Jury Member Report – Doctor of Philosophy thesis.

Name of Candidate: Evgenii Baraboshkin

PhD Program: Petroleum Engineering

Title of Thesis: Automated core description based on computer analysis

Supervisor: Professor Dmitry Koroteev
Co-supervisor: Dr. Denis Orlov

Name of the Reviewer:

I confirm the absence of any conflict of interest	
	Date: 12-01-2023

Reviewer's Report

To whom it may concern.
I am writing to support Evgeny Baraboshkin to be awarded a doctorate in Petroleum Engineering after thesis revisions. Computer vision, machine learning, and deep learning techniques have revolutionised many industries yet their application for core analysis remains lacking. The submitted thesis provides a wide breadth of analyses of various modern techniques to classify and segment rock core images. The strength of the work was on the evaluation of many modern techniques with a focus on applications that would impact the oil and gas industry. The potential to automate labour intensive workflows is outstanding and the translation of the presented research to deployable workflows were covered. The main weakness of the thesis is that clear objectives are not explained at the start. After reading the thesis, the objectives 'come out' but the objectives are not explicitly explained anywhere. It would be best to provide the thesis objectives at the end of Chapter 1 and to further explain these objectives within the broader context of the published works. This would really help with understanding the impact of the work.

I have various comments within the provided annotated manuscript that should be addressed.

Overall, a substantial amount of new knowledge and novel innovations were achieved, which warrants a PhD. But the presentation of the work(s) can be improved – as explained in the annotated manuscript.

Provisional Recommendation

☐ *I recommend that the candidate should defend the thesis by means of a formal thesis defense*

X *I recommend that the candidate should defend the thesis by means of a formal thesis defense only after appropriate changes would be introduced in candidate's thesis according to the recommendations of the present report*

☐ *The thesis is not acceptable and I recommend that the candidate be exempt from the formal thesis defense*

Skoltech
Skolkovo Institute of Science and Technology

Skolkovo Institute of Science and Technology

AUTOMATED CORE DESCRIPTION BASED ON COMPUTER ANALYSIS

Doctoral Thesis

by

EVGENY BARABOSHKIN

DOCTORAL PROGRAM IN PETROLEUM ENGINEERING

Supervisor
Professor Dmitry Koroteev

Co-supervisor
Denis Orlov

Moscow - 2022

I hereby declare that the work presented in this thesis was carried out by myself at Skolkovo Institute of Science and Technology, Moscow, except where due acknowledgement is made, and has not been submitted for any other degree.

Candidate (Evgeny Baraboshkin)

Supervisor (Prof. Dmitry Koroteev)

Abstract

Currently, a geologist should manually fill an electronic table while he describes rocks or scratch a description on a blanket. This takes a lot of time and effort. Lots of researchers have developed their approach to autonomous rock classification. Unfortunately, all of them are based on a core column investigation. Most used a traditional approach to image classification based on a statistical analysis of images and machine learning techniques. In this work, modern deep learning techniques were applied to the problem of classification and segmentation of core images. An important note here is that previous works cover only core column examination, while this work presents an approach to extract core columns from a core box. Thus, a geologist shouldn't spend time on a manual description of the core or extraction of separate images out of a core box. This can be done by applying the developed algorithms described in this work. The algorithms were also included in a system which became worldwide available software.

Keywords: automated rock description, core column extraction, sedimentology, mining, geology, deep learning, machine learning.

Publications

Journal papers

Baraboshkin, E.E., Baraboshkin, E.Y., Ismailova, L.S., Orlov, D.M., Zhukovskaya, E.A., Kalmykov, G.A., Khotylev, O.V., Koroteev, D.A., 2020. Deep convolutions for in-depth automated rock typing. Comput. Geosci. 135. https://doi.org/10.1016/j.cageo.2019.104330

Baraboshkin, E.E., Demidov, A.E., Orlov, D.M., Koroteev, D.A., 2022. Core box image recognition and its improvement with a new augmentation technique. Comput. Geosci. 162. https://doi.org/10.1016/j.cageo.2022.105099

Patents

Baraboshkin, E.E., Koroteev, D.A., Orlov, D.M., Ismailova, L.S., 2019. RU2706515C1 - System and method for automated description of rocks.

Conference papers

Baraboshkin, E., Ismailova, L., Orlov, D., Koroteev, D., 2019. Machine Vision Methods in the Application for Core Image Segmentation, in: Progress'19. European Association of Geoscientists & Engineers, pp. 1–5. https://doi.org/10.3997/2214-4609.201953061

Baraboshkin, E., Orlov, D., Koroteev, D., 2020. Tools for Automated Rock Description, in: First EAGE Digitalization Conference and Exhibition. European Association of Geoscientists & Engineers, pp. 1–5. https://doi.org/10.3997/2214-4609.202032061

Baraboshkin, E., Panchenko, E., Demidov, A., Sharipova, Y., Gatina, N., Koroteev, D., 2022. Automated core description system application for sedimentological analysis, in: 21st International Sedimentological Congress.

Baraboshkin, E.E., Demidov, A.E., Orlov, D.M., Alekseev, V.V., Tkachev, A.Y., Nyazhemetdinov, R.R., Korneeva, K.N., Kostoev, R.S., Koroteev, D.A., 2021a. An intellectual system for fast and reliable automated core description development, in: EAGE Annual. 82nd EAGE Conference and Exhibition. Amsterdam.

Baraboshkin, E.E., Demidov, A.E., Panchenko, E.A., Gatina, N.N., Hahina, A.N., Mamaev, D.A., Alekseev, V.V., Nyazhemetdinov, R.R., Tkachev, A.Y., Orlov, D.M., Koroteev, D.A., 2021b. Automated Full-Bore Core Description Application for Production Purposes. From an Idea to IT-Product, in: Data Science in Oil and Gas 2021. European Association of Geoscientists & Engineers, pp. 1–6. https://doi.org/10.3997/2214-4609.202156016

Baraboshkin, E.E., Ivchenko, A. V, Ismailova, L.S., Orlov, D.M., Baraboshkin, E.Y., Koroteev, D.A., 2018. Core photos lithological interpretation using neural networks, in: 20th International Sedimentological Congress.

Baraboshkin, E.E., Panchenko, E.A., Demidov, A.E., Gatina, N.N., Hahina, A.N., Mamaev, D.A., Alekseev, V.V., Nyazhemetdinov, R.R., Tkachev, A.Y., Orlov, D.M., Koroteev, D.A., 2022b. Automated full-bore core description system application in production. Use case example, in: Ways of Realization of Oil and Gas Potential of West Siberia. Khanty-Mansiysk, pp. 293–299.

Ivchenko, A. V, Baraboshkin, E.E., Ismailova, L.S., Orlov, D.M., Koroteev, D.A., Baraboshkin, E.Y., 2018. Core Photo Lithological Interpretation Based on Computer Analyses, in: IEEE Northwest Russia Conference on Mathematical Methods in Engineering and Technology. pp. 425–428.

Acknowledgements

Table of Contents

List of Symbols, Abbreviations

PCA - Principal Component Analysis.

HOG - Histogram of Oriented Gradients.

PNNF – features extracted with pretrained network

CNN – convolutional neural network.

SVM - Support Vector Machines.

RF - Random Forest.

AC – agglomerative clustering.

FH - Felzenszwalb-Huttenlocher.

QS - Quick Shift.

SLIC - Simple Linear Iterative Clustering.

RAG - Region Adjacency Graph.

SSh - Selective Search.

GTD-1 and GTD-2 – geotechnical description datasets described in sec. 4.1.1.

TLA - template-like augmentation technique described in sec. 3.4.

List of Figures

11

12

15

List of Tables

Chapter 1.   Introduction

1.1. Problem overview

The scientific approach requires data collection on a studied object. A geological object is presented by a large area formed in a certain way (physical weathering by winds and rivers, temperature changes, volcanic and tectonic processes, etc.) during a certain time (millions of years). To study this object, a geologist may apply different methods that constantly improve and arise. It could be well logs, seismic, thin section analysis, and many others. Since the beginning of geology, one of the ways to acquire data was to manually describe the rocks which are formed during the Earth's history. This data can be used to understand many things: how rocks were formed, how they can be used, what valuable components are stored in the area, and what an explorer should do to extract these components efficiently. This data and other characteristics are stored in a geological model. While humanity developed technologies, the depth of such investigation has grown. A special methodology was developed to get the rocks from depth, which allowed to extract samples by drilling. Such drilled sample is called a core. Currently, most manual rock descriptions are performed on core samples extracted from the Earth by drilling.

The number of cores extracted for one drilling season could be different in different industries. For example, in 2020 according to the annual reports Novatek (oil and gas company) drilled 45.4 km for exploration with core extraction and 640 km for production without core extraction, while Polymetall (mining company) drilled 189 km with core extraction for exploration, the number of production cored intervals is unknown. The drilling depth and complexity cause differences in core extraction rates between industries. The drilling depths in the oil industry are around 2 to 3 km and up to 500 – 1 km in the mining industry.

Various conclusions and decisions rely on such data, so sometimes, companies hire several experts to describe the same drill core. The first expert may describe general and technical information right after extraction, and the second may dive into details to build a proper geological model. A reliable and precise description is essential to decrease all the uncertainties during reserves calculation.

17

There are several cases when the description required for proper modelling:

1) Geological modelling.
2) Hydrological modelling based on upscaled results of geological modelling.
3) Geotechnical modelling.

Each case requires a different amount of data and various information from the core description. Thus, an expert should sometimes describe the core to get the results. As other experts may have different views on describing core, different companies developed a set of standards.

The main document, which included most of the best practices for core description and analysis, was prepared by American Petroleum Institute (1998). The main workflow is the following:

1-6) Check the core layout, numbering, quantity, and any damage.
7) Compare the core gamma-ray log.
8) Assess the cored sequence before description to seek out large-scale features (sequences, contacts, etc.).
9) Record major features (9 features).
10) Record centimetre-by-centimetre features.
11) Record non-reservoir rocks.
12) Record fracture information.

As the preparation process before the core description can also differ, a special section devoted to core imaging techniques exists in the recommendations (American Petroleum Institute, 1998). The imaging can be performed with different tools and under different spectra (daylight, ultraviolet, x-rays, etc.). In this work, specific types of images are used – those which are taken under natural light. Unfortunately, no special methodology is described in (American Petroleum Institute, 1998) to get such images in natural light.

In recent documents ("Appendix no 5. To the regulations on 'Core investigation' (In Russian)," 2017) several additions to imaging techniques described like color scheme persistence. Also, some modern types of analysis are proposed, like color

distribution based on images taken with a camera. Thin (2 cm) slabs stored in epoxy can be taken from the full-bore core column.

One way or another – all the extracted cores are usually described manually, which is time-consuming and requires experts' attention. A description's speed and precision may vary depending on the geologist's experience, the purpose of the research and the available budget. Various methods are used to describe core (Marjoribanks, 2010): prose logging, graphical scale logging, and analytical spreadsheet logging. All the methods have their benefits and pitfalls.

Prose logging can be helpful when a quick review of freshly extracted rock is needed when there is no time to investigate the core precisely and the object is unknown. The main problem is that each expert can formulate the description differently. There could be a standard for the formulation. In most cases, the descriptions are hard to compare as the depth intervals are not fixed, and each expert may highlight different features. Prose logging is usually applied in reports.

Graphical scale logging is useful when there is some time available for an expert to investigate the core on a predefined scale properly (e.g., 1:5, 1:10, 1:100, etc.). With a set scale, the expert fills a blanket with different characteristics. Each characteristic may be either numerical (e.g., representing angles, sample numbers, etc.) or graphical. Also, a separate comment section usually appears to describe some features that are not fit in the table. Such reports are easier to compare. The number of characteristics may differ in different companies and regions. As some characteristics may not be included in such a log, and some may be way smaller than the chosen scale – the level of detail from such a log may be insufficient.

The analytical spreadsheet logging is the most time-consuming and detailed as they are formed in a table. To fill the table, special codes are used to indicate the core description parameters called geocodes. Geocodes and analytical spreadsheets are usually applied in the mining industry (Marjoribanks, 2010). The geocodes may represent different prior-defined categories like color, mineral percentage composition, etc. The expert may use a special catalogue with barcodes to scan each lithotype and

automatically record it into a system or manually choose each code in a special software.

Some common characteristics should be counted, such as discontinuities (for rock quality designation (RQD) calculation) and total core recovery in both industries. It is common to describe the core within certain intervals bounded to the coring run or description scale. The coring run is the rod length drilled into the ground before the core inner tube recovering.

A core description is applied to various tasks. The time spent on the description of a core could be various. Typically, a geologist who wants to describe the core thoroughly spends around 8 hours for 30 to 100 meters of the core, depending on the deposit's complexity.

The description process requires an expert presence in the core storage. Meanwhile, in most cases, the core stored in the storage has a digital representation with an image. The expert can visually inspect the core without a core storage visit. As the expert is valuable, it is not in the company's interest to spend his time on manual core description. Thus, different methods were developed to automate this process and help the expert describe the core faster. The following subsection will discuss these attempts and approaches.

1.2. Literature review

The rock texture classification is a common problem which can be found either in geology or different fields, like decoration (Laurenge and Borduà, 2004) or tile quality control (Ferreira and Giraldi, 2017) and autonomous rovers' guidance (Gor et al., 2001). Here all related to rock image interpretation through computer vision methods works presented. As rock type definition can be performed in various scales following classification applied to describe all the scales based on the way of the data acquisition and analysis: long-distance feature mapping (airborne, satellite imagery), visual rock investigation without special equipment (which is around 2 to 15 cm), thin-section investigation, and digital rock investigation (e.g., tomography, scanning electronic microscope). A list of problems which may appear during data gathering and analysis with traditional techniques will be described first, and then the description of

the reference will be given. An almost thorough reference description will be given in the visual rock investigation section, as it is the principal subject of this research.

1.2.1. Long-distance feature mapping

This section united all the image processing performed over a long-distance imaging technique (the scale is more than 15 cm) and aimed to classify such images to a certain type of rock or geological unit. Such distance measurements could be performed with satellites, special aerial equipment (drones, planes), or ground tools. The methods of interpreting such data require understanding the rocks' physical characteristics. The interpreter (e.g., geologist) should also know how the relief forms and how it may be changed due to different geological dynamic processes. As the relief may form in different ways, the region's history may also be important for the interpretation (e.g., detection of certain deposits). There is a list of problems which an expert may run into:

1) The observed characteristics may not be connected to only one object.
   A set of layers may generate and increase the observed physical characteristics. In the same way, they can compensate each other.

2) The searched object may not generate any high physical field.

3) Some objects (like diamond tubes) may not be observed thought physical fields as they are like rocks in their characteristics. For such cases, a set of additional methods like satellite images should be observed to detect other important features like lineaments.

4) Each case is special.
   As the Earth was formed over a long period of time and different stochastic (in terms of our knowledge) processes happened – we cannot exactly know what kind of things happened to the observed place. The main problem that at the same time and place, several processes could happen, e.g., some rivers could be covered by other sediments or destructed by erosional processes. A geologist may only propose some most noticeable versions based on the available information (the region's history, the observed characteristics, etc.).

Thus, many features should be counted in to develop a generalized algorithm that will work in any place on the earth. As the data is gathered at different times with various methods, there may be quite a lot of missing data, which hardens the algorithm construction situation. Still, it is possible to construct an algorithm for a certain region. For satellite imagery, several approaches were developed to detect large-scale features.

A work by (Cracknell and Reading, 2014) demonstrates the abilities of different machine learning algorithms to extract features and improve manual interpretation results. A set of algorithms described: Naive Bayes (NB), k-Nearest Neighbors (kNN), Random Forests (RF), Support Vector Machines (SVM), and Artificial Neural Networks (ANN). The study covers an area of ~160 km^2 located near Broken Hill, far western New South Wales, Australia. The region includes 13 different types of rock (please, refer to the publication for a thorough description): retrograde micaceous schist; non-graphitic meta-sediment (pelite– and psammite–psammopelite); garnet-bearing quartzo-feldspathic gneiss; medium to fine-grained quartz-plagioclase-K-feldspar-biotite garnet gneiss; metasediments ranging from well bedded pelitic/psammopelitic schists, to psammopelitic or psammitic/pelitic metasediments; basic gneiss, lenticular masses of garnet-bearing quartzo–feldspathic gneiss, and "lode horizon" rocks; metasediment and metasedimentary composite gneiss; garnet rich. Sporadic bodies of basic gneiss and quartz-gahnite; medium-grained saccharoidal leucocratic sodic plagioclase quartz+K-feldspar biotite rocks; psammopelitic to psammitic composite gneisses or metasediments; plagioclase-quartz rocks and minor granular quartz-iron oxide/iron sulfide "lode" rocks; medium to coarse-grained quartz–feldspar–biotite+garnet gneiss; sedimentary quartz–feldspar–biotite–sillimanite-garnet-cordierite composite gneiss. A digital 1:250,000 geological map was used to prepare the training dataset. 10% of the data (~6500 samples) were used to train an algorithm. The sample is represented by a 256x256px crop of a map which is equal to a 50x50m region. A set of 18 geophysical data sources was used: airborne geophysical data were supplied with X and Y spatial coordinates, digital elevation model (DEM) (m ASL), total magnetic intensity (TMI) (nT), and three gamma-ray spectrometry (GRS) channels comprising potassium (K%), thorium (Th ppm), and uranium (U ppm), Landsat ETM+ data consisted of band 1 blue, band 2 green, band 3 red, band 4 near-

infrared, band 5 short-wave infrared, band 6 thermal low gain / high gain, band 7 mid-infrared, excluding band 8 panchromatic. Also, 9 different features were generated from Landsat images by dividing 3rd band to 1,2,5,7 bands, dividing 5th band to 1, 2, 4, 7 and by multiplying the division of 5/4 and 3/4 bands. A set of experiments was conducted with different inputs for training and classification. Different training data sampling techniques were used to train the algorithm: a unified area, several areas (32 data clusters) and multiple randomly chosen areas (1024 clusters). The maximum resulting accuracy for each algorithm on a test dataset: 0.506 - Naive Bayes, 0.635 - k-Nearest Neighbors (kNN), 0.762 - Random Forests (RF), 0.671 - Support Vector Machines (SVM), 0.627 – Artificial Neural Networks (ANN). As a result, the authors concluded that Random Forest is a good first-choice machine learning algorithm for multiclass inference using widely available high-dimensional multi- source remotely sensed geophysical variables.

The on-ground scanning of the rocks is also may be done. While satellites and airborne surveys aim to catch large areas, on-ground measurements may be valuable for detailed mapping of quarries and outcrops. To catch the whole outcrop a special equipment used like drones and lidar/laser – scanning (Enge et al., 2007). The GPS data, various places scans and photographs are united in one project. This data is further processed and cleaned. Up to 50% of data may be decimated. Finally, the data is triangulated, and a digital elevation model is created. The model may be textured with image data. Such a model is very helpful for a thorough investigation by an expert, but it involves a lot of manual processing.

The next step after the analysis of long-distance data is to analyze the data which has some material form, like rock samples.

1.2.2. Visual rock investigation

The visual description of rocks (from 10 meters to 10 cm) could be done with traditional methods by hand. This method requires only visual recognition and a fresh piece of rock. The traditional instruments required for the description are a hammer, acid (e.g. HCL for carbonates detection), magnification tool, and sometimes – UV light (e.g. for mineral and oil stain detection). In rare cases, a geologist may use a needle and

some elements from the Mohs scale of mineral hardness. Special equipment used for modern investigation includes XlrF-scanners.

A set of problems may also be presented during the investigation:

1) The scale of the observed data may be much larger than the expert could see.

This is a common case when the expert has only a core. For example, suppose a boulder is greater than the core size. In that case, it could be interpreted as a layer if an expert wouldn't see special elements that will point out this difference (e.g., a sharp boundary or an unexpected lithology change that doesn't fit the region's environmental conditions). The same problem could be met in the field when the observed object is big and doesn't fit the expert's environmental process understanding.

2) The researcher may not know some special topics to understand the material.

This is a common case for researchers who have little experience in studying the material or the information about the studied object is insufficient. In that case, some features may be guessed. A nice illustration of such a case may be trace fossils counted as some animals or plants for a long time.

3) The interpretation of the material may change depending on the case.

The initial appearance of the rocks may have a dual meaning as some processes (e.g. flows, chemical processes, etc.) may look the same in different conditions (e.g. salt deposits may appear in oceans and salt lakes). Rocks may be changed the same way in different conditions. For example, quartzite (metamorphized sandstone) may form in case of heating and pressure deep underground or heated by intrusion near the surface.

The classification of the image itself to a certain class or feature characteristic may be done with different algorithms. The rock description automatization problem was touched by different scientists in various ways and for various purposes.

One of the earliest works on the theme of geological features automatic characterization was published by Jonatan Hall with his colleagues (Hall et al., 1996). They proposed a methodology to extract sedimentological and structural, petrophysical and reservoir description information from wireline images and core photographs. The main objective was to classify known geometric structures and textures and abstract them into geological processes. The second objective was upscaling and downscaling the described characteristics to produce reliable descriptions. A set of human and machine virtues and failings was defined by (Hall et al., 1996). Humans can recognize features from noisy data and adapt by learning new knowledge, but it can be subjective and biased. A human tends to be inconsistent in applying knowledge and interpretation and making hasty interpretations, but he also can formalize models to explain observations and interpretations. According to (Hall et al., 1996), a machine has virtues. It can untiringly load and process large volumes of data, provide summarized statistics and provide formalized uncertainty evaluation measures. Several methods were applied to extract information from the images(Hall et al., 1996). For the planar events like boundaries, calcite concretionary nodules, fractures and many others, edge detection followed by Hough transform is performed. The extracted features were analyzed with different statistics for various shapes like ellipsoids. It takes around 40 minutes per meter to calculate the statistic. For textural information characterization like rock fabric, several steps are performed. Firstly, a statistic based on co-occurrence matrices was calculated. Secondly, linearity is defined as the distance from the origin and direction as the angular position on the cross plot calculated. Finally, an unsupervised neural network called Competitive and Selective Learning (CSL) classifies the texture based on calculated statistics. The same statistics are used for the calculation of True Dip and Azimuth. The computation time for a window length of 0.5 m with a step of 0.075m is 6 seconds per meter.

(Singh et al., 2004) tested several algorithms for the autonomous classification of rocks to help civil engineers make decisions. Seven feature extraction methods were tested due to a consensus that no single texture analysis method is suitable for all image data types. The methods are based on texture analysis. Each of the methods extracts a different number of features or coefficients: Autocorrelation – 24, Co-occurrence

Matrices [4,5] – 25, Edge frequency – 25, Law's Mask Features [13] – 125 features gained with the application of 25 Law's masks, Run Length – depends on the number of runs with 5 groups of features, Texture Operators – 5 groups of features, Texture Spectrum – 5 groups of features. 110 samples were used from igneous (27), sedimentary (61) and metamorphic (22) types of rocks. These samples were manually imaged in a similar lightning and background condition. The features were extracted from grey-scale images and classified with the k-nearest neighbour classifier with cross-validation. Maximum accuracy of 94.21 % was gained with k = 7 on features extracted with Laws masks.

(Paclík et al., 2005) designed a system for an autonomous rock classification system. It contains tree steps: local texture derivation of an informative representation, local patch classifier, and combination of per-patch decisions into a single decision. The work is based solely on gray-level textures analysis. The work uses a co-occurrence matrix (CM). A classifier derives the decision. Several classifiers are used to compare their performance. Two types of rocks were used. The minimum error is 0.271 gained with Kullback-Leibler divergence due to the non-linearity introduced by this algorithm.

(Wu et al., 2012) developed a system which allows a user to upload an image of a rock sample to a remote server with a mobile device and receive a description of it. They used color, directionality, and granularity features of an image. The color image transformed to HIS (Hue, Saturation, Intensity) and only hue and saturation were used for color features. Ix and Iy derivatives are used to calculate directions for a given pixel (x, y), and the finite difference is used to approximate the derivative. After derivatives for all pixels were calculated, a histogram of directions was used to estimate the direction. The histogram is close to a uniform distribution for a texture without direction. The granularity is evaluated over a small patch of an image. A difference in the quantity of gray level extremes between the original patch and the median-filtered patch was calculated. Similar numbers indicate low granularity because the filter does not change the image too much, and vice versa. The classification is performed with ANFIS (adaptive-network-based fuzzy inference system based on fuzzy inference system and neural network. 375 images were evaluated in experiments performed by Wu et al. Based on the features, several membership functions were developed. Two

26

membership features distinguished between colored and achromatic rocks. And four functions for further colored rocks classification. The membership functions for directionality and granularity were established similarly. The success rate is 77.6%.

