Natural Numerical Networks on Directed Graphs in Satellite Image Classification

Karol Mikula[1,2], Michal Kollár[1,2], Aneta A. Ožvat[1,2](✉), Mária Šibíková[3], and Lucia Čahojová[3]

[1] Department of Mathematics, Slovak University of Technology in Bratislava,
Radlinskeho 11, Bratislava 810 05, Slovakia
karol.mikula@gmail.com, aneta.ozvat@stuba.sk

[2] Algoritmy:SK s.r.o., Šulekova 6, 81106 Bratislava, Slovakia

[3] Plant Science and Biodiversity Center, Slovak Academy of Sciences,
Institute of Botany, Dubravska cesta 9, Bratislava 845 23, Slovakia

Abstract. Natural numerical networks on directed graphs as a new supervised deep learning PDE-based classification algorithm are proposed in this work. The Natural numerical network (NatNet) is based on a forward-backward diffusion model, where the points of the given clusters are attracted together by the forward diffusion, and in contrast, the backward diffusion repulses points of different clusters from each other. First, the network is trained on the labelled data to achieve the highest possible accuracy on the learning dataset. Then, the method is applied to the classification of Sentinel-2 satellite optical data to automatically identify the protected oak habitat in Western Slovakia due to its threatened status. To that goal, the relevancy map, one of the outputs of the Natural numerical network, is created efficiently; its construction is significantly speed up thanks to the new NatNet formulation on directed graphs.

Keywords: Forward-backward diffusion · Partial differential equations on graph · Numerical methods · Data classification

1 Introduction

A new concept of Natural numerical network (NatNet) on directed graphs is introduced in this paper. The NatNet as a new supervised deep learning PDE-based classification method was presented in [13]. It introduces a forward-backward diffusion model on undirected graphs and its numerical discretisation to get the classification algorithm. The forward diffusion attracts the points of the given clusters together while the backward diffusion repulses the points of different clusters from each other. Such approach is inspired by the recent ODE and PDE-based deep learning methods from [4,10], and attraction-repulsion strategies are used also in other clustering applications such as high-dimensional data visualization [3]. For an interesting overview of PDE-based and variational approaches

Supported by grants APVV-19-0460, VEGA 1/0436/20 and ESA contract 4000140486/23/NL/SC/rp.

L. Calatroni et al. (Eds.): SSVM 2023, LNCS 14009, pp. 339–351, 2023.
https://doi.org/10.1007/978-3-031-31975-4_26

on graphs for high-dimensional data classification we refer also to [2]. In [10], the relation between a successful deep learning model, the so-called Residual Neural Network (ResNet) [4,11], and the numerical solution of the system of ordinary differential equations using the forward Euler method is shown. The authors then designed parabolic and hyperbolic networks for deep learning classification based on corresponding types of partial differential equations. The NatNet [13] uses another type of PDE, the nonlinear forward-backward diffusion equation, which is natural for supervised deep learning classification.

In [13], the NatNet supervised deep learning classification method was applied to the Sentinel-2 multispectral optical data to obtain in an automated way a spatial appearance of Natura 2000 [7] protected habitats in Slovakia and along the Danube river in Central and South-Eastern Europe. There exist various recent studies dealing with the classification of land cover classes by using the multispectral satellite images, and all standard classification algorithms, such as the Random Forest, k-Nearest Neighbour, and Support Vector Machine, were used, reported, and compared [5,16,18]. In general, the standard methods provide a meaningful classification of land cover classes, but a higher accuracy, exceeding 90%, is only reached, when simple categories are identified, e.g. water bodies, meadows, forests, fields and urbanised areas. They fail to classify accurately different forest types using purely the Sentinel-2 multispectral optical bands information, in such case, the classification accuracy drops to about 80% [18]. On the other hand, forest habitats defined in the Natura 2000 classification system are complex plant communities with variable species composition, and commonly one habitat cannot be defined by one dominant tree species. Providing classification of such detailed classes has been a challenging task with high demand on new reliable classification methods based on widely available multispectral satellite data. The NatNet designed in [13] provides a first successful approach to solve this task.

The trained NatNet can be represented by the forward-backward diffusion on undirected graphs [13], but for the classification of a large number of new observations, the directed graph concept is useful due to its computational efficiency. The NatNets classification output is given, together with the cluster membership of any pixel of the satellite image, by the so-called relevancy map. The relevancy map is a greyscale image giving information on the relevancy of cluster membership for every pixel and it has the same dimension as the satellite image. In order to create the relevancy map we have to let evolve the new observation by the dynamics of the trained network and classify it. In the undirected graph approach, we have to do it one by one solving as many times a small system of equations as there are pixels in the relevancy map. By using the directed graph concept, all pixel values in the relevancy map are computed at once by solving one system of equations which speeds up the computation many times, proportionally to the number of graph vertices in the learning dataset, and makes it possible to include the relevancy map computation directly into the NaturaSat software [15]. In this paper, we explain the NatNet on directed graphs in detail and apply it to the classification of protected oak habitats in Western Slovakia by Sentinel-2 satellite images.

