Seabed Sediment Classification of Side-scan Sonar Data Using Convolutional Neural Networks

Tim Berthold*, Artem Leichter*, Bodo Rosenhahn†, Volker Berkhahn* and Jennifer Valerius‡
*Institute for Risk and Reliability
Leibniz University of Hanover
Email: {berthold,leichter,berkhahn}@irz.uni-hannover.de
†Institute for Information Processing
Leibniz University of Hanover
Email: rosenhahn@tnt.uni-hannover.de
‡Federal Maritime and Hydrographic Agency
Hamburg
Email: jennifer.valerius@bsh.de

***Abstract*—Spatially high-resolution information on the seabed sediment is import for many applications in the fields of oceanic engineering, coastal engineering, habitat mapping, and others. The seabed sediment is typically described by information based on the grain-size distribution, which are derived from sediment samples collected from the seafloor. For covering large areas side-scan sonar systems are typically used, which measure the backscatter intensity. From this information the sediment types can be derived. We propose a model for the automatic sediment type classification of the side-scan sonar data, which is based on convolutional neural networks (CNN). A big advantage of CNN is that they provide an end-to-end training: the CNN derives appropriate features automatically during the training process, which are then used for classification. The approach is based on a patch-wise classification using ensemble voting. The approach is evaluated on real world side-scan sonar data, which have been labelled using four classes (fine, sand, coarse, and mixed sediment) by experts. While the prediction of sand achieves an accuracy of 83 percent, the accuracy for fine sediment is very poor (11 percent).**

I. Introduction

Spatially high-resolution information on the seabed sediment has recently become more and more important for many applications in the fields of oceanic engineering, coastal engineering, habitat mapping, and monitoring of the marine environment [1]. In order to describe the surface sediment samples are typically collected from the seabed by a grab in the region of interest. The composition of a sediment sample is then described in terms of a grain-size distribution, which is obtained from some grain-size analysis. Although the measured information is usually of good quality this kind of measurement is very time-consuming and can be performed pratically only at selected locations. In order to get information on the seabed sediment covering a large area a common approach is to use side-scan sonar systems. This technique is based on measurements of the intensity of transmitted acoustical signals that have been reflected by the seafloor. Since the backscatter intensity is amongst others influenced by the composition of the surface this information can be used to infer information on the seabed sediment [2, p. 16]. A manual interpretation can become extensive for large areas so that an automatic interpretation is desirable.

A lots of approaches ([3], [4], [5], [6]) have been developed in the last decades to address this problem, where most of them are based on the classical classification system: first, a set of characteristic parameters (features) have to be defined, which can then be separated by a (trainable) classifier. Here, the main problem is to find appropriate features, which is a time consuming trial and error process in the most cases. Convolutional neural networks (CNN) are a special type of neural networks that are able to extract the features during the learning process and have been applied successfully to many pattern recognition tasks in the last years (for a summary of the successes see [7] for example). CNN seems to be a promising approach for the classification of side-scan sonar data as well.

This paper presents an approach for the automatic classification of seabed sediment on the basis of side-scan sonar images using CNN. Extensive side-scan sonar data is available from the project Aufmod [8] that has been labled by experts. In section II the available data are described first. The structure of CNN and the overall approach is described in section III. The application of the model to the real world data as well as the results are presented in section IV followed by a discussion of the results. Section V summarized the contribution of this paper and addresses future work.

II. Data

In the years between 2010 and 2013 the AufMod project was carried out having the goal to setup models for the analysis of the long term morphodynamic evolution in the German Bight [8]. In order to obtain spatially high resolution information concerning the seabed sediment several measurement campaigns were undertaken within this project in specific regions. These measurements have been made available by the

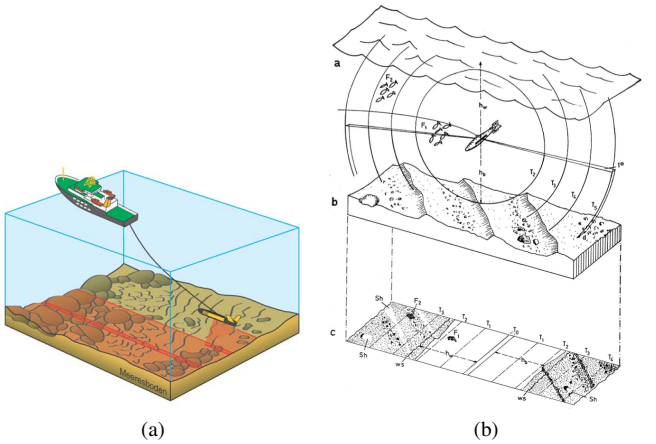
978-1-5386-2726-6/17/$31.00 ©2017 IEEE

BSH[1] and are used for first investigations dealt with in this paper. The selected investigation area for this paper is a part of the tidal race called Hever which is located in the wadden sea in the north east of the German Bight, south to the island Pellworm. In this area all data collection and processing during the AufMod project have been performed by FTZ Büsum. The given data consist of sonary images that have been produced by side-scan sonar systems and corresponding maps – the labels – containing information of the seabed sediment. In the following, the side-scan sonar data as well as the labels themselves and their procedure of creation will be described in more detail.

