Object Comparison using PDE-based Wave Metric on Cellular Neural Networks

István Szatmári[1]

Abstract – The paper investigates PDE-based dynamic phenomena for comparing objects and introduces a spatio-temporal nonlinear wave metric. This metric is capable of comparing both binary and gray-scale object pairs in a parallel way. Spatio-temporal waves are initialized and controlled to explore the quantitative properties of objects. In addition to spatial data, even "hidden", time related information is also extracted and used for evaluating differences and similarities. The detailed analysis of the proposed metric shows that this wave-based approach can outperform well-known metrics such as Hausdorff and Hamming metrics in selectivity and sensitivity. The approach in question can be efficiently implemented on massively parallel architectures, e.g., on Cellular Neural/Nonlinear Networks (CNN), providing solutions either for real time applications.

Key words: object comparison, metrics, nonlinear waves, reaction diffusion, PDE

1. Introduction

The paradigm of Cellular Neural/Nonlinear Networks (CNN) [1-3] and CNN Universal Machines (CNN-UM) [4] offers a flexible framework for theoretic investigation and modeling of complex nonlinear dynamics but also serves as a high-speed, parallel device for practical applications. In the paper, we will investigate how this framework can be utilized for quantitative object comparison. Choosing a proper metric for comparing objects requires careful planning because an inappropriate distance measure might give completely false result. Methodologies based on constrained diffusion models have been turned out to be a powerful tool in many image processing related tasks. In the field of edge detection, for instance, anisotropic diffusion was proposed in [5] for adaptive and controlled smoothing and edge enhancement. The PDE-based model was used for defining a CNN framework for image segmentation and edge detection presented in [6]. The wave approach (modeled by a reaction-diffusion process) has already been considered for object analysis and comparison, and the autowave principle was proposed for image processing, see, e.g., [7, 8]. In the paper, an earlier methodology for comparing binary objects will be completed first outlined in [25], and will be extended also to gray-scale objects. A generalized nonlinear diffusion model defines the framework in which nonlinear spatio-temporal waves explore objects and determine the distance between them. The novelty in this approach lies in the fact that along spatial features, as a new dimension, time-related information can be extracted by investigating the "hidden" properties of objects. The proposed dynamical approach is of great importance, namely, it defines a framework for quantitative measurements on the basis of PDE techniques. In addition to object comparison, this

---

[1] Analogical and Neural Computing Laboratory of Computer and Automation Research Institute of Hungarian Academy of Sciences, address: Kende u. 13-17, Budapest, Hungary, H-1111
e-mail: szatmari@sztaki.hu

framework makes it possible to analyze also video flows and measure, e.g., similarities among image frames, optical flow, etc.

2. Metrics and motivations

In this section we extend the binary interpretation of Hamming and Hausdorff metrics also to gray-scale objects and show their limitations through simple examples. A function *d(A, B)* of two variables defined on a set (Metric Space) *S* is called a metric function (between A and B) provided

(1)

1. $d_{AB} = d(A,B) \geq 0$ *and* $d_{AB} = 0$ *if and only if A = B. (Positiveness and Identity)*
2. $d_{AB} = d_{BA}$ *(Symmetry)*
3. $d_{AB} + d_{BC} \geq d_{AC}$ *(Triangle Inequality)*

If condition (3) in Eq. 1 is not satisfied, (1) and (2) alone define a distance function.

2.1 Distance metrics on binary objects

Several possible metrics and distance functions for images can be found in [9]. The well-known Hamming distance between two binary images (or objects) is *the measure of symmetrical difference (number of different points)* and can be defined formally as

(2)
$$d_H = \left| \{ (A \cup B) \setminus (A \cap B) \} \right|.$$

This is the result of a pixel-wise XOR operation on binary images. If we consider binary images represented by 0 and 1, white and black pixels, respectively, the Hamming distance is equivalent to the L_1 (Manhattan, or Minkowsky distance with *r=1*), i.e.,

(3)
$$d_H = \sum_{i=1}^{n} |A_i - B_i|,$$

where A_i stands for intensity value at position *i*.
Remark: This type of definition will help us to extend Hamming distance also to gray-scale images. One important property of Hamming distance is that it cannot take the shape information into account because it measures the differences only.

Another frequently used metric, called Hausdorff distance, measures *the maximum distance of a set to the nearest point in the other set*, defined formally as

(4)
$$d_{HS}(A, B) = \max(h(A, B), h(B, A)),$$
$$\textbf{where } h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

Function *h(A,B)* is called the *directed* Hausdorff distance from *A* to *B*. It identifies the point $a \in A$ that is farthest from any point of *B* and measures the distance from *a* to its nearest neighbor in *B* (using the given norm ‖.‖). The *h(A,B)*, in effect, ranks each point of *A* based on its distance to the nearest point of *B*, and then uses the largest ranked point as the distance (the most mismatched point of *A*). Note that, in general, *h(A,B)* and

h(B,A) may result in very different values (the directed distances are not symmetric). The Hausdorff distance $d_{HS}(A,B)$ is the maximum of *h(A,B)*, and *h(B,A)*. Thus, it measures the degree of mismatch between two sets by measuring the distance of the point of *A* that is farthest from any point of *B* and vice versa.

Intuitively, if $d_{HS}(A,B) = d$, then each point of *A* must be within the distance *d* of some point of *B* and vice versa. This metric is computable in polynomial time and the problem of computing the Hausdorff distance between geometric entities has been considered in the area of computational geometry (see, e.g., [10, 11]). Unfortunately, Hausdorff distance is extremely unstable in the presence of noise. The appearance of a noisy spot, however small it can be, will drastically change the distance.

2.2 Distance metrics on gray-scale objects

First, we extend the previous two metric functions to gray-scale objects, as follows. Let us consider a gray-scale image, represented by interval [0, 1], as a surface. Zero intensity will be considered as background, therefore, objects can take intensity values from interval (0, 1]. On binary objects, the Hamming distance gives back the number of different pixels, i.e. it measures the area of different parts. We define *Volumetric Hamming distance* for arbitrary objects (either binary or gray-scale) as

$$(5) \qquad d_{HV} = \sum_{i=1}^{n} |A_i - B_i| ,$$

where A_i stands for intensity value at position *i*. This distance function measures the *volume difference* instead of area but equals to Hamming distance in case of binary images. The interpretation of Hausdorff distance on gray-scale objects is straightforward, but we will modify the two sets to be compared. Regarding noise suppression, the definition of the Hausdorff distance on binary objects was modified in [7]. Distances are only allowed to be measured among points which form closed contiguous region with $A \cap B$. In this case, a separated noisy spot will be excluded from the distance computation. Similar consideration led us to define new initial sets for the computation of both Volumetric Hamming and Hausdorff distance. Instead of computing distance between the two original images, we define the two initial sets as follows.

$$(6) \qquad \begin{aligned} I_{\max}^{A,B}(i) &= \max(A_i, B_i), \\ I_{\min}^{A,B}(i) &= \min(A_i, B_i) \end{aligned}$$
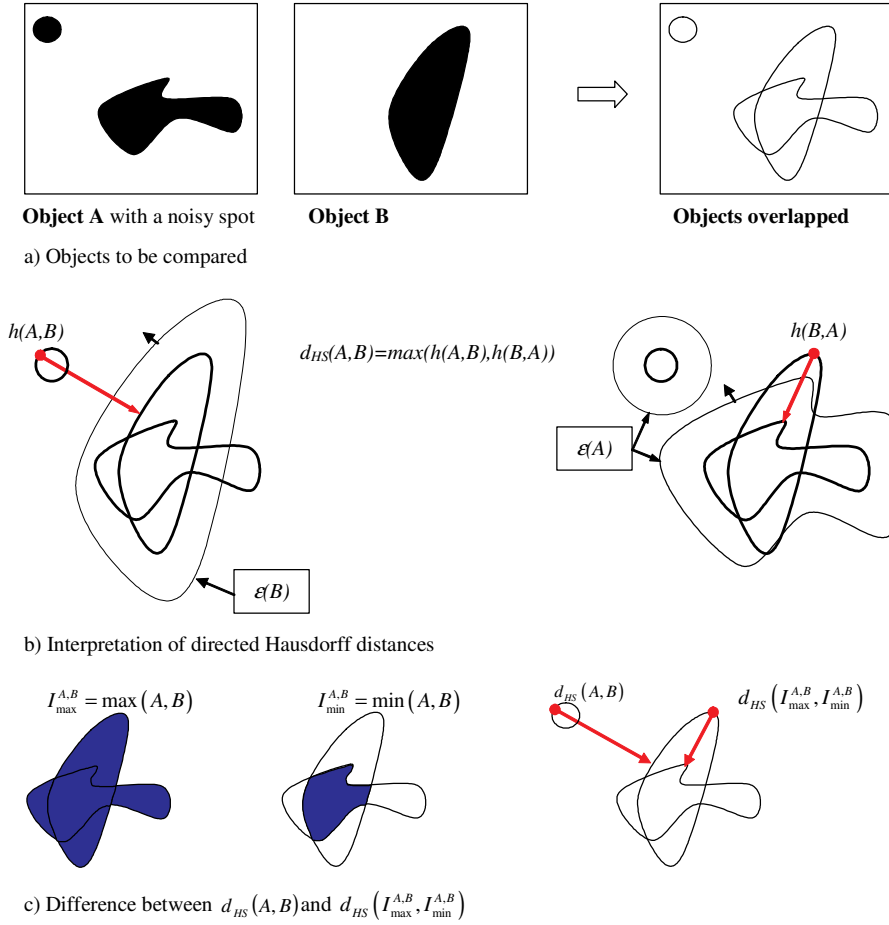
Regarding Hamming distance, we have the following modified definition:

$$(7) \qquad d_{HV} = \sum_{i=1}^{n} \left(I_{\max}^{A,B}(i) - I_{\min}^{A,B}(i) \right)$$

Note this will produce the same result ($\max(A,B) \equiv A \cup B$ and $\min(A,B) \equiv A \cap B$) regarding both binary and gray-scale objects. Evaluating Hausdorff distance, the advantage is that we do not need to compute two directed Hausdorff distances, only one as follows.

$$(8) \qquad d_{HS}(A,B) \equiv h(I_{\max}^{A,B}, I_{\min}^{A,B}) = \max_{a \in I_{\max}} \min_{b \in I_{\min}} \|a - b\|$$

For an example, see Fig. 1 showing two binary objects overlapped onto each other. It shows the interpretation of directed Hausdorff distances and a variant of them defined by Eq. 6 and 8. Another advantage of this modification is that it will simplify the PDE-based definition of nonlinear wave metric.