(Prince and Chitale, 2008) implemented color distribution analyses method. Authors made thresholding (separating by some value) of grayscale images in HSI (Hue, Saturation, Intensity) for the pay estimation. The authors defined two classes – pay (sand) and non-pay (shale), and this method was tested on high-resolution images.

In (2011), Thomas et al. proposed a new method for automatic lithology classification. This method is called object-based image analysis. The main aim was to detect pixels which significantly differ from each other. Four lithotypes were used – sand, shale, carbonate and no core. First, a threshold automatically separates different lithotypes from the core image. After that, the nearest neighbor classifier (Arya et al., 1998) was applied to the small data for supervised learning. The nearest neighborhood classification is based on the comparison of all datasets in n-dimensional space. Different classes were separated and placed at a maximum distance from one other. The training dataset is made from several examples of different lithotypes from the well. The following classification is being done on the samples from the same well.

The trained classifier was applied to the whole dataset to classify lithotypes. And the expert manually checked the results. Additional samples were added or removed from the nearest neighborhood model if the results were not reasonable. This method works with good performance if applied in one object as all the data are from the same distribution (core image collection). The accuracy of this method reached 94.29%. The misclassification took place in the images with empty cores or shaded areas.

(Harinie et al., 2012) used Tamura Feature Extraction and Tamura Feature Classification based on coarseness, contrast, directionality, line-likeness, regularity, and roughness, to describe low-level statistical properties of textures to determine eight types of rocks: peridotite, phyllite, pegmatite, rhyolite, aphanitic, andesite, diatomite, conglomerate. The algorithm was trained on 50 256x256 images and tested on 60 images. The test set prediction gave 87% accuracy.

(Shang and Barnes, 2012) used local colour histograms and the first (mean) and second (standard deviation or std) order colour statistics (2 features for each pixel) based on both R, G, B, and H, S, and V channels (12 features) features. Unsupervised Feature Selection Based on Data Reliability and Information Gain-based Feature Ranking used for feature selection. Data reliability is based on feature distance measurement. A feature is considered reliable (or relevant to the problem) if its values are tightly grouped. Support vector machine, K-Nearest Neighbours, and decision trees are used to classify the features. 14 unnamed types of rocks are used to train and test the algorithm with 10-fold cross-validation. A maximum accuracy rate of 93.58% is gained with an SVM-based classifier trained on 20 reliability-based selected features.

The improved color distribution analysis approach was shown in aeries of works by Khasanov et all (Khasanov, 2015, 2014, 2013; Khasanov et al., 2016; Postnikova et al., 2017). This method is based on the difference in lithology colors in HSI color space (limestones may be white or yellowish-white, sandstones – yellowish-white to green, etc.). Besides that, other petrophysical characteristics like oil saturation, porosity, permeability, cementation, etc., also can be seen through the color distribution. This approach requires several hours to choose the right color distribution range for each type of characteristic. After that, it is necessary to determine which components are responsible for the color distribution (lithology, porosity/ permeability, etc.). The authors proposed to make a database of such color distributions and train an artificial neural network by color intensities to simplify this process.

Another method was based on extracting features with Principal Component Analysis (PCA) (Wieling, 2013) and color distribution analyses. This research was aimed to evaluate bedding direction, lithology, grain size, and permeability. This work was mainly based on various statistical tools. The lithology, bedding direction, permeability and grain size determination was done on the core segmented to centimeter intervals. The detection was performed with an auto-covariance function. Information about each core image was represented in RGBD (Red, Green, Blue, Darkness) color space. This information was extracted by a centered log ratio (CLR) transformation followed by PCA.

After PCA, the information about bedding was measured by the correlation of a bedding image and two parallel lines drawn on the image. Then k-means clustering was applied to the data to gather all corresponding pixels. The classification by lithology was made by Multivariate Gaussian (MG) distribution model followed by the Minimum Covariance Determinant and Quadratic Decision Boundary (QDB). The performed operations are accomplished by linear regression of the Auto-Covariance properties for grain size and permeability estimation. Two lithology types were chosen for separation: sandstone and other rocks (coal, mudstone=shale (not to be confused with carbonates) and siltstone). The accuracy of the QDB classification model was 91% and 88% for the MG. Thus, the boundary between sandstone and other rocks is defined clearly. To separate other rocks, the contrast between siltstone and other coupled lithology classes was set by a linear decision boundary. The ability to separate this kind of data was very poor as soon as siltstone, shale, and coal were all dark by color.

In Chatterjee et al. works (Chatterjee, 2013; Chatterjee et al., 2010, 2008; Patel et al., 2017a, 2017b, 2016) crushed rocks from different mines were analyzed. Each boulder from the box was separately segmented from the image. Various features of texture and structure were extracted from the image by different methods. These features were reduced to the smaller dimensional space by the principal component analyses (PCA) method (till the 2010s) or by genetic algorithms (GA). There was no dimension reduction in the latter works (Patel et al., 2017a). Only 18 features of color distribution and intensity were used for the demands of online system development.

These features were placed into a multiclass support vector machine (SVM) – a method based on mapping the non-linearly separated problem (many features present non-linearity) to linear space. After the mapping, it can be solved as a separate linear problem. Both approaches to separate the data were used: one-versus-all and one-versus-one. This method is precise (from 90% to 99%, depending on the method and the number of lithotypes).

A massive amount of work is also denoted to ore classification problems thoroughly reviewed(Patel et al., 2017a). For example, (Khorram et al., 2017) used SVM for the classification of three different types of carbonates. The SVM method made this separation with an accuracy of up to 89%.

(Sharif et al., 2015) used 3.5x3.5 cm greyscale patches to classify the rock with Bayesian classification approach. The main goal was to develop a system for a mars-rower. 30 images of different rock types (some of them of the same origin) were taken within 50 cm distance in a similar condition to save textural information. The images were converted to greyscale, and three patches of 256x256 px were extracted from each image for analysis. Simple image statistics such as the maximum average, standard deviation, and skewness of the intensity distribution did not work for rock sample differentiation. Haralick feature extraction was used to sufficiently describe the image for an algorithm. The features were classified with a Bayesian function. Such an approach allows having a 1/30 chance that a sample is correctly classified. 8191 possible combinations of Haralick parameters were tested, and the highest score attained was 24.1 (80.3%) using 6 Haralick parameters out of 25.

(Francis et al., 2014) worked on an autonomous system for rock outcrop surface segmentation for rovers. Mars Exploration Rovers (MER) and Mars Science Laboratory (MSL) missions. These works were dedicated to the recognition of images which are received from mars. A set of methods and a pipeline for exploring a geological environment are presented. The main accent of applying such methods is in different autonomous scenarios without human guidance.

(Li and Han, 2018) developed a deep learning system for predicting 3 types of rocks: granite, phyllite and breccia. 571 samples were used for transfer learning with the Inception-v3 model. The testing set contains 9 images for each class. A total accuracy was over 80%.

(Shu et al., 2017) proposed an unsupervised method for autonomous rock image classification for Mars exploration programs. Two unsupervised approaches were tested: K-means feature learning and self-taught learning. 9 different rock types were imaged, and 700 textural images were generated from these types. Each type contained 80 textural areas with size of 128x128 px and about 1-2 cm across. Features were extracted both manually using first and second-order statistics (carefully hand-crafted by computer scientists without any geological meaning) and autonomously by unsupervised feature learning based on K-means. One-vs-all linear SVM (support

vector machine) was used to classify the features with 5-fold cross-validation. Maximum accuracy of 96.24% for manually extracted features 96.71% for feature learning and 90.32% for self-taught learning were gained.

(Valentín et al., 2017) used Haralick Textural Features, along with other textural features, extracted from each image and a Naive Bayes classifier. Each image was filtered with 4 different filters: gaussian, canny-edge, entropy and variance. A set of 104 features for each image (filtered and the original one) was calculated. Thus, 520 features were extracted for each image. With genetic optimization and principal component analysis, it was possible to achieve a 92.27% success rate for 9 rock types prediction.

(Zhang et al., 2018b, 2018a) used neural networks to identify various structures from geological images with K-nearest neighbors, artificial neural network (ANN), XGboost (extreme gradient boosting) algorithms, 3-layer CNN and pretrained Inception-v3 convolutional neural network. They used 2206 geological structure images with 12 labels, including anticline, ripple marks, xenolith, scratch, ptygmatic folds, fault, concretion, mud cracks, gneissose, boudin, basalt columns, and dike. KNN, ANN, and XGboost had poor performance, with an accuracy of less than 40.0%, 3-layer CNN overfitted on the training data, and Inception-v3 reached 83.3% top-1 accuracy.

(Harraden et al., 2019) implemented an automated core logging technology based on a high-resolution Hyperspectral Core Imager Mark-III scanner, including 3d depth distribution. The fractures are calculated automatically with a special computer vision technique. The mineral composition is calculated from the hyperspectral analysis.

(Ran et al., 2019) collected 2290 field images (the size is either 5760 × 3840 or 4000 × 3000px) belonging to six rock types. The images were cropped to 512x512px areas. The final number of training set images is14,589 for training, 4863 images for testing and 4863 images for validation. The accuracy is 97.96% gained with a 3-layered convolutional neural network.

(Liu et al., 2020) proposed application of Faster R-CNN architecture based on VGG16 convolutional neural network to classify multiple rocks in one image. 1034

images were used collected for 8 classes and further amplified to 78143 by different types of augmentation. The accuracy for multiple rocks detection is 80%.

(Fan et al., 2020) developed a lightweight convolutional neural network model. The dataset consists of 28 kinds of rock with 2566 images in a training set, 330 rock images in a validation set, and 312 rock images in a testing set. The image scale is about 10 to 20 cm. The size of an image is 224x224 px. MobileNet and SqueezeNet convolutional neural networks are used to reduce the network size and inference time. The average recognition accuracy of the two models was 94.55% and 93.27% on a test set, respectively.

(Cai et al., 2021) used ResNet convolutional neural network architecture to classify high-resolution images into 8 classes: black coal, gray black mudstone, gray argillaceous, siltstone, gray fine sandstone, light gray fine sandstone, dark gray silty mudstone, dark gray mudstone. The training dataset contained 322 images of high resolution (up to 4096x3000), which were split into 224x224 images 51,600 experimental images. A set of 33 test images were predicted by optimized ResNet with 0.996 accuracy.

(Alférez et al., 2021) had six plutonic rock types: gabbro, diorite, tonalite, granodiorite, monzodiorite, granite. The dataset consists of 71 high-resolution images split into various size images. In the end, 423 images for training and 423 images for validation were gained. Various combinations of rock images were used for classification and testing. The maximum accuracy gained with a self-constructed convolutional neural network was 0.95.

(Xu et al., 2021) trained an algorithm for full-scale rock samples from an outcrop with size of about 10 cm. A data fusion method was implemented using images and portable X-ray fluorescence spectrometers. Initial datasets consist of 13 classes with image size up to 2085 × 2390 px. The images were randomly cropped to 224x224 px; thus, 19,000 images were gained with relevant elemental data, including Fe, Ca, K, Al, Si, Mg and the mixed content of other elements. ResNet-50 and ResNet-101 models were used for image identification and data fusion. 6700 images with relevant elemental data were used as the testing dataset. The ResNet-50 model has a test accuracy of

84.99%, and the data fusion model has a test accuracy of 94.62%. ResNet-101 has a test accuracy of 86.65%, and the fusion model has a test accuracy of 94.44%.

(Alzubaidi et al., 2021) used deep convolutional neural networks for small tile classification. The authors distinguish three images: limestone, sandstone, shale, and non-core. They used various convolutional neural network architectures. A total of 76,500 images (25,500 images per each rock type) were created as cropped images ranging from 240x240 pixels to 300x300 pixels in size. A scale of 2 cm was used to crop 2x2 cm images. The dataset was then divided into 54,000 for training, 13,500 for validation and 9,000 for testing. 88.74% accuracy was received before smoothing of the label prediction and 92.00% after smoothing with ResNeXt-50 architecture.

(Alzubaidi et al., 2022) used convolutional neural networks to calculate rock quality designation (RQD) splitting the image into 5 classes: rock, fractures, crushed core, non-rock objects, and empty tray regions. The final model was trained on a training set of 6400 225x225px size images. The maximum accuracy gained with a self-constructed convolutional neural network was 98%.

(Abashkin et al., 2020; Seleznev et al., 2020, 2019) developed statistical methods for image analysis to classify different types of rocks. Various features are calculated over images like color, which may be interpreted into a rock.

(Timmer et al., 2021) used 13,545 tiles of 128x128px size are used to train a DL algorithm to automatically predict if a core photograph contains evidence of bioturbation. The final network accuracy is 88%.

(Ayranci et al., 2021) developed a model which determines 6 bioturbation indexes (BI). They used 1303 images with various BI. The final number of classes was set to 3: 0; 1–2; 3–6. With the use of VGG-16 the maximum accuracy gained was 94.7%.

(Fu et al., 2022) collected dataset with 10 lithologies: diabase; diorite; gneiss; granite; limestone; marble; monzonite; mudstone; shale; and siltstone. A total of 15,000 images were obtained. Various architectures were tested, but ResNeSt-50 gained the best result of 99.60% accuracy on untrained shots.

1.2.3. Thin section investigation

A thin-section investigation could be performed with an optical microscope. The images taken in this scale can have different scales. The microscope may have different equipment for further investigation: analyzer, uv-light, diaphragm. Several problems should be kept in mind while studying thin sections:

1) Some minerals may have similar optical characteristics in a static position. Optical characteristics are viewed in different lightning conditions.

The minerals are identified by features like color, relief, twinning, pleochroism, and extinction. Minerals like feldspar and quartz or different calcite minerals like dolomite and carbonate are differentiated by a set of detected features. Also, to easily distinct some minerals from each other, it is important to look at them in both polarized and crossed-polarized polar views; sometimes, it should be done while the stage is rotating.

2) Camera characteristics and light conditions may differ drastically.

Various cameras may have different resolutions, and the user may set various settings during imaging, including lightning, focus, and additional filters.

3) We can only observe small and distinct characteristics like grain size, composition, or distribution.

As we are looking at a small scale, the sample's naming may only be constructed based on initial characteristics.

(Młynarczuk et al., 2013) introduced a method for thin-section automated classification by machine learning algorithms based on color spaces analysis. Four pattern recognition methods were compared: the nearest neighbour algorithm, the K-nearest neighbor, the nearest mode algorithm, and the method of optimal spherical neighbourhoods. Different color spaces were used to test the methodology: RGB, CIELab, YIQ, and HSV. 9 different thin section types were used. The whole dataset consisted of 2700 images. The maximum accuracy gained on the CIELab color space and the nearest neighbor algorithm with the result of 99.8%.

(Cheng and Guo, 2017) used 4800 samples under various colour spaces: HSV, YCbCr and RGB to recognize the granularity of input image samples. Three types of rock images were used: course feldspar sandstone, medium granular feldspar sandstone, and fine feldspar sandstone. In RGB colour space, the classification accuracy achieves 98.5% with the application of a self-constructed neural network.

(Budennyy et al., 2017; Bukharev et al., 2018) created a deep-learning watershed algorithm to perform grain analysis accurately and evaluate the composition of rock. A grauwacke was segmented with about 90% accuracy.

(Wei et al., 2014) used K-means and probabilistic neural network to detect 3 classes within 600 images (400 for training): rock thin images, background images and pore target images. The average correct classification rate is around 95.12%

(Koeshidayatullah et al., 2020) used 4000 images of carbonate petrographic thin sections of different scales to train two algorithms for carbonate components detection (coated grains; bioclasts; micrite; calcite cement; replacement dolomite; and porosity) and object detection (13,000 individual carbonate components were manually labelled): ooids; peloids; foraminifera; molluscs; other skeletal grains; micrite; calcite cement; replacement dolomite; and porosity. Inception-ResNet showed a good result of 0.91 for image classification and faster R–CNN with a result of 0.84 for object detection.

1.2.4. Digital rock investigation

The highest scale (up to 1 micron) is produced by scanning electronic microscopy (SEM) and tomography imaging techniques. They are based on computer imaging of the rocks by scanning either by CT-scan or with SEM. A quantitative evaluation of minerals (QEMSCAN) may be added to SEM which allows to get the mineral map equal to thin section. Still, there are some important things to keep in mind during data processing of such data:

1) The data is different from the other scales.

   Visual rock and thin section scale investigation data gathered from the physical sample differs from the SEM and CT-scan data as it is collected on a scale where

not all details are visible to the human eye. The minerals are detected either by shape, density or the detected elements.

2) It is hard to upscale and downscale the data.

As the scale is very high not all the data should be used for modelling as some of the details may influence the result, causing only slight differences (Pickup and Hern, 2002) in reservoir simulation, most of such micrometre-scale information is usually neglected due to high computational cost. But such information will be valuable in the case of oil production intensification techniques.

3) A lot of factors may influence the scanning results.

The operator experience, aim of the study, and the equipment's quality directly influence the quality of the data received for interpretation. Compared to optical imaging, the final data interpretation may be corrupted due to some physically induced process that may not be directly visible. In CT imaging, the result may change depending on the reconstruction parameters.

(Chauhan et al., 2016) made a comparison of different segmentation algorithms for CT images. Both unsupervised (k-means, fuzzy c-means (FCMs) and self-organized maps (SOMs) and supervised (feed forward artificial neural network (FFANN) and LS-SVM) machine learning techniques were used for segmentation of pore, mineral and matrix phases). The averaged porosity obtained for various samples (andesite, Berea sandstone, Rotliegend sandstone and the synthetic sample) is in very good agreement with the respective laboratory measurement data and varies by a factor of 0.2.

(Wang et al., 2021) compared various convolutional neural networks for segmentation of CT-scan with SEM and QEMSCAN data. Voxelwise accuracy is relatively high for all tested models (85%–95%), but for each mineral, the accuracy highly differs form architecture to architecture.

1.2.5. Summary

Different authors have done lots of work to help an expert make his life easier. Various approaches have been tested, most of the reviewed works are based on measurements taken on a test set, which is similar to the training set, which is not always

the case. There are a lot of problems which may appear during testing the same approaches on a new set of data. So, the problems should be further investigated. After that, the prepared algorithms may be included in a system which will help the researcher to use the developed models.

While a researcher looks through the literature, some time may be spent to understand the terminology used. The following section will introduce some general terms to help a researcher in his investigation.

1.3. Terminology review

As the work combines both the geology and machine learning/computer vision (which are united in computer sciences) fields of knowledge, some general terms of both scientific fields should be described to ease the accessibility of this work and introduce the problems which may appear during the study of these directions. As geology and computer vision are continuously developing scientific areas, various definitions from different sources are collected in the following subsections to show different point of views on some of the terms and reveal the ambiguous moments that may confuse a researcher during his study. The ambiguity appears not only because of the multiple uses of the same term in different situations but also because of some linguistic barriers.

1.3.1. Geology

Geology is the scientific study of the composition, structure, and history of the Earth. More recently, many branches of geology have become established as disciplines in their own right (e.g., geochemistry, geomorphology, geophysics, mineralogy, and paleontology) (Allaby, 2013).

Due to various people's views about geology, there are a few terms where several geologists may be certain in mutual definition (such as mineral, fossil, cement). Even geologists of the same organization may define some terms differently. Due to an overwhelming number of regulatory documents, finding the mutual definition of specific terms is hard. Here is a short description, history of definitions and reasoning behind the author's opinion provided for the most commonly confusing terms which may be used in this work. As the work assumes that some of the words used are well

defined in <mark>wildly</mark> available wordbooks (such as Oxford Advanced Learner's Dictionary (*Oxford Advanced Learner's Dictionary*, 2020)) the author will refer the reader to such a dictionary in case of any misunderstanding.

This work is generally devoted to the study of rocks. The rock is a type of sediment usually formed from a composite of minerals. It can be solid or soft. The origin of rocks may be various, the same as the classification of such origin. Here a classification based on the nature of processes ("British Geological Survey (BGS) Rock Classification Scheme," n.d.) is used, which includes sedimentary, igneous and metamorphic rocks. Some researchers may separate consolidated sediment as a rock with a separate name (like siltstone) and loose sediment with another name (e.g., silt). As this work is based on image analysis of the core, the image in most cases may not represent the sediment's looseness, but the grain size – silt and siltstone will be referred to as the same class.

Lithology, lithotype, lithogenetic type and **facies.** A few definitions of both facies and lithotype exist:

1. **Lithology** is a **physical characteristic** of rock, including colour, composition, and texture (*Collins COBUILD Advanced Learner`s Dictionary*, 2018).
2. **Lithology** (from Greek *lithos* – rock, *logos* – to study) is a fundamental part of geology aimed for **composition, formation and genetical study** of sedimentary rocks (Yapaskurt, 2016).
3. **Lithology** is a geological term used to describe the **types of rock** formations (Guo, 2019). Three main types are commonly defined: sedimentary, metamorphic, and igneous. In reservoir analysis, the lithology is identified by geologists using core samples taken from the exploration wells (Guo, 2019).
4. A **lithotype** is a term which describes **coal types**. Lithotype descriptions are primarily used to characterize coal beds in geological investigations (Malvadkar et al., 2004). Each coal lithotype is a macroscopic

component of coal that is sufficiently distinguishable from other coals (like vitrain, clarain, durain or fusain).

5. Lithological type or **lithotype** – **it is a textural** (granulometry) type of rock with several **first and second-order genetic** (environmental) **attributes** (Sirotin et al., 2004a). Lithogenetic type includes one or several lithological types with specific genetic characteristics (Sirotin et al., 2004b).

6. As stated in VSEGEI dictionary (Petrov, 2011), lithotype is a geological body that includes layers or several layers with common sedimentological and diagenetic attributes that point to the environmental conditions.

7. As stated in GOST (*GOST R 53375-2016. Oil and gas wells. Geological-technological logging. General requirements.*, 2016) standard, a **lithotype** is a **type of rock** identified by a number of lithological attributes.

8. The term **lithofacies** may be shortened to **facies** by some experts. Lithofacies is defined as a body of rock with certain specified attributes that distinguish it from other rock units (Leeder, 1982).

9. The **facies** (the environmental facies model) is the whole set of attributes possessed by the deposited sediment laid down in a particular environment (Leeder, 1982).

In Russia **lithology** term has only one meaning which is the study of rocks. English terminology has a dual meaning. In this work, lithology means the study of rock. Several concepts were used to split the use of lithology terms: rock type and lithotype.

The term **rock type** is used to describe a rock's mineral composition, including its original formation (basalt, sandstone, limestone, etc.). The rock type can be a soft or hard rock that has a fixed structure and texture at the current time.

Here is the definition of **lithotype** from GOST (*GOST R 53375-2016. Oil and gas wells. Geological-technological logging. General requirements.*, 2016) standard describes rock type and structure.

The facies is an overused term (Moradi et al., 2019). The term "facies" introduced by A. Gressly in 1838 (Gressly, 1838) originally meant the lateral change of a stratigraphic unit's rocks, structures, and fauna. Its meaning has been broadened to express a wide range of geologic concepts: environment of deposition, lithologic composition, biodiversity interchange, or tectonic and metamorphic associations. Also, an experienced sedimentologist may mix facies and lithofacies as he can propose a specific environment for lithofacies on the go, which may cause a misunderstanding for a newcomer. In this work, the term **facies** will not be used. The term **lithofacies** will be used instead. Several lithotypes may exist in one lithofacies.

The structure has a reversed translation, "текстура" (which is pronounced as texture) in Russian. It means an arrangement and organization of interrelated elements in a material object or system, or the object or system organized so. A **sedimentary structure** (Allaby, 2013) is the external shape, the internal structure, or the forms preserved on bedding surfaces, generated in sedimentary rocks by sedimentary processes or contemporaneous biogenic activity.

