, $\mathcal{P} \subseteq F^{(n)}$ an arbitrary $n-D$ physical object. The function $f_{\mathcal{P}}: \mathbb{R}^r \rightarrow \mathbb{R}$ is called an object $n-D$ filter of object $\mathcal{P} \subseteq F^{(n)}$ only and if only, when exists a mapping $C(\mathbf{t}): \bigotimes_{i=1}^n \{-\varepsilon_i; \dots; 0; \dots; \varepsilon_i\} \rightarrow \mathbb{R}$ such that

$$f_{\mathcal{P}}(\beta(\mathbf{F}_1); \beta(\mathbf{F}_2); \dots; \beta(\mathbf{F}_r)) = \sum_{\mathbf{t} \in O_{\varepsilon}(\mathbf{F}_i) \cap \mathcal{P}} C(\mathbf{t}) g(\mathbf{F}_i - \mathbf{t}) \quad (4.1.)$$

(for logical object by analogy).

Object $n-D$ filter is different from space $n-D$ filter. We sum not through all surrounding $O_{\varepsilon}(\mathbf{F}_i)$, but through intersect this surrounding with object - $O_{\varepsilon}(\mathbf{F}_i) \cap \mathcal{P}$.

Binary valuation ${}_F \beta: F^{(n)} \rightarrow \{0; 1\}$ of physical space determine the physical object ${}_F \mathcal{P} \subseteq F^{(n)}$. This fact is a principle the simplest $3-D$ object reconstruction method - the voxel reconstruction method. Physical voxel is modelled as a box, which is only and if only, when ${}_F \beta(\mathbf{F}) = 1 \Leftrightarrow \mathbf{F} \in {}_F \mathcal{P}$.

In the left hand part of Fig. 2. we can see the voxel reconstruction of a protozoon Paramethyum organella part. This reconstruction is quite raw in itself but it is useable as a ground for more accurately technique

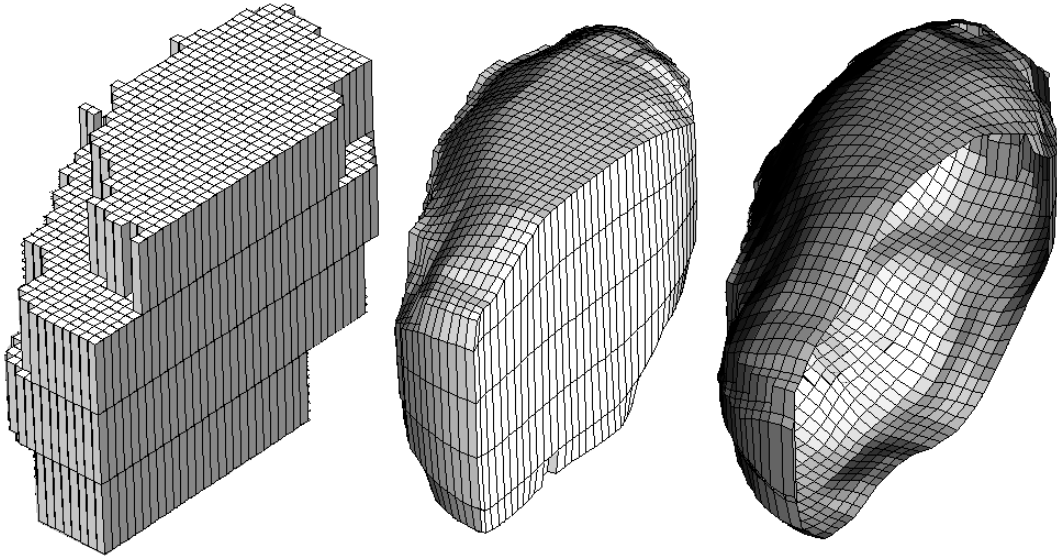


Fig. 2. Object moulding by using the $3-D$ object filters

Let $\mathcal{P} \subseteq F^{(n)}$ be arbitrary $n-D$ object, $\beta_X; \beta_Y; \beta_Z: F^{(3)} \rightarrow \mathbb{R}$ are valuation of this space such that for all $\mathbf{F}_{ijk} \in F^{(3)}$ are $\beta_X(\mathbf{F}_{ijk}) = i$, $\beta_Y(\mathbf{F}_{ijk}) = j$, $\beta_Z(\mathbf{F}_{ijk}) = k$. Using the object linear $3-D$ filter according to def. 4.2. where $C(\mathbf{t})$ is given by (3.1.) we smooth a object surface. We can see this shoothing in the middle part of Fig 2.