Object A with a noisy spot   **Object B**   **Objects overlapped**

a) Objects to be compared

$h(A,B)$

$d_{HS}(A,B)=max(h(A,B),h(B,A))$

$h(B,A)$

$\varepsilon(A)$

$\varepsilon(B)$

b) Interpretation of directed Hausdorff distances

$I_{\max}^{A,B} = \max(A,B)$ $I_{\min}^{A,B} = \min(A,B)$ $d_{HS}(A,B)$ $d_{HS}\left(I_{\max}^{A,B}, I_{\min}^{A,B}\right)$

c) Difference between $d_{HS}(A,B)$ and $d_{HS}\left(I_{\max}^{A,B}, I_{\min}^{A,B}\right)$

Fig. 1   Hausdorff distance computation (examples concern binary objects). a) Two objects to be compared. Object A has a separate spot, while Object B forms a single contiguous closed area. The third image shows overlapped objects. b) Interpretation of directed Hausdorff distances. Term $\varepsilon(B)$ stands for ε-vicinity of B. The minimal ε-vicinity of B which covers completely set A is the directed (asymmetric) Hausdorff distance of A to B, i.e. $h(A, B)$ and vice versa. c) Difference between $d_{HS}(A,B)$ and $d_{HS}\left(I_{\max}^{A,B}, I_{\min}^{A,B}\right)$. Some advantages of the second case are 1, filters out noisy spots, 2, there is no need to compute two directed Hausdorff distances, only one computation is required.

2.3 Selectivity problems of metrics

In this section, simple examples will be shown to demonstrate some constraints of metrics introduced in the previous section. The examples are constructed in such a way that they could be treated analytically. Fig. 2 and Fig. 3 show examples regarding binary objects and gray-scale objects, respectively.

$$\varphi(x) \;=\; \frac{1}{\sigma\sqrt{2\pi}}\,e^{\frac{-x^2}{2\sigma^2}}$$
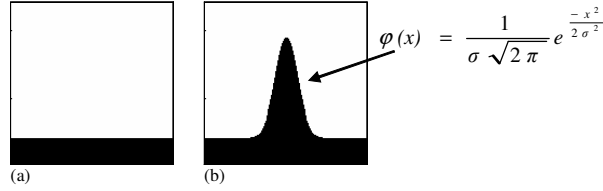
Fig. 2   Example to demonstrate the limitations of different metrics on binary objects. Object (a) contains the reference object, while (b) is an element of image series where object properties are modified via function parameters. Image size is 256x256.

$$\varphi(x,y) = \frac{1}{2\pi\sigma^2}\,e^{-\frac{x^2+y^2}{2\sigma^2}}$$
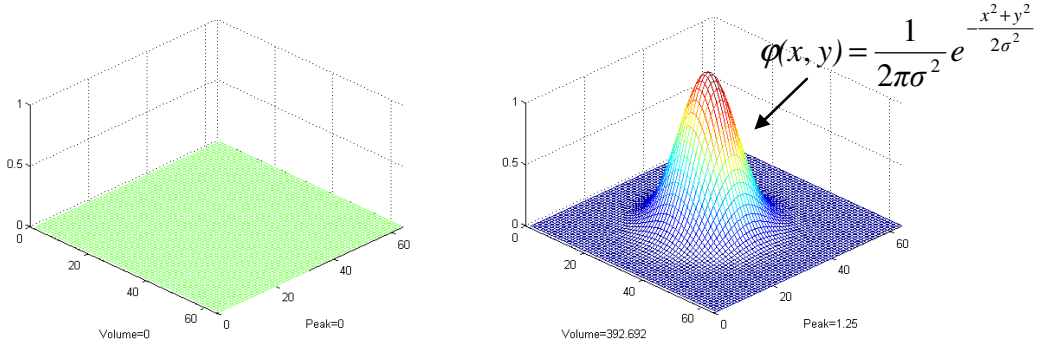
Fig. 3   Example to demonstrate the limitations of different metrics on gray-scale objects. Object (a) contains the reference object, while (b) is an element of image series where object properties are modified via function parameters.

The results are similar regarding both binary and gray-scale images. Here only the gray-scale case will be presented (considering binary situation, we refer to [25]). As Fig. 3 shows, the reference object will be the zero plane (zero intensities) and image objects have intensities of Gaussian functions. Three series will be examined as follows, also showing analytical metric evaluation (using continuous type description). Image intensities are given by values of a two-dimensional Gaussian probability density function defined as

$$(9) \qquad \phi(x,y) = \frac{1}{2\pi\sigma^2}\cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \;,$$

where $\phi(x,y)$ denotes the gray level value at point (x,y). If image intensities are given by a function, $f(x,y)$, then Hamming distance equals to its integral, while Hausdorff distance is the maximum of the function. Table 1 shows analytical values regarding our example.
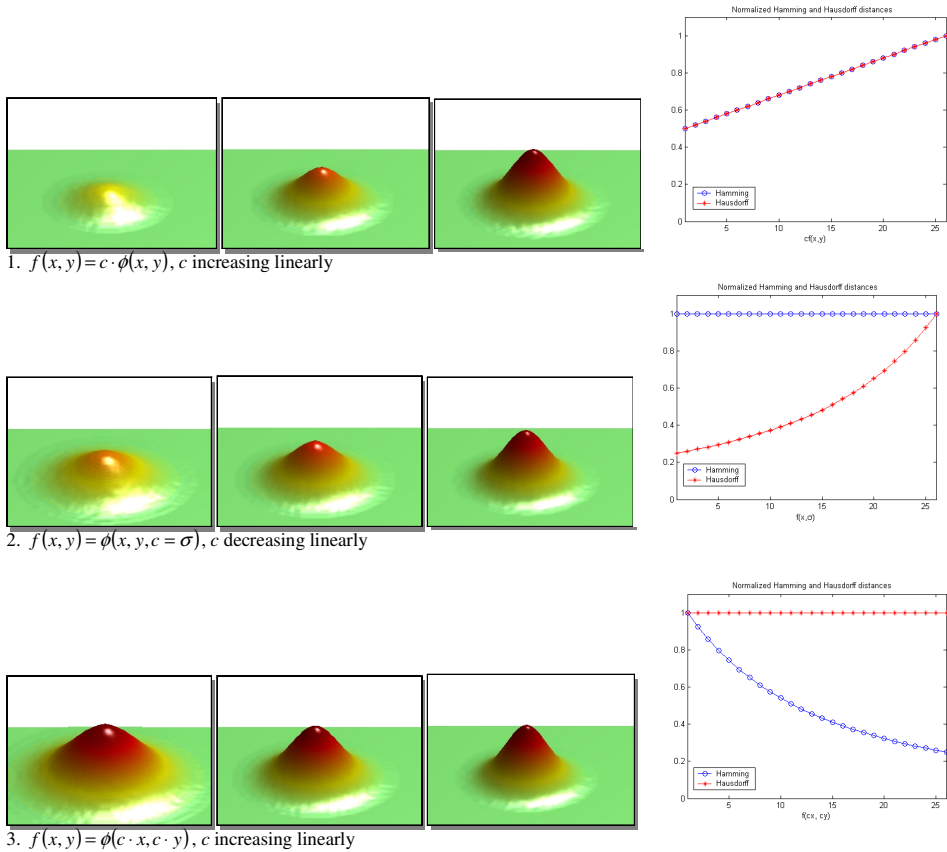
Function	Hamming distance	Hausdorff distance
$f(x,y)$	$d_H = \int\limits_{-\infty}^{+\infty} f(x,y)dxdy$	$d_{HS} = \max(f(x,y))$
$\phi(x,y) = \dfrac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$	$d_H = \Phi(-\infty, +\infty) = 1$	$d_{HS} = \phi(0,0) = \dfrac{1}{2\pi\sigma^2}$

Table 1   **Hamming and Hausdorff distances values of images described by two-dimensional Gaussian probability density functions**

We will examine three situations by using two-dimensional Gaussian probability density functions. Table 2 shows computed values. In the first case, Gaussian functions are multiplied by parameter c ($c \geq 1$) resulting in linearly increasing Hamming distances (volume) and Hausdorff distances (maximum), as well. In the second case, parameter c is the variance of the Gaussian function. It has no effect on the integral, consequently Hamming distances are constant. Nevertheless, images are very different as shown in Fig. 4. Hausdorff distances yield a quadratic dependence on parameter c. In the third case, independent variables are multiplied by parameter c resulting constant Hausdorff distances (i.e., maxima are constant), and Hamming distances show a quadratic dependence.