The same as structure, **texture** has a reversed translation, "структура" (pronounced as structure), and in sedimentology, it means the relationship between the materials of which a rock is composed (Allaby, 2013). In petrology, the meaning is similar sizes and shapes of particles in the rock and their mutual interrelationships (Allaby, 2013). In pedology, it is the proportions of sand, silt, and clay in the fine earth of a soil sample, which give a distinctive feel to the soil when handled, and which are defined by classes of soil texture (Allaby, 2013).

The structure and texture will be used "as is" as defined in English literature.

Summary

Basically, there are several known to every geologist term, but others may be interpreted differently depending on the background and experience of the geologist.

That happens due to various deposits and environments presented to the geologist, its understanding of formations and sometimes due to linguistic barriers. This work will use several general terms: lithotype, rock type, structure and texture.

1.3.2. Computer science

Like in geology, computer science has various definitions for various terms. Even the definition of computer science can differ (McGuffee, 2000) from one author to another. This may happen either because different authors want to distinct or include various applications to the research or because some new methods and applications have developed over time. Also, in most cases, it is hard to find a strict definition of any term in the book or paper. In most revised books, each term described here had a separate paragraph discussing what this term is about without a definition.

The definition of computer science (the discipline of computing) is the systematic study of algorithmic processes that describe and transform information. Their theory, analysis, design, efficiency, implementation, and application could be different (Denning et al., 1989). According to the Oxford dictionary (Butterfield et al., 2016), it includes: "programming; information structures; software engineering; programming languages; compilers and operating systems; hardware design and testing; computer system architecture; computer networks and distributed systems; systems analysis and design; theories of information, systems, and computation; applicable mathematics and electronics; computing techniques (e.g. graphics, simulation, artificial intelligence, and neural networks); applications; social, economic, organizational, political, legal, and historical aspects of computing".

This work mainly uses the applicable mathematics area of computer science, including computer vision and machine learning techniques and methods.

Computer vision is the transformation of data from a still or video camera into either a decision or a new representation to achieve some particular goal (Bradski and Kaehler, 2008). Various information may be analyzed together with the image to help the computer understand the picture. Important to note that the computer only operates with numbers, so various methods are applied to help it understand the image, like machine learning techniques.

Machine learning is about extracting knowledge from data (Müller and Guido, 2016). It is a research field at the intersection of statistics, artificial intelligence, and computer science. And a set of methods that can automatically detect patterns in data, and then use the uncovered patterns to predict future data, or to perform other kinds of decision making under uncertainty (Murphy, 2012).

A separate term is **machine vision** which might confuse a reader as it is very similar to computer vision. According to (Fisher et al., 2014) it is: "a general term for processing image data by a computer; often synonymous with computer vision. There is a slight tendency to use "machine vision" for practical vision systems, such as industrial vision, and "computer vision" for more exploratory vision systems or for systems that aim at some of the competences of the human vision system."

Summary

Some terms in computer science may have similar definitions, which may intersect as they developed simultaneously from various fields of application. The same as in geology, computer science techniques were formed rapidly, which caused some pluralism of the opinions for various terms. Still, many methods developed during the time are widely used and united within computer science. A part of these methods is described in the next section.

Chapter 2.   Methods.

This work uses different mathematical and computational algorithms based on computer vision, machine learning and deep learning techniques applied for image analysis. The work doesn't ~~pretend to have a~~ solid mathematical basis for applied algorithms as it is based on well-developed methods, so only the algorithm's intuition and important notes are described. For strong mathematical formulation, the reader can refer to the works described here. The author will only use the formulas in cases where it is necessary.

There are several steps in image processing. Each followed subsection will describe and reference a workflow and intuition developed by highly qualified experts in the field. Firstly, a feature from the image should be extracted. Secondly, the feature can be preprocessed by different dimensionality reduction methods. Finally, the feature can be sent for classification to a machine learning or deep learning algorithm. Next, the classification can be further processed by different statistical algorithms. Computer vision includes all the steps. All these steps should be implemented within various programs connected with special communication protocols for production purposes.

2.1. Feature extraction

There are broad methods to extract different features from images. The feature can describe image granularity, corners, and many other useful characteristics a human can read from an image.

The following feature extraction methods were used:

- Principal component analysis (PCA) (Tipping and Bishop, 1999; Wold et al., 1987).
- Histogram of oriented gradients (HOG), like in (Dalal and Triggs, 2005; Lowe, 2004; Viola et al., 2003).
- Pretrained convolution neural network feature extraction (PNNF) (Intrator, 1990; Wiatowski and Bolcskei, 2018).

The principal component analysis is based on a data projection onto a principal subspace that minimizes the square reconstruction error.

The histogram of oriented gradients is based on a calculation of gradients over the image. The gradients are calculated in small cells over the image.

Pretrained convolutional neural networks can have features like common extractors. In most cases, they extract much more useful information as they can be trained for specific extraction tasks.

2.1.1. Principal component analysis

The method minimizes the squared reconstruction error when it finds a certain basis with certain properties onto which the d-dimensional data vectors variance under projection is maximal. PCA is nothing other than a low-rank matrix approximation (Halko et al., 2011; Wold et al., 1987). If we apply it to an image, the PCA reduces (compresses) the matrix into a lower dimension and can restore it with data loss (Figure 1). During compression, only high-frequency patterns may be stored.
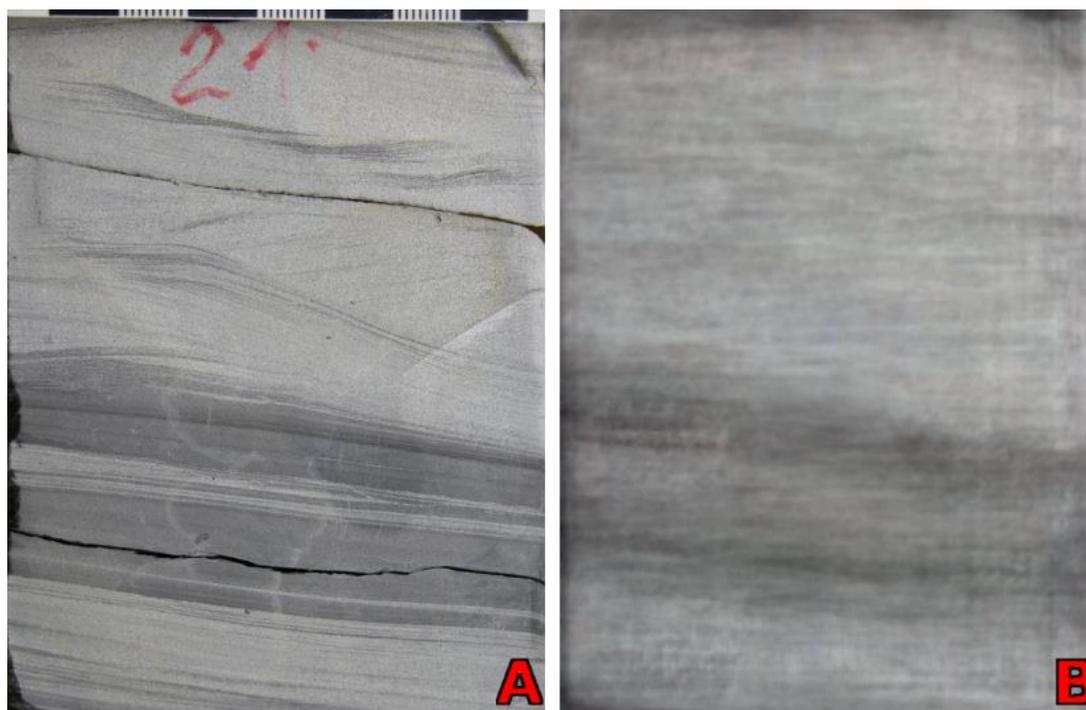
Figure 1. A principal component analysis image compression example. A – the original image, B – PCA compressed image with a 0.95 information retrieval. The principal component may be extracted from an image used for the classification algorithm. The image demonstrates how the principal components work. This algorithm aimed to store high-frequency data only, so details like the rule (on the top of an image) and small structures were not preserved.

2.1.2. Histogram of oriented gradients

A histogram of oriented gradients (HOG) goes through the image and calculates gradients over a small sliding window (e.g., for a 128x128 px image, an 8x8 filter is applied). A normalization over the smaller regions is used, representing the result as normalized vectors over the image. The vectors are concatenated in a longer feature vector, which reduces the dimension of an image. The vectors are usually supplied as a 2d- matrix to the algorithm, so the spatial gradient distribution doesn't account for explicitly.
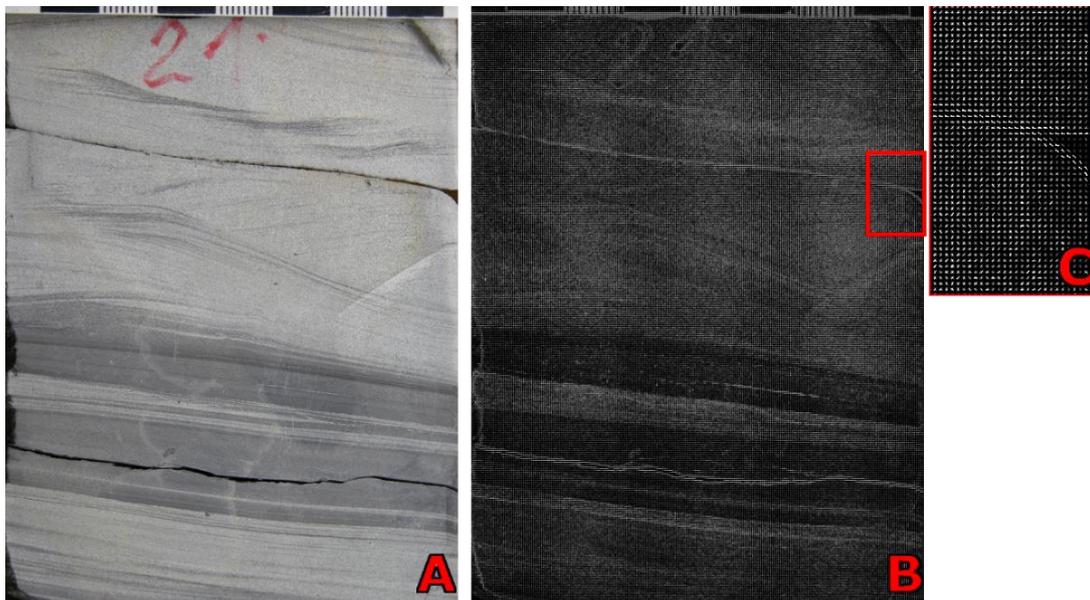
Figure 2. A Histogram of oriented gradients application example. A – the original image, B – visualization of gradients extracted by HOG, C – enlarged area with gradients drawn with white arrows. HOG calculates the gradient between the two nearest regions was calculated. In comparison to PCA (figure 1), various structures were preserved except for color.

2.1.3. Pretrained convolution neural network feature extraction

The idea to use an artificial neural network (ANN) as a feature extractor comes from an approximation of principle component analysis with ANN (Intrator, 1990), which is shown by Oja (1982). The feature extracted with a pre-trained neural network may be better suitable for any task in the case of computer vision due to its nonlinearity. Convolutional neural networks (CNNs) are built with a set of filters which may extract various information about the image (they will be described in section 2.3.1.). As it

meant by name – they should be trained first. After training, the CNN will be used to extract features.

2.2. Machine learning

Machine learning is a set of methods that allows a computer to provide an algorithm that will solve problems without human guidance. Machine learning uses mathematical models and computer science to construct such an algorithm. A supervised and unsupervised task can be solved with this method. The supervised task is based on a dataset consisting of different data and labels (answers). An unsupervised task has no labels in the dataset.

2.2.1. Training a machine

The machine should know how to differentiate positive and negative cases to train an algorithm in supervised ways. In the case of unsupervised learning – it should know how to measure distance or form some groups by some rules. Another algorithm optimizes a set of objective functions to give such information. During the optimization (training), the model is iteratively going through the dataset to predict the correct answer or group the data into meaningful form with an objective function. The process of model training is usually called model fitting.

Objective function

The objective function (also known as a criterion, in case of minimization – cost function, loss function, or error function) is a function which penalizes the model for making a wrong decision on a prediction. Only loss function and objective terms will be used hereinafter. The model can fit itself to the dataset only with the use of the loss function as it measures the model's success rate. A set of loss functions can be used in different cases. For the classification task (Wang et al., 2020): Cross Entropy, Negative Log Likelihood, Margin Classifier, L1 Hinge Error, Mean Square, Cosine Error.

For regression (Wang et al., 2020): Square loss, Absolute loss, Huber loss, Log-cosh loss, Quantile loss, ε-insensitive loss, Square error, Distance error, Reconstruction error, and Negative variance.

For semantic Segmentation (Jadon, 2020): Binary Cross-Entropy (Ma et al., 2004), Weighted Binary Cross-Entropy, Balanced Cross-Entropy, Focal Loss, Dice Loss, Tversky Loss, Exponential Logarithmic Loss, Distance map derived loss penalty term, Correlation Maximized Structural Similarity Loss, Hausdorff Distance Loss, Log-Cosh Dice Loss, Sensitivity-Specificity Loss, Shape aware loss, Combo Loss.

Each function has different properties and was designed for specific tasks in machine and deep learning. During machine learning algorithm training, humans aim to find the best loss function that will result in the best performance on a given data set, which may result in many experiments. Still, not only the loss function does the job, as the data should be verified, and a bunch of optimization methods, the same as hyperparameters, should also be selected carefully.

Optimization

Optimization is an operation which finds an argument value which maximizes or minimizes an objective function. The optimization algorithms can be different. They all have their pitfalls in various cases. The pitfalls can be caused by saddle points, local minimum or local maximum, which appear during the optimization process. In machine learning, the optimization function tries to reach the global minimum of a function, but it is not guaranteed to do that at the end (Kawaguchi et al., 2019). That's why machine learning algorithms are continuously improving. The optimization algorithms in machine learning can be (Sun et al., 2020): 1) first-order optimization methods, represented by stochastic gradient methods; 2) high-order optimization methods, and 3) heuristic derivative-free optimization methods. The optimisation is usually done with a gradient descent algorithm or modification. The most known optimization algorithms for classification are (Ruder, 2016): Stochastic Gradient Decent (Robbins and Monro, 1951), Adagrad (Duchi JDUCHI and Singer, 2011), Adam (Kingma and Ba, 2014), Nesterov accelerated gradient (Nesterov, 1983).

A convergence of an optimization algorithm is the main problem a researcher can stumble into, as not every task can finally converge to a sufficient point. That depends not only on the data but also on those initial points called hyperparameters.

Hyperparameters

Each machine learning algorithm and a set of parameters a researcher should tune to get the best performance of an algorithm (Smith, 2018) are called hyperparameters. The parameters could differ. Any parameter which can be changed inside an algorithm is a hyperparameter, even a loss function and optimization algorithm. A researcher should spend a lot of time and perform experiments searching for optimal hyperparameters, especially if the task he solves is new.

Types of predictions

The whole training process is aimed at getting some results. This could be some class (classification task) which will look like a set of numbers that can be transformed into meaningful names or a number (regression task) which may also be further processed. The other forms of predictions are mainly derived as an expansion of either classification or regression task (for segmentation, it is a whole image pixel-by-pixel classification, for time series – to predict a series of numbers, etc.).

2.2.2. Classification

The classification algorithms which were used are Random Forest (RF) (Breiman, 2001) and Support vector machine (SVM) (Fan et al., 2008). For semi-supervised classification Mean shift algorithm (Comaniciu and Meer, 2002), Feature Agglomeration (Duda et al., 2001) and K-mean clustering (Lloyd, 1982; Macqueen, 1967) were used.

Random forest

Random forests grow decision trees based on the training set. Each iteration develops a new leaf of such trees based on some rule generated during training. Different methods for creating such trees are described in (Breiman, 2001). It can be grown randomly by selecting both values and branching or propagating the branches using the Adaboost algorithm (Breiman, 2001). Some authors (Breiman, 2001) argue that random forests cannot overfit. Still, they have a limited generalization error which depends on the strength of the individual classifiers in the forest and their correlation.

Support vector Machine

The method constructs support vectors which separate the data on different classes and works like PCA. Instead of approximation, it tries to maximize the distance between data objects to a hyperplane which separates the objects (Fan et al., 2008).

Mean shift

It is a clustering algorithm aimed to help analyse a complex multimodal feature space and delineate arbitrarily shaped clusters in it (Comaniciu and Meer, 2002). It is based on features density estimation, representing the empirical probability density function (p.d.f.) of the represented parameter. Dense regions in the feature space thus correspond to local maxima of the p.d.f., which are the modes of the unknown density. Once the location of a mode is determined, the cluster associated with it is delineated based on the local structure of the feature space (Comaniciu and Meer, 2002).

It can be used for any type of data to create clusters either at a pixel level or with a feature extracted from images.

Feature Agglomeration

It is a hierarchical clustering algorithm based on Ward's agglomerative clustering method that is based on a classical sum-of-squares criterion, producing groups that minimize within-group dispersion at each binary fusion (Murtagh and Legendre, 2014). Important to note that the user may manually set the number of clusters; if not – a distance threshold should be set, which will eventually separate different regions into distinct clusters.

K-means

It is a centroid-based clustering algorithm which minimizes the within-cluster sum of squares (or variance) and separates different clusters based on that rule. Initial points for clusters may are defined either randomly or with a special rule. A user defines the cluster number. The most used algorithm for cluster centroids seeding is kmean++ algorithm which optimizes the seeding to prevent minim (Arthur and Vassilvitskii, 2007).

2.2.3. Segmentation

Following unsupervised superpixel segmentation algorithms were used: Felzenszwalb-Huttenlocher (FH) based on graph construction (Felzenszwalb and Huttenlocher, 2004), Quick Shift (QS) (Vedaldi and Soatto, 2008) comparing pixels in a reduced dimensionality, Simple Linear Iterative Clustering (SLIC) (Achanta et al., 2010) which use k-mean clustering, Region Adjacency Graph (RAG) (Tremeau and Colantoni, 2000) which unites two previous methods, and unites the most similar colors in a graph, Selective Search (SSh) (Uijlings et al., 2013) similar to RAG algorithms with larger features number (structure, objects size, etc.).

Felzenszwalb-Huttenlocher

The algorithm was designed to be broadly useful for image segmentation, like edge detection algorithms (Felzenszwalb and Huttenlocher, 2004). Two main aspects of a segmentation algorithm were established:

1) It should capture perceptually important regions, which often reflect global aspects of the image.
2) It should be computationally efficient, running in time nearly linear to the number of image pixels.

The method is based on selecting graph edges, where each pixel is a node in a graph. Weights on each edge measure the dissimilarity between pixels. Each pixel compares to the other with a pairwise region comparison predicate. A segmentation produced by this algorithm obeys the properties of being neither too coarse nor too fine (Felzenszwalb and Huttenlocher, 2004).

Quick Shift

The method uses the kernel medoid shift algorithm as a driver to create clusters of data and quick shift to move each data point to the nearest neighbor. The algorithm unites all the data in a tree first, then separates the branches of the tree that are longer than a manually set threshold. It can balance the under and over-fragmentation of the clusters by choosing an actual parameter (Vedaldi and Soatto, 2008).

Simple Linear Iterative Clustering

SLIC performs a local clustering of pixels in the 5-D space defined by the LAB values of the CIELAB color space and the x, y pixel coordinates) (Achanta et al., 2010). It has only one parameter – the number of superpixels which should be placed on an image. A new distance measure that considers superpixel size was introduced in the paper. It enforces color similarity and pixel proximity in this 5-D space such that the expected cluster sizes and their spatial extent are approximately equal.

Region Adjacency Graph

The algorithm is based on the region's growing process. Firstly, it coarsely pre-segments the image with either region growing, watershed or image of local second-order moments algorithm. Secondly, it refines the pre-segmentation previously obtained with Region Adjacency Graph. Each segment unites in a group of simple- and second-order- connectivity views. The groups unite if their color distribution is sufficiently close. This merging provides a new partition of the image plane – "a fortiori" (Tremeau and Colantoni, 2000).

Selective Search

The method is based on a hierarchical understanding of an image. The algorithm's primary goal is to generate a class-independent, data-driven, selective search strategy that generates a small set of high-quality object locations (Uijlings et al., 2013). It uses a bottom-up approach for object groping. To get the first groups of pixels Felzenszwalb-Huttenlocher algorithm was used. These groups are united by a greedy algorithm to iteratively group the most similar regions. At the end of iterations, only one region exists. A special diversification by complementary colour spaces, complementary similarity measures and starting regions methods is applied to support the grouping and get separate regions. Finally, the produced regions are ranked and compared to each other. If the regions are similar – they are united (Uijlings et al., 2013).

2.3. Deep Learning

Deep learning is a subset of machine learning algorithms. There is no exact boundary when machine learning turns to deep learning. Still, an algorithm can be safely regarded as deep learning as the study of models that involve a greater amount

of composition of either learned functions/concepts than traditional machine learning does (Goodfellow et al., 2016). Most researchers refer to an algorithm as a deep learning algorithm when they use an artificial neural network (perceptron-based or other) with a layer depth of more than three.

Different algorithm types may be used for creating a deep learning system. Like machine learning algorithms they can be used for regression and classification.

As deep learning algorithms are large and complex, the dataset for training of such algorithms should also be sufficiently large (thousands and thousands of data points with relevant answers).

In this work, convolutional neural networks were used for both segmentation and classification tasks.

2.3.1. Convolutional neural networks

The convolutional neural networks are based on a linear operation created for image filtering called convolution. The convolution helps extract different information types from an image, like edges.

Mathematically, the convolution (Figure 3) used in a CNN and most machine learning algorithms is a cross-correlation (Goodfellow et al., 2016; Rosebrock, 2017).

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n) \ (1)$$

The CNN usually contains a large number of convolutions (up to hundreds) randomly initialized at the beginning, and the class prediction is random. During training, the filters "learn" to extract different types of features from the data to predict the correct label that the data have.
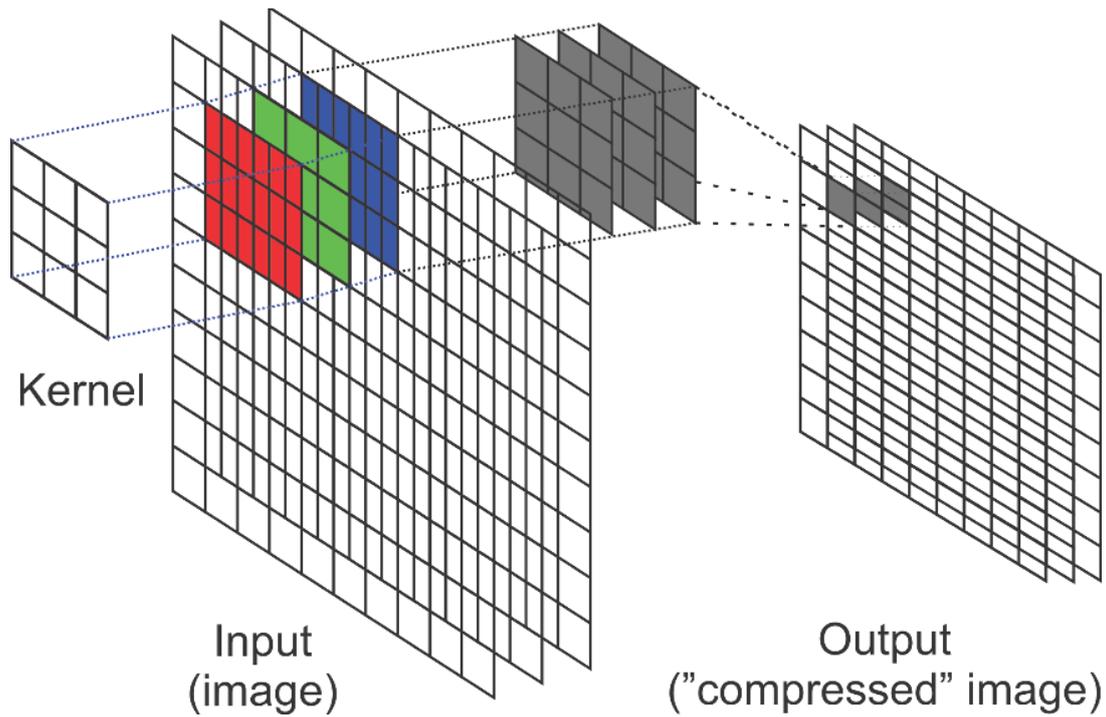
Figure 3. An example of convolution happening in each convolution layer inside CNN. The information from an image is transformed from a human-readable format (RGB channels) into a machine-readable format (pixels with condensed details). After the first layer – the RGB channels are not used, and each next layer further "compresses" the information.