A. Side-scan Sonar Data

Although backscatter information can also be collected by other sonar systems, like the single-beam or multi-beam echosounders, the side-scan sonar[2] is a common approach for high-resolution seabed mapping due to its capability covering much larger areas at once. Typically, such a system is mounted on a tow fish carried behind a vessel in the water (see Fig. 1a). Two sound beams are transmitted by the projector, one on the left and one on the right side, the beams being narrow along-track and wide across-track. The signal is reflected by the seafloor (sometimes by other objects as well), a portion is scattered back in direction of the side-scan sonar and there received by the hydrophones. The intensity of the received signal is influenced by many factors, amongst others by the geometry of the sensor-target system, the physical characteristics of the surface, and the intrinsic nature of the surface. Altogether, in the presence of interfering factors there is a rough relation between the backscatter intensity and the surface characteristics: the backscatter intensity of fine sediment is typically lower than the intensity of a signal reflected at a surface with coarse material. Essentially, this relation is more complex. Finally, the received information is mapped to a ground position which can be determined by the propagation time of the signal and the altitude of the sonar system as outlined in Fig. 1b. By moving the ship ahead the seabed is scanned stripe by stripe.

Usually (and as was done here), the data are postprocessed by stiching the collected data together in one image (so-called backscatter mosaic), which is then saved as a georeferenced grayscale image. Here, high intensity is coded as color black, low intensity as color white. The color depth of the image is $8\,\mathrm{bit}$. The width and height of a pixel corresponds to $0.25\,\mathrm{m}$ each. Fig. 2a shows the mosaic for the area of interest. Two features of the mosaic can be seen clearly, first, the edges, where the stripes have been stitched together, and second, the thin bands in the middle of the stripes. These regions are located directly under the vessel during data recording, where no data are available due to the setup of the measurement system.



(a) (b)

Fig. 1. Measuring backscatter intensities using a side-scan sonar system. (a) the side-scan sonar is mounted on a towfish and the seafloor is sensed stripe by stripe (adopted from [1]). (b) functioning of the side-scan sonar: the received intensities are mapped to a position at the seafloor according to the delays of the signals. (adopted from [9]).

B. Labels and Mapping Procedure

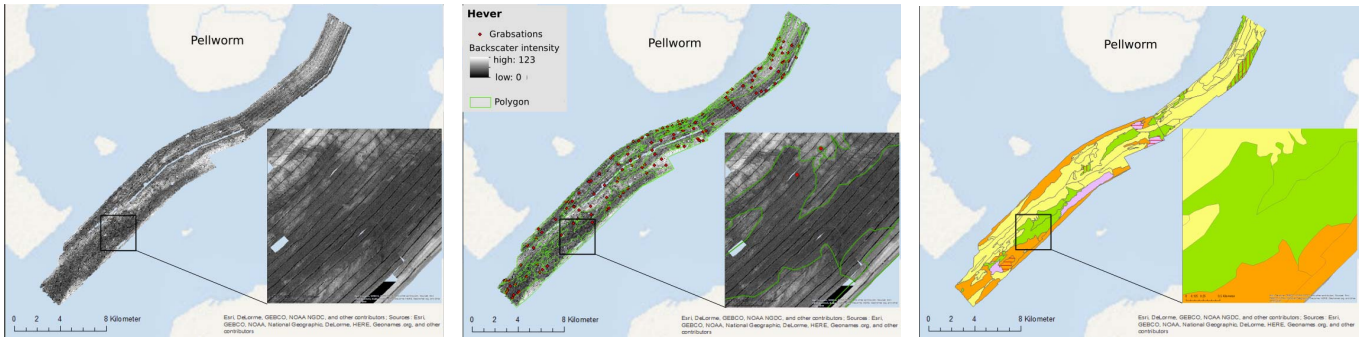
Besides the side-scan sonar images, their interpretation is also available in terms of labelled polygonal regions (see Fig. 2c). Information on the sediment type is assigned to each polygon so that the images are labelled pixelwise. The mapping procedure was performed manually and is based on expert knowledge and grain-size distributions of ground truthing sediment samples that have been collected at specific locations within the investigation area (see Fig. 2b).

The sediment samples were collected by a Van Veen Grab. Afterwards a grain-size analysis was performed using sieves for particles with diameter greater than $63\,\mu\mathrm{m}$ and the laser defraction method for smaller particles respectively. Based on the grain-size distributions the relative content of the fractions for clay/silt ($< 63\,\mu\mathrm{m}$), sand ($63\,\mu\mathrm{m}$ to $2000\,\mu\mathrm{m}$), and gravel ($> 2000\,\mu\mathrm{m}$) was determined. The classification according to Folk was then used to specify the sediment type. Fig. 3 depicts the Folk triangle, where the sediment type is defined in dependency of the relative amount of gravel, sand, and silt/clay (mud). Four coarse classes are defined based on the Guidline for Seaflor Mapping in German Marine Waters [1]: coarse sediment (orange), mixed sediment (lavender), sand (yellow), fine sediment (green). A finer graduation is given by using the eleven subclasses: gravel (G), sandy gravel (sG), gravelly sand (gS), sand (S), muddy gravel (mG), muddy sandy gravel (msG), gravelly mud (gM), gravelly muddy Sand (mgS), mud (M), sandy mud (sM), and muddy sand (mS).