Using the same technique linear filter, for which is

$$\left[C(\mathbf{t}) = -\frac{1}{(2\varepsilon+1)^n} \Leftrightarrow \mathbf{t} \neq [0;0;\dots;0] \right] \wedge \left[C(\mathbf{t}) = \frac{(2\varepsilon+1)^n - 1}{(2\varepsilon+1)^n} \Leftrightarrow \mathbf{t} = [0;0;\dots;0] \right] \quad (4.2.)$$

we are able to detect and construct the surface of 3-D object. We can see this detection in the right hand part of Fig 2.

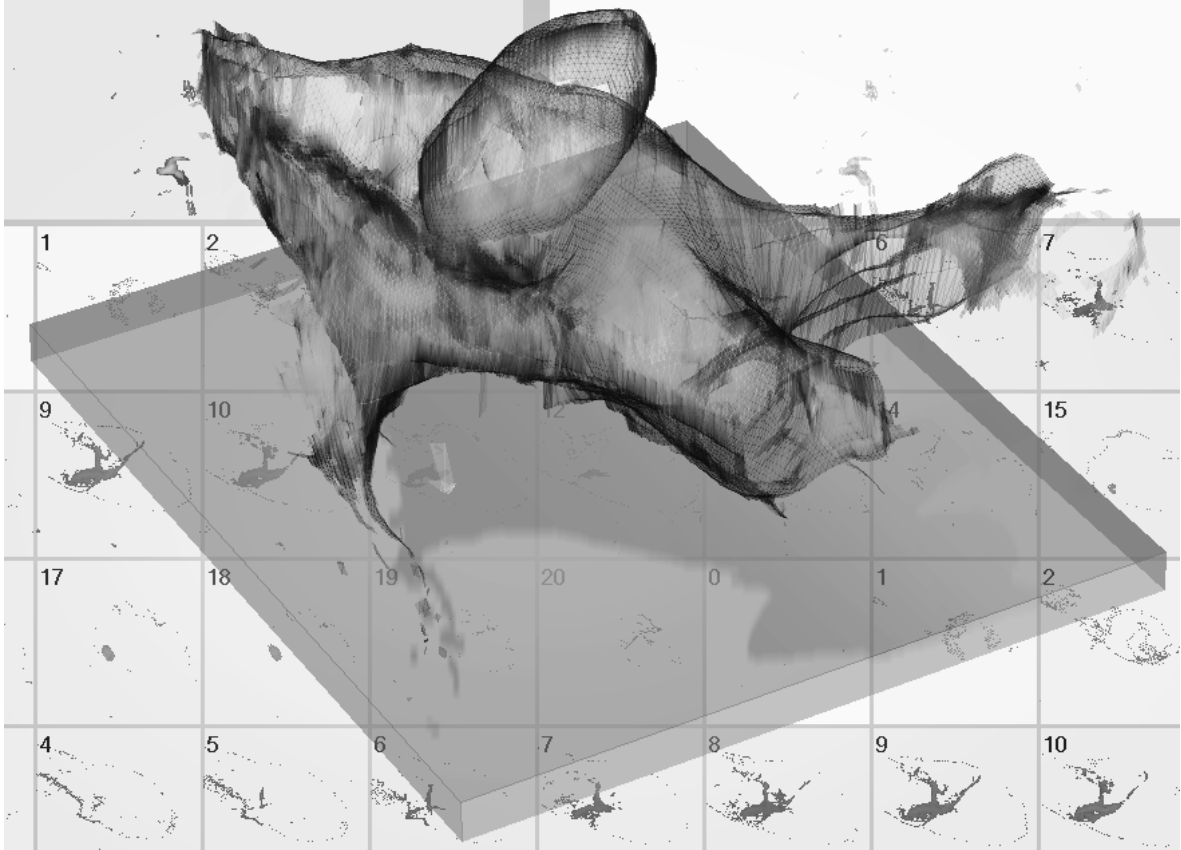


Fig 3: Object reconstruction by using of 3-D filters

The using of 3-D filters to 3-D data reconstruction is more effectively than nowadays used methods (marching cubes, marching triangles, dividing cubes etc.) and presented comparable results.

I have used data scanned by Prof. MUDr. Roman Janisch, DrSc. from LF MU Brno. I thank him for his cooperation.

This project is supported by research design CEZ: J22/98: 261100009 "Non-traditional methods for investigating complex and vague systems".

REFERENCES

- 1) Martišek, D.: Mathematical principles of graphics systems, in: <http://martisek.webzdarma.cz>
- 2) Lorensen W. E., Cline H. E.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm, Computer Graphics, 21(4): 163-169, 1987
- 3) Hilton A., Illingworth J.: Marching Triangles: Delaunay Implicit Surface Triangulation, University of Surrey, 1997