2.4. Evaluation

2.4.1. Metrics

There is always a question of how well an algorithm performs. This question can be answered with a specific function that can measure success's value. This function is called metrics and can be created for any purpose and task.

The results were evaluated on testing set using various metrics (Fawcett, 2006; Metz, 1978) and by visual screening of new unlabeled images as an application of an algorithm in the real world. A part of the visual screened results was additionally labelled to evaluate the performance with metrics.

The metrics used are:

- $precision = \frac{TP}{TP+FP}$ (Fawcett, 2006; Metz, 1978) (1)

- $recall = \frac{TP}{P}$ (Fawcett, 2006; Metz, 1978) (2)

- f − score $F_{\beta} = \frac{(1+\beta^2)*precision*recall}{(\beta^2*precision)+recall}$ (Sasaki and Fellow, 2007) (3)

53

- intersection over union $IOU = \frac{TP}{(FP+TP+FN)}$ (Rahman and Wang, 2016) (4),

where TP – true positive, TN – true negative predictions, FP – false positive, FN – false negative labels to predict (the number of examples).

Precision (1) describes an amount of data predicted correctly out of all predicted values. Recall (2) represents the amount of data detected out of all data. The f1 score (3) means an equal contribution of precision and recall (β=1), and f2 (3) implies inclination to recall (β=2). Metrics 1-3 were used both for classification and segmentation tasks. Some metrics will not work with segmentation tasks (Rahman and Wang, 2016) as areas of labels are compared. An area-comparing technique is used for segmentation called IOU. IOU (4) describes the prediction and target mask intersection. A demo example was created to illustrate the calculation (Figure 4).
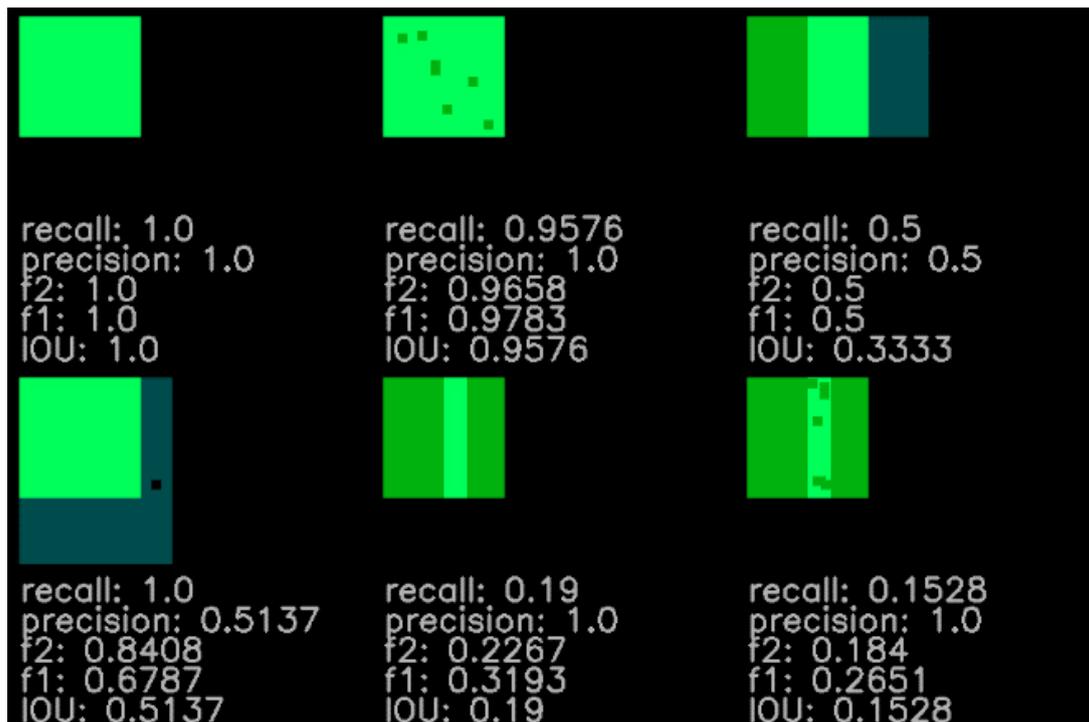
Figure 4. Examples of metrics calculation for segmentation task. The ground truth is marked as green, the intersection with the mask is light green, and the mask is light blue. The calculated metrics for different types of intersections are below the squares.

2.4.2. Unboxing black box

Machine learning algorithms are usually called a "black box", as during the training process, human is not directly involved in the tuning of the model (Ribeiro et al., 2016; Ruder, 2016; Rudin, 2019). The algorithm receives some data and answers the researcher. There is no specific way to understand the decisions of such algorithms. But there are a few ways to make the decisions being made during prediction a little bit clearer for a human.

The first way is to build a separate algorithm that could be trained to extract different features from the data and recursively give it to the model to see how the predictions changes if the feature changes (Ribeiro et al., 2016). These algorithms may be represented by different low-level interpretable models like decision trees. Ribeiro and his co-authors (2016) designed a special algorithm called Local Interpretable Model-agnostic Explanations (LIME) for such an operation. Such an algorithm may help to understand which parts of data are used for classification and how they coincide with human perception.

Another way which can be applied to neural networks is to use feature maps (Chollet, 2017) extracted from a neural network. The feature map in the case of CNN is a 3D tensor with width and height and a number of channels (depth) produced by the hyperparameters of the network. Its depth can be arbitrary since the output depth is a layer parameter. It stands for what is called "filters". Filters encode specific aspects of the input data. A single filter could encode different concepts, like the presence of a crack or structure on the image and other important characteristics which could be valuable for the final decision.

2.5. Programming languages frameworks

A set of computer languages were used to do scientific research on this thesis and provide a system based on the research.

A research part was written in Python (version 3.7) (Van Rossum and Drake, 2011). Different Python libraries were used:

- OpenCV (4.1.1 (Bradski, 2000))

- Numpy (1.18.1 (Travis, 2006))

- Augmentation library Albumentation (0.4.1 (Buslaev et al., 2018)).

- The Segmentation Models library (1.0.0 (Yakubovskiy, 2019)) was used to test a hypothesis based on the use of convolutional neural networks (CNN) applied for segmentation.

- Keras library was used to build CNN's architecture (2.3.1 (Chollet et al., 2015)) with Tensorflow backend (1.15) (Martín Abadi, Ashish Agarwal, Paul Barham et al., 2015) and Pytorch (1.3 (Ketkar, 2017)) with the use of one Nvidia 1080TI GPU instance.

- Matplotlib (3.1.1 (Hunter, 2007)) was used for plotting the results.

- Scikit-Learn (0.22, (Pedregosa et al., 2011)) was used for model tests.

A system for production purposes was based on Golang (backend), Apache Kafka (message broker), PostgreSQL (scheme-oriented database), MinIO (object-oriented database), and Vue.js (frontend).

Chapter 3.   Dataset preparation

The dataset is the greatest value of a machine learning algorithm. Two tasks were solved: classification and segmentation. A classification task requires a dataset with labelled images (each image has a class label). The segmentation task requires a mask based on an image. Each pixel of this mask has some class.

Each dataset was collected separately from different data sources. Most of the data is collected from NOPIMS (National Offshore Petroleum Information Management System, IODP (Integrated Ocean Drilling Program), and RosGeolfond (Russian federal geological fund). The dataset consists of images of full-bore cores, which are placed in core boxes (Figure 5). Over 10000 full-bore core box images were collected.
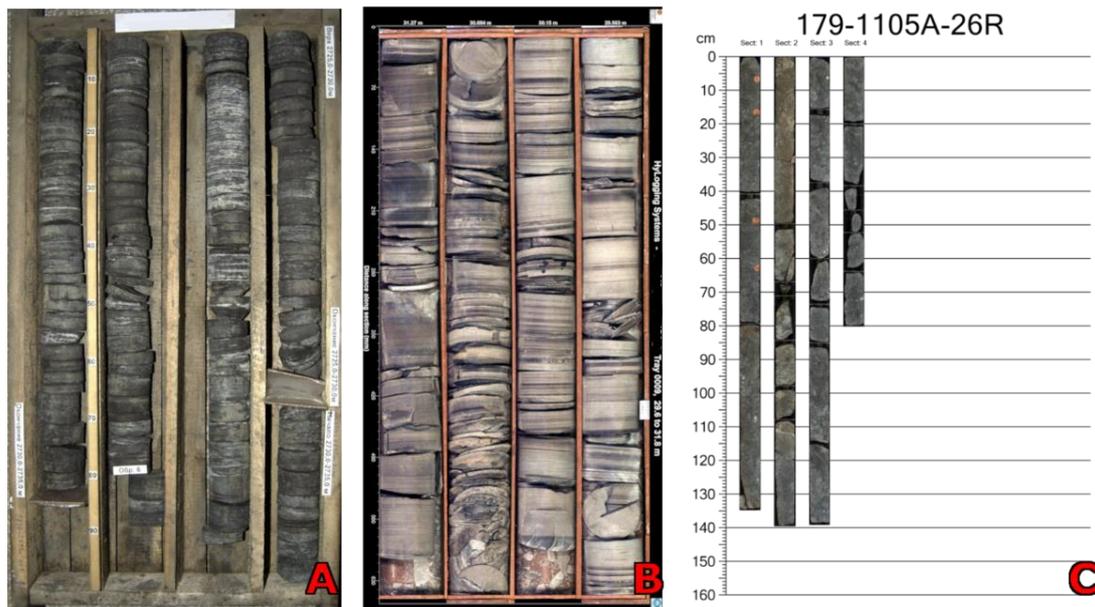
Figure 5. Datasets examples. A - RosGeolfond (Russian federal geological fund), B - NOPIMS (National Offshore Petroleum Information Management System, C - IODP (Integrated Ocean Drilling Program).

The images were stored in a form of document (.doc, .docx), table (.xlsx, .xls), vector graphic drawings (.cdr), and as image (.jpg, .tiff). A set of preprocessors and extensions were developed in Python and VBA programming languages (Figure 6) to convert all files into a unified image format (.jpg) as the most commonly used format with high compression quality rate to store images.

After converting, core box images were broken down into separate core columns, further sliced to 10cm scale images, and placed in one folder.
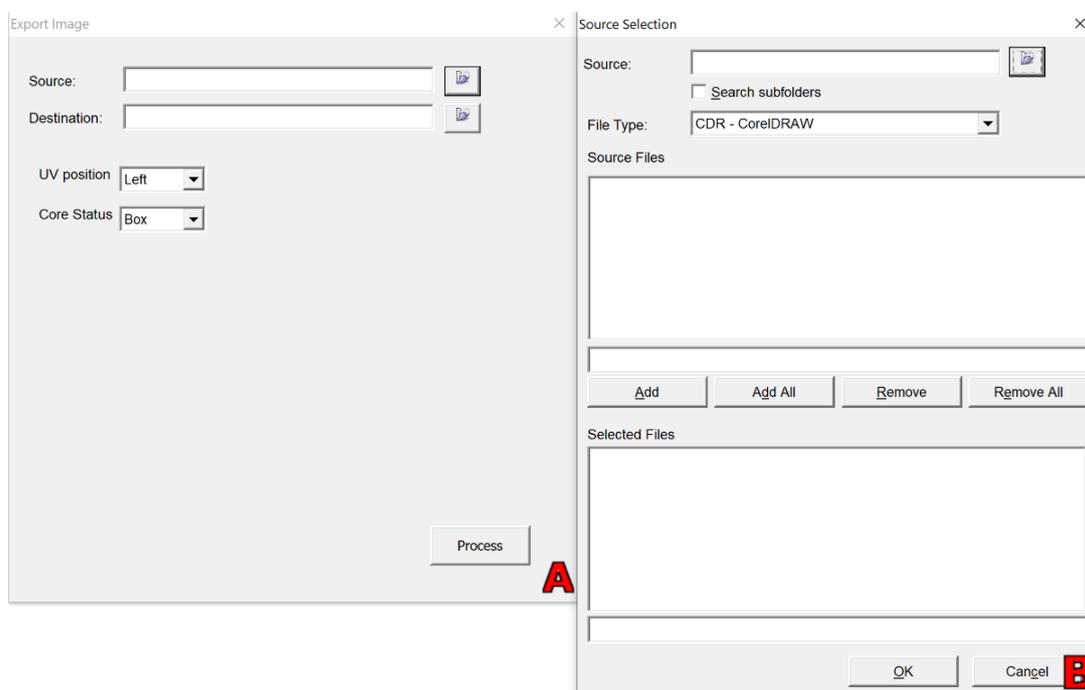
Figure 6. A preprocessor for .cdr file was created to extract images as a file (.jpg) from the vector graphic files. A – a window to select the source folder, which contains folder and subfolders with .cdr files; B – a window to select files from those which were found to convert into image (.jpg).

In addition to images a lithological description for images was collected in a form of reports (.doc, .docx, .pdf) and tables (.xlsx, .csv, .xls, .txt, .wam, .waw). These descriptions were also processed manually and with python scripts to a unified format of tables (.xlsx).

The description is usually performed in intervals (e.g. from 10 to 11.5 m – sandstone, 11.5 – 12.9 – limestone, etc.). These intervals should be transformed into continuous logs with a discretization of 10 cm to connect the description and core columns images. After such transformation, another problem may appear – some of the data can be wrongly correlated with depth due to wrong depth referencing of the description or image. A unique preprocessing tool was developed to ensure data depth consistency (Figure 7).

After such data preparation, it can be further labelled for classification and segmentation. The following labelling process differs for these tasks as each task

requires different types of labels. The classification task requires classes and segmentation – pixel labels in the form of an image-sized mask.
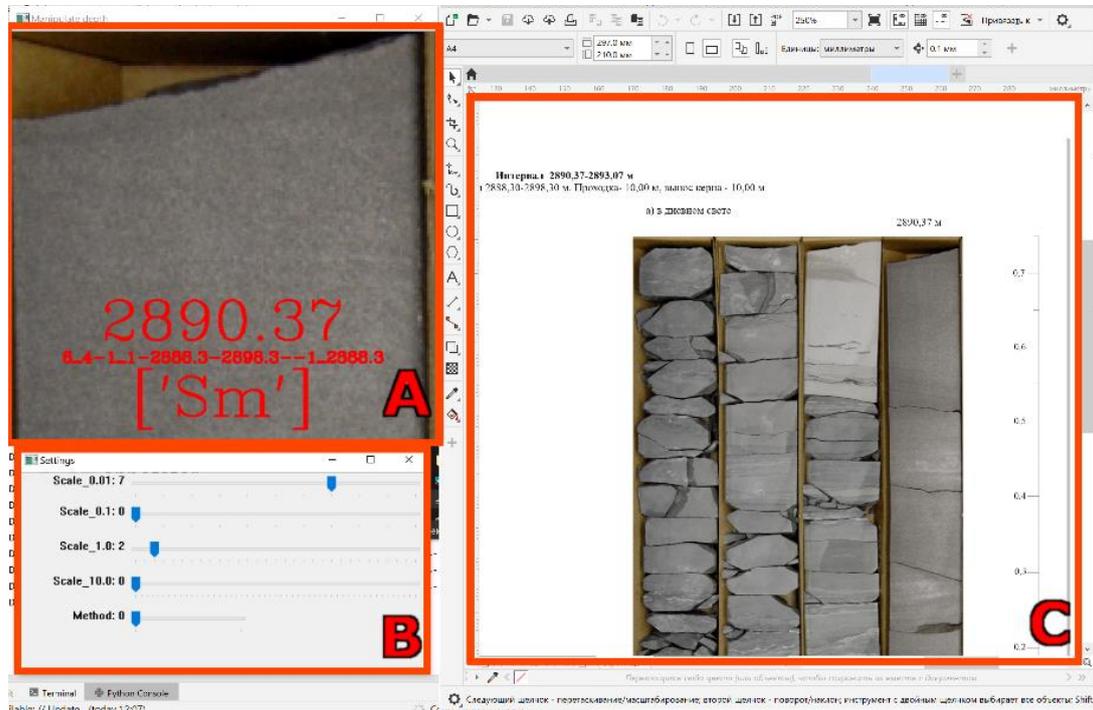
Figure 7. Developed a tool for description depth matching to a core image. A – current 10cm part of a core with depth referencing information; B – a window to add or subtract from the image depth; C – current image for depth matching.

3.1. Classification

For the classification task, the 10 cm data samples should be rearranged from folder to directories according to lithology or class from the description.

Machine learning algorithms require a well-prepared set of data. A special classification task data validation tool was developed (Figure 8). It allows an expert to process all the folders inside a path as a class. It enables a user to compare the current image with an expert-ensured database to check the final quality and relabel pictures in case some data was mislabeled. Each dataset has been checked with this program.
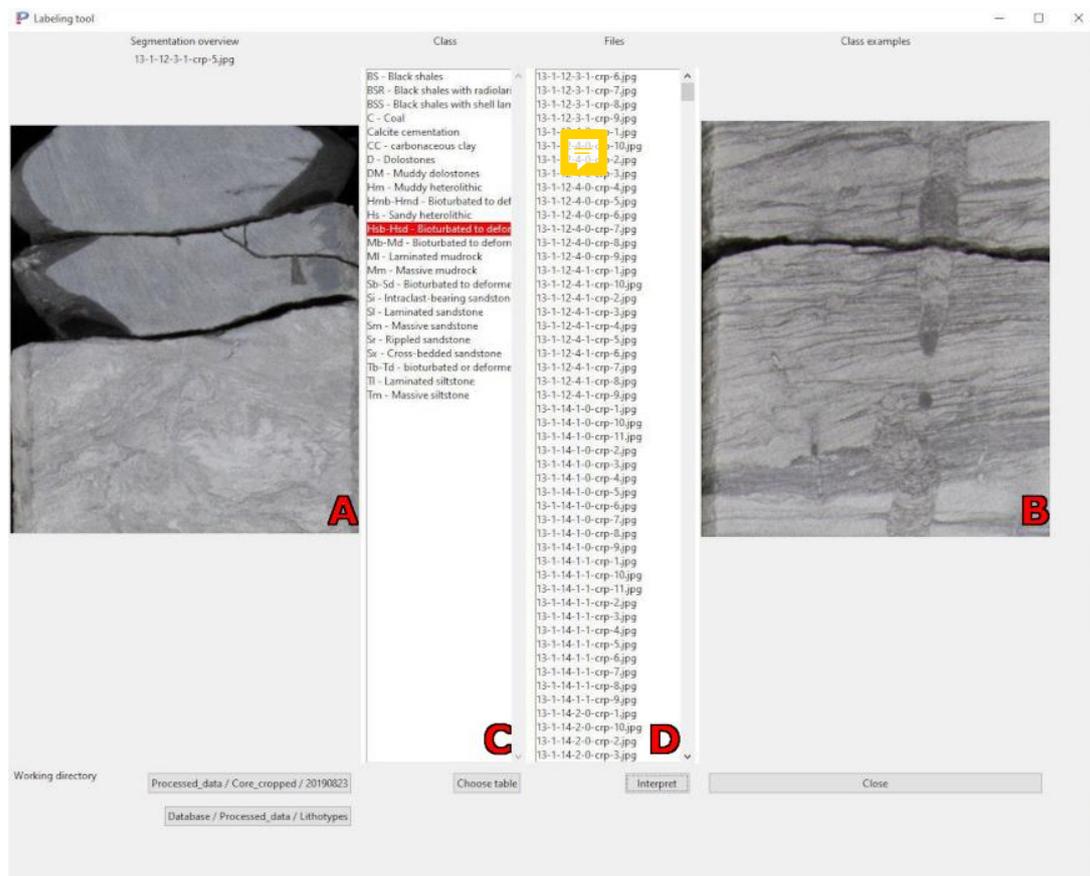
Figure 8. A tool developed within this work for the final expert data quality check. A – window with the current image; B – window with an expert-defined class example; C – window with classes available; D – list of files available for check.

3.2. Segmentation

The extracted table data from the database is required only as a reference for this task. The masks, in most cases, should be created with special tools.

For rock-type segmentation (5.1.2. Rock segmentation) a PixelAnnotationTool (PAT) (Bréhéret, 2017) used. A watershed algorithm described above automates the labelling of the rest of an image in a semi-supervised way. To segment the image, a user should load the dictionary (key-value pairs) with labels he wants to mark on it and manually draw them on an image. Only small regions sufficiently distinct from each other may be marked on the image. The other regions will be automatically labelled with a watershed algorithm.

For core column segmentation (5.1.2. Box segmentation), a GIMP (GNU image manipulating program) was used (The GIMP Development Team, n.d.). The principle of use is the same as in PAT, except watershed algorithm. The masks are created manually with only one label – the box. An additional layer for mask was created on image and colored into two colors; one color meant background, and the other – was foreground (the core).
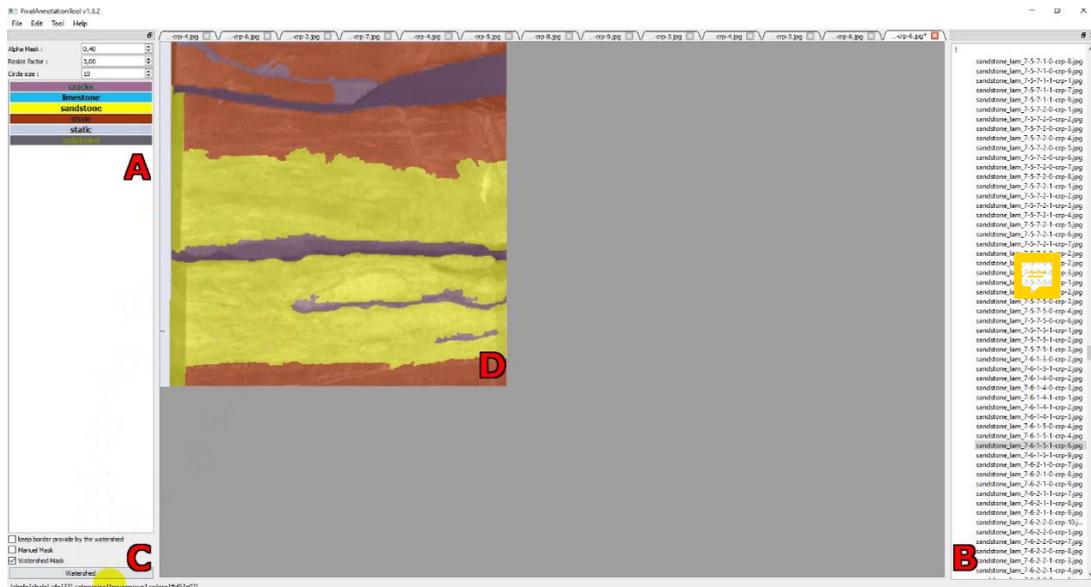
Figure 9. PixelAnnotationTool (PAT) (Bréhéret, 2017) *usage example for the creation of pixel-accurate rock type and structure (cracks) labels. A – available labels, B – available images to segment, C – watershed button, D – current image with mask hovered over it.*

3.3. Dataset automated gathering

Due to a massive problem of data storage, sometimes the databases may be stored in an unstructured way. In one laboratory, the data storage can have a data structure that holds the images in separate folders, and the other laboratory may store the images with other data. That's why a special framework was developed for automated datasets search. It is aimed at searching for images whenever the data is stored on a hard drive without a specially facilitated object-oriented database.

The framework is inspired by relational databases such as PostgreSQL (Dimitri, 2019). It has structured tables that rely on one another but can still be used separately.

There are several storage tables at the first run:

Depth – the table where all parsed depth for each image is saved.

Ini_names – table with a path to initial data and the name of the parsed file.

Images – table with image names.

The idea of the automated gathering is to collect all the available image data in the selected folders and subfolders in one place to manually process it.

3.4. Data balancing techniques

As machine learning algorithms are based on database analysis, a concept rule "garbage in, garbage out" (Lidwell et al., 2010) is applicable. Suppose the data is insufficient or inappropriately labelled. In that case, the algorithm either won't get good accuracy on a new set of data or will not be able to overfit the data. It is a nice practice to overfit the algorithm on the available data first to check the data quality.