The mapping procedure was carried out in two steps, first, the spatial segmentation, and second, the assignment of characteristic classes/values to the regions [8]. The segmentation of the mosaic images was carried out by dividing the images into regions with similar sediment characteristics based on

[1]The Bundesamt für Seeschifffahrt und Hydrographie (BSH, Federal Maritime and Hydrographic Agency) is a higher federal authority in Germany coming under the jurisdication of the Federal Ministry of Transport and Digital Infrastructure.

[2]The technical description of side-scan sonar systems is based on [2, ch. 2].



(a) (b) (c)

Fig. 2. The given data in the investigation area around the tidal race called Hever. (a) backscatter intensities in terms of a grayscale mosaic. (b) manually segmented regions and locations of sediment samples. (c) labels according to the Folk classification (see Fig. 3) determined on the basis of the mud, sand, and gravel content of the sediment samples.

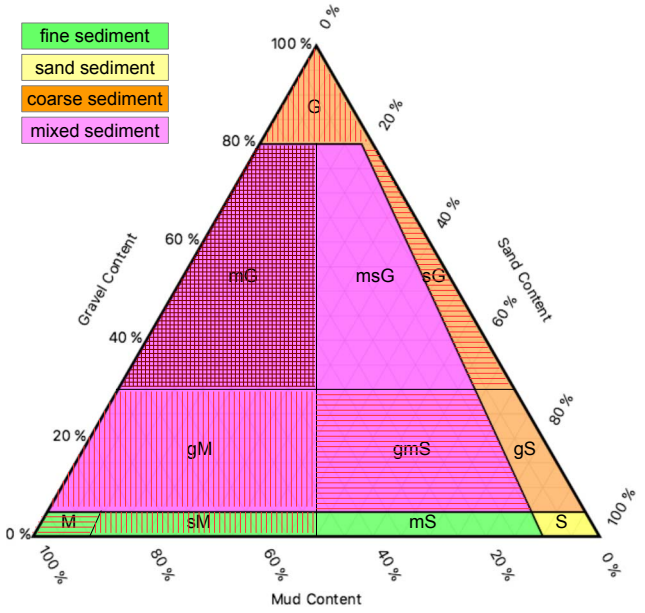


Fig. 3. Sediment type classification according to Folk [10]. The sediment type is determined by the content of mud, sand, and gravel. 11 types of sediment are distinguished, which can be grouped into the four types of fine, sand, coarse, and mixed sediments.

the backscatter intensities themselves and additional information like geological conditions, water depth, and underwater video data. The regions were manually outlined by polygons. Afterwards, for each region the sediment type according to the folk classification was determined and assigned to the whole region. Besides the type also the relative content of gravel, sand and silt/clay was assigned to the regions. For the determination of the sediment parameters of a region all the sediment samples falling into this region were taken into account. In case of mupltiple samples within a region, the arithmetic mean of the grain-size distributions was used to determine the sediment parameters and type. For regions containing no samples at all, labels were copied from regions exhibiting similar properties.

Fig. 2b and Fig. 2c depict the locations of the grabstations and the regions as well as the assigned labels (folk classification) respectively.

III. CNN-based Model

The sediment consists of particles with sizes ranging from the scale of micrometers to several centimeters. Almost all of the particles are too small to be detected as single particles in the side-scan sonar image due to the insufficient resolution. Nevertheless, the aggregation of many particles produces different backscatter intensities in the side-scan sonar image depending on the composition of the particles regarding their size. Single values in the backscatter image do not contain much information, but the combination of the pixel values in the local neighbourhood produces characteristic patterns varying depending on the sediment type [4].

Infering sediment information from the backscatter images is a pattern recognition task. The classical approach for pattern recognition consists of two steps [11]: first, transforming the input data into a lower dimensional representation in terms of a feature vector. Appropriate features should be 'relatively invariant with respect to transformations and distortions of the input' [11, p. 2279] and representative for each pattern type. In a second step a (trainable) classifier is used to seperate the feature vectors. The main work is to find appropriate features, which is often a costly trial and error process.

Convolutional neural networks (CNN) are a special type of feedforward networks and have been very succesful in recent years in the domain of pattern recognition tasks [11], [12], [13], [7]. One main advantage is that their special structure enables an end-to-end training, i.e. the feature extraction and classification module is incorporated into one model. This enables the model to extract the relevant features on its own and not being restricted to a set of handcrafted features given by the designer. Furthermore, CNN are modeled in a way that especially the spatial structure of the given data is used to extract relevant information by the use of convolutional layers. CNN have been successfully applied to pattern recognition tasks in recent times, especially – but not restricted to – the do-

main of image recognition. Typically, the raw data images are used as input without any (significant) preprocessing, which have often been recorded under varying conditions regarding exposure, perspective, image quality, etc. Given an acceptably large set of images, CNN have proved to cope well with such miscellaneous data. In terms of the sonary images the situation retains similar. The information of the seabed sediment in the backscatter image is usually overlayed with interference resulting from measurement conditions. These facts and the availability of large data sets make CNN a promising tool for the classification of seabed sediment information.