2 Natural Numerical Network (NatNet)

2.1 Mathematical Model

A directed graph G consists of a non-empty finite set $V(G)$ of elements called vertices and a finite set $A(G)$ of ordered pairs of distinct vertices called arcs or directed edges [1]. We denote the number of vertices of the directed graph G by N_V. In general, for an arc $(u, v) = e_{uv}$, the first vertex u is its tail, and the second vertex v is its head, which means that the arc (u, v) leaves u and enters v. The head and tail of an arc are its end-vertices [1]. In the sequel, we will use a semi-complete directed graph G, where semi-complete means that there exists an arc between every pair of vertices $V(G)$.

Let us have the function $X : G \times [0, T] \to \mathbb{R}^k$ representing the Euclidean coordinates $X(v, t) = (x_1(v, t), \ldots, x_k(v, t))$ of the vertex $v \in V(G)$ in time $t \in [0, T]$. The index k represents a dimension of the feature space $\mathbb{R}^k$. A diffusion of the function $X(v, t)$ on the directed graph G is formulated as a partial differential equation (PDE)

$$\partial_t X(v, t) = \nabla \cdot (g \nabla X(v, t)), \qquad v \in V(G), \qquad t \in [0, T], \tag{1}$$

where g represents the diffusion coefficient, see also [9]. We consider Eq. (1) together with an initial condition $X(v, 0) = X^0(v)$, $v \in V(G)$. The boundary conditions are not necessary to prescribe because in our model diffusion occurs between all vertices of the semi-complete directed graph G.

We consider the diffusion coefficient g depending on the distance between two vertices v and u of the directed graph G. It will give a nonlinear diffusion model on the directed graph. We consider Eq. (1) with diffusion coefficient g in the form

$$g(e_{uv}) = \varepsilon(e_{uv}) \frac{1}{1 + \sum_{i=1}^{k} (K_i \ l_i^2(e_{uv}))}, \qquad K_i \geq 0, \quad i = 1, \ldots, k, \tag{2}$$

where K_i represents weights for each coordinate $l_i(e_{uv})$, $i = 1, \ldots, k$, of the vector

$$\begin{aligned} l(e_{uv}) = (l_1(e_{uv}), \ldots, l_k(e_{uv}))^T = X(v, \cdot) - X(u, \cdot) = \\ = (x_1(v, \cdot) - x_1(u, \cdot), \ldots, x_k(v, \cdot) - x_k(u, \cdot))^T, \qquad v, u \in V(G), \end{aligned} \tag{3}$$

and allow us to control the diffusion speed in each direction of the k-dimensional feature space. If the sum in the diffusion coefficient is large, the diffusion coefficient g is close to 0, which means that the diffusion process will be slow and the points do not diffuse by averaging. If the sum in the diffusion coefficient is small, the diffusion coefficient is close to 1, the diffusion process is faster, and the points are moving fast by diffusion.

The value of $\varepsilon(e_{uv})$ in the diffusion coefficient depends on the type of diffusion applied in each arc between each pair of vertices. For applying the forward diffusion, we choose $\varepsilon(e_{uv})$ as a positive constant, in all computations presented in this paper $\varepsilon(e_{uv}) = 1$. In our application, forward diffusion causes a moving, and thus clustering of points together. On the other hand, backward diffusion is represented by a negative diffusion coefficient, $\varepsilon(e_{uv})$ is a small negative value, and in our application it gives a repulsion of the points belonging to different clusters. Such a model with a combination of forward and backward diffusion is a suitable tool for supervised learning.

In the case of two directed edges between two vertices u and v, we consider $g(e_{uv}) = g(e_{vu})$.

In Fig. 1 we illustrate the behaviour of the model (1)–(2) on two given clusters where by light blue arrows we plot some of the links of forward diffusion and by red arrows some of the links of backward diffusion. This figure depicts the basic features and behaviour of the Natural network. The points inside a given cluster are attracted by forward diffusion, while there is a repulsion of points of different clusters by backward diffusion. The model allows the directed graph to have arcs with the same end-vertices, and in this figure all vertices have two directed edges, which means that the diffusion influence occurs in both directions. It is important to realise that the model does not allow pairs of arcs with the same tail and the same head (parallel arcs) or arcs whose heads and tails are the same vertex (loops).