If the data quality is sufficient, another problem may appear – some classes may not be represented the same way as others which is called undersampling. To fight such cases, a set of techniques are available. Also, even if data is sufficient, some outliers may still exist. In the case of machine vision – such outliers may be overlapping of several objects on an image or differentiation in object color. Such issues may be solved with data augmentation

3.4.1. Standard augmentation

Augmentation is a set of image permutations which can be applied to an original image to increase the generalization ability of an algorithm. The algorithm may classify the same image differently if the image slightly changes. Augmentation is a standard tool to improve generalization by machine learning algorithms (DeVries and Taylor, 2017).

A set of augmentations were used to increase the performance of algorithms (Figure 10).
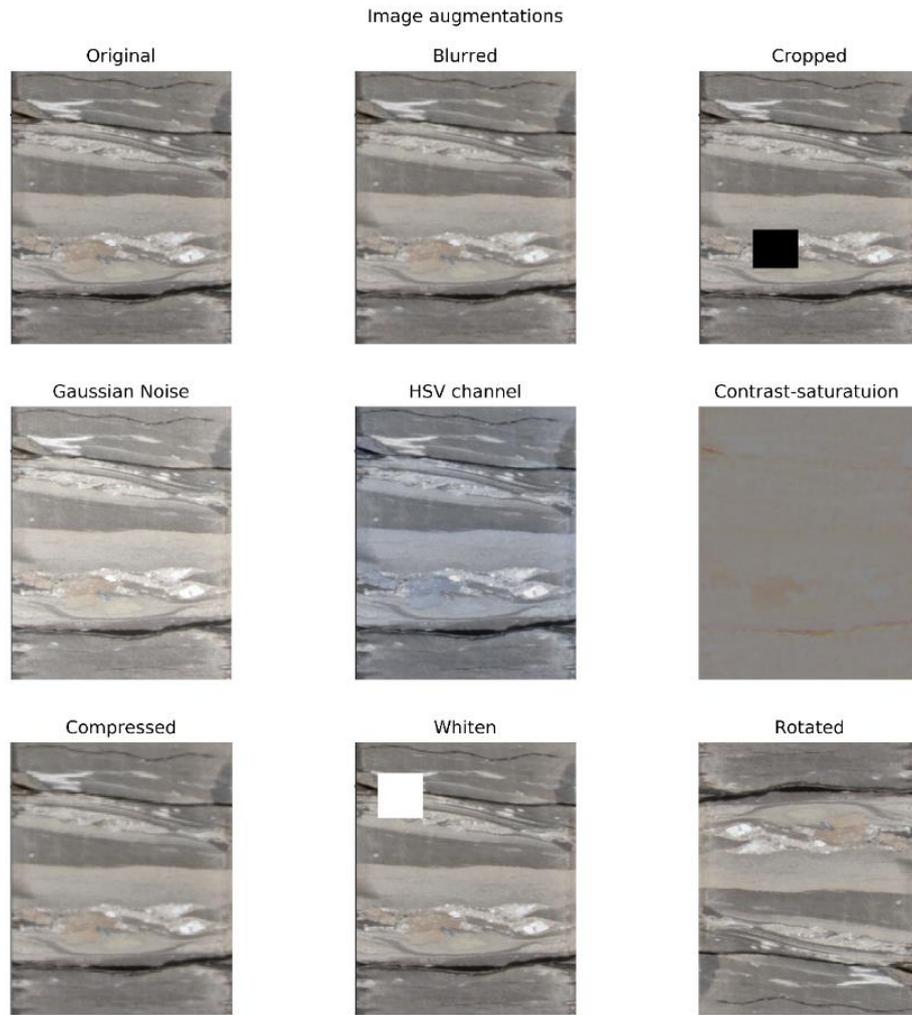
Figure 10. Different types of image augmentation. Various augmentations were applied to prevent overfitting and improve the generalization ability of the developed algorithms

3.4.2. Template-like augmentation

The existing augmentation techniques only work with the prepared image. This reduces the number of actions which can be performed over an image. A unique augmentation framework for the segmentation task was developed and described in (E. E. Baraboshkin et al., 2022a). A template and data should be placed together to run such an augmentation. The framework changes parts of a segmented image on a template to emulate different imaging environments (Figure 11). As seen in the figure, the core inside the core box was almost entirely changed. Such action improves the results of segmentation described in section 5.1.2.

63
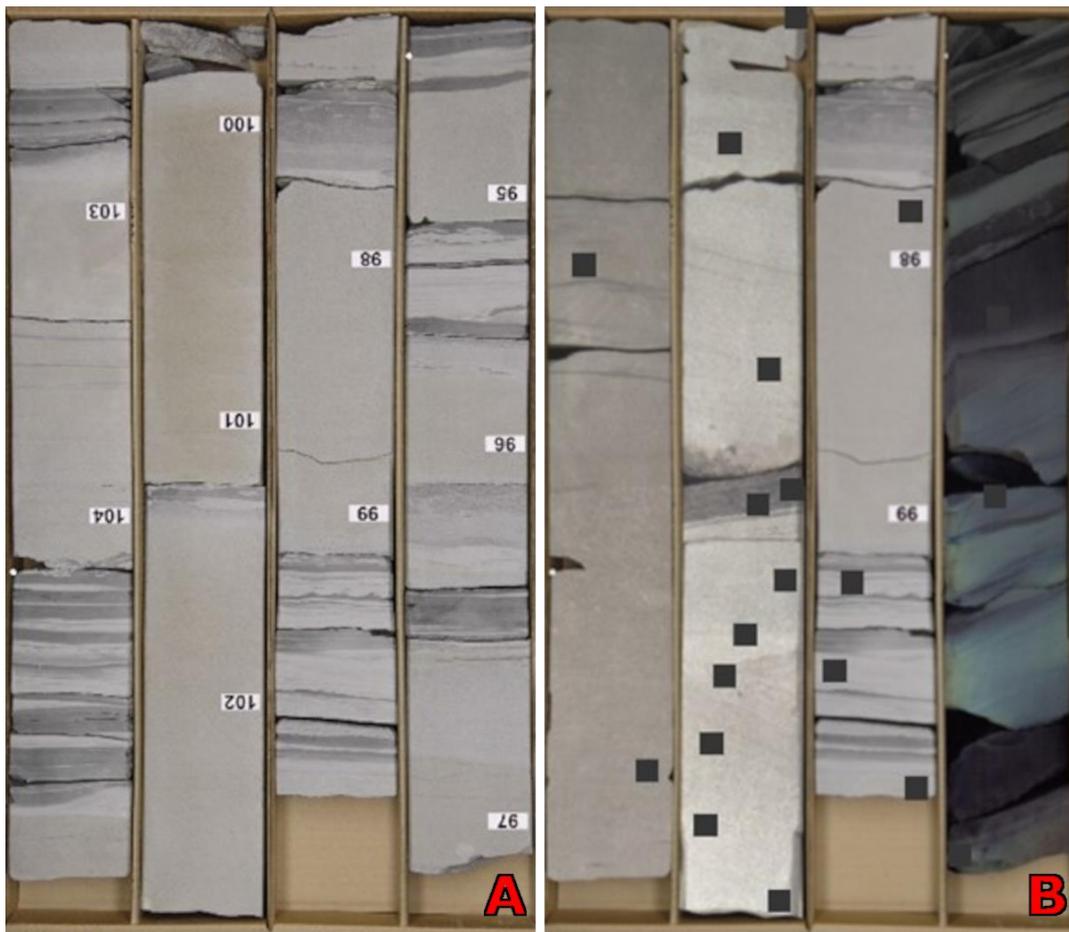
Figure 11. Comparison of the initial image (A) and augmented images (B). Some cores in the box changed (B); parts are cut out (B).

3.4.3. Other techniques

A set of techniques like Oversampling (Chawla et al., 2002), Undersampling (Liu et al., n.d.), sample weighting (Cui et al., n.d.), Different loss functions (e.g. Focal loss), Label smoothing (Müller et al., n.d.) were applied to produce appropriate data distribution during the training.

Chapter 4.   Classification

Classification is an important task that can be solved by machine learning. Unfortunately, to build a reliable classification model based on supervised machine learning, it is vital to have a specific task to solve. The study should be clear and straight, the same as data which is supplied to an algorithm. Geologists like to build different classifications based on various features which they can get from observations. The features may be different: structure, texture, layering direction, mineral composition, etc., which depends on the study purpose and scale. The classifications applied may also differ depending on the task required to complete: find a certain deposit or create a specific model in terms of genesis. The data and the task may be ambiguous and contain several hidden objectives a geologist tries to solve.

The objective of a researcher is to understand what a geologist wants to get from the data and which exact tasks and subtasks he solves to reach the final decision on an object.

The presented research is aimed to describe several essential characteristics for further geological modelling. First, a general description of rocks was developed to understand the general rock composition, and then a more sophisticated model to get a sedimentological feature was created. Finally, a technical core condition description was developed to build a reliable model for fracturing.

4.1. Results and discussion

Several models to predict rock and lithological types were developed. The first model is developed for a general description of rock types. The second is for a sedimentological description of rocks, and two more for a geotechnical description. The following sections describe the datasets, the conducted experiments and the classification results.

4.1.1. Datasets

Rock typing dataset

The dataset for rock typing consists of 590 core box images (approximately 1750 m of core). This data was cropped into 17500 images with size 10x10cm and px size

from 185 to 707 px (~ 47 – 180 dpi). The images were manually labelled into 6 classes (Figure 12): argillite, sandstone, laminated sandstone, siltstone, granite, and limestone. Each class define a type of rock.
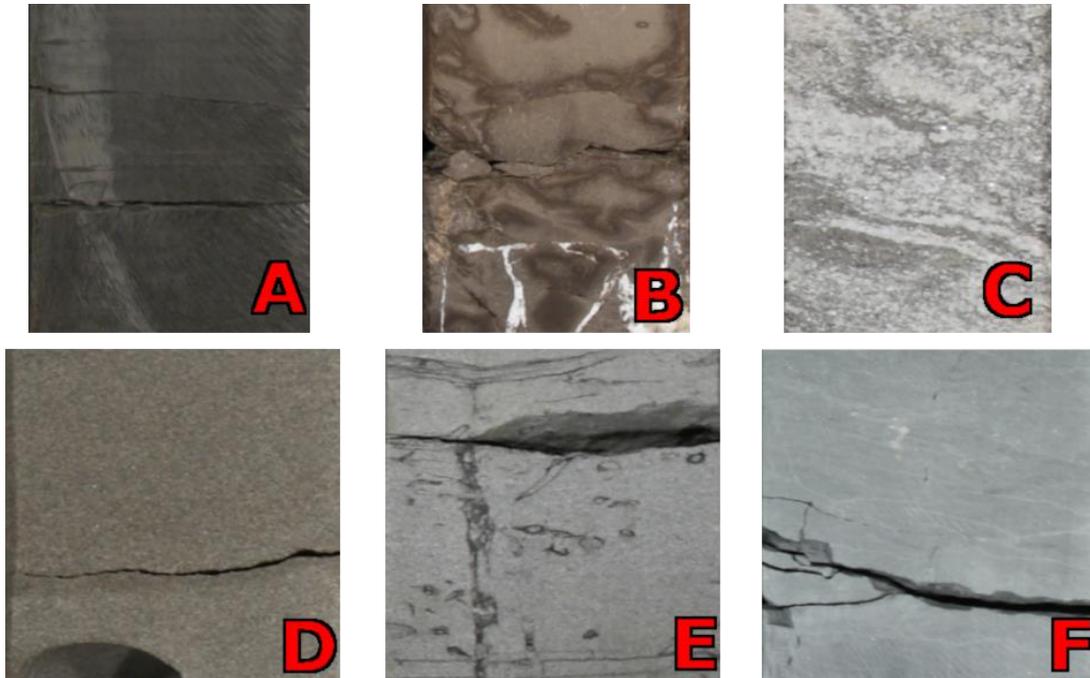
Figure 12. Manually labelled dataset for rock typing: A - argillite, B - limestone, C - granite, D - massive sandstone, E - laminated sandstone, F – siltstone.

Sedimentological description dataset

The dataset for sedimentological description had 350 images (median size 2300x4000 px) of core boxes and an initial label which was stored in a table. The core was cropped into smaller 10x10 cm images with average size of 506px (~128 dpi) up to 564px (~143 dpi). The dataset consists of 20 lithotypes (Figure 13): laminated shaly sand (Hs), laminated shaly sand with bioturbation (Hsb), laminated sandy shale (Ht), laminated sandy shale with bioturbation (Htb), deformed sandy shale (Htd), laminated mudrock (Ml), massive mudrock (Mm), bioturbated sandstone (Sb), deformed sandstone (Sd), intraclast-bearing sandstone(Si), laminated sandstone (Sl), massive sandstone (Sm), rippled sandstone (Sr), cross-bedded sandstone (Sx), bioturbated siltstone (Tb), deformed siltstone (Td), laminated siltstone (Tl), massive siltstone (Tm), rippled siltstone (Tr), coal (C).
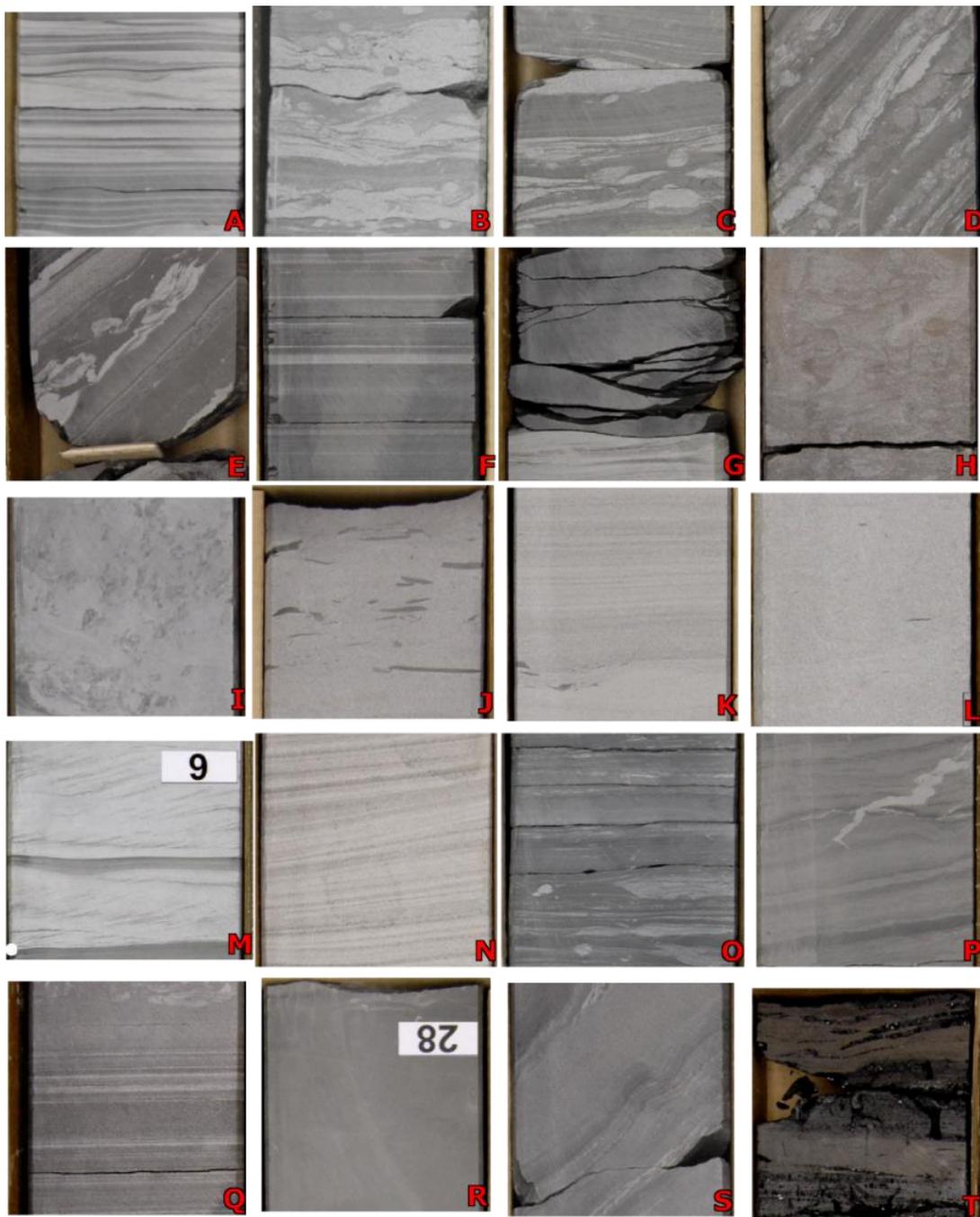
Figure 13.Various lithotypes for sedimentological description: A – laminated shaly sand (Hs), B – laminated shaly sand with bioturbation (Hsb) , C – laminated sandy shale (Ht), D – laminated sandy shale with bioturbation (Htb), E – deformed sandy shale (Htd), F – laminated mudrock (Ml), G – massive mudrock (Mm), H – bioturbated sandstone (Sb), I – deformed sandstone (Sd), J – intraclast-bearing sandstone(Si), K – laminated sandstone (Sl), L – massive sandstone (Sm), M –rippled sandstone (Sr), N – cross-bedded sandstone (Sx), O – bioturbated siltstone (Tb), P – deformed siltstone (Td), Q – laminated siltstone (Tl), R – massive siltstone (Tm), S – rippled siltstone (Tr), T – coal (C).

Each lithotype defines a type of rock and a structure which is formed during a deposition process. The lithotypes are used for further depositional environment and detailed deposits geological model reconstruction and lithofacies analysis.

After preparing the dataset with a table description, the dataset was manually checked, as described in section 2.5. 78% of data which previously didn't have a particular class due to errors in depth referencing was classified (Figure 14). 2 to 15% were relabeled to other classes. The final dataset consists of 6104 images. An exact number of images for each class can be found in Figure 15.
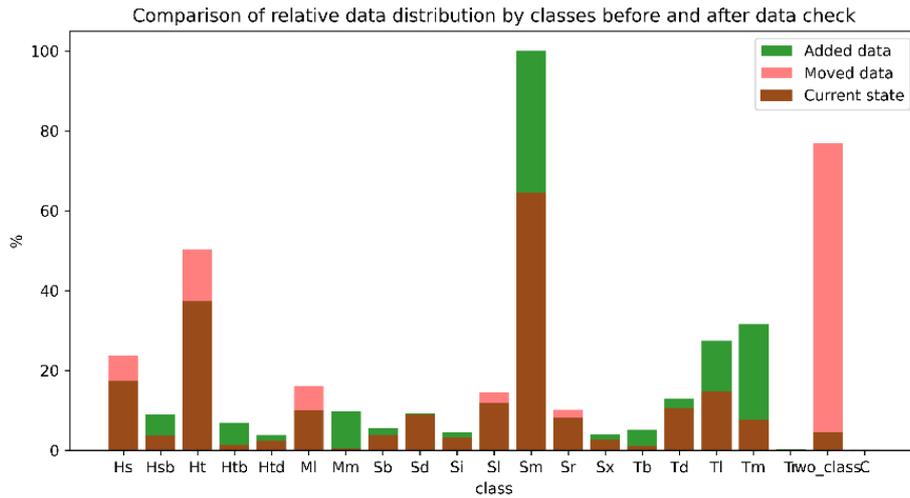
Figure 14. Data check results. 75% unclassified data was classified to different classes. Caption for the abbreviations can be found on figure 13 and figure 15.
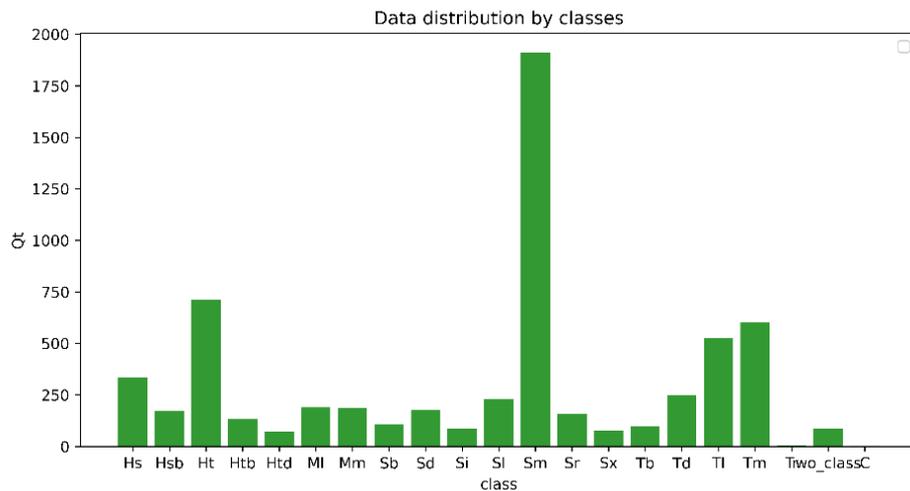
Figure 15. Final data quantity for each class. The distribution is not equal. Hs - laminated shaly sand (333), Hsb – laminated shaly sand with bioturbation (171); Ht – laminated sandy shale (711); Htb – laminated sandy shale with bioturbation (132); Htd – deformed sandy shale (73); Ml – laminated mudrock (188); Mm – massive mudrock (185); Sb – bioturbated sandstone (108); Sd – deformed sandstone (175); Si – intraclast-bearing sandstone (84); Sl – laminated sandstone (227); Sm – massive sandstone (1909); Sr –rippled sandstone (156); Sx – cross-bedded sandstone (76); Tb – bioturbated siltstone (98); Td – deformed siltstone (246); Tl – laminated siltstone (524); Tm – massive siltstone (602); Tr – rippled siltstone (4); C – coal (2).

Geotechnical description

The geotechnical description dataset was made to describe core intervals for the mining industry. The dataset consists of 390 core box images with a coring size of 6,6 cm. It was separated into smaller crops 5x6,6 cm The dataset was built in two iterations.

The first iteration only determined the core condition (1509 images in total): crushed core (907), intact core (223), and cracked core (379). It will be referenced as **GTD-1**.

The second data labelling iteration contained 5 fault rock types (Figure 16) and an arbitrary one (3310 images in total): Cataclasite (Kt, 79), Mylonite (Ml, 80), Quartz (Q, 65), Non-tectonized (2266), Fracture zones (Z, 764) and Non-classified (56). Each rock state reflects a level of tectonic restructuring except for Quartz, which pointed out the quartz veins, which usually contain gold. The second dataset will be referenced as **GTD-2**
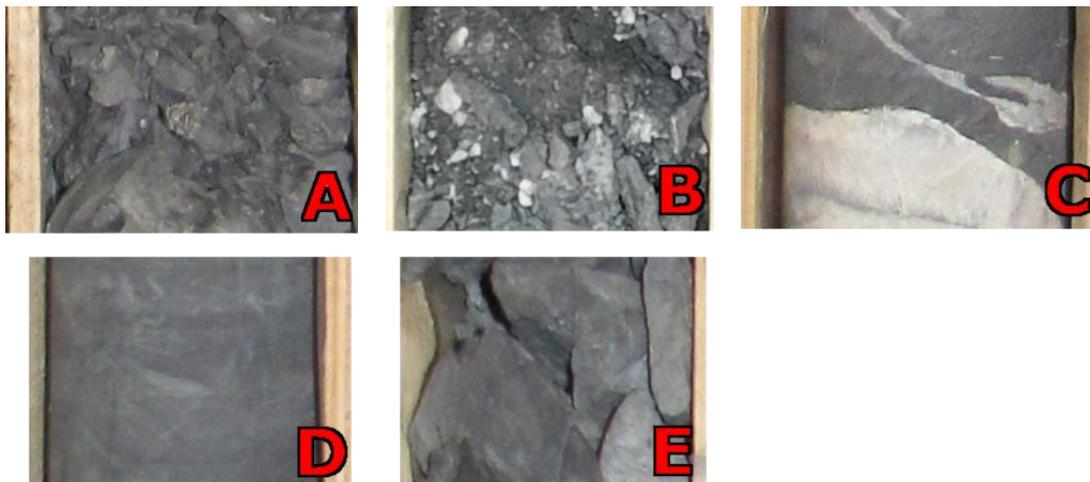
Figure 16. Various fault rock types for GTD-2 dataset: A – cataclasite (Kt), B – mylonite (Ml), C – quartz (Q), D – non-tectonized, E - fracture zones (Z).