In the following we will briefly summarize the structure of CNN and then describe the overall approach we use in this paper.

A. Structure of Convolutional Neural Networks

Basically, the idea of artificial neural networks (ANN) is to learn a mapping $\mathbb{R}^n \to \mathbb{R}^m$ from a n-dimensional input space to a m-dimensional output space. Usually, the mapping is only known at discrete points represented by the samples that have been observed, and one is interested in an approximation of the underlying mapping in a generalized way. In order to use ANN as a prediction model, one has to perform three phases: first, the network structure has to be defined and the parameters of the network have to be initialized, second, the network parameters are being adjusted in the training process, and third, the network can be applied to unknown data in the prediction phase.

The network is assembled by neurons, single units that compute an output based on a given input. The neurons are connected with each other in terms of nodes in a graph. The connections are weighted ones, and the input of a neuron is computed by the weighted sum of the output values (activations) of its predecessor neurons. The output is then calculated by applying a (usually nonlinear) function; this is known as the activation function. A feedforward network corresponds to a graph without cycles and the neurons are typically arranged in layers. In such a network the information is propagated from the input layer to the output layer in a directed way. The weights significantly influence the output values of the neurons and usually are the parameters that are adjusted during the training phase.

Convolutional neural networks (CNN) are a special type of ANN, an overview of the structure of a typical CNN with the key elements is given in Fig. 4. The network can be devided into two main parts: the feature extractor that is able to extract (hierarchical) features directly from the raw data, and the classifier part which is basically a fully connected feed forward network connected to the last layer of feature extractor part.

The whole network is arranged in layers of different types, each type performing a specific task. The information is processed through the network layer by layer, starting from the input layer, passing through the layers of the feature extractor as well as the classifier, and finally ending up at the output layer. The basic elements are explained in the following (in order of workflow) using the classification of two dimensional gray-scale image data as example.

1) The input data in terms of an $w \times h$ image is presented to the network, where w and h denote the width and the height of the image respectively. The values given at each pixel of the image are used as input values for the next layer.
2) The convolutional layer is used to extract low level features of its (2d) input. It performs convolutions of the input image with different kernels. For each feature a set of neurons is used, which are arranged in a plane again (as the input image). The number of neurons in that plane corresponds to the width and height of the input image. These neurons (their activation values) assemble the feature map, which is the 2d output signal of the input image convolved with a 2d kernel. For this purpose each neuron is connected to a spatially restricted region of the input, also called the receptive field. Typically, the region is $k \times k$ pixels. The receptive fields of two neurons in the feature map are shifted corresponding to their neighbourhood relation. The set of weights of each neuron in one and the same plane is forced to be equal and thus defines the dimensions of the kernel and the kernel weights. Since the weights are adapted during the training process, relevant features are extracted automatically.
   Each plane of neurons performs the extraction of one feature. In a convolutional layer several kernels are used in parallel to extract several features of the input. Altogether, there are $w_c \times h_c \times n$ neurons in a convolutional layer, where n is the number of kernels and w_c and h_c denote the dimensions of the resulting feature maps.
   Remark: the input of a convolutional layer is 3d in general (for multi-channel images as well as subsequent convolutional layers). Hence, the convolution is performed in 3d using a 3d kernel. The third dimension of the kernel is equal to the depth of the input data (which corresponds to the number of channels for an image or the number of kernels used in a previous convolutional layer).
3) A pooling layer is used to reduce the number of parameters by performing a downsampling of the input data, which decreases the width and height of the input data. There are different approaches, a commonly used one is max pooling. Here, the output is calculated by the maximum of all values within the defined neighbourhood. Pooling is performed for each slice (regarding the depth) of the input volume separately.
4) Usually, several convolutional and pooling layers are used successively. This way hierarchical features are extracted from the input data. While the size of the feature maps is reduced by the pooling, the number of kernels is typically increased in the rear layers. Finally, the feature extractor ends up with a layer that holds the activations for several features with subject to the input