Function parameter	Hamming distance	Hausdorff distance
1. $f(x,y) = c \cdot \phi(x,y)$	$d_H = c$	$d_{HS} = \dfrac{c}{2\pi\sigma^2}$
2. $f(x,y) = \phi(x,y,c=\sigma)$	$d_H = 1 = $ constant	$d_{HS} = \dfrac{1}{c^2} \cdot \dfrac{1}{2\pi}$
3. $f(x,y) = \phi(c \cdot x, c \cdot y)$	$d_H \approx \dfrac{1}{c^2}$	$d_{HS} = \dfrac{1}{2\pi\sigma^2} = $ constant

Table 2   **Dependence of metrics distances on parameter c. Three cases are examined. In the second and the third cases, one of the metric distances is constant, i.e., inadequate to measure differences.**

1. $f(x, y) = c \cdot \phi(x, y)$, c increasing linearly

2. $f(x, y) = \phi(x, y, c = \sigma)$, c decreasing linearly

3. $f(x, y) = \phi(c \cdot x, c \cdot y)$, c increasing linearly

Fig. 4   Three different object series and results of their metrics measurement. Test images are generated via two-dimensional Gaussian probability density functions. Both Hamming and Hausdorff metrics can measure the first case. In the second case, the volume of surfaces are constant, therefore, Hamming distances are constant too. In the third case, surfaces have equal peak values consequently Hausdorff distances produce constant values. In the two latter cases, the corresponding metric function is inadequate for measuring differences.

What does this measurement tell us about these metrics? As we have seen, Hamming distance measures *area-* or *volume differences* (binary or gray-scale case) and it cannot separate objects having equal area or volume. Each difference gives the same contribution to the accumulated distance without grading among values and positions. At a given area or volume difference, there might exist many very different shapes. Hausdorff distance, on the contrary, takes only the farthest point (although the most important) between two sets and do not measure other points. Again, within the ε-vicinity of the measured set there might be several different arrangements of pixels (values). In the next section, we will show how PDE-based dynamics can overcome these problems and define a framework for quantitative distance measurement.

3. PDE-based nonlinear wave metric

There exist numerous examples for dynamical approaches in the field of image processing and content modification. For pattern analysis and recognition, the synergetic approach was proposed in [7] where self-organization processes [12,13] are used for pattern formation. The idea behind this approach was that an active medium is

7

related to the image space and spreading autowaves [14] explore these objects. Other dynamical approaches for image analysis and processing were proposed as deformable surfaces named as snakes by [15], level set method by studies of [16,17], front propagation in [19,20], and flame propagation in [21]. Anefficient CNN implementation of these mechanisms can be found in [23]. Also a CNN type implementation of Hausdorff distance measurement based on trigger wave propagation was proposed in [25]. Here, a constrained diffusion approach will be introduced as a general framework for object comparison.

The nonlinear PDE-based wave mapping and metric computation consist of three parts as follows.

(10)
$$D = [\underset{I.}{W}, \ \underset{II.}{\{\Psi, \Phi, G, B\}}, \ \underset{III.}{d}]$$

1. *W(.)* is a constrained diffusion mapping generating a wave map.
2. $\{\Psi, \Phi, G, B\}$ intermediate processing produces a specific distribution, gray-scale or binary morphology computation.
3. $d(.)$ identifies the final distance via a real function projection.

The three parts correspond to a three-step transformation. The first projection extends spatial information to spatio-temporal, since time is also involved to spatial dimensions. Transformations, thereafter, compress spatio-temporal information by reducing spatial and temporal dimensions into a single real number that gives, finally, the result of the metric computation.

To define a proper *W(.)* constrained diffusion mapping, let us consider image intensities as forming mountains and valleys on the surface of a ball. When we compare two images, they can be considered as two concentric balls. Comparison takes place in such a way that we stretch the elastic surface of the inner ball to the outer ball's rigid surface. During stretching, we may record its dynamics and from this information, we can derive distance measures encoding this dynamical process. This process can be modeled by a constrained reaction diffusion process as follows. Keeping the analogy of concentric balls, two images are constructed. As to the initial set (inner elastic ball), we assign the pixel-wise minimum of the images (objects) to be compared, while the constraint (outer rigid ball) will be the pixel-wise maximum of these images. Stretching will be implemented by two driving forces, the first one will be horizontal, the second one will work vertically, see Fig. 5. The horizontal force can be realized by the diffusion term of the reaction-diffusion equation. As to vertical force, a control function will govern time evolution of wave propagation.
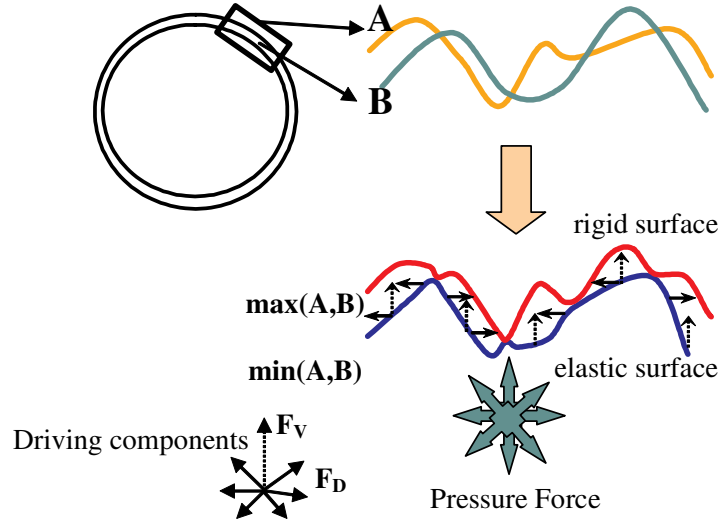
Fig. 5 **Wave mapping between two images may be considered as stretching an elastic inner ball to the rigid surface of the outer ball. The initial set of the transformation (elastic surface) is the minimum values of intensity pairs. The final set (rigid surface) is the pixel-wise maximum of intensity values. The wave mapping records this morphing and encodes spatio-temporal dynamical information in a spatial map.**

3.1 The PDE model

We define an image (either binary or gray-scale) as a real function $I(x, y)$ where it denotes intensity value at the point (x, y). Object intensities are in the region of (0, 1]. Zero intensity stands for background. The dynamical equation producing wave map for comparison of object $A(x,y)$ and $B(x,y)$ is a nonlinear partial differentiation equation (PDE) derived from the ball analogy.

(11)
$$\left.\begin{array}{l} \dfrac{\partial I_1(x,y,t)}{\partial t} = D \cdot \Delta I_1 + f_v\left(I_1, I_{\max}^{A,B}\right) \\[2mm] \dfrac{\partial I_2(x,y,t)}{\partial t} = f_w\left(I_1, I_{\max}^{A,B}\right) \end{array}\right\}$$
, with the following initial conditions:

(12)
$$I_1(x,y,0) = I_{\min}^{A,B}(x,y) = \min\left(A(x,y), B(x,y)\right),$$
$$I_2(x,y,0) = 0,$$
$$I_{\max}^{A,B}(x,y) = \max\left(A(x,y), B(x,y)\right)$$

Operator Δ is the *continuous Laplace operator* ($\dfrac{\partial^2}{\partial x^2} + \dfrac{\partial^2}{\partial y^2}$) in the two-dimensional case. Regarding control functions, the simplest choice for $f_v(.)$ and $f_w(.)$ can be as follows:

(13)
$$f_v(x,y):c_v\cdot\left(I_{max}^{A,B}(x,y)-I_1(x,y)\right)$$
$$f_w(x,y):w\cdot\left(I_{max}^{A,B}(x,y)-I_1(x,y)\right)$$
, where $c_v>0$, and $w>0$.

The vertical force $f_v(.)$ ensures that image intensities will be growing from $I_{min}^{A,B}$ upto $I_{max}^{A,B}$, while it also realizes a negative feedback to guarantee the process stopping at $I_{max}^{A,B}$. Weighting function $f_w(.)$ produces the wave map on the second layer with growing intensities at each position *(x,y)* until wave front reaches $I_{max}^{A,B}(x,y)$ on the first layer, i.e., $I_1(x,y,\infty)=I_{max}^{A,B}(x,y)$. We define the final wave map as the steady state solution of the second layer:

(14)
$$W_{AB}(x,y,\infty)=I_2(x,y,\infty),$$
$$\frac{\partial I_2}{\partial t}=0\Big|_{I_1(\infty)=I_{max}}$$

Note the second layer encodes the time evolution of the first layer's dynamics. Several quantitative measurements can be defined by using the wave map. For object comparison, for instance, we can define the following measure between object A and B, as follows.

(15) $\quad d_W(A,B)=\displaystyle\int_{D_{AB}}W_{AB}(x,y,\infty)dxdy$, where

the domain of the integration is:

(16) $\quad D_{AB}=\left\{(x,y)\in A\cup B\,\big|\,(x,y)\text{ is arcwise connected to }A\cap B\text{ in }A\cup B\right\}$

Remark 1: The definition of integration domain is necessary if several object pairs are compared together. Otherwise it can be the whole image space (considering compact objects, W_{AB} is zero outside the domain). Regarding noise suppression, domain D_{AB} should contain only those points which form closed, contiguous region with $A\cap B$. Here, we used the term *connectedness* of topology.

Definition of Connectivity: A point set *D* is connected if any two points in *D* can be connected by a curve lying wholly within *D*.

The difference measurement defined by (15) relies on each point of the difference's set in which each position gives its specific value to the distance. The farther the point is from the minimum, the higher the weight is contributing to the distance. It is opposite to the Hamming distance where each point has the same weight and it is also opposite to the Hausdorff distance where the distance of the farthest point only results in the distance measure.