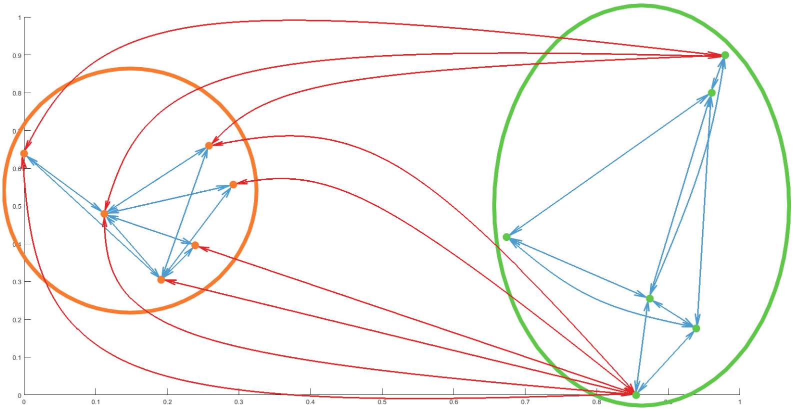

Fig. 1. Randomly generated 2D points in two clusters and some links of forward diffusion (light blue arrows) inside the clusters, and some links of backward diffusion (red arrows) between points from different clusters. (Color figure online)

Furthermore, in Fig. 2 we illustrate the situation that arises in the supervised learning and application phases of the classification method when a new observation is added to the network. Only forward diffusion is applied to all links of the vertex representing the new observation, which means that the new vertex is the head for directed edges connecting it with all other vertices. The forward diffusion links are depicted by the dark blue arrows connecting the new observation (black square) to every other point. Thus, this new observation is attracted by a certain diffusion speed to all existing clusters, which themselves are subject to the forward-backward diffusion as described before. The dynamics of the network decides on the cluster membership of the new observation.

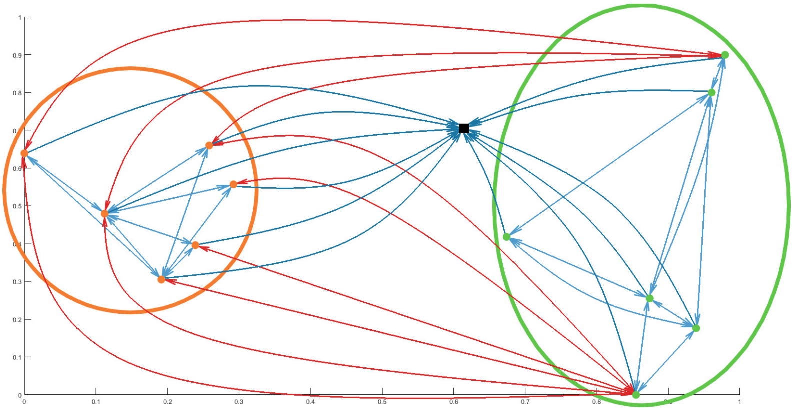

Fig. 2. Randomly generated 2D points in two clusters with one new observation (black square). The light blue arrows represent some of the forward diffusion links inside the clusters. The forward diffusion links from all other points to the new observation are represented by the dark blue arrows. The red arrows represent the links of backward diffusion between points from different clusters. (Color figure online)

We can reduce the influence of forward diffusion on the new observation vertex $w \in V(G)$ by using the diffusion coefficient in the form

$$g(e_{vw}) = \max(\varepsilon(e_{vw})\frac{1}{1+\sum_{i=1}^{k}(K_i\ l_i^2(e_{vw}))} - \delta,\ 0), \qquad \varepsilon(e_{vw}) > 0 \qquad (4)$$

at all directed edges entering w, where δ is a parameter of the size of the "diffusion neighbourhood". The aforementioned modification causes that only the points for which the diffusion coefficient is larger than δ, attract new observation point w in the classification process.

2.2 Numerical Discretisation

To discretise the equation (1), we use *i)* the balance of diffusion fluxes (inflows and outflows) at each vertex $v \in V(G)$ and *ii)* the approximation of the diffusion flux to the vertex v along its arcs.

Let us define the diffusion flux approximation, which depends on the difference of the values of the function X at the vertices v and u, as

$$\mathcal{X}(v, e_{uv}, t) = g_{e_{uv}}(X(u,t) - X(v,t)), \tag{5}$$

for each directed edge (u, v), where $g_{e_{uv}}$ represents the diffusion coefficient on the directed edge (u, v). If $\mathcal{X}(v, e_{uv}, t) > 0$, it represents the diffusion inflow, while if $\mathcal{X}(v, e_{uv}, t) < 0$, it represents the diffusion outflow. Then the balance of diffusion fluxes at the vertex v is expressed by the equation

$$\partial_t X(v,t) = \sum_{\substack{u \in V(G)\,:\\ e_{uv} \in A(G)}} \mathcal{X}(v, e_{uv}, t). \tag{6}$$

By the substitution of the approximation of the diffusion flux (5) into the balance equation (6), we obtain the so-called "graph-Laplacian" with a measure equal to 1 which is a common choice in the graph theory. A more detailed description can be found in the paper [13].