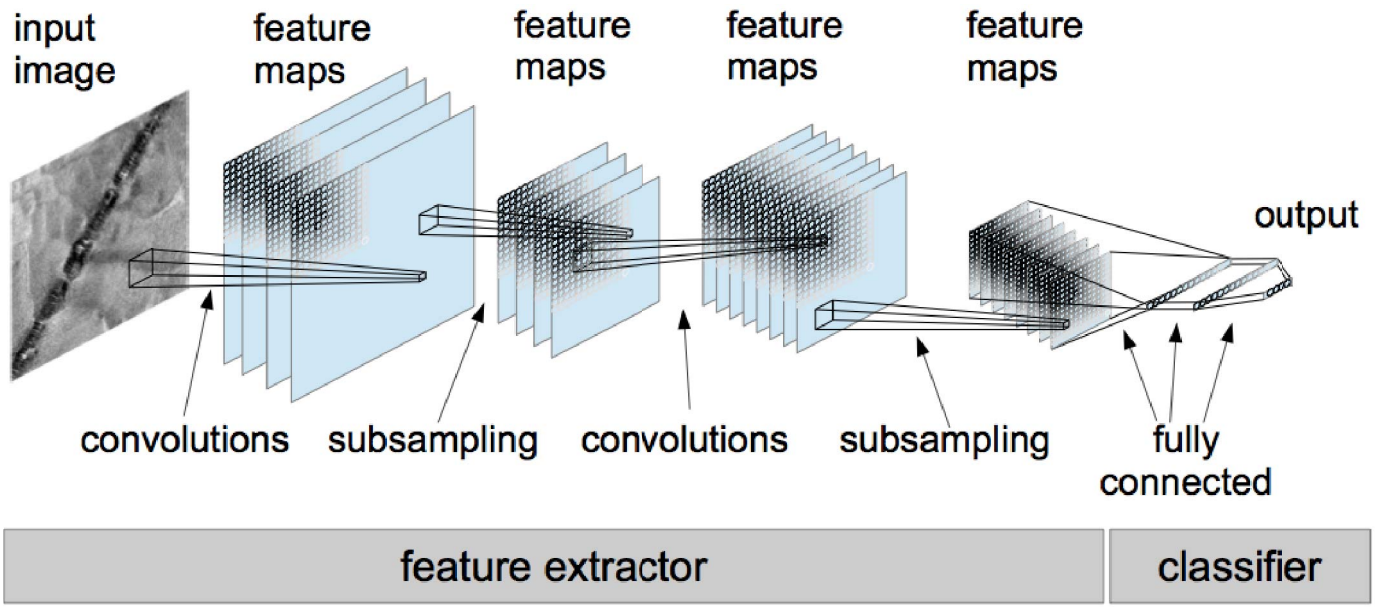


Fig. 4. Basic structure of CNN with the feature extractor and classifier part. The neurons are arranged in layers of special types, like convolutional, pooling, and fully connected layers.

image.

5) These features are then separated by the classifier part of the network, which is directly connected to the last feature map. It consists of some layers, which are fully connected to the previous layers. This part is equivalent to a multilayer perceptron architecture.
   In order to prevent the network from overfitting, methods like dropout [14] can be used. Here, randomly chosen neurons are dropped out by setting their output temporarily to zero during the forward pass and the backpropagation.
6) The output layer consists of as many neurons as classes have to be separated, each neuron representing one class.
7) If the activation functions used to calculate the output values of the neurons are nonlinear, they increase the flexibility of the overall mapping of the network. In the most cases monotonic nonlinear functions are used and in the scope of CNN especially rectified linear units (ReLU) are very common. A ReLU is a unit (neuron) that uses the activation function $f(x) = \max(0, x)$, which has a low computational effort and avoids the problem of vanishing gradients.

B. Overall Approach

The given input data are the side-scan sonar images that have been generated by composing the recorded data after some postprocessing procedure. The dimensions of the resulting image can be very huge (thousands of pixels in width and height respectively) and depend on the processed resolution of the recorded data, the size, and the shape of the investigation area. Generally, the architecture of the CNN is designed according to the input dimensions, so that a standardized size of the input image is preferable. This is the reason why we choose a patch-based classification approach, described as follows.

The original image is divided into patches so that the whole area is covered by patches. Each patch has the same size in order to guarantee comparability among one another with respect to the training and classification procedure. The generated patches are labeled according to the sediment types given by the polygons and together they form the training data. As a matter of fact there are different ways of generating the training data. Some central points have been object of preliminary investigations and will be discussed in the following.

- Orientation: the grayscale mosaic is composed of the stripes that have been recorded step by step. Due to the recording setup the stripes have a characteristic structure: the data located in the middle (this location corresponds to the position directly under the towfish) contain no significant information, but can be seen as faulty data. Furthermore, the intensity is typically decreasing from the inner part towards the outer part of the stripe. This decrease is tried to be reduced during post processing, but cannot be reduced completely in general. Hence, there might be a dependency with respect to the moving direction of the vessel. Preliminary investigations suggest that using oriented data is helpful, but on the other hand the additional information of oriented patches can be compensated by augmenting the data by applying transformations to the patches.
- Patchsize: the patchsize has not been studied extensively yet. It depends on the architecture of the CNN. It turned out that using the standard patch size worked well for AlexNet [15] as well as GoogLeNet [13].