The GTD-2 dataset had a table description of different fault rock types. 3 types were chosen by an expert as most common and important: mylonite, fracture zone, cataclasite. They were processed and manually cleaned up as described in section 2.5. 15 to 40% of data was moved within each class (Figure 17), as some types were described within meter intervals. After processing a geologist manually added two more classes to the dataset (Figure 18). The final GTD-2 dataset contains 1254 images.
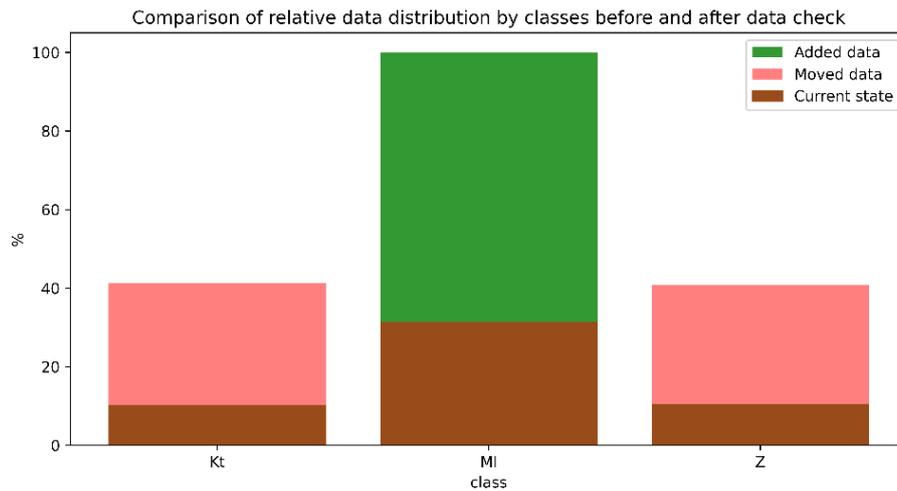
Figure 17. Data check results in GTD-2 dataset. A large amount of data was relabeled for mylonite as it was previously described in a scale of 1:100 cm, while the dataset has scale of 1:5 cm. Caption for the abbreviations can be found in Figure 16 and Figure 18.
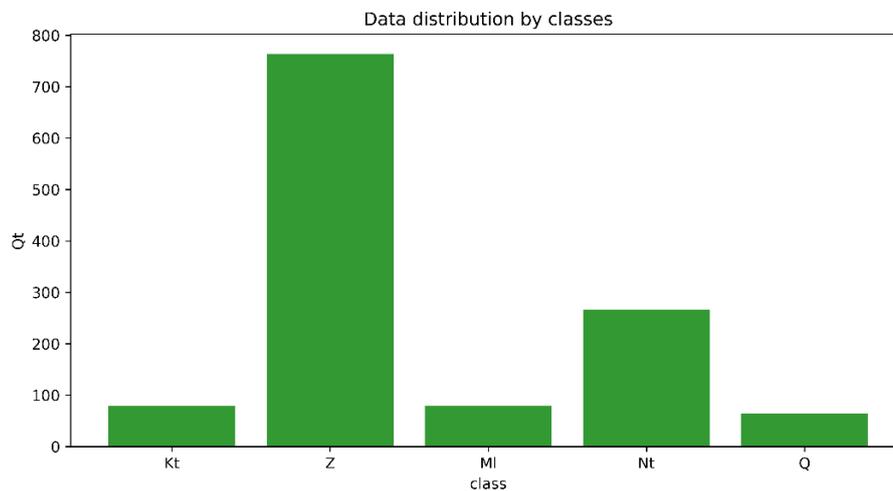
Figure 18. Final data quantity for each class in the GTD-2 dataset. The distribution is not equal: Kt - cataclasite (79), Z - fracture zones (764), Ml – mylonite (80), Nt – non-tectonized (266), Q – quartz (65).

4.1.2. Feature-based classification

The feature-based classification was performed on the rock typing dataset as it is an excellent toy example to understand the algorithm's performances on certain rock types. Features were extracted from each image out of the dataset. The following feature extraction methods were used: Principal Component Analysis (PCA), Histogram of Oriented Gradients (HOG), and Pretrained artificial networks (PNNF). Each method allows extracting a different number of features: PCA – 456, HOG – 156816, and PNNF – 4608.

Supervised classification

To understand how the data is distributed ~~a~~ 3 randomly selected features were projected to a 3-dimensional space (Figure 19). It can be noted that PCA and HOG methods produce sparse features, while PNNF features are condensed in a small area. An additional PCA algorithm was applied for HOG and PNNF features to reduce dimensionality further. Unfortunately, it worked only with PNNF data as ~~convolutional neural networks~~ are based on linear convolutions the same as PCA performs reduction in linear space.
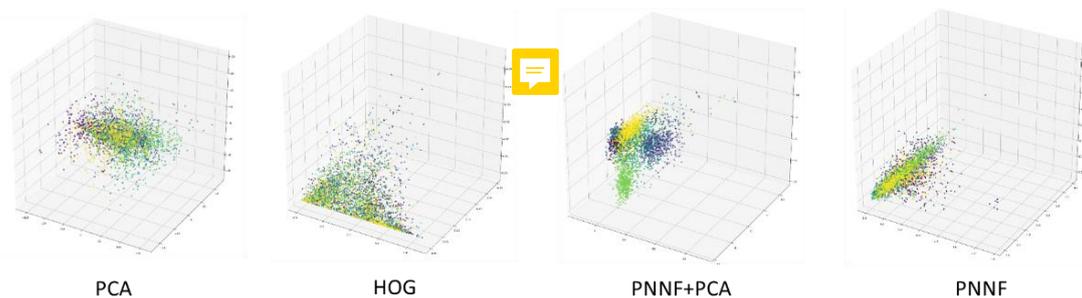
Figure 19. A feature representation is based on the features extracted from different images. Feature extraction methods: Principal Component Analysis (PCA), Histogram of Oriented Gradients (HOG); Pretrained networks (PNNF). Each color represents a different class from the rock typing dataset.

Two machine learning algorithms were used to classify the extracted features – SVM and Random Forest (Figure 20). As a reference algorithm for classifying images, convolutional neural networks (CNN) were used.
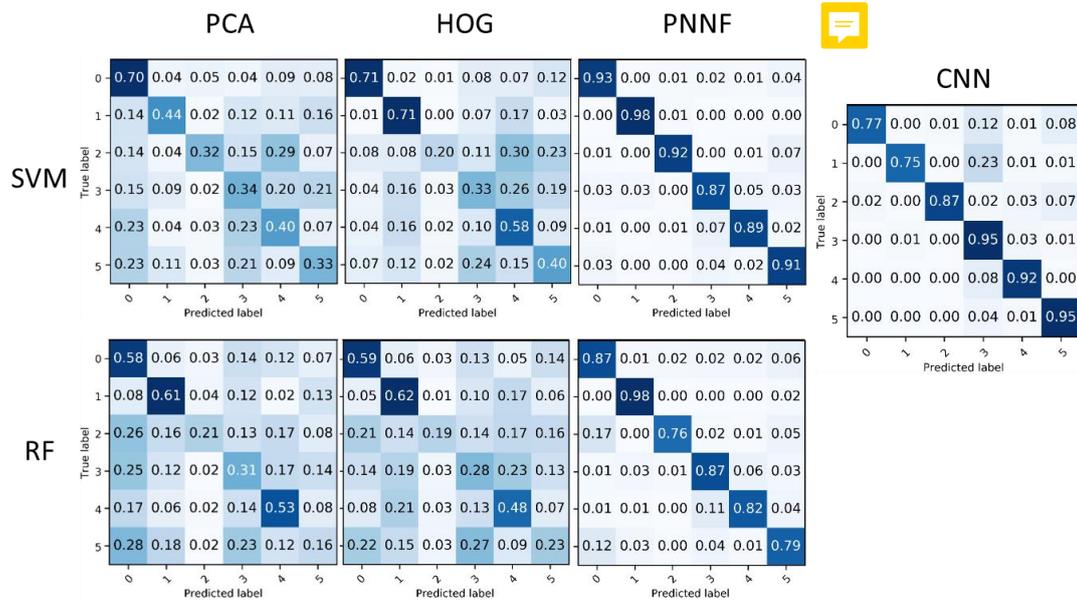
Figure 20. A set of classifications based on different features classified with Support Vector Machines (SVM) and Random Forest (RF). 0 – argillite, 1 – granite, 2 – limestone, 3 – laminated sandstone, 4 – massive sandstone, 5 – siltstone. Feature extraction methods: Principal Component Analysis (PCA), Histogram of Oriented Gradients (HOG); Pretrained networks (PNNF)

After a thorough analysis of the results presented in figure 20, it was found that the SVM-based classification of features extracted with pretrained networks outperforms the convolutional neural networks in the classification of argillites, granites, and laminated sandstones. Random forest classification beat convolutional neural networks only in argillites and limestones. The other algorithms' combination failed to achieve such results.

A hard-to-classify for CNN data interval was chosen (figure 22) to approve the gained results. The interval consists of 20 segments of rock, each sliced to 10x10 cm (the section length is 2 m). The interval causes a problem due to the rock's colour (Figure 21) – it is black-colored with highly hydrocarbon-saturated material - kerogen. As the initial dataset doesn't contain such cases - the trained CNN misclassified it as argillite or siltstone, while originally, it is a massive sandstone. Important to note that if a rookie expert classifies the image, he may also make such a misclassification.

Figure 21. Examples of highly hydrocarbon-saturated massive sandstones.

All the algorithms classified the interval. The results of the classification can be found in figure 22. As it can be seen, the most agreed with ground truth classification is HOG followed by SVM classification (8 segments were detected correctly), the next top prediction rate is SVM and RF on PNNF classification (4 segments), and NN prediction itself (3 segments). Interesting to note that RF classification similarly classifies the images as a rookie expert would. Most of the black-colored image parts are marked as argillite. The RF classification made the least mistakes in terms of rock type classification. It classified all the rocks as a terrigenous type. It may be a result of underfitting caused by a tree's lack of generalization ability, as explained in section 2.2.2.
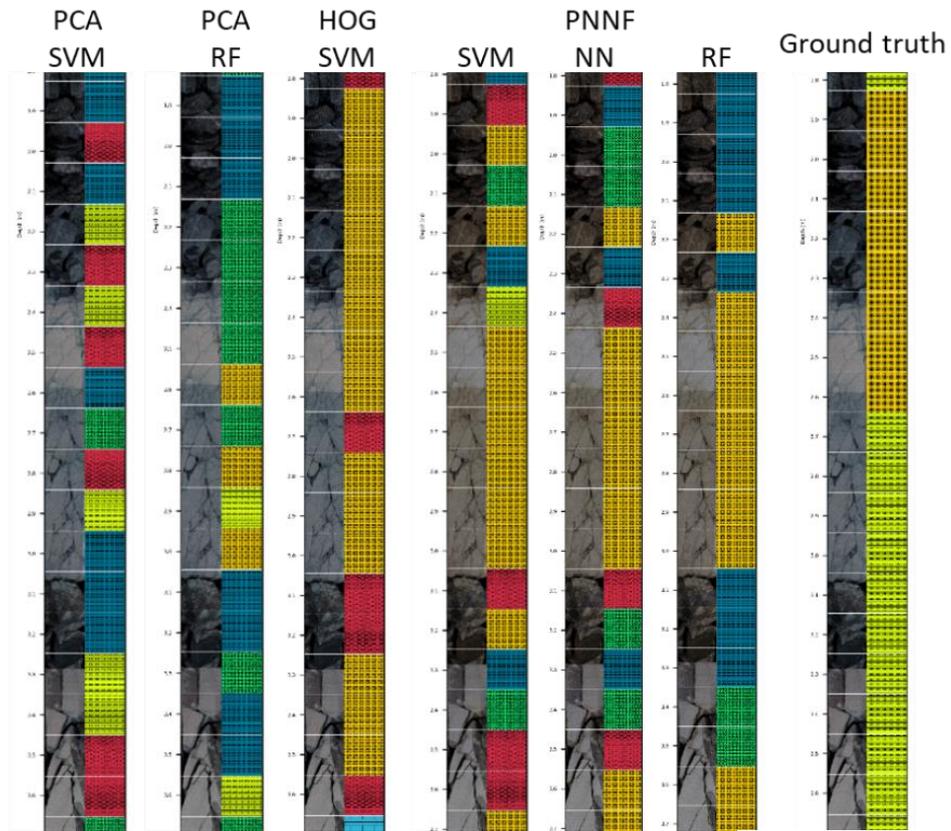
Figure 22. A comparison of different feature-based classification algorithms. Several machine learning algorithms (Support Vector Machines (SVM) and Random Forest (RF)) compared to convolutional Neural Network (NN) classification. The classification was performed on features extracted in different ways: Principal Component Analysis (PCA), Histogram of Oriented Gradients (HOG), Pretrained networks (PNNF) and Neural Network (NN) as a reference value.

Unsupervised classification

A set of experiments was performed to create an unsupervised classification (Figure 23).

PNNF, K-mean

	0	1	2	3	4	5
0	0.55	0.24	0.10	0.06	0.04	0.01
1	0.43	0.27	0.23	0.05	0.01	0.01
2	0.53	0.27	0.08	0.06	0.05	0.01
3	0.50	0.22	0.14	0.04	0.03	0.08
4	0.45	0.22	0.13	0.05	0.13	0.01
5	0.50	0.23	0.09	0.08	0.04	0.06

HOG, K-mean

	0	1	2	3	4	5
0	0.36	0.12	0.21	0.12	0.19	0.00
1	0.12	0.01	0.04	0.06	0.76	0.00
2	0.16	0.00	0.31	0.05	0.48	0.00
3	0.21	0.01	0.14	0.25	0.35	0.05
4	0.07	0.01	0.35	0.03	0.52	0.02
5	0.24	0.00	0.18	0.20	0.25	0.12

PNNF, AC

	0	1	2	3	4	5
0	0.07	0.73	0.19	0.00	0.01	0.00
1	0.01	0.00	0.01	0.94	0.04	0.00
2	0.00	0.96	0.00	0.02	0.02	0.00
3	0.00	0.01	0.03	0.01	0.87	0.08
4	0.01	0.02	0.02	0.00	0.26	0.69
5	0.00	0.38	0.00	0.01	0.60	0.00

HOG, AC

	0	1	2	3	4	5
0	0.23	0.25	0.28	0.20	0.01	0.02
1	0.57	0.03	0.02	0.37	0.02	0.00
2	0.53	0.00	0.34	0.11	0.02	0.00
3	0.24	0.02	0.23	0.34	0.07	0.10
4	0.51	0.02	0.36	0.08	0.03	0.00
5	0.22	0.02	0.25	0.28	0.16	0.07

Assigned label

Figure 23. Results of unsupervised classification produced by various algorithms and feature extraction methods. Applied algorithms: K-mean – k-mean algorithm, AC – agglomerative clustering. Feature extraction methods: HOG – histogram of oriented gradients, PNNF – pre-trained neural network. No meaningful results were gained within 5 classes chosen according to the class labels..

As shown in figure 23 – the unsupervised classification has failed to classify the rock types correctly. The clusters were analyzed for each class, and it was found that the clusters are separated mainly by the texture and color attributes. As seen in figure 24, the images have some lamination caused either by cracks and mechanical damage or by natural reasons (dissolutions and laminations). Different rock and structure types were set in the same cluster.
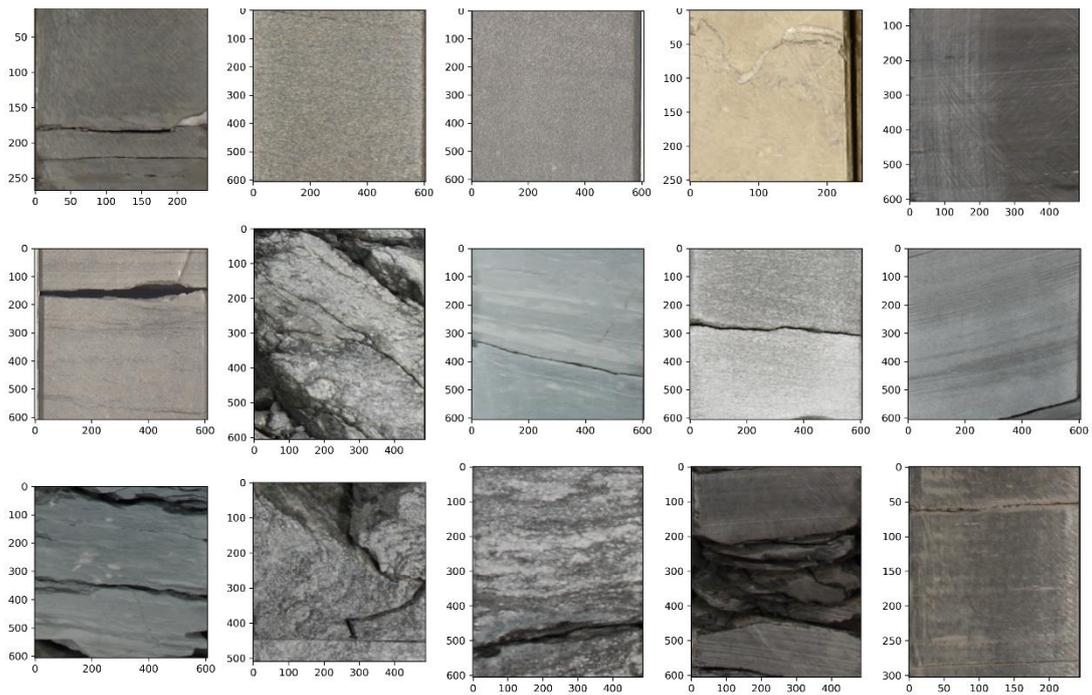
Figure 24. Example of classification based on AC clustering with Neural Network features from one class. All of the images belong to the same class produced by an algorithm. As can be seen, various images and textures are placed in the same class.

The clustering results were also applied to hard-to-classify for CNN part data (Figure 25). It can be noted that both algorithms made the right decision and separated oil-saturated parts of the crop from other samples. Unfortunately, they both included a shadowed laminated sandstone sample to the same group, meaning the separation was primarily made by color and is unsuitable for further application.
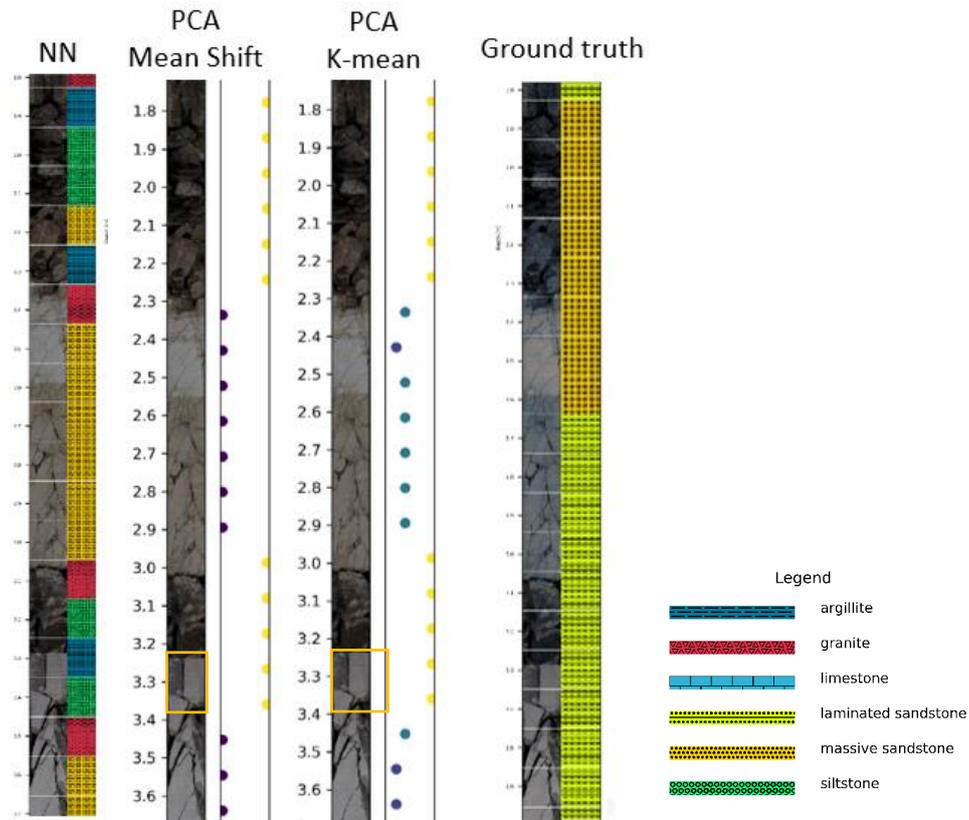
Figure 25. A comparison of different feature-based clusterization algorithms on a hard-to-classify for CNN part data. Clusterization methods (Mean Shift and K-mean) compared to convolutional Neural Network (NN) classification. The clusterization was performed on features extracted with Principal Component Analysis (PCA). The orange box on the mean shift and k-mean algorithm results highlight misclassification regions.

4.1.3. Deep learning-based classification

Several neural networks were trained for each of the datasets. Each time a learning rate, set of augmentations and optimizer were selected during a set of experiments.

Each architecture was trained on the same set of data. A set of parameters observed to verify the stability in results determination and generalization of the networks:

1) The loss curves for training and validation sets match.

2) A test set classified with the same or higher metrics rates.

3) The prediction results on a new data set are reasonable at an expert's first glance.

Rock typing classification

The first set of experiments aimed to search for the best architecture that could fit the classification problem. During the search, several architectures were assembled (figure 26): AlexNet (Krizhevsky et al., 2012), ResNet (He et al., 2015), GoogLeNet (Szegedy et al., 2015), VGG16 (Simonyan and Zisserman, 2014).
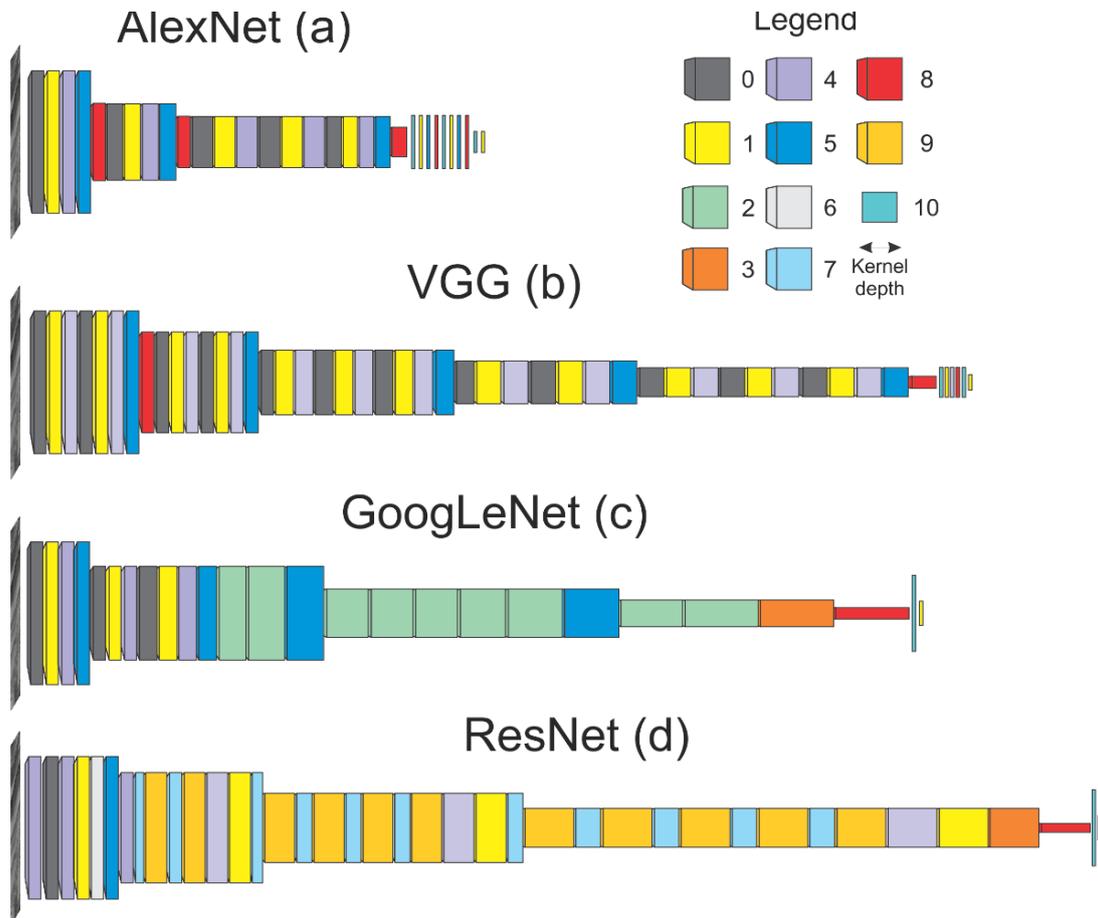
Figure 26. Sketch-ups of different architectures: 0 – convolution layer, 1 – activation layer, 2 – inception module, 3 – average pooling, 4 – batch normalization, 5 – max pooling, 6 – zero padding, 7 – residual module, 8 – drop-out layer, 9 – layer composition, 10 – dense layer.