For the time discretisation, we use the semi-implicit approach, see e.g. [14]. The finite difference method is used for the approximation of time derivative. Since the diffusion coefficient $g_{e_{uv}}$ at the directed edge e_{uv} can depend on the unknown quantity X, see (2) and (4), and thus can change over time, we take its value from the previous time step. In the case of classification of the data from the k-dimensional feature space, we get in each time step k systems of linear equations

$$\begin{aligned}(1 + \tau \sum_{\substack{u \in V(G)\,:\\ e_{uv} \in A(G)}} g_{e_{uv}}^{n-1}) x_i^n(v) - \tau \sum_{\substack{u \in V(G)\,:\\ e_{uv} \in A(G)}} g_{e_{uv}}^{n-1} x_i^n(u) = x_i^{n-1}(v),\\ i = 1, \dots, k, \quad v \in V(G),\end{aligned} \tag{7}$$

which are interconnected by the diffusion coefficient $g_{e_{uv}}^{n-1}$, which depends on all $x_i^{n-1}(v)$, $x_i^{n-1}(u)$, $i = 1, \dots, k$ and can be written in the form

$$g_{e_{uv}}^{n-1} = \varepsilon(e_{uv}^{n-1}) \frac{1}{1 + \sum_{i=1}^{k} (K_i \; l_i^2(e_{uv}^{n-1}))}, \qquad K_i \geq 0. \tag{8}$$

This system of equations is represented by a full matrix and, as we have said before, for considered semi-complete directed graphs, it is not necessary to define any boundary condition.

The Eqs. (7)–(8) represent our network, where the points inside the given clusters are moving together, and the clusters themselves are keeping away. The illustration of that dynamic can be found in the paper [13]. In the learning phase, and also in the application phase, the dynamics is modified in such a way that all other points are moving by (7)–(8) but for the new observation $w \notin C_i$, $i \in \{1, \dots, N_C\}$, the diffusion coefficient is set to

$$g_{e_{uw}}^{n-1} = \max(\varepsilon(e_{uw}^{n-1}) \frac{1}{1 + \sum_{i=1}^{k} (K_i \; l_i^2(e_{uw}^{n-1}))} - \delta, \; 0), \tag{9}$$

$\varepsilon(e_{uw}^{n-1}) \geq 0,\ K_i \geq 0,\ \delta > 0$ are given constants, see also (4). It is crucial to realise that the vertex w is the head of the directed edges connecting it with the neighbouring vertices, so the directed edges enter w, and there do not exist arcs leaving w.

To clearly conceive the role of the diffusion coefficient $g_{e_{uv}}$ in the classification algorithm, we describe and schematically show the system matrix of the semi-implicit scheme. As an example, we use the directed graph from Fig. 1, where the number of clusters is $N_C = 2$. For that directed graph, the matrix (10) is constructed by (7)–(8) representing the fundamental dynamics of the NatNet. The matrix contains four blocks. The first block B_1 corresponds to the vertices from the first cluster, and because the vertices are from the same cluster, only forward diffusion on their arcs is used. The vertices influence each other in the same way, so the first block is symmetric $B_1 = B_1^T$. Similarly, the fourth block B_4 corresponds to the vertices of the second cluster, and again the forward diffusion is applied to their arcs in a symmetric way, thus $B_4 = B_4^T$. Moreover, the diagonal of these blocks is positive while out of diagonal elements are negative. The entries in the second B_2 and the third B_3 block are given by the values of diffusion coefficient on the directed edges between the vertices of different clusters (in Fig. 1 red arrows). The backward diffusion with a small negative value of $\varepsilon(e_{uv}) = -10^{-2}$ is applied to these directed edges and cause the small positive values in the second and third block. The blocks are symmetric to each other $B_2 = B_3^T$, which means that the vertices from the different clusters affect themselves symmetrically.

$$\left(\begin{array}{c|c} B_1 & B_2 \\ \hline B_3 & B_4 \end{array}\right) \tag{10}$$