Another point is the strategy of labelling the patches. Since

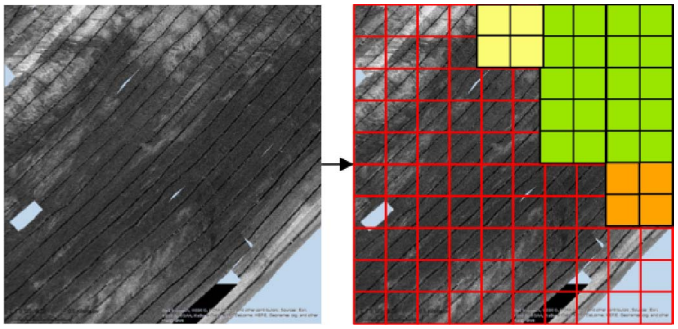


Fig. 5. Patchwise classification approach: the mosaic is devided into patches of equal size, which are labelled using a single class for each patch.

a single patch may span over different polygons, possible options are assigning a single label to each patch or multiple labels. A special case of assigning multiple labels is pixelwise labelling. For three reasons we choose to use single labels as a first approach: 1) *Simplicity*: in this study we want to investigate if CNN are an appropriate tool for the classification of seabed sediment using side-scan sonar data. The examination of different models (like semantic segmentation CNNs) is out of the scope of this paper. 2) *Spatial resolution*: using patches of the given size already gives a good spatial approximation of the underlying sediment patterns for large areas. 3) *Multi voting*: a higher spatial resolution as well as a realization of gradual changes in the classification map can be accomplished by using overlapping patches and weighting the classification results.

As a first approach, we use patches that are generated regardless of their orientation and are labelled using one class for each patch (single labels) as shown in Fig. 5.

IV. Application and Results

In order to evaluate the approach we apply it to a part of the given data in the domain arround the tidal race Hever. As a first step we use the architecture of GoogLeNet [13] as prediction model and train it on the data which we created as described below. We used the Caffe framework [16] for training the models.

Some priliminary investigations revealed that the prediction accuracy highly depends on the validation set. For this reason we apply a cross validation of the model and devide the investigation area into 10 subsets as depicted in Fig. 6. Each subset has approximately the same size of useable data.

A. Generation of the Training Data

A review of the given data shows that there seems to be some inconsistency in the given labels. The labelled classes of some polygons do not correspond to the given sediment sample data, so that we skipped the corresponding polygons (compare Fig. 6a). Looking at Fig. 2c reveals that the area covered by the different classes differs highly. While big areas are marked as sand other classes like gravelly muddy sand are underrepresented. Hence, we use the coarse classification system where the given classes are summarized into the four classes fine sediment, sand sediment, mixed sediment, and coarse sediment (compare to Fig. 3). The resulting labelling is illustrated in Fig. 6b. The four rough classes are still unbalanced regarding to the covered area. We address this problem by augmenting the training data and then choosing the same number of patches for each class randomly (undersampling). The patches for the training data are generated as follows: patches of size 360×360 (pixels) are used where neighbouring patches overlap by $50\,\%$ in each direction. This way, we generally have four patches covering one pixel. Only those patches are chosen which have one class at least covering $60\,\%$ of the whole patch. Those patches are then augmented using six transformations (identity, vertical flip, horizontal flip, rotations by $90°$, $180°$ and $270°$). During training a 224×224 region of the image is cropped randomly as done in the original paper [13]. The patch generation is performed for each subset separately so that there is no overlap among neighbouring subsets. Fig. 6c shows the spatial coverage of the training data resulting from the undersampling procedure for the training data where subset 3 is left out for validation by example.

B. Training and Validation Strategy

By applying the cross validation to the given szenario we have ten different configurations for training. For each configuration one subset is left out for validation. The number of the left out subset is used as the identifier for the configuration (for example: configuration 3 denotes the training where subset 3 is left out for validation). For each configuration ten different networks are trained, each network having the structure described in [13]. All the networks are trained 200 epochs, starting with a learning rate of 0.01, which is reduced after $1/3$ and $2/3$ of the iterations by factor 0.1 each.

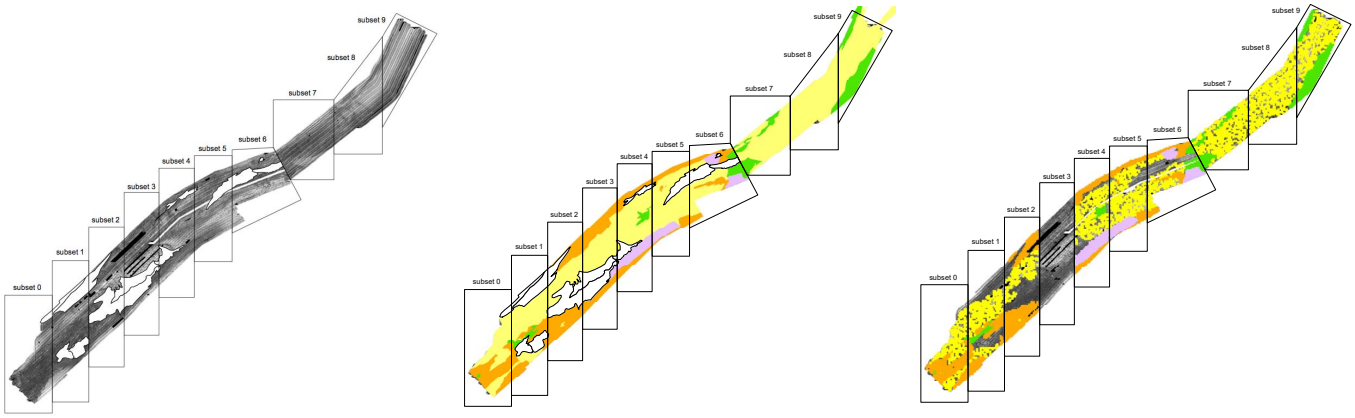
The patches for validation are generated simlilarly as for training but differing in the following way: a) a patch may have $40\,\%$ of background (no label) at maximum. b) no transformations are applied to the patches.