These architectures were exhaustively researched. Over 100 experiments were launched. Different hyperparameters were tested and modified, including loss functions, optimizations and layers. Each set of experiments concluded with inspecting classification results on a validation set. In addition, a set of randomly chosen features extracted from different layers was visually analyzed (Figure 27).

After the analysis of extracted features and validation metrics calculation, the images were sent to classification with the LIME package. Random image parts are consequently sent to the CNN, and a region that gives the most classification value is evaluated (Figure 28).

Only AlexNet activated a limited number of filters from a large number in all lithotypes (only one filter out of four was activated in figure 27). In other architectures, the filters have the same weight sign for most lithotypes, but each filter selects features for each lithotype. ResNet and VGGNet concentrate on the grain size of the lithotype and a texture, as each filter emphasizes the granularity of an image. ResNet does not extract information from images when the size of grains is small (the siltstone and argillite images do not have significant weight fluctuation). However, VGGNet attempts to extract information in all cases. GoogLeNet activations concentrate more on texture classification. The granularity of intermediate layers is small in GoogLeNet; it can be seen only in the massive sandstone and granite. More examples can be found in a published article (Baraboshkin et al., 2020).

Regarding classification area countering - all the networks detect the lithotype based on several different-size image regions (Figure 28), which sometimes can cause misclassification. All CNNs classified the presented examples on the first row as laminated sandstone. In contrast, the second row has been classified as siltstone by most architectures except VGGNet, which classified it as laminated sandstone. That happened because most of the examined patches of an image are like siltstone. The VGGNet made such a prediction because the patches it examined contained sandstone parts and fine-grained siltstone.
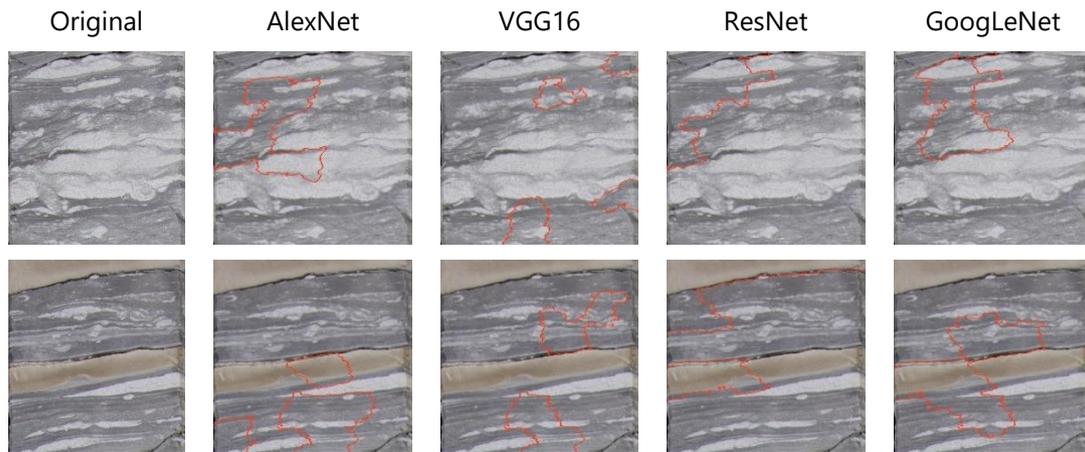
With such an experimental setup, the best optimization method was Adam (Adaptive Moment Estimation), with the learning rate starting at a level of 0.01. As a result of experiments, the following average metrics were gained on the validation set with the best-chosen hyperparameters (Table 1):

Table 1. Metrics gained with different CNN architectures on a test set.

Metric	AlexNet	VGG16	GoogLeNet	ResNet
Precision	0.95	0.96	0.95	0.93
Recall	0.95	0.96	0.95	0.93
F1	0.95	0.96	0.95	0.93

The neural network performances were compared based on new data with an expert classification checkup within 44 m of the core. The metrics were calculated based on the expert's classification check and can be found in table 2.

Table 2. Metrics gained with different CNN architectures on new data with expert validation.

Metric	AlexNet	VGG16	GoogLeNet	ResNet
Precision	0.65	0.63	0.72	0.70
Recall	0.53	0.51	0.53	0.60
F1	0.58	0.56	0.57	0.56

As can be seen, most data were classified correctly with GoogleNet and ResNet architectures. The highest recall rate has ResNet architecture, meaning it has the highest true positive rates.
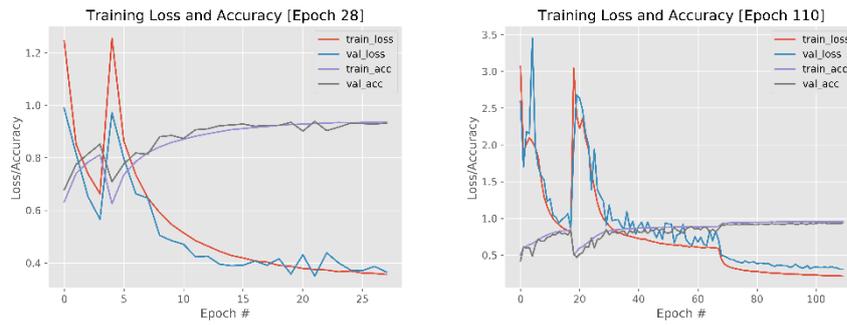
Figure 29. Training loss curves for AlexNet and ResNet architectures.

Also, it appeared during experiments that ResNet - a type of architecture doing well in the classification of various rock types in case of a small number of examples (below 100 images). As ResNet is robust for small datasets and provides the highest true positive rate, it will be used further in such tasks.

Sedimentological description

Different ResNet architectures were compared during model preparation. The final metrics for ResNet-101 architecture are presented in table 3. The main issue during the training was a highly imbalanced class. To fight this issue, oversampling and class weighting techniques were used, described in section 3.4.3.

Table 3. The best metrices for the sedimentological dataset gained with ResNet-101 architecture on a test set.

Metric	ResNet-101
Precision	0.778
Recall	0.771
F1	0.770

The performance of an algorithm was measured on a test set (Figure 30). The analysis confusion matrix provided exciting insights into the data. As seen in figure 30 – some classes are misclassified within the same rock type (e.g., some Htb are referred to as Hs), and some cases are mixed between rock types (e.g., Sd referred to as Hs). Such cases were analyzed, and it appeared that these classes are very similar to the predicted one (e.g., Hs), and in 88% of cases, they shouldn't be counted as errors as these images contain both classes (Figure 31).
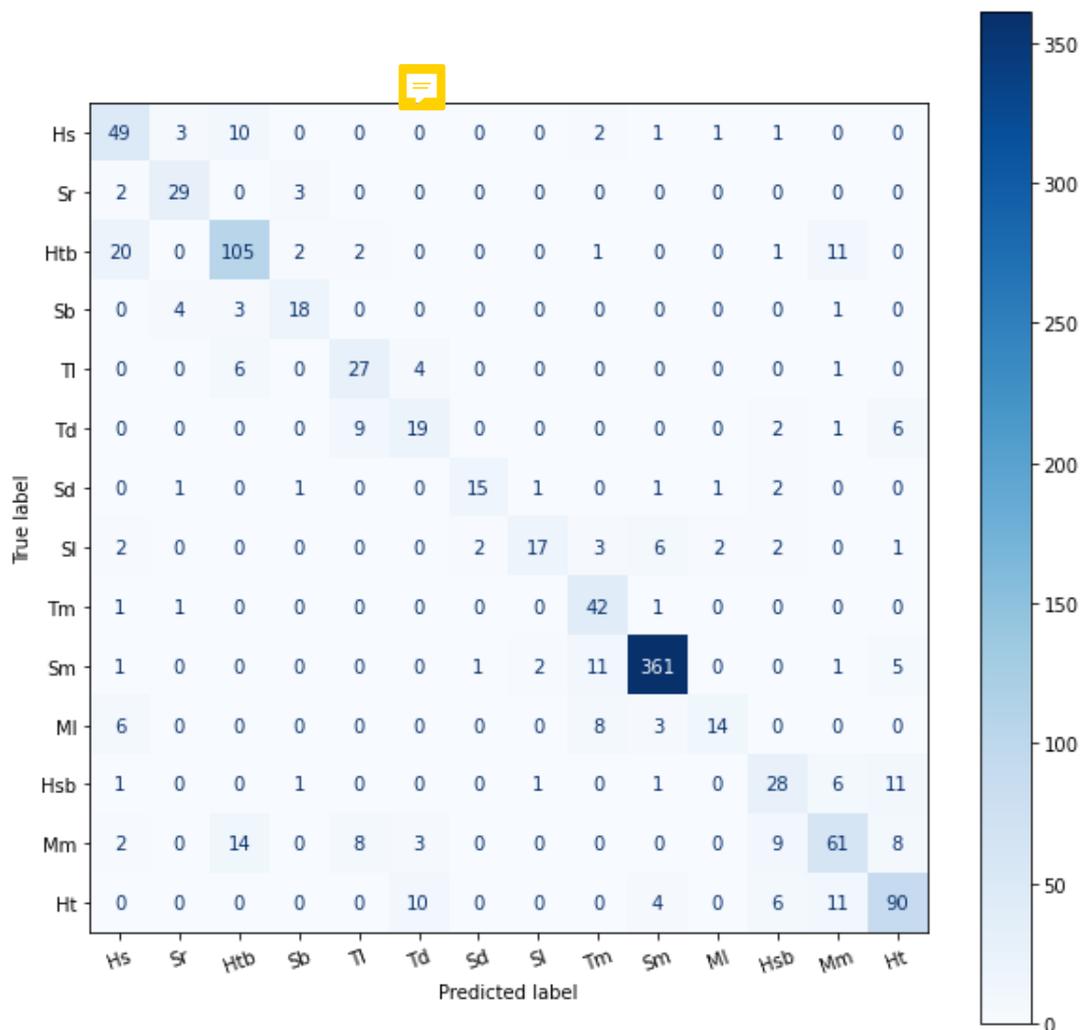
Figure 30. A confusion matrix for the ResNet-101 lithotyping algorithm. Hs – laminated shaly sand; Hsb – laminated shaly sand with bioturbation; Htb – laminated sandy shale with bioturbation; Ht – laminated sandy shale; Ml – laminated mudrock; Mm – massive mudrock; Sb – bioturbated sandstone; Sd – deformed sandstone; Sl – laminated sandstone; Sm – massive sandstone; Sr –rippled sandstone; Td – deformed siltstone; Tl – laminated siltstone; Tm – massive siltstone.

Such misclassification could be overcome with label smoothing (Müller et al., n.d.). For each label with several classes, a label should be smoothed with equal probabilities distribution. Another way to overcome the problem is to create a set of algorithms and get a mean classification result with maximum votes from a certain class.
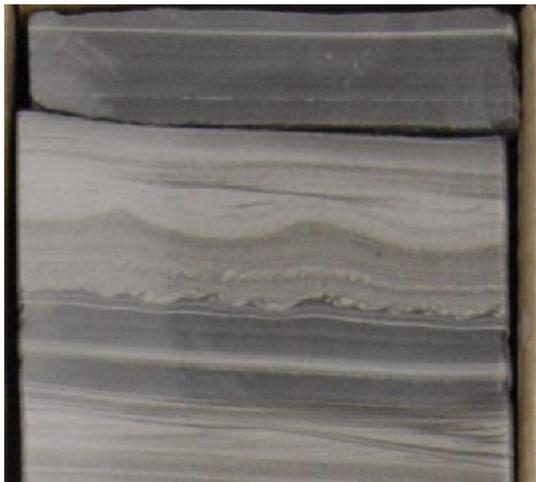
Ht (labelled) – Hs (predicted)
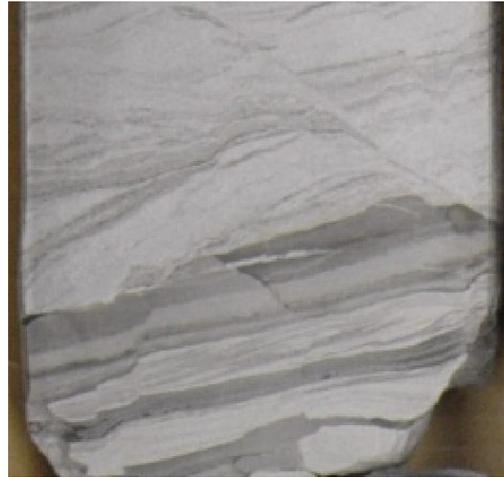
Sb (labelled) – Hs (predicted)
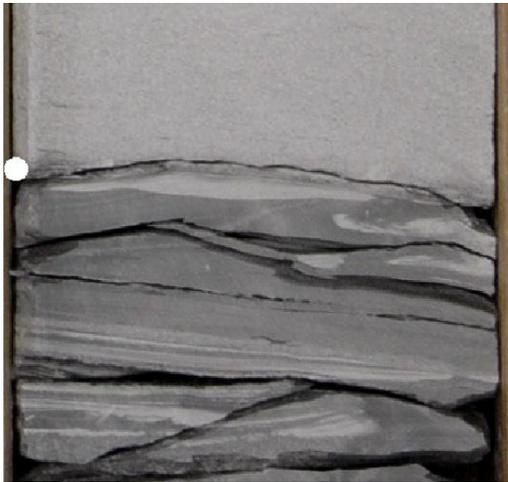
Ht (labelled) – Hs (predicted)
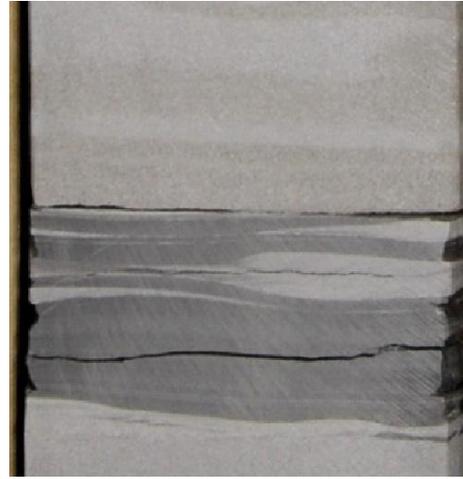
Hs (labelled) – Hs (predicted)

Figure 31. Different samples were defined as laminated shaly sand (Hs) lithotypes by the ResNet-101. Hs – laminated shaly sand, Ht – laminated sandy shale, Sb – bioturbated sandstone. These examples truly contain both classes as they have multi-labels.

New data was interpreted by an algorithm to check the final accuracy of a model. The data consisted of a new core with an overall length of 50 m. An expert reviewed the results and made some changes in the interpretation (added layers, changing boundaries, etc.) (figure 32).
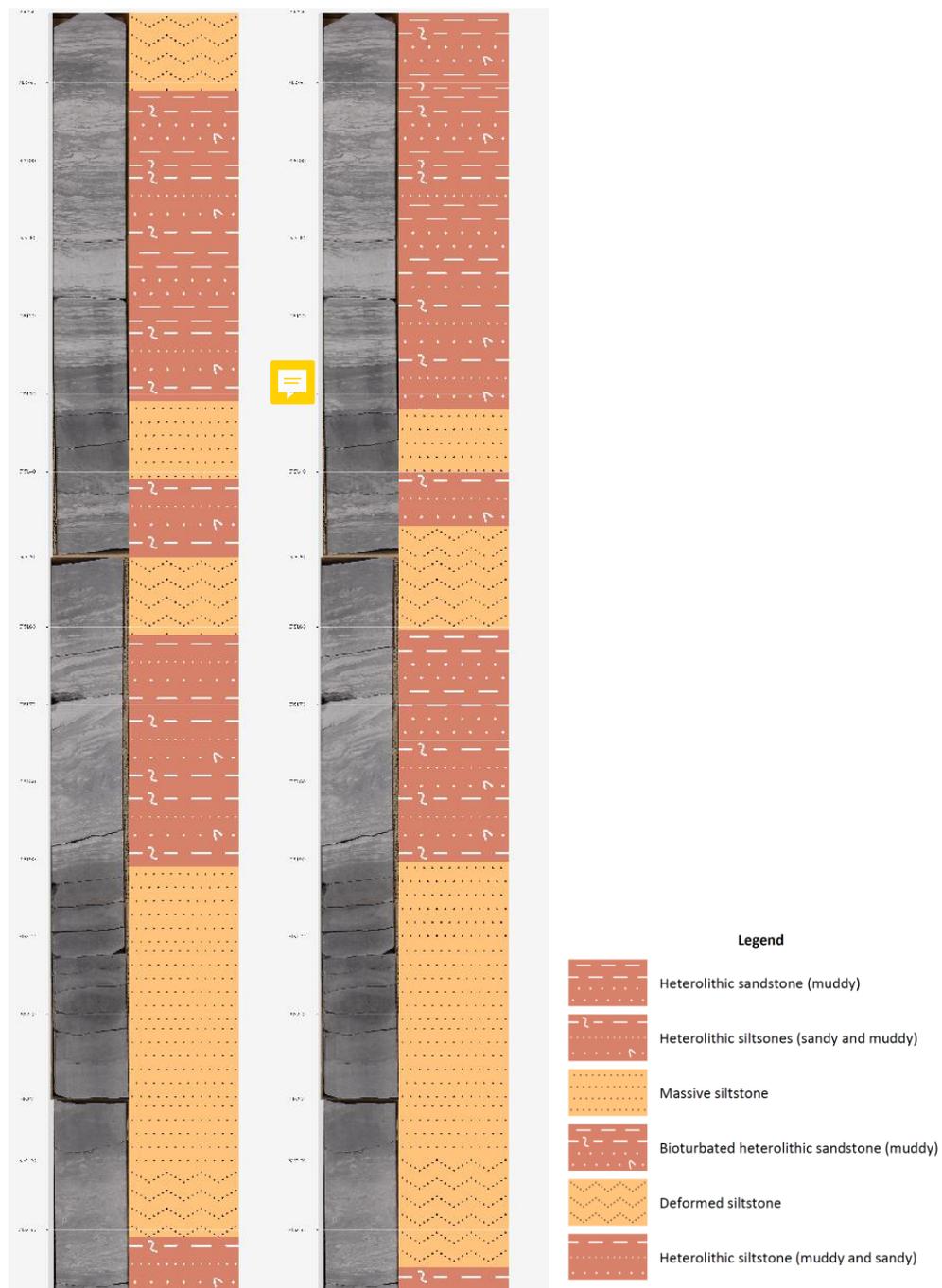
Figure 32. Comparison of algorithm (left) classification and expert's (right) classification correction.

Total estimated time for classification with an expert's changes application – 40 minutes, depth referencing included. The traditional way of such classification would take 5 hours from an expert. The results of the evaluation are presented in table 4.

Metric	ResNet-101
Precision	0.72
Recall	0.70
F1	0.71

As seen, an expert made the same number of corrections as in the validation set. After analysis of a confusion matrix based on an expert, the same trend was observed in misclassification. Most errors were made within a structure, but not within a rock type.
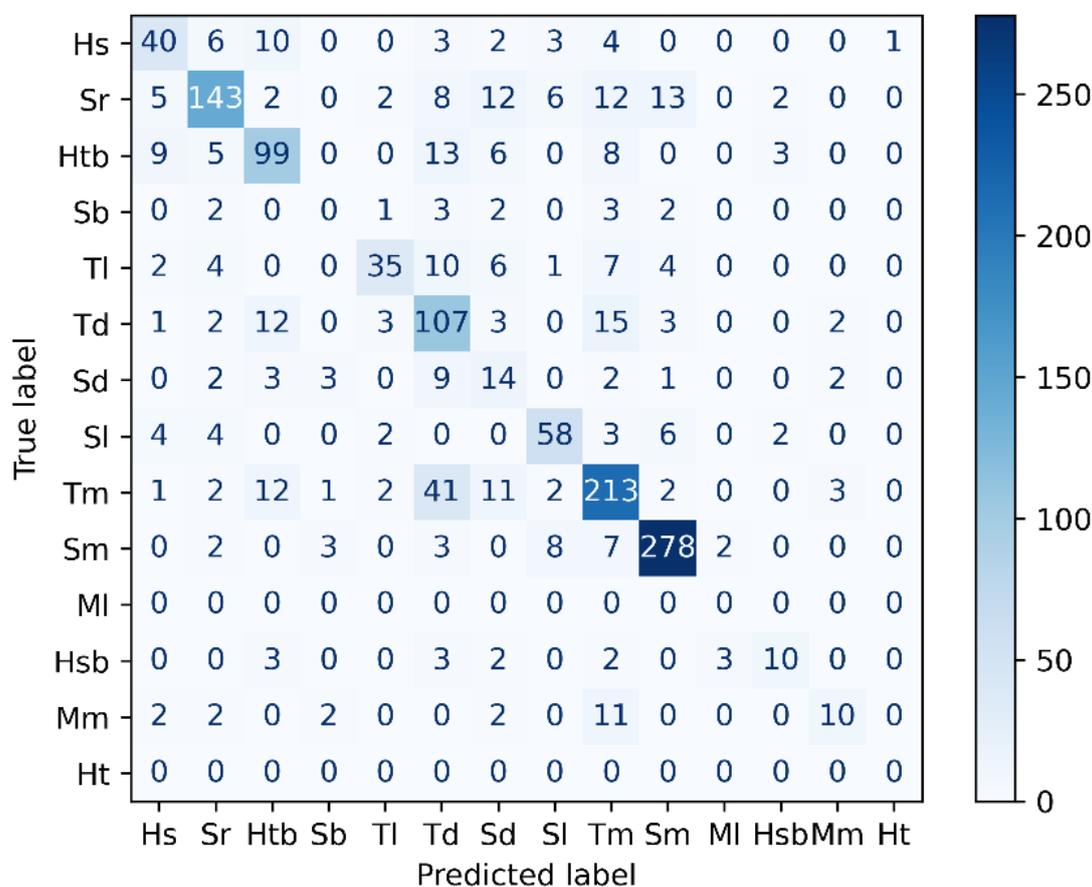
Figure 33. Confusion matrix for expert classification made over CNN classification. The true label represents the final expert decision made after classification by an algorithm. Hs – laminated shaly sand; Hsb – laminated shaly sand with bioturbation; Htb – laminated sandy shale with bioturbation; Ht – laminated sandy shale; Ml – laminated mudrock; Mm – massive mudrock; Sb – bioturbated sandstone; Sd – deformed sandstone; Sl – laminated sandstone; Sm – massive sandstone; Sr – rippled sandstone; Td – deformed siltstone; Tl – laminated siltstone; Tm – massive siltstone.