Modifying the diffusion coefficient for the new observation by using (9) in the directed graph approach changes the matrix of the system in the following way. Let us consider the same example as in the description of the matrix for fundamental network dynamics but with one new observation added to the directed graph, see Fig. 2. Adding the new observation in the directed graph enlarges the size of the matrix (10) by one row and column. Due to one-sided arcs between new observation and other vertices the system matrix is non-symmetric. Consequently, the matrix (11) for the directed graph enriched with a new observation differs only in the last row and column from the system matrix (10). The values in the row $\boldsymbol{N}$ for the new observation are calculated by (9), which means that if the new observation has some vertices in the "δ-diffusion neighbourhood", the non-zero value of diffusion coefficient is set in the intersection of the row of the new observation and the column of that vertex in the matrix. In the last column corresponding to the new observation there are only zero values because the new observation is not affecting any other vertex of the directed graph.

$$\left(\begin{array}{c|c|c} B_1 & B_2 & \multirow{2}{*}{$\mathbf{0}$} \\ \cline{1-2} B_3 & B_4 & \\ \hline \multicolumn{3}{c}{\boldsymbol{N}} \end{array}\right) \tag{11}$$

The aforementioned change in the Natural numerical network concept leads to a significant reduction in CPU time. In paper [13], the classification process is run for any new observation sequentially, which means that the new observations are added to the undirected graph one by one and dynamics of the network are run for every new observation independently. In theory, the CPU time for such an approach is proportional to $P(N_V + 1)$, where P are the number of new observations. With a new concept of the NatNets on directed graphs, we can classify all new observations simultaneously because the new observations do not affect any other vertex of the graph. In this case, theoretically, the CPU time is proportional to $N_V + P$, and the speed-up of the computation is thus proportional to N_V. To test the speed up in practice, we calculated the relevancy map of dimension 2000×2000, so we have $4 \cdot 10^6$ new observations to be classified by NatNet. For the approach from [13], we obtained CPU time 801 s for such large-scale classification task, while for the new concept presented in this paper, we obtained CPU time 25.2 seconds, which means the speed up 31.78 times. The computations were performed on the machine with AMD Ryzen Threadripper PRO 3975WX 32-Cores processor, RAM 256 GB DDR4 and using OpenMP parallelisation in both approaches.

A histogram stopping criterion is applied in the network dynamics. It is based on the calculation of the number of occurrences (frequency) of evolving points in prescribed spatial cells in every time step. For more details about the stopping criterion, see the paper [13].

2.3 Construction of Relevancy Maps

The relevancy map is a grayscale image with the same size as the images from satellite optical channels. The square $A(p, r)$ is created in every image pixel p with Chebyshev radius r. For each p of the square $A(p, r)$, the statistical characteristics (the mean, the standard deviation, the minimum value and the maximum value) are computed and considered and added in the directed graph G as new observation $w(p)$. Every new observation $w(p)$ is classified by the Natural network and its relevancy coefficient $R(w(p))$ is computed. Finally, depending on the Chebyshev radius r of the square $A(p, r)$, the relevancy map M_i^r, $i = 1, \ldots, N_C$, is defined for every cluster C_i, $i = 1, \ldots, N_C$, in every pixel p as follows

$$\begin{aligned} M_i^r(p) &= R(w(p)), && \text{if } w(p) \text{ is classified into } C_i, \\ M_i^r(p) &= 0, && \text{if } w(p) \text{ is not classified into } C_i. \end{aligned} \tag{12}$$

The definition of the relevancy coefficient $R(w(p))$ is given by

$$R(w(p)) = \mathcal{L}(1 - \frac{l_1(w(p))}{l_1(w(p)) + l_2(w(p))}), \quad \mathcal{L}(x) = \frac{1}{1 + e^{\lambda(0.5 - x)}} \tag{13}$$

and it depends on the distance between the new observation and the centroid of the cluster $\mathcal{C}_a(w(p))$ to which it is assigned by the network dynamics,

$$l_1(w(p)) = \mid X(w(p), 0) - \mathcal{C}_a(w(p)) \mid , \tag{14}$$

and the average distance of the new observation to all other cluster centroids,

$$l_2(w(p)) = \frac{1}{N_c - 1} \sum_{\substack{i=1 \\ i \neq a}}^{N_c} \mid X(w(p), 0) - \mathcal{C}_i(w(p)) \mid \ . \tag{15}$$

The relevancy coefficient $R(w(p))$ has the nonlinear character and is assigning the values close to 1 for the new observation $X(w, 0)$ close to the centroid of the cluster to which it is classified, while in the other case it is low.

3 Natural Numerical Network in Nature Protection

The Natural numerical network is applied in ecology and nature conservation tasks. It is a suitable tool for identification and classification of protected forest habitats by using the remote sensing. In our experiments we are focused on the data from the Sentinel-2 satellite of the European Space Agency (ESA) [8]. The Sentinel-2 offers optical imagery in high spatial resolution with 17 channels. In addition to these 17 channels, we calculate one more, the normalized difference vegetation index (NDVI) [6] which quantify the vegetation. Thus, for the feature space construction, we use 18 channels in which we compute the statistical characteristics in a prescribed image subarea A. Therefore, the feature space is the 72-dimensional Euclidean space, i.e. $k = 72$.