The wave map (W_{AB}) shown in Fig. 6 is the result of the comparison of objects in Fig. 1. As we can see, this map includes both the Hamming and Hausdorff distances as special
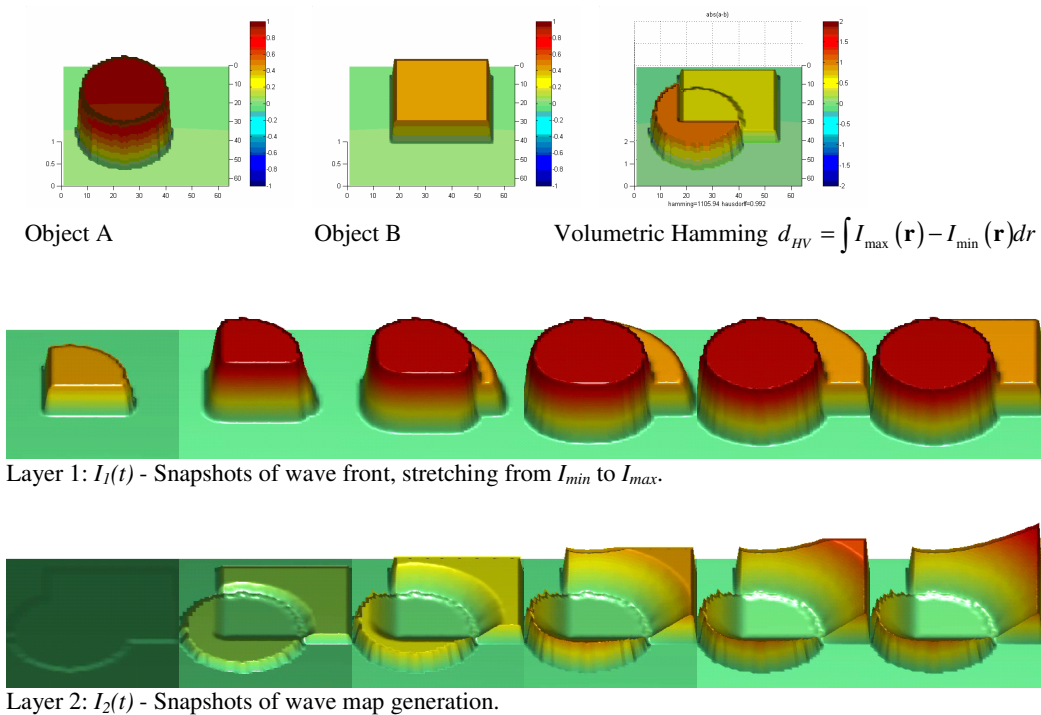
cases. It should be noted that, while the Hausdorff distance can only be determined at the end of the process, the Hamming distance is already known at the beginning.
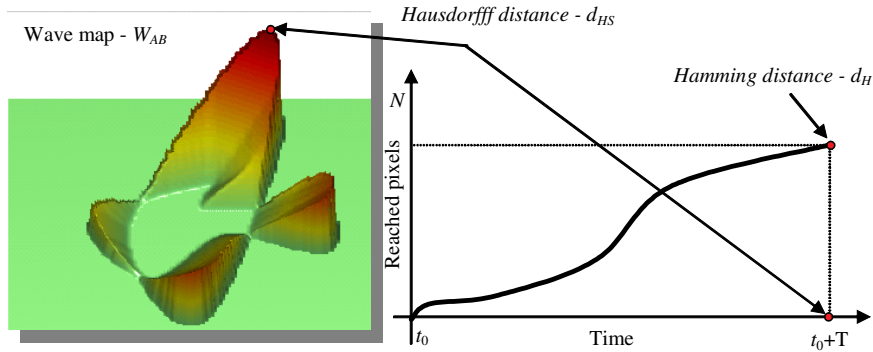
Fig. 6 **Result of constrained diffusion-based wave map generation regarding objects shown in Fig. 1. The plot of pixels reached by propagating waves, versus time, shows the properties of wave propagation. The number of pixels at the end point gives the Hamming distance, while the time for reaching all pixels is equal to the Hausdorff distance.**

We may interpret the intensity values of this wave map as a *local Hausdorff distance*, i.e., the ε-vicinity measure of the initial set (I_{min}) with the constraint of the final set (I_{max}).

As a second example, Fig. 7 shows constrained diffusion-based wave transition and wave map generation.

Object A   Object B   Volumetric Hamming $d_{HV} = \int I_{max}(\mathbf{r}) - I_{min}(\mathbf{r}) dr$

Layer 1: $I_1(t)$ - Snapshots of wave front, stretching from I_{min} to I_{max}.

Layer 2: $I_2(t)$ - Snapshots of wave map generation.

Fig. 7   Example showing PDE-based wave map generation. Object A and Object B are compared. Constrained diffusion-based wave propagation takes place on the first layer, while wave map generation is produced on the second layer.

For our examples shown in Fig. 4, the dynamical system produces a propagating wave front (as to binary objects trigger wave was used) from the elastic surface and time is recorded at each location that is necessary for the wave front to reach the rigid surface. Corresponding measurements are shown in Fig. 8. As it can be seen, PDE-based dynamical measure can distinguish all the three series unlike the Hamming and the Hausdorff distances. It is worth mentioning that like in binary object comparison, several gray-scale object pairs can simultaneously be compared to each other. In these examples, wave maps form poling shapes with a concentric line of local maxima with or without a local maximum in the centre of these shapes. This is the result of wave propagation driven by both vertical and horizontal forces. Note that local maxima correspond the locations where the wave front reached the rigid surface for the last time. Thus, contrary to the Hausdorff measure, not only the height of the Gaussian surface was measured but also its width.
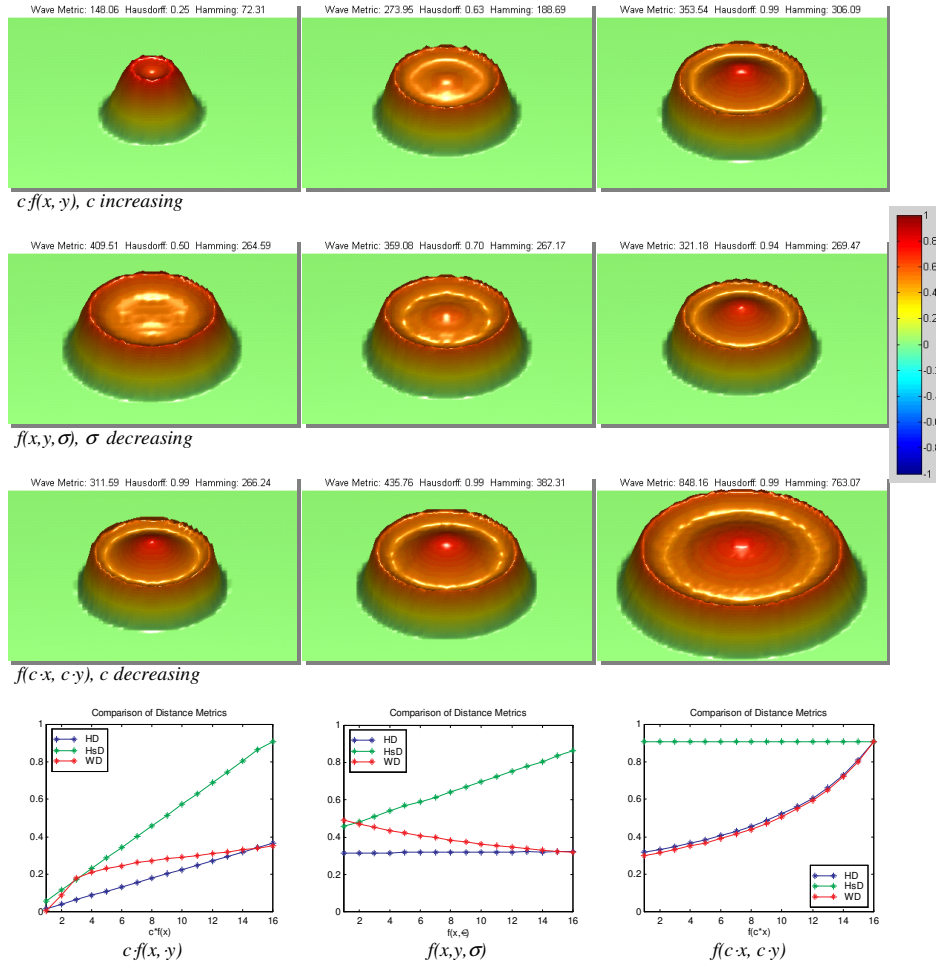
Fig. 8 **Wave maps and the computed wave metric distances for the three series of two dimensional Gaussian functions (see Fig. 4). Wave based comparison is capable for measuring differences contrary to Hamming and Hausdorff metrics.**

4. Discussion

The interpretation of PDE-based wave metric

On the basis of the previous discussion, we can give two interpretations of PDE-based metric computation as follows.

- *Weighted Hamming*: Hamming distances are weighted and summed where the intensity values of the wave map stand for these weights. The higher the intensity, the farther the distance of the point is from the initial position. It means that farther points are more important than near points.

- *Integrated Hausdorff*: The intensity values of the wave map stand for *"local"* Hausdorff distances. Local means that instead of the Hausdorff distance, in which the distance of the farthest point is computed, distances for each pixel are determined and stored (as if they were the farthest points). Summing these values (in

continuum it is the integration, i.e., the volume of the wave map) gives the measure of the difference between two objects.

For the latter interpretation, we can define local Hausdorff distance regarding binary objects as

(17) $d_{LHS}^{A,B}(x,y) = h((x,y), A \cap B)$, where

$h((x,y), A \cap B)$ is the directed Hausdorff distance of point (x,y) to $A \cap B$. Regarding gray-scale case, if a point $p(x,y)$ is given with intensity p at coordinate (x,y) the local Hausdorff distance can be defined as

(18) $d_{LHS}^{A,B}(p(x,y)) = h(p(x,y), \min(A,B))$. Note the gray-scale definition includes the binary case as well (in binary case $p=1$, and $\min(A,B) = A \cap B$).