For prediction, the trained networks (at epoch 200) are used as an ensemble in the following way: for each pixel all patches are determined that cover this pixel (which are generally four patches per pixel). Each patch is classified by each instance of the networks, so that we have 40 predictions per pixel in general. The overall classification corresponds to the label that is voted most (maximum voting). If more than one class have the most votes the result is set to *undetermined*.

C. Results and Interpretation

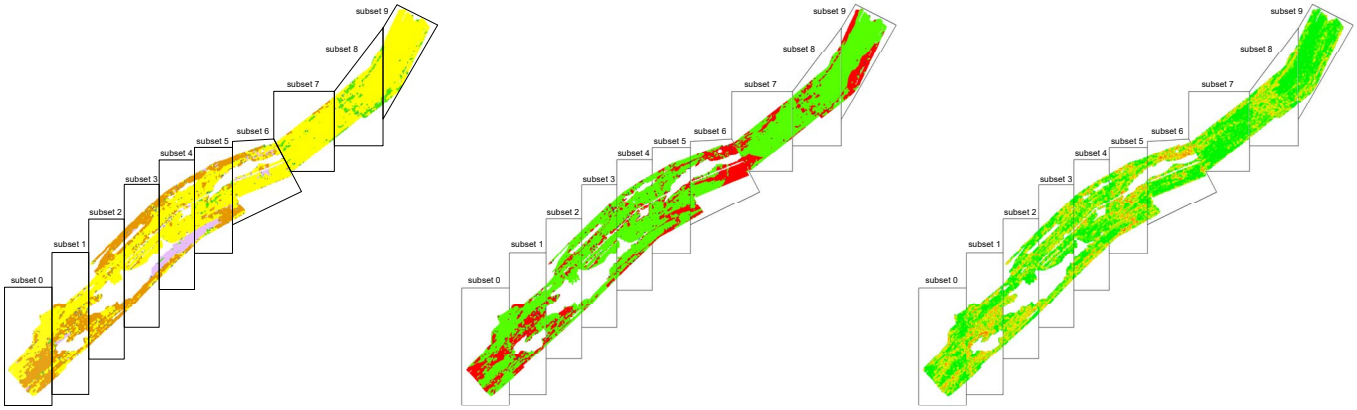
The results of the overall (ensemble) classification are given for each subset individually as well as a for the whole area composed of the validation sets for each configuration 0 to 9. Graphical results are given in terms of a classification map (Fig. 7a), an error map (Fig. 7b), and a confidence map (Fig. 7c). The prediction accuracy is summarized in table I.

For the whole investigation area $74\,\%$ of the pixels are classified correctly by the model (mean accuracy per pixel). The classification of sand sediment works quite well ($83\,\%$ accuracy), while the prediction accuracy of fine sediment



(a) (b) (c)

Fig. 6. Prepared training data used for the application. The investigation area has been partitioned into 10 subsets $(0, \ldots, 9)$ for cross validation. Polygons that obviously had wrong labels have been left out (marked white in 6a and 6b). 6a side-scan sonar image as input. 6b groundtruth. Labels have been combined to four main classes (fine sediment (green), mixed sediment (purple), sand sediment (yellow), coarse sediment (orange)). 6c spatial coverage of training data patches resulting from data augmentation and undersampling (exemplary for the training set, where subset 3 is used for validation).



(a) (b) (c)

Fig. 7. Results of the prediction of the validation sets (composed of all 10 configurations). (a) classification map (colored the same way as the labels and undetermined predictions are marked blue). (b) error map (red: false classification, green: true classification, blue: undetermined classification). (c) confidence map (color scale ranging from red (low confidence) over yellow to green (high confidence)).