The classification by NatNets is applied to two groups of forest habitats, QC forest - segments dominated by Quercus cerris (habitats 91M0 Pannonian-Balkanic turkey oak-sessile oak forests and 91I0 Euro-Siberian steppic woods with Quercus spp.) and QP forest - segments with the dominance of Quercus petraea (habitat 91G0 Pannonic woods with Quercus petraea and Carpinus betulus). The habitats 91M0, 91I0, and 91G0 are part of Natura 2000 protected network [7]. The motivation for the classification and identification of such oak forests is that the forests of 91M0 and 91I0 habitats are very endangered in Slovakia. The wood of the Quercus cerris (turkey oak), which is dominant in these habitats, is considered to be of lower quality compared to other oak species in Slovakia, and therefore the turkey oak was often eliminated in favour of other Quercus species.

3.1 Training of the Network

The vegetation scientists use automatic segmentation methods in NaturaSat software [15] to segment 42 areas of QC and QP forests in Western Slovakia, see Fig. 3 for areas examples of QC segmented areas. We denote the segmented areas as S_i, where $i = 1, \ldots, N_S$, $N_S = 42$, and in each segmented area, we choose randomly a square $A_i = A(p_i, r)$ centred in a pixel $p_i \in S_i$. The values for Chebyshev radius r are equal to 5 for large segmented areas, while for small areas, r can be smaller, equal to 4 or 3. The statistical characteristics are computed for every square A_i. The statistical characteristics of squares A_i represent the vertices of the initial network directed graph G. Since we have multi-dimensional feature space, we apply the Principal Component Analysis (PCA) [12,17] to reduce the data dimension but retain the maximum amount of information in the data. We observed experimentally, that using the first two principal components is sufficient, further coordinates do not help to differentiate clusters and can be omitted. Thus the dimension of the problem is reduced to $k = 2$ which gives at the same time the computationally tractable task and sufficiently accurate classification, exceeding 95%, in the training phase of the model. Because we have two types of forests, we have two clusters in the classification, $N_C = 2$, and each vertex in the directed graph is labelled by the cluster number to which its segmented area belongs.

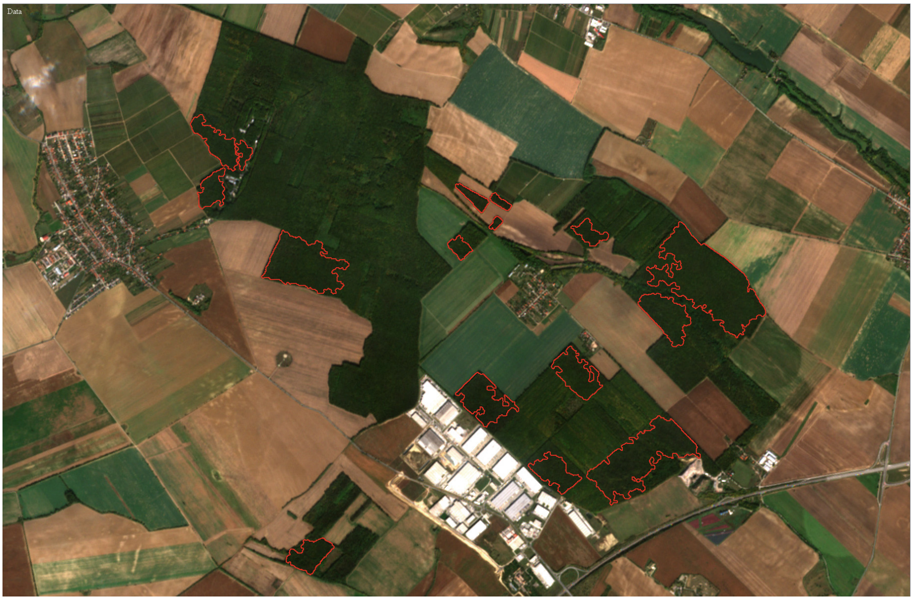

Fig. 3. The subregion of Western Slovakia with segmented areas of protected QC forests (red curves). (Color figure online)

Network training aim is to tune the parameters of the model (7)–(9) to achieve the highest possible classification accuracy for the learning dataset. To attain that aim, we subsequently remove the cluster label from each vertex of the directed graph G, representing the learning dataset, and set it as the new observation. Then we classify it using the NatNet. We analyse the results and choose the model parameters with the highest success rate N_B/N_V, where N_B is a number of correctly classified observations.