The PDE-based distance computation can be expressed via local Hausdorff distances regarding binary objects:

$$(19)\, d_W(A,B) = \int_{D_{AB}} d_{LHS}^{A,B}(x,y)\, dxdy = \int_{D_{AB}} h((x,y), A \cap B)\, dxdy\ ,$$

where the domain of the integration is

$$(20)\, D_{AB} = \{(x,y) \in A \cup B \,|\, arcwise\ connected\ to\ A \cap B\ in\ A \cup B\}$$

The PDE-based distance regarding gray-scale objects is

$$(21)\, d_W(A,B) = \int_{D_{AB}} h(I_{\max}^{A,B}(x,y), \min(A,B))\, dxdy\ ,$$

where $I_{\max}^{A,B}(x,y) = \max(A(x,y), B(x,y))$. Note the gray-scale case includes the binary case as well.

Checking metric conditions
Let A, B, and C be three either binary or gray-scale objects to be compared. We will show that the first two conditions in (1) hold trivially while regarding the third condition to be true wee need the following definitions. We define initial set as $S_{\min} = \min(A,B,C)$ with condition that $\min(A,B,C) \neq \{0\}$ and final sets as

$S_{AB} = \max(A,B)$, with integration domain

$D_{AB,C} = \{(x,y) \in A \cup B \,|\, S_{AB}(x,y)\, arcwise\ connected\ to\ A \cap B \cap C\ in\ A \cup B \cup C\}$

$S_{BC} = \max(B,C)$, with integration domain

$D_{BC,A} = \{(x,y) \in B \cup C \,|\, S_{BC}(x,y)\, arcwise\ connected\ to\ A \cap B \cap C\ in\ A \cup B \cup C\}$

$S_{AC} = \max(A,C)$, with integration domain

$D_{AC,B} = \{(x,y) \in A \cup C \,|\, S_{AC}(x,y)\, arcwise\ connected\ to\ A \cap B \cap C\ in\ A \cup B \cup C\}$.

Let us compute three wave based distances as

$$d_{wAB} = d_w\left(S_{AB}, S_{\min}\right)\big|_{D_{AB,C}},$$

$$d_{wBC} = d_w\left(S_{BC}, S_{\min}\right)\big|_{D_{BC,A}}, \text{and}$$

$$d_{wAC} = d_w\left(S_{AC}, S_{\min}\right)\big|_{D_{AC,B}}$$

THEOREM: The wave mapping based distance measure of objects defined by (15) satisfies the metric axioms among distances d_{wAB}, d_{wBC}, and d_{wAC}.

Proof:

(1)   *Identity and Positiveness*. Regarding any two objects, if $A = B$ the wave map will be the zero plane so will be $d_w(A, A) = 0$. If $A \neq B$ the mapping will produce wave map with non-negative values at each position of objects resulting in $d_w(A, B) > 0$.

(2)   *Symmetry*. This can be simply proved by swapping variables *A, B* in equation (21), therefore we can write the following:

$$d_w(A, B) = d_w\left(\max(A, B), \min(A, B)\right) = d_w(B, A).$$

(3)   *Triangle inequality*. Let us suppose we can write the following:

$$d_{wAB} + d_{wBC} \overset{?}{\geq} d_{wAC}$$

$$d_w(S_{AB}, S_{\min}) + d_w(S_{BC}, S_{\min}) \overset{?}{\geq} d_w(S_{AC}, S_{\min})$$

$$\int_{D_{AB,C}} h\left(S_{AB}(x, y), \min(A, B, C)\right) dxdy + \int_{D_{BC,A}} h\left(S_{BC}(x, y), \min(A, B, C)\right) dxdy \overset{?}{\geq} \int_{D_{AC,B}} h\left(S_{AC}(x, y), \min(A, B, C)\right) dxdy$$

, which can be written as *(a), regarding binary objects*

$$\int_{D_{AB,C}} h\left(1(x, y), \min(A, B, C)\right) dxdy + \int_{D_{BC,A}} h\left(1(x, y), \min(A, B, C)\right) dxdy \overset{?}{\geq} \int_{D_{AC,B}} h\left(1(x, y), \min(A, B, C)\right) dxdy$$

That is equivalent to

$D_{AB,C} \cup D_{BC,A} \supseteq D_{AC,B}$ which is always true. Therefore, we can write $d_{wAB} + d_{wBC} \geq d_{wAC}$.

(b), regarding gray-scale objects

the condition is equivalent to

$$\int_{D_{AB,C}} \max(A, B) dxdy + \int_{D_{BC,A}} \max(B, C) dxdy \overset{?}{\geq} \int_{D_{AC,B}} \max(A, C) dxdy.$$ We can distinguish seven cases (see Fig. 9).

Case 1: $D_1 = \{A \cap B \cap C\}$ we get $\int_{D_1} \max(A, B) dxdy + \int_{D_1} \max(B, C) dxdy \geq \int_{D_1} \max(A, C) dxdy$

Case 2: $D_2 = \{A \cap C \setminus B\}$ we get $\int_{D_2} A(x, y) dxdy + \int_{D_2} C(x, y) dxdy \geq \int_{D_2} \max(A, C) dxdy$

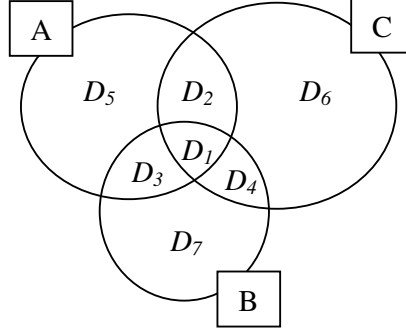Case 3: $D_3 = \{A \cap B \setminus C\}$ we get $\int_{D_3} \max(A, B) dxdy + \int_{D_3} B(x, y) dxdy \geq \int_{D_3} A(x, y) dxdy$

Case 4: $D_4 = \{B \cap C \setminus A\}$ we get $\int_{D_4} B(x, y) dxdy + \int_{D_4} \max(B, C) dxdy \geq \int_{D_4} C(x, y) dxdy$

Case 5: $D_5 = \{(A \setminus B) \setminus C\}$ we get $\int_{D_5} A(x, y) dxdy + \int_{D_5} 0(x, y) dxdy \geq \int_{D_5} A(x, y) dxdy$

Case 6: $D_6 = \{(C \setminus A) \setminus B\}$ we get $\int_{D_6} 0(x,y)\,dxdy + \int_{D_6} C(x,y)\,dxdy \geq \int_{D_6} C(x,y)\,dxdy$

Case 7: $D_7 = \{(B \setminus A) \setminus C\}$ we get $\int_{D_7} B(x,y)\,dxdy + \int_{D_7} B(x,y)\,dxdy \geq \int_{D_7} 0(x,y)\,dxdy$

Finally, we can write $d_{wAB} + d_{wBC} \geq d_{wAC}$.■

Fig. 9   The interpretation of different integration domains

If there are more objects than three the construction of initial set can be extended in two different ways. Either the initial set can be defined as the minimum of all objects (its drawback that distances change as a new object is included) or it can be a fixed set of points in the image space (practically related to the application but it should not depend on objects). In this latter case, the distance calculation can be defined as

$$(22)\, d_W(A,B) = \int_{D_{AB}} h\left(I_{max}^{A,B}(x,y), I_{ref}\right) dxdy \; , \text{ where}$$

$I_{ref}(x,y)$ is the fixed set of points in the image space. This definitions includes (19) and (21) as special cases. The distance calculation defined in (22) satisfies the metric axioms. The proof is very similar to the presented one. Regarding the PDE implementation, the only change is in the initial condition of (12), i.e., $I_1(x,y,0) = I_{ref}(x,y)$.

If we interpret the wave mapping based calculation as a weighted Hamming distance we can define (among others) the following distance:

$$(23)\, d_W(A,B) = \int \underbrace{\left|A(x,y) - B(x,y)\right|}_{\textbf{Hamming}} \cdot \underbrace{\left(1 + h\left(I_{max}^{A,B}(x,y), I_{ref}\right)\right)}_{\textbf{Weight}} dxdy \; .$$

Equation (23) satisfies the metric axioms if the weight function is strictly positive. The domain of the integration can be the whole image space because the term Hamming is zero outside the objects.

16

An other variant of weighted Hamming can be defined as a weighted surface integral of the ε-vicinity of I_{ref} with the constraint that $\varepsilon \le h\left(I_{max}^{A,B}, I_{ref}\right)$.

$$(24)\; d_W(A,B) = \int\limits_0^{h\left(I_{max}^{A,B}, I_{ref}\right)} \int_{S\left(\varepsilon\left(I_{ref}\right)\right)} \varepsilon \, da d\varepsilon \;,\; \text{where}$$

$S\left(\varepsilon\left(I_{ref}\right)\right)$ is the surface of the ε-vicinity of I_{ref}. In case of binary objects, this definition is equivalent with (19). Fig. 10 shows graphically the interpretations of different wave-based metrics.