TABLE I
RESULTS OF THE PREDICTION ACCURACY.

conf	fine	coarse	mixed	sand	mean (class)	mean (pixel)
0	0.31	0.66	–	0.65	0.54	0.65
1	0.05	0.66	–	0.65	0.45	0.61
2	–	0.74	–	0.79	0.77	0.76
3	0.01	0.85	0.92	0.85	0.66	0.85
4	0.00	0.73	0.91	0.91	0.64	0.86
5	0.01	0.81	0.69	0.76	0.57	0.76
6	0.01	0.32	0.08	0.78	0.30	0.43
7	0.07	–	–	0.95	0.51	0.81
8	0.86	–	–	0.84	0.85	0.84
9	0.12	–	–	0.99	0.56	0.71
all	0.11	0.70	0.61	0.83	0.56	0.74

is very poor (11 %). As already observed in preliminary investigations, the accuracy varies notably when applying the model to the different configurations (validation sets). The worst classification accuracy is achieved for configuration 6 (43 % per pixel and only 30 % for the mean prediction per class). In this region the side-scan sonar image reveals a characteristic structure, which does not appear in any other subset. This might be faulty data. Other validation sets achieve a classifcation accuracy of 84 % and 85 % respectively (configuration 8). The confidence map illustrates the percentage of votes, which contributed to the class with maximum votes. The color scale ranges from green (100 %, i.e. all votes belong to one class) to red (25 %, i.e. all four classes have been voted equally). This is a measure for confidence. It is interesting to see that the confidence decreases typically in those regions, where the classifcation is wrong (marked red in the error map). But there are also regions, where the prediction is wrong and

the confidence is high (like in the eastern part of subset 6 and some parts of subset 9).

Alltogether the prediction accuracy of the model is not very high, which might rely on the following points:

- Since the backscatter intensities depend on many factors, the influence of errors might be to strong to extract a clear mapping.
- The sediment type was mainly determined on the basis of the sediment samples. The samples contain local information, but this information has been projected to sometimes very large regions. Maybe the given labels contain to much contrary data.
- For regions that contained more than one sediment sample, the given information was averaged on the basis of all samples. Contrary sediment samples could result in false information.
- Regions without any sediment sample received a label from a region with similar characteristics. This can also lead to false information.

V. Conclusion

In this paper we proposed a CNN-based approach for the classfication of sediment types on the basis of side-scan sonar images. The approach uses a patch-based classification, where the patches are classified using an ensemble of networks, which have the architecture of the well-known GoogLeNet [13]. While the classification accuracy of sand sediment is quite good, the prediction of fine sediment is nearly impossible. The overall accuracy could be improved by addressing several issues. Most of the points addressed in section IV-C result from discretizing the information. The influence of the different discretization levels should be investigated in future work. In order to avoid possible discretization errors a promising approach could use training data primarily near the locations where the sediment samples have been collected.

References

[1] C. Propp, A. Bartholomä, C. Hass, P. Holler, M. Lambers-Huesmann, S. Papenmeier, P. Richter, K. Schwarzer, F. Tauber, and M. Zeiler, "Guideline for seafloor mapping in German marine waters using high-resolution sonars," BSH, Tech. Rep., 2016.

[2] P. Blondel, *The handbook of sidescan sonar*. Springer Berlin Heidelberg New York, 2009.

[3] L. Atallah, P. J. P. Smith, C. R. Bates, and L. Atallah Probert Smith, P.J. and Rates, C.R., "Wavelet analysis of bathymetric sidescan sonar data for the classification of seafloor sediments in Hopvagen Bay-Norway," *Marine Geophysical Researches*, vol. 23, pp. 431–442, 2002.

[4] P. Blondel, L. Parson, and V. Robigou, "TexAn: textural analysis of sidescan sonar imagery and generic seafloor characterisation," *IEEE Oceanic Engineering Society. OCEANS'98. Conference Proceedings (Cat. No.98CH36259)*, vol. 1, pp. 419–423, 1998. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=725780

[5] B. Bourgeois and C. Walker, "Sidescan sonar image interpretation with neural networks," *in Proc. IEEE Int. OCEANS Conf., Honolulu, HI*, pp. 1687–1694, 1991.

[6] D. Buscombe, P. E. Grams, and S. M. C. Smith, "Automated riverbed sediment classification using low-cost sidescan sonar," *Journal of Hydraulic Engineering*, p. 06015019, sep 2015. [Online]. Available: http://ascelibrary.org/doi/10.1061/%28ASCE%29HY.1943-7900.0001079

[7] J. Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.neunet.2014.09.003

[8] H. Heyer and K. Schrottke, "Aufbau von integrierten Modellsystemen zur Analyse der langfristigen Morphodynamik in der Deutschen Bucht: AufMod; gemeinsamer Abschlussbericht für das Gesamtprojekt mit Beiträgen aus allen 7 Teilprojekten," Bundesamt für Seeschifffahrt und Hydrographie (BSH) (u. a.), Hamburg (u. a.), Tech. Rep., 2013.

[9] E. Seibold, *Der Meeresboden*, 1974.

[10] R. L. Folk, "The distinction between grain size and mineral composition in sedimentary-rock nomenclature," *The Journal of Geology*, vol. 62, no. 4, pp. 344–359, 1954. [Online]. Available: http://www.jstor.org/stable/30065016

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Arxiv.Org*, vol. 7, no. 3, pp. 171–180, 2015. [Online]. Available: http://arxiv.org/pdf/1512.03385v1.pdf

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June, pp. 1–9, 2015.

[14] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," pp. 1–18, 2012. [Online]. Available: http://arxiv.org/abs/1207.0580

[15] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," pp. 1–9, 2012.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.