Now let us consider the directed graph G having $N_V = 42$ vertices described above and denote it LDS42. We run the training of the network on the LDS42 dataset, the results are shown in the first row of Table 1. We achieved the success rate of 37/42. Since we have randomly chosen the squares $A(p_i, r)$, it is not the optimal approach. To increase the success rate, we adjust the learning dataset by a spatial shifting of representative squares inside the segmented areas S_i based on the relevancy maps computed using LDS42. We try to find the new square $A(p_i, r)$ for each segmented area S_i such that $M_a^r(p_i) >> 0$ for a new square centre $p_i \in S_i$ by analysing the relevancy map M_a^r of the cluster to which the segmented area S_i belongs. If we find a pixel p_i with high relevancy, we can construct the new square $A(p_i, r)$ and replace the randomly chosen square from LDS42 with the new one. In this manner, we construct the adjusted learning dataset LDS42adj and run the training of the network again. The result of the training is shown in the second row of Table 1. We achieve the success rate of $40/42 = 0.9524$, which is sufficiently high and allows us to use the trained NatNet for the identification of oak forests in Western Slovakia outside the areas used in the training of the network.

Table 1. The results of the learning phase on datasets LDS40 and LDS40adj.

Dataset name	Correctly classified	Incorrectly classified	Outliers	Success rate
LDS42	37	5	1	88.09%
LDS42adj	40	1	1	95.24%

3.2 Application of the Trained Network

By a successfully trained NatNet, we can classify satellite images and identify areas of target habitat in the examined territory. In our case, the target habitats are that dominated by Quercus cerris (turkey oak) due to their endangered status in Slovakia. The relevancy maps are computed for that purpose. We work with two habitats, thus we obtain two relevancy maps. Figure 4 depicts the area of Martinsky les special protected area with the segmented areas of QC forests (red curves). On the left part in Fig. 4, there is the image from the Sentinel-2 satellite, and on the right part, there is the relevancy map for the QC forests. The relevancy map shows bright colours in the interior of the segmented areas, which means high relevancy coefficient in the pixels and reflects the correct classification of the segmented area. We can notice that there is also apparent bright colour outside the segmented areas, which leads to field reviewal by vegetation scientists. The result of the field research is that the NatNet correctly classified a given territory. There are further areas of 91M0 and 91I0 habitats in such territory validated by fields visit and comparing them with forestry maps. The second relevancy map for QP forests is depicted in Fig. 5. This relevancy map shows the regions of appearance of QP forests in bright colour pixels, thus having a high relevancy coefficient. When we are focused on the segmented curves of the QC forests, we can conclude that the interior is quite dark, which expresses low

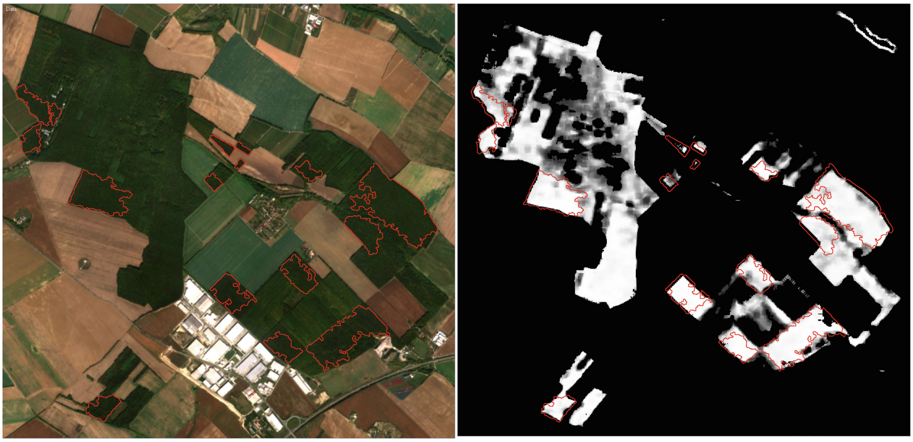

Fig. 4. The segmented areas of QC forests (red curve) plotted on the Sentinel-2 image (left) and on the relevancy map for QC forests (right). (Color figure online)