$$d_W(A,B) = \int\limits_{D_{AB}} h\left((x,y), A \cap B\right) dxdy$$

$$d_W(A,B) = \int\limits_{D_{AB}} h\left(I_{max}^{A,B}(x,y), \min(A,B)\right) dxdy$$

$$d_W(A,B) = \int \left|A(x,y) - B(x,y)\right| \left(1 + h\left(I_{max}^{A,B}(x,y), I_{ref}\right)\right) dxdy$$

$$d_W(A,B) = \int\limits_0^{h\left(I_{max}^{A,B}, I_{ref}\right)} \int_{S\left(\varepsilon\left(I_{ref}\right)\right)} \varepsilon \, da d\varepsilon$$
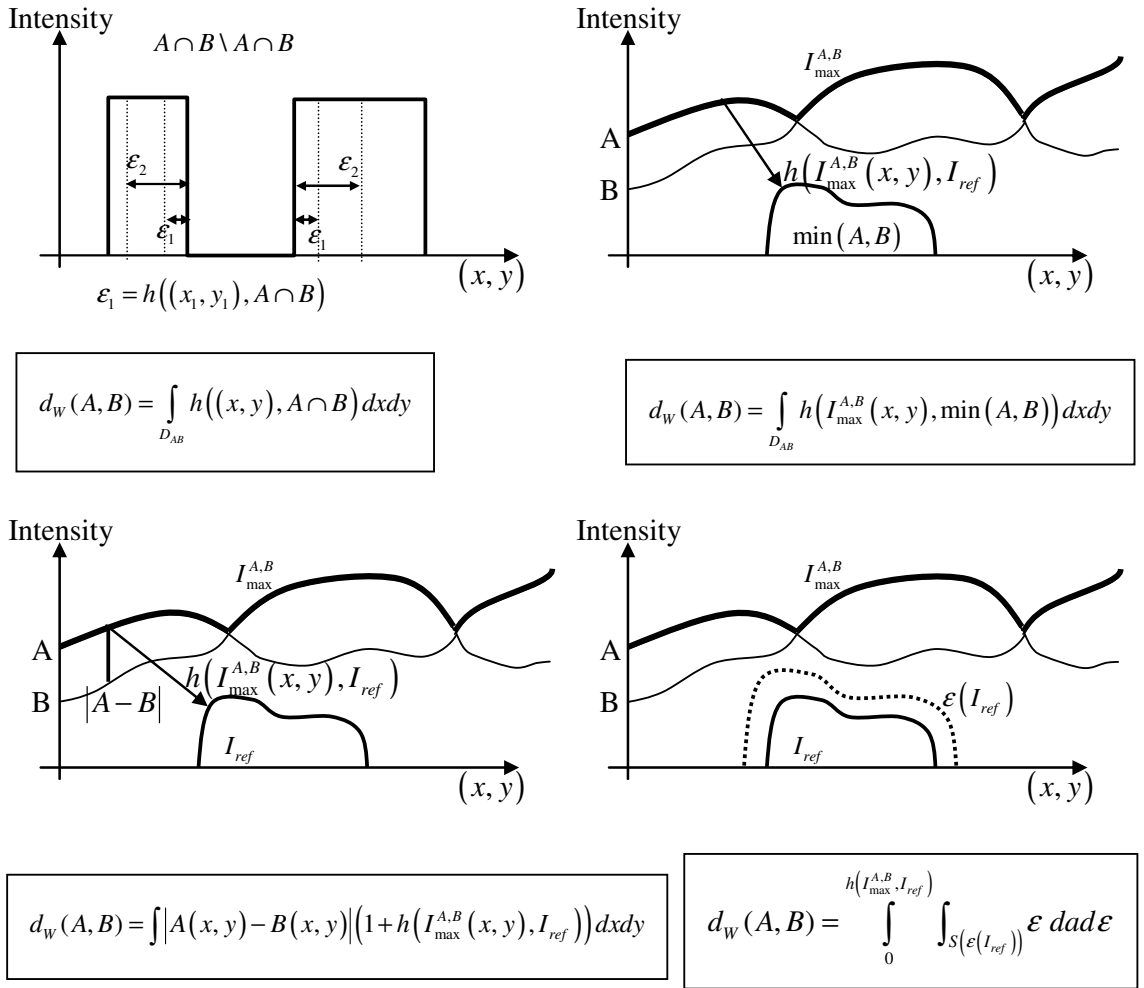
Fig. 10 Variants of wave-based metrics. *a)* Integrated local Hausdorff distance for binary objects, *b)* integrated Hausdorff distance for gray-scale objects (gray-scale includes binary case as well). *c)* Weighted Hamming distance by Hausdorff. *d)* Weighted surface integral of the ε-vicinity of reference set with constraint of *max(A,B)*. Regarding binary objects, the last definition is equivalent with *(a)*.

Next, properties and connection between wave mapping, time distribution, and histogram calculation will be discussed.

Connections between distribution and histogram calculation

Wave processing: $W : W_{AB}(x, y, t)\big|_{I_{min}^{A,B} \rightarrow I_{max}^{A,B}}$
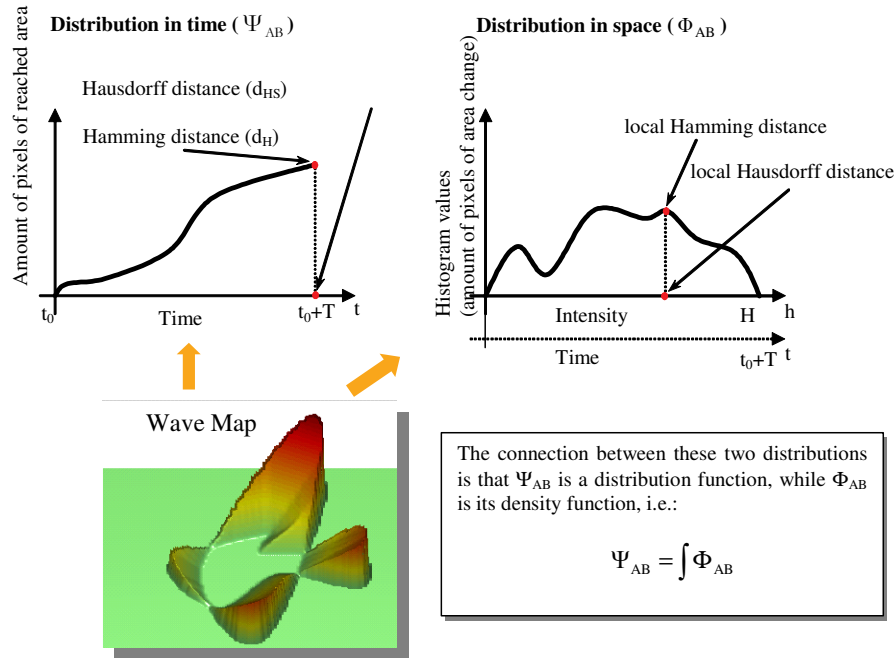
For intermediate processing, we introduce two functions projecting wave map to space distribution and time distribution.

(25)   *Time distribution of the wave map:* $\Psi_{AB} \triangleq \Psi(W_{AB}, t)$

This denotes the distribution of the wave map related to the time. It means that at each time t function $\Psi_{AB}(t)$ has the area of saturated region to the rigid surface by the wave process, i.e., the number of reached positions in discrete space.

(26)   *Spatial distribution of the wave map:* $\Phi_{AB} \triangleq H(W_{AB}) \equiv \Phi(W_{AB}, t)$

This denotes the histogram of the wave map. Because during the wave process an intensity value encodes the corresponding propagation time, this definition holds if the distribution is expressed versus time (see Fig. 11).

The connection between these two distributions is that Ψ_{AB} is a distribution function, while Φ_{AB} is its density function, i.e.:

$$\Psi_{AB} = \int \Phi_{AB}$$

Fig. 11   Projections of wave map to time distribution and to space distribution. The wave map was constructed from the example shown in Fig. 1.

Different distance calculations can be expressed from the distribution functions.

$$\text{Hamming distance}: d_H = \int_0^H \Phi_{AB}(h)\,dh \equiv \frac{1}{T} \cdot \int_{t_0}^{t_0+T} \Phi_{AB}(t)\,dt$$

(27) $\text{Hausdorff distance}: d_{HS} = H°T$

$$\text{PDE- based wave distance}: d_W = \int_0^H h \cdot \Phi_{AB}(h)\,dh \equiv \int_{t_0}^{t_0+T} t \cdot \Phi_{AB}(t)\,dt$$

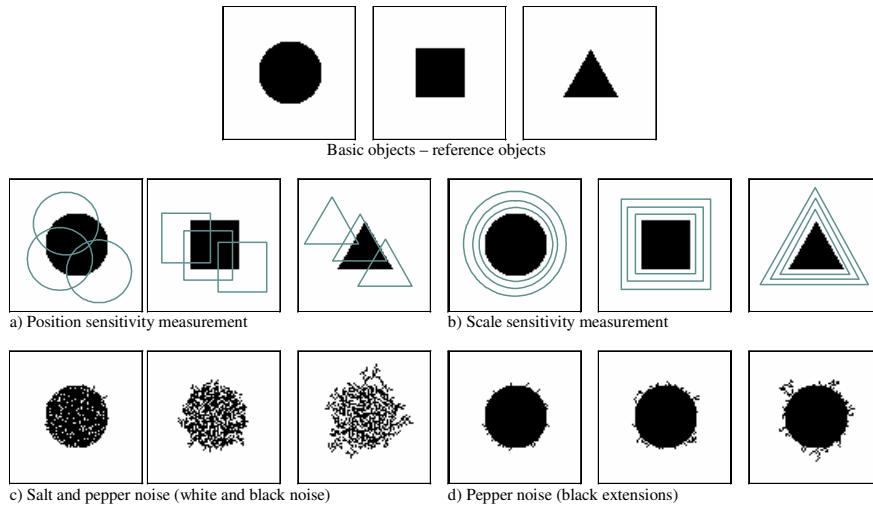**Connections between time and spatial distribution**
As it is shown above, PDE-based wave metric can be computed already from the intermediate process by using distribution calculation. This is important from the point of view of implementation (for more details, see Sec 4.).

(28) $$d_W = \int_{t_0}^{t_0+T} t \cdot \Phi_{AB}(t)\,dt = \int_{D_{AB}} W_{AB}(x, y, \infty)\,dxdy$$

For instance, if the iterative type implementation of the wave map generation is used, the distribution function can be computed at each time step very simply by using the first type of equation. The area integration belongs to the dynamic type implementation.

The sensitivity and selectivity of PDE-based wave metric
An interesting question is the sensitivity of PDE-based distance calculation regarding position errors, noise, scaling, etc. Using the term *sensitivity* from the field of electrical engineering [22], *distance metric sensitivity* can be defined as the measure of the change in distance measurement affected by a given perturbation (noise, position, scale, etc). Examples are shown in Fig. 12. The sensitivities of different distance calculations were evaluated and measurements are shown in Fig. 13. In all cases, PDE-based wave metric computation showed the largest selectivity and sensitivity. In the example, noise sensitivity measurement means the assessment of the extent of noise level, rather than the effect of noise itself. As mentioned earlier, a single noisy spot has great affect in Hausdorff distance but not in PDE-based wave metric calculation.

Basic objects – reference objects

a) Position sensitivity measurement

b) Scale sensitivity measurement

c) Salt and pepper noise (white and black noise)


d) Pepper noise (black extensions)

Fig. 12 Basic shapes to test distance sensitivity by using different types of perturbation. Disturbances may alter position, scale, rotation, and may cause different noise effects.

Position
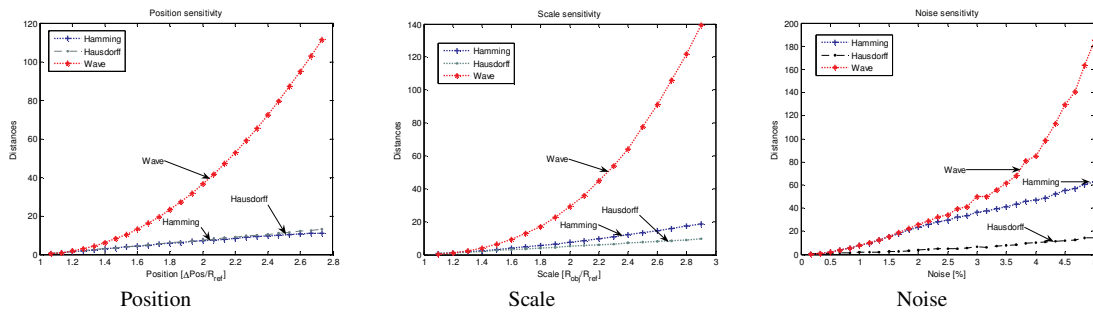
Scale

Noise

Fig. 13 Distance sensitivity measurements by using different types of perturbation models. PDE-based wave metric, Hamming, and Hausdorff metrics have been evaluated. Perturbation models: a) position, b) scale, c) noise. In all the three cases, PDE-based wave metric showed the largest sensitivity and selectivity.
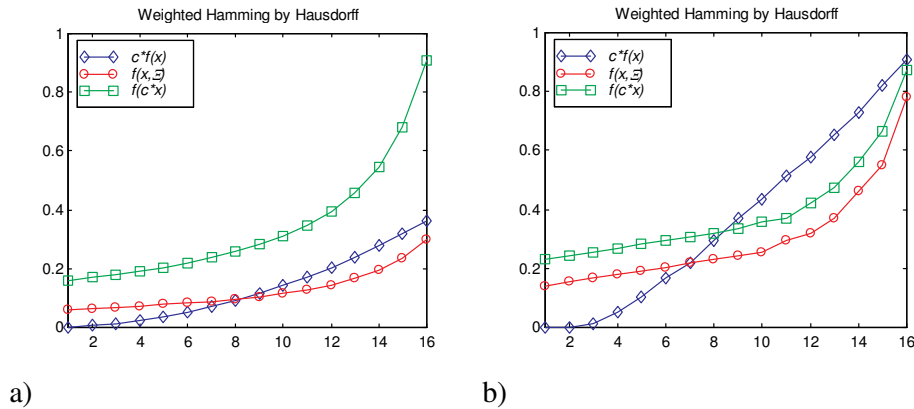
5. CNN implementation

An iterative and also a dynamical type implementation of wave-based metric concerning binary object comparison were presented in [25]. The iterative approach was implemented on the CNN-UM chip, ACE4K [24]. An example is shown in Fig. 14.
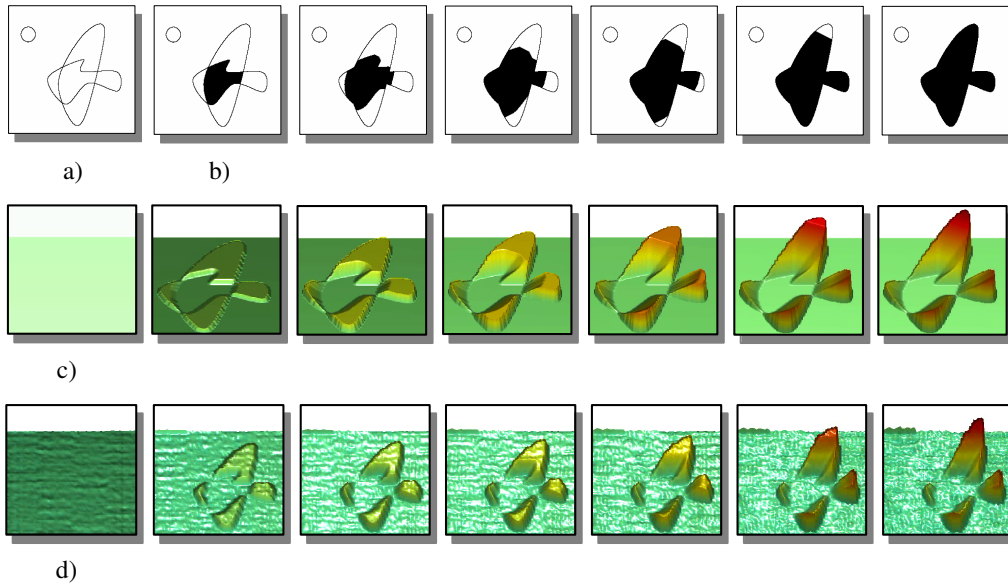
Fig. 14 CNN-UM implementation of wave map generation. a) Outlines of two partially overlapping point set, b) Propagating wave spreads from the intersection through the union of contiguous parts of point sets until all the points become triggered, c) Wave map generated by increasing intensities of pixels until wave front reaches them, *simulation result* **d) Consecutive steps of generating the wave map on the 64x64 I/O CNN-UM chip (ACE4K). Note: intermediate steps are shown with demonstration purpose.**
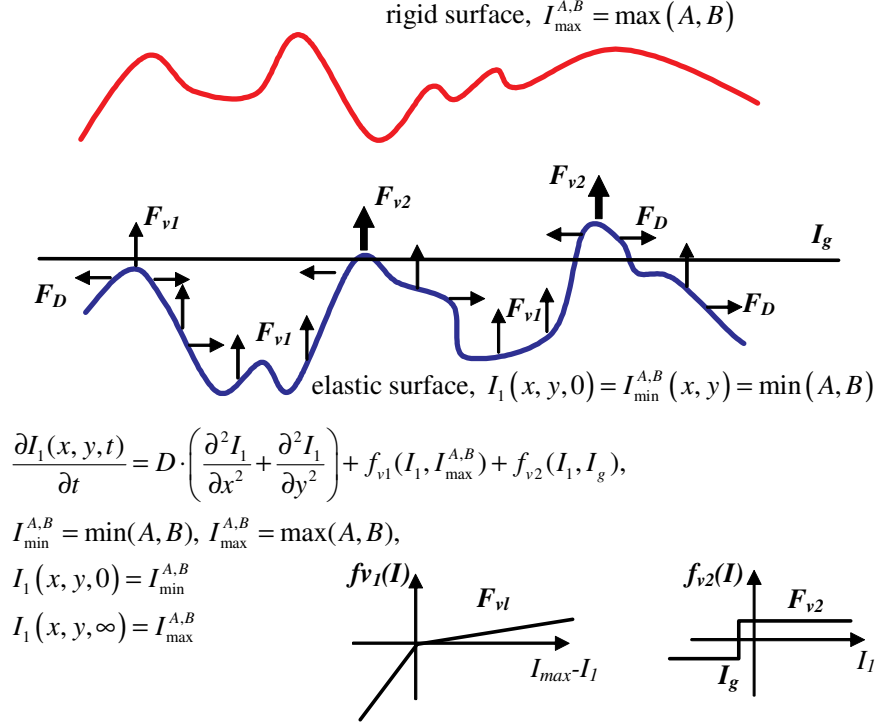
Regarding the example presented in Fig. 2, the PDE-based distance calculation produces monotonic function for all the three cases both in simulation and on ACE4K chip (Fig. 15).

Fig. 15 PDE-based metric values for the three cases shown in Fig. 2. a) simulation results, b) results of the iterative implementation of wave-based distance calculation on the 64x64 I/O CNN-UM chip made in Seville (ACE4K, [24]).

As we have seen, concerning gray-scale object comparison the wave generation should be modified and cannot be treated as the propagation of saturated states. Equation (11) defines the general framework capable for object comparison both in binary and gray-scale cases, as well. In case of binary objects, this framework returns the same result

obtained by the previous version presented in [25]. Here, constraints are incorporated into control functions instead of using fixed state maps (masks). As to vertical force, two nonlinear functions have been defined (the modified version of function $f_v(.)$ in Eq. 13). This has advantage from the point of view of VLSI chip implementation.

rigid surface, $I_{max}^{A,B} = \max(A,B)$

elastic surface, $I_1(x,y,0) = I_{min}^{A,B}(x,y) = \min(A,B)$

$$\frac{\partial I_1(x,y,t)}{\partial t} = D \cdot \left(\frac{\partial^2 I_1}{\partial x^2} + \frac{\partial^2 I_1}{\partial y^2} \right) + f_{v1}(I_1, I_{max}^{A,B}) + f_{v2}(I_1, I_g),$$

$$I_{min}^{A,B} = \min(A,B), \; I_{max}^{A,B} = \max(A,B),$$

$$I_1(x,y,0) = I_{min}^{A,B}$$

$$I_1(x,y,\infty) = I_{max}^{A,B}$$

Fig. 16   **Reaction-diffusion type wave process for gray-scale object comparison on CNN. The stretching of elastic surface to the rigid one is realized by three components. The diffusion part behaves as a horizontal force and the two nonlinear functions ensure vertical forces. Function $f_{v1}(.)$ realizes also a feedback term to guarantee that process will stop at the rigid surface. Function $f_{v2}(.)$ speeds up the process.**

The nonlinearity of function $f_{v1}(I)$ ensures two different properties. The positive part forces propagation to the rigid surface even if diffusion does not. The negative part realizes a strong negative feedback to ensure that the waves should stop at the rigid surface. Function $f_{v2}(.)$ speeds up the propagation to the rigid surface. Advantage of this implementation is that there is no need for a special mask for stopping this wave process, this is already included into the cell interaction.