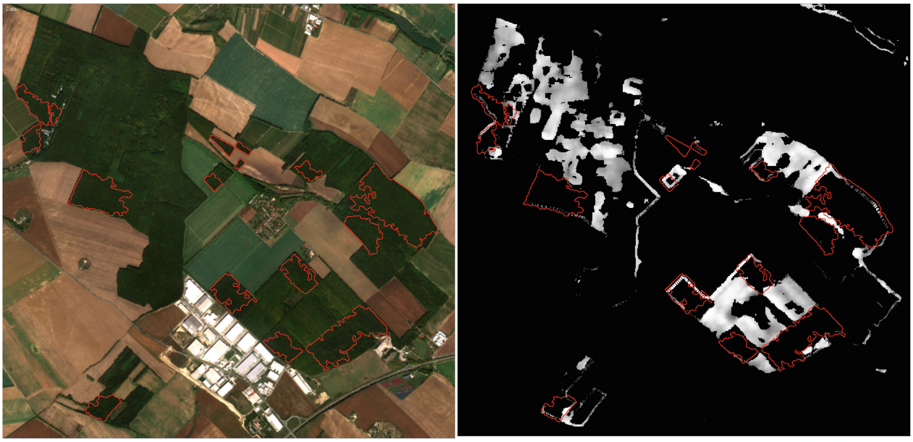

Fig. 5. The segmented areas of QC forests (red curve) plotted on the Sentinel-2 image (left) and on the relevancy map for QP forests (right). (Color figure online)

or no relevancy of the occurrence of the QP forest. The Natura 2000 habitats are complex compositions of various types of species, and it is impossible to have a homogeneous relevancy map for one habitat. Nevertheless, we can observe that the relevancy map for the QC forests and the QP forests complement each other.

References

1. Bang-Jensen, J., Gutin, G.R.: Digraphs: Theory, 2nd edn. Algorithms and Applications. Springer-Verlag, London (2010)
2. Bertozzi, A.L., Flenner, A.: Diffuse interface models on graphs for classification of high dimensional data. SIAM Rev. **58**(2), 293–328 (2016). https://doi.org/10.1137/16M1070426
3. Böhm, J.N., Berens, P., Kobak, D.: Attraction-repulsion spectrum in neighbor embeddings. J. Mach. Learn. Res. **23**(95), 1–32 (2022)
4. Chang, B., Meng, L., Haber, E., Ruthotto, L., Begert, D., Holtham, E.: Reversible architectures for arbitrarily deep residual neural networks. In: 32nd AAAI Conference on Artificial Intelligence, AAAI 2018. vol. 32, pp. 2811–2818 (April 2018), https://ojs.aaai.org/index.php/AAAI/article/view/11668
5. Cheng, K., Wang, J.: Forest type classification based on integrated spectral-spatial-temporal features and random forest algorithm-a case study in the qinling mountains. Forests **10**(7), 559 (2019). https://doi.org/10.3390/f10070559
6. Earth Observing System: Normalized difference vegetation index. https://eos.com/ndvi/ (2020)
7. European Environmental Agency: The natura 2000 protected areas network. https://www.eea.europa.eu/themes/biodiversity/natura-2000/the-natura-2000-protected-areas-network (2020)
8. European Space Agency: Sentinel 2. https://sentinel.esa.int/web/sentinel/missions/sentinel-2/data-products (2020)
9. Friedman, J., Tillich, J.P.: Calculus on graphs. CoRR cs.DM/0408028 (2004)
10. Haber, E., Ruthotto, L.: Stable architectures for deep neural networks. Inverse Probl. **34**(1) (2018). https://doi.org/10.1088/1361-6420/aa9a90
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (June, 2016). https://doi.org/10.1109/CVPR.2016.90
12. Jolliffe, I.T.: Principal Component Analysis. Springer-Verlag, New York, 2nd edn. (2002). https://doi.org/10.1007/b98835
13. Mikula, K., Kollár, M., Ožvat, A.A., Ambroz, M., Čahojová, L., Jarolímek, I., Šibík, J., Šibíková, M.: Natural numerical networks for natura 2000 habitats classification by satellite images. Appl. Math. Model. **116**, 209–235 (2023). https://doi.org/10.1016/j.apm.2022.11.021
14. Mikula, K., Ramarosy, N.: Semi-implicit finite volume scheme for solving nonlinear diffusion equations in image processing. Numer. Math. **89**(3), 561–590 (2001)
15. Mikula, K., et al.: Naturasat-a software tool for identification, monitoring and evaluation of habitats by remote sensing techniques. Remote Sens. **13**(17) (2021). https://doi.org/10.3390/rs13173381
16. Noi, P.T., Kappas, M.: Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using sentinel-2 imagery. Sensors (Switzerland) **18**(1), 18 (2018). https://doi.org/10.3390/s18010018
17. Rencher, A.: Methods of Multivariate Analysis. Wiley-Interscience (02 2002). https://doi.org/10.1002/0471271357
18. Waśniewski, A., Hoscilo, A., Zagajewski, B., Mouketou-Tarazewicz, D.: Assessment of sentinel-2 satellite images and random forest classifier for rainforest mapping in gabon. Forests **11**(9), 941 (2020). https://doi.org/10.3390/f11090941