6. Conclusion and possible applications

As it has been shown, computation of constrained diffusion-based wave metric can be used for both binary and gray-scale object comparison. The advantage of the proposed method is that several object pairs can be measured simultaneously. The approach based on spatio-temporal processes gives qualitatively novel way for object exploration and in addition to this it also provides investigation tool for quantitative measurements. Dynamics of this process is recorded during the transient resulting new, time related information in addition to spatial properties. All this information is encoded in a special

wave map which can be used for extracting several measures based on the requirements of different applications. Next some possible applications are mentioned where PDE-based metrics can be exploited.

### *Condition-based maintenance*

An application case study was presented in [26]. In that work, the wave metric was applied to the bubble debris separation problem where a huge number of objects was to be classified in a very short time.

Evaluation of contour estimations

Wave-based distance calculation was applied for evaluating errors in different active contour techniques [27]. This technique can also be used for the assessment of contour estimation in the tracking of heart movement in echocardiography applications.

Boundary and surface roughness measurement

Another possible application could be the boundary- or surface-roughness measurement of objects. The deformation is projected to the boundary (surface) of the reference object. Let us examine a simple example shown in Fig. 17. The second case in Fig. 17 has the advantage that the roughness values are projected to the line of reference, therefore, the analysis is much easier. For visualization purposes, it is quite convenient to draw it as a one-dimensional function where function values related to the boundary of the reference object, thus, showing the difference of the rough object. Similar techniques may be applied for surface roughness measurement.
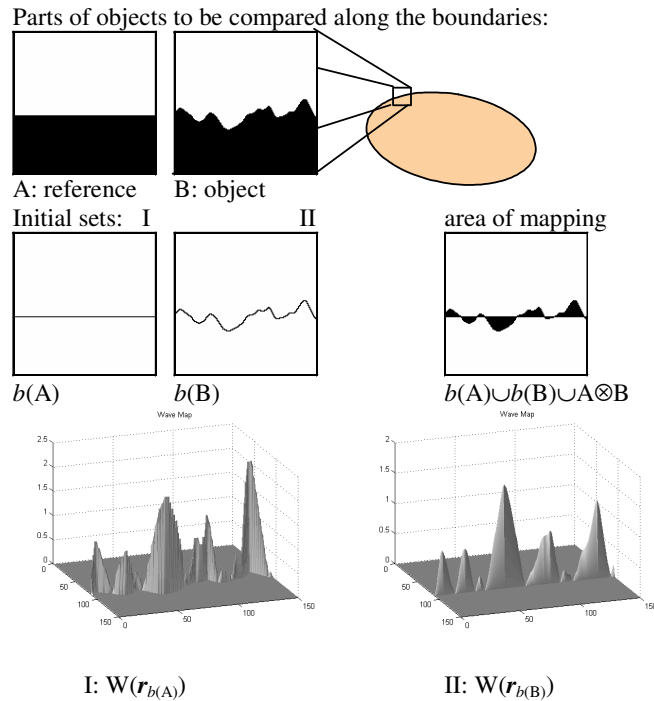
Fig. 17 The proposed method applied to roughness measurement. Here, wave maps are restricted to boundaries.

Trace tracking of dynamical processes

PDE-based wave mapping may have controlling input not only from a wave-front traveling among objects. If we consider, for instance, the two-layer dynamical implementation of mapping, then it is quite clear that on the first layer any dynamical process may run. The second layer records this process and encodes its time evolution as a spatial map. By proper settings, any periodic or even chaotic process can be tracked. Similarly to a Poincare map, the second layer will encode trace information about the process.

Optical flow estimation

Another promising possibility is using wave mapping for estimating optical flow. While on the first layer the image flow can be observed, the second layer can compute – within a considerable short time – the difference between two consecutive snapshots of the image flow based on wave map calculation. The local maxima of the wave map encode these differences and also the ramps of these "local maps" encode directions of local displacements.

References

[1] Chua L. O, and Yang L. Cellular Neural Networks: Theory, *IEEE Trans. on Circuits and Systems*, Vol. 35, October (1988), 1257-1272.

[2] Chua L. O, and Yang L. Cellular Neural Networks: Applications, *IEEE Trans. on Circuits and Systems*, Vol. 35, October (1988), 1273-1290.

[3] Chua L. O, and Roska T. The CNN Paradigm, *IEEE Trans. on Circuits and Systems*, Vol. 40, March (1993), 147-156.

[4] Roska T, and Chua L. O. The CNN Universal Machine: an Analogic Array Computer, *IEEE Transactions on Circuits and Systems II* 1993; Vol. 40, No. 3, March 163-173

[5] Perona P, Malik J. Scale-space and edge detection using anisotropic diffusion, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1990; 12:629–639.

[6] Rekeczky Cs. CNN architectures for constrained diffusion based locally adaptive image processing, *Int. J. Circ. Theor. Appl.* 2002; 30:313–348.

[7] Biktashev V. N, Krinsky V. I, and Haken H. A Wave Approach to Pattern Recognition (with Application to Optical Character Recognition), *International Journal of Bifurcation and Chaos*, Vol. 4, No 1 (1994), 193-207.

[8] Krinsky V. I, Biktashev V. N, and Efimov I. R. Autowave Principles for Parallel Image Processing, *Physica D49* (1991), 247-253.

[9] Rosenfeld A, and Pfaltz J. Distance Functions in Digital Pictures, *Pattern Recognition*, Vol. 1 (1968), 33-61.

[10] Huttenlocher D, Klanderman G, and Rucklidge W. Comparing images using the hausdorff distance, *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol 15(9) (1993), 850–863.

[11] Huttenlocher D, Kedem, K, Efficiently Computing the Hausdorff Distance for Point Sets under Translation, *Proceedings of the Sixth ACM Symposium on Computational Geometry* (1990), 340-349.

[12] Haken H. *Synergetics*, Springer, Berlin, 1978

[13] Haken H. Synergetics: From Pattern Formation to pattern Analysis and Pattern Recognition, *International Journal of Bifurcation and Chaos*, Vol. 4, No. 5 (1994), 1069-1083.

[14] Krinsky V, ed., *Self-Organization. Autowaves and Structures Far from Equilibrium. Synergetics*, Vol. 28, Springer, Berlin, 1984

[15] Kass M, Witkin A, and Terzopoulos D. Snakes: Active Contour Models, *International Journal of Computer Vision* (1988), 321-331.

[16] Sethian J. A. Curvature and Evolution of Fronts, *Comm. in Mathematical Physics*, Vol 101 (1985), 487-499.

[17] Sethian J. A. Numerical Algorithms for Propagating Interfaces: Hamilton-Jacobi Equationsand Conservation Laws, *Journal of Differential Geometry*, Vol. 31 (1990), 131-161.

[18] Malladi R, Sethian J. A, and Vemuri B. C. Shape Modeling via Front Propagation: a Level Set Approach, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol 17, No 2. (1995)

[19] Osher S, and Sethian J. A. Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulation, *Journal of Computational Physics*, Vol. 79 (1998), 12-49.

[20] Sapiro G. Contrast Enhancement via Image Evolution Flows, *Graphical Models and Image Processing* Vol 59, No 6, November (1997), 407-416.

[21] Chorin A. J. Flame Advection and Propagation Algorithms, *Journal of Compatational Physics*, Vol 35 (1980), 1-11.

[22] Dorf R. C. *The Electrical Engineering Handbook*, CRC Press, 1993.

[23] Rekeczky Cs, Chua L. O. Computing with Front Propagation: Active Contour and Skeleton Models in Continuous-time CNN, *Journal of VLSI Signal Processing Systems*, Vol. 23, No. 2/3, November-December (1999), 373-402.

[24] Linan G, Espejo S, Dominguez-Castro R, and Rodriguez-Vasquez A. ACE4K: an analog I/O 64x64 visual microprocessor chip with 7-bit analog accuracy, *Int. Journal of Circuit Theory and Applications*, Vol 30, No. 2/3, March-June 2002, pp. 89-116.

[25] Szatmári I, Rekeczky Cs, and Roska T, A Nonlinear Wave Metric and its CNN Implementation for Object Classification, *Journal of VLSI Signal Processing*, Special Issue: Spatiotemporal Signal Processing with Analogic CNN Visual Microprocessors, Vol 23, No 2/3, November-December (1999), 437-447.

[26] Szatmári I, Schultz A, Rekeczky Cs, Kozek T, Roska, T, and Chua L., O. Morphology and Autowave Metric on CNN Applied to Bubble-Debris Classification, *IEEE Transaction on Neural Networks*, Vol. 11, No. 6, November (2000), 1385-1393.

[27] Hillier D, Vilarino D, and Rekeczky Cs. Topographic Cellular Active Contour Techniques: Theory, Implementations and Comparisons, *Circuit Theory and Applications*, special issue, 2005, (in this issue)