To further understand the abilities of ResNet-101 architecture, features from the last convolution layer were extracted and analyzed for different classes in the dataset

(figure 34). As seen in a figure, each filter extracts different information from the image. The filters on the first-row extract texture information, and the filters on the second row – collect structure information.
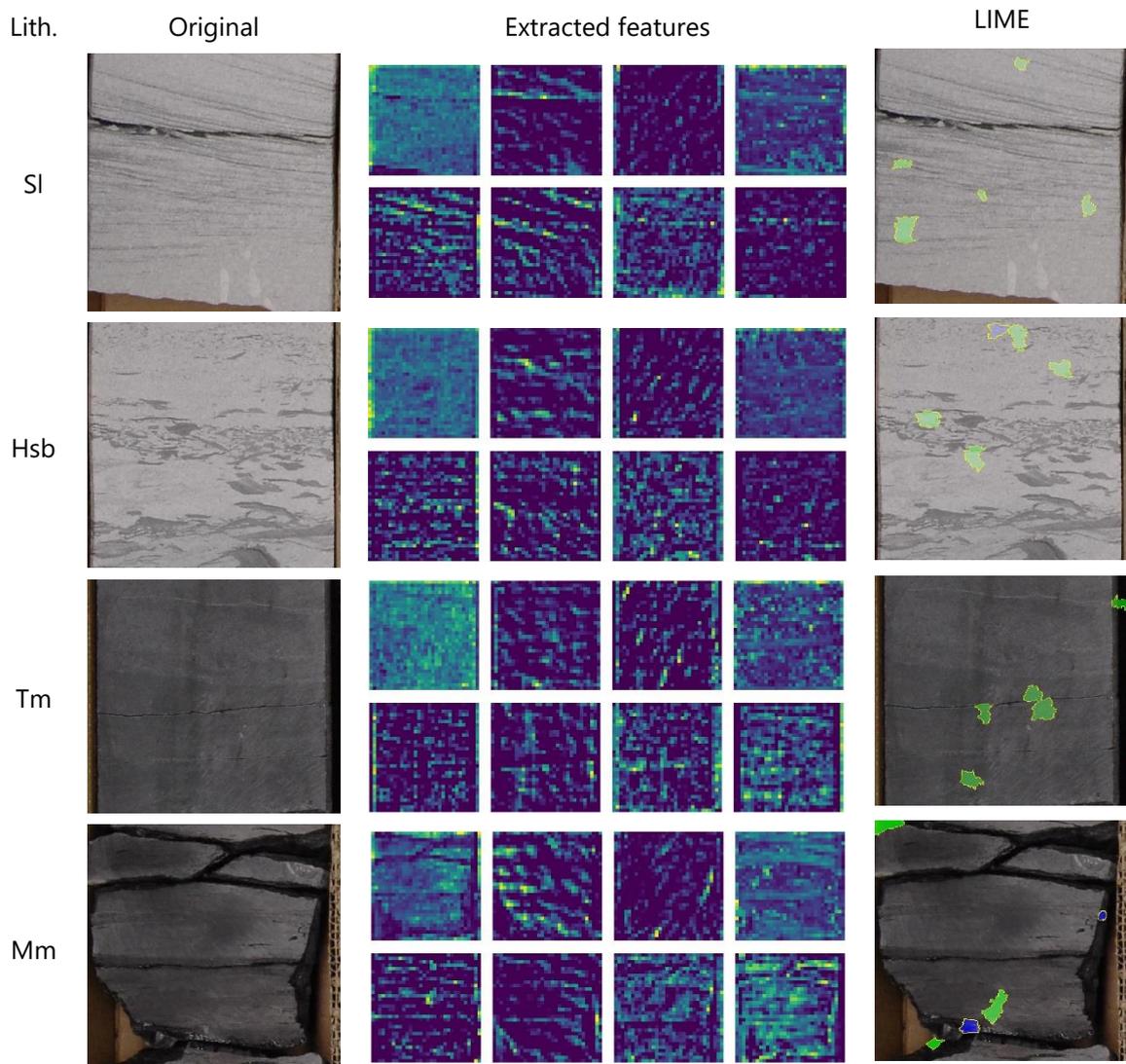
Figure 34. Some feature extraction examples in different lithotypes classified by ResNet-101. Hsb – laminated shaly sand with bioturbation, Mm – massive mudrock, Sl – laminated sandstone, Tm – massive siltstone. The following columns persist Lith. – lithotype, Original – original image, extracted features – randomly selected feature maps, LIME – image processing with application of Local Interpretable Model-agnostic Explanations, green areas indicate positive input for classification, blue – negative ones

Geotechnical description

A ResNet architecture was also used for geotechnical description. Two models were developed with it.

3-class geotechnical description

The first model trained on the GTD-1 (geotechnical description – 1) dataset aimed to detect the core condition. It contained three classes: crushed core, intact core, and cracked core. The trained algorithm performed well on test and validation sets (table 5, figure 35).

Table 5. The best metrics for the first iteration of the GTD-1 dataset gained with ResNet-101 architecture on new data.

Metric	ResNet-101
Precision	0.887
Recall	0.885
F1	0.886

As seen in figure 35, the same as in the sedimentological dataset, the CNN classify some examples differently. It is also sometimes mixing intact core and cracked core.
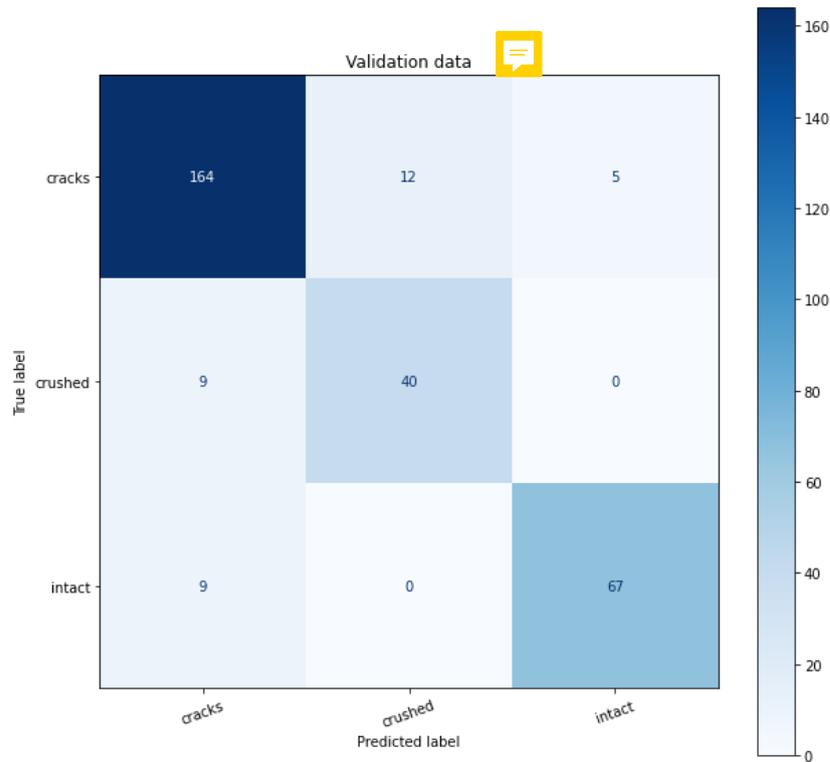
Figure 35. A confusion matrix for the ResNet-101-based GTD-1 interpretation algorithm inference on a validation set.

These performance issues should be eliminated over time when more data is acquired (at least more than 1000 examples per class). The developed algorithm was applied to different data and proved it stability in many cases. Although it was trained on 6,6 cm core samples, it also works on 10 cm core data from the sedimentological dataset. This model is domain-agnostic as the features developed by the layers are pretty similar for any rock image.
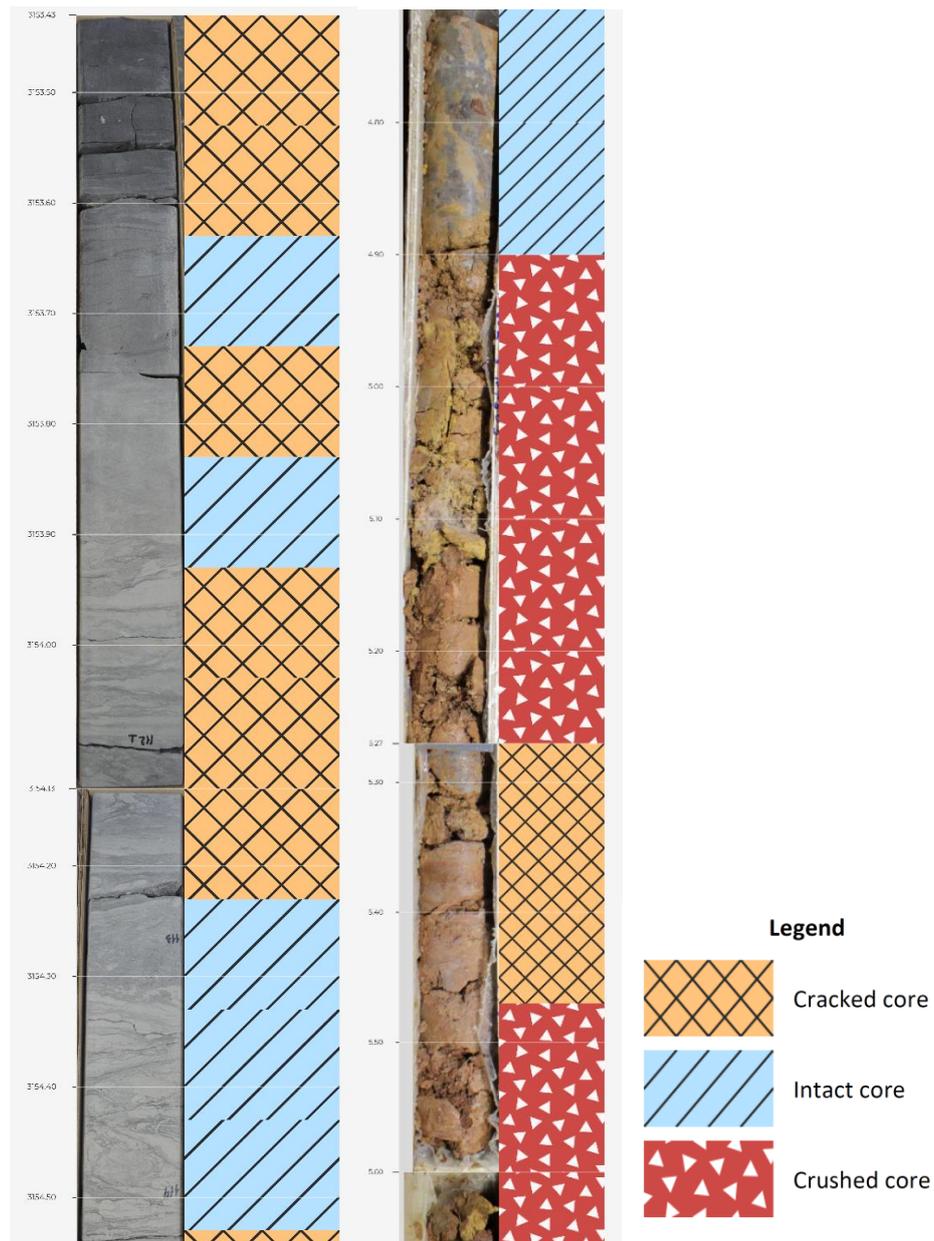
Figure 36. The GTD-1 model is applied to new data from different domains. Sedimentological-like dataset to the left, a new dataset to the right

5-class geotechnical description

The second model trained on the GTD-2 dataset was also validated with the results described in table 6.

Metric	ResNet-101
Precision	0.849
Recall	0.839
F1	0.832

The validation results on new data were lower than on a test set and presented in figure 37 and table 7.

Figure 37. A confusion matrix for the ResNet-101-based GTD-2 interpretation algorithm inference on a validation dataset.

The confusion matrix (figure 37) shows that some labels were misinterpreted. Some of the "Mylonite" was labelled either as cataclasite (4 cases) or fracture (2

samples). The same is for "Cataclasite". As can be seen, most of the "Quartz" and "Non-tectonized" labels were classified correctly. The weighted metrics for this validation set a presented in table 7.

Table 7. The metrics for the GTD-2 dataset gained with ResNet-101 architecture on a validation data

Metric	ResNet-101
Precision	0.77
Recall	0.76
F1	0.76

So, the validation didn't go well as in the GTD-1 dataset. This is caused by multi-label (dual sense) classes (the image has several labels which are related and equally correct) which especially can be noted when new data was used to check the trained network (Figure 38).
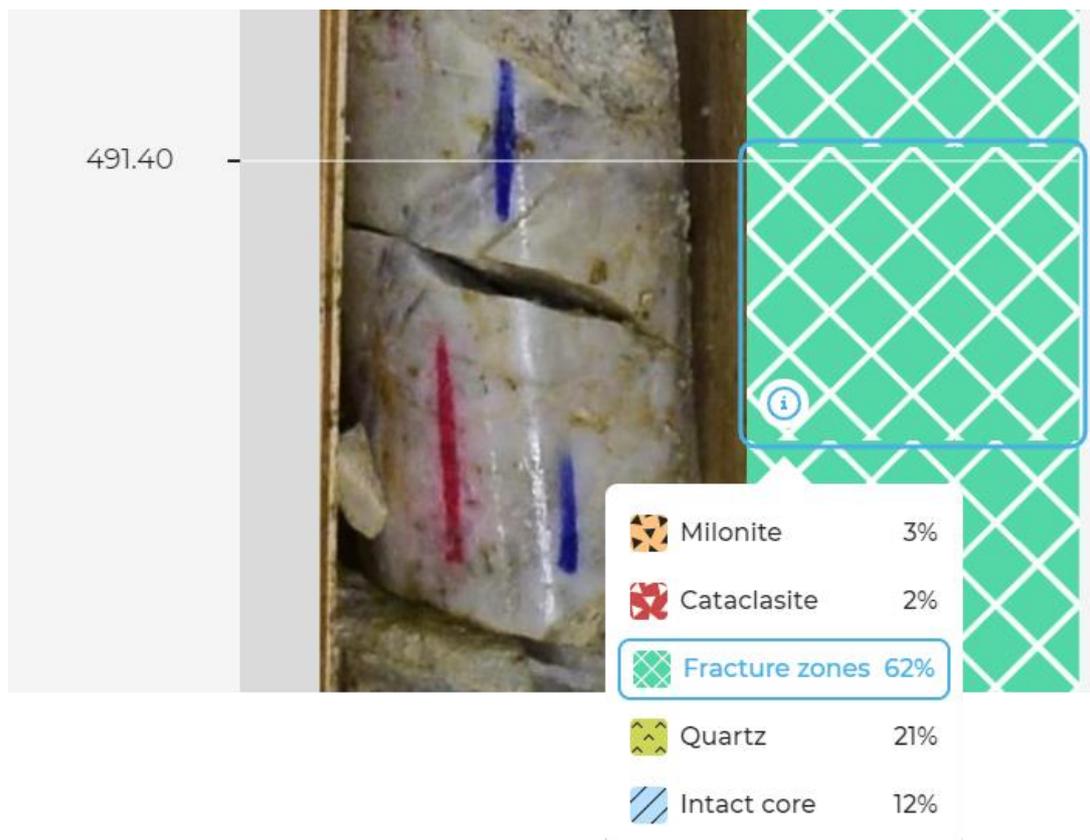
Figure 38. An example of classification result with CNN trained on GTD-2 dataset. The percentage shows probabilities. The image contains both quartz (the core is white) and a fracture zone (there are several cracks on the image), so the image has multiple labels.

The idea of a multi-label class is well-illustrated in figure 38. The probabilities are distributed in the way they should be. First, the algorithm "sees" the fracture in the

middle, then it detects that the fracture is in the quartz vein, and finally, two intervals between it are presented as an intact core. Suppose the dataset will be relabeled in two or three ways with a different meanings. In that case, it will be easier to train several simple yet stable algorithms (e.g., one is for detecting the core condition, the second is detecting quartz, and the last is for mylonite \ cataclasite detection).

The features from the algorithm were also analyzed, and a different set of filters were observed (figure 39). A filter for detecting vertical structure is presented in the extracted features' first row, and the texture is detected in the second row. Unlike in the sedimentological dataset – the LIME algorithm could detect the regions which could cause misclassification (blue colored in figure 39)
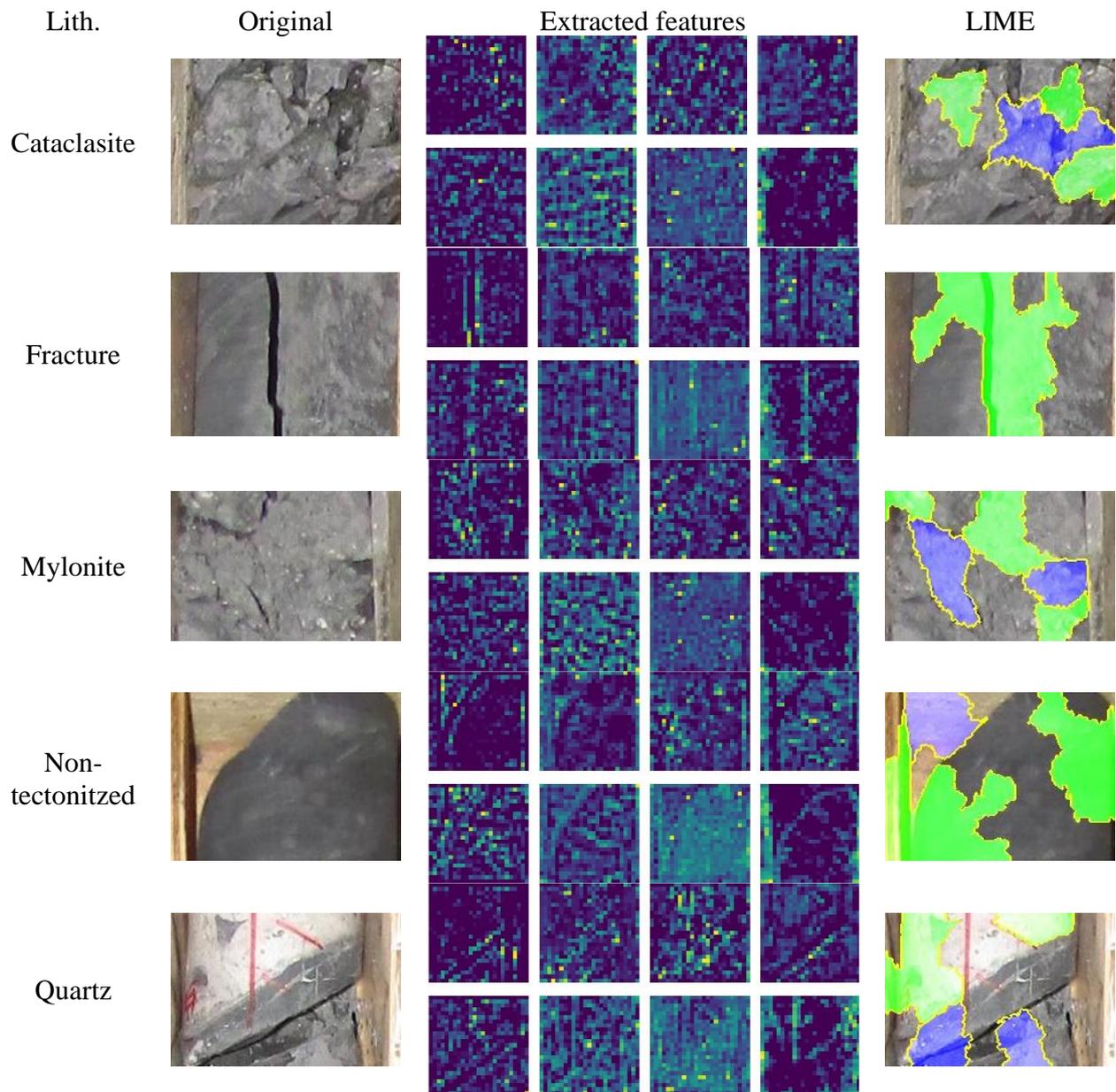
Figure 39. Some feature extraction examples in different lithotypes. Lith. – lithotype, Original – original image, Extracted features – randomly selected feature maps, LIME – image processing with an application of Local Interpretable Model-agnostic Explanations, green areas indicate positive input for classification, blue – negative ones

Chapter 5. Segmentation

The segmentation task may be solved in different ways. A traditional approach to the segmentation problem is to create superpixels, regions of similar color connected with semantic meaning. Another algorithm may further classify those regions. A set of methods is presented here to make a semi- and supervised segmentation of images compared to the traditional approach. In this work, the segmentation was used to solve several tasks. First – to extract the core column from an image; second – to classify different features on an image. The core extraction is aimed to help in those cases when a user provides a core box with some amount of core presented in columns on an image. These columns should be marked within a mask to allow the ~~machine~~ "understand" the image and extract the columns from a box and reference depths. Similarly, another task was solved – to classify the image into different rock types.

5.1. Results and discussion

Two models were developed during the research. One is for core column detection inside the core box image, and the other is for sand and shale detection, which is used for NTG (net-to-gross) calculation. Such datasets usually absent open data sources and should be manually labelled.

5.1.1. Datasets

The dataset previously collected for classification was reused for manual pixel-by-pixel labelling.

The labels are stored in the form of a mask –a greyscale image with the same height and width as the original image. Each pixel in that greyscale image represents the label. For binary cases – they are two types of pixels – the darkest (traditionally – it equals 0) and the brightest (255).

Box segmentation

Usually, the core column image is stored inside a core box image the last should be analyzed to extract the image from the box. Regardless of image size and resolution, all of them were labelled, as shown in figure 40.

Figure 40. Examples of the labelled datasets for core column extraction. A – the original image; B – the produced mask.

A set of 1000 images ~~was~~ labelled.

Rock segmentation

In the same way as core boxes, the 10x10 cm images from the rock typing dataset (see chapter 4.1.1) were labelled. Semi-supervised methods can be applied. The dataset was labelled with the use of a watershed algorithm implemented in PixelAnnotationTool (PAT) (Bréhéret, 2017) (Figure 41). Each type of rock has its own label 15 for sandstone and 33 for shale.
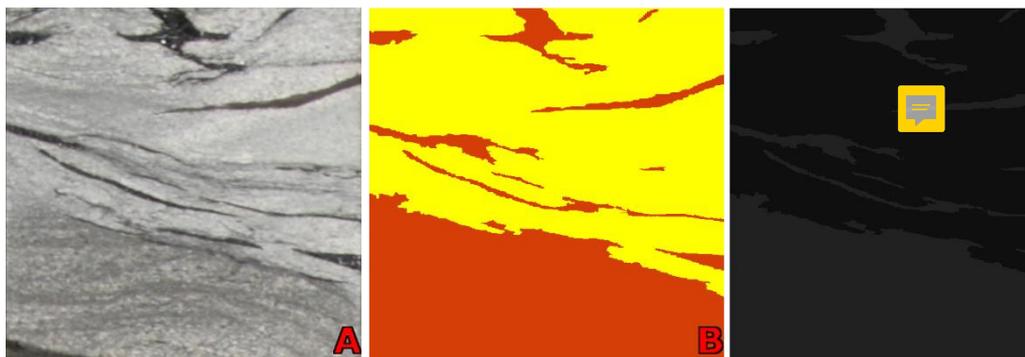
Figure 41. Dataset labelling example for rock type segmentation task gained with PixelAnnotationTool (Bréhéret, 2017). A – original image, B – human-readable format (yellow – sandstone, orange – shale, C – machine-readable labelling format (barely visible grey – sandstone, black – shale). The size of the sample is around 10 cm.

A total amount of 400 images were labelled this way.

5.1.2. Core box segmentation

As experiments with the classification dataset proved that the deep-learning algorithm could be successfully applied to rock-type detection, the new datasets were also used to train CNNs for semantic segmentation. A U-Net architecture was used with the Res-Net backend. Both architecture and backend (as discussed in section 4.1.3.) proved to have a great generalization ability on a small dataset.

Preprocessing

The data should be represented in HxWxC form to train a CNN for semantic segmentation, where H – is the height of an image, W – is the width, and C is for channels. In a color image, channels are represented in RGB format. In the case of classification, each channel should be presented as only one label, as the channels represent classes.

Thus, a special framework for data preparation was developed. Various datasets can have different labels. Each label covers some number of pixels of the image. It is important to preprocess such labels correctly into a multidimensional array with a number of channels and classes equal to the number of objects segmented plus one more dimension for background; that is what the framework does.

Box segmentation

The results of this section were published in (E. E. Baraboshkin et al., 2022a). The usual solution for the problem of automated rock classification described in Chapter 4 is based on initially prepared core images. ~~Sadly~~, the preparation process may include a lot of human work, including image gathering, acquiring depth information, cropping of the image, and extracting the core columns. Thus, much time should be spent on this, even if only 15 core box images are processed. A good way to optimize the time spent on this manual work is to build an algorithm which will do this work for the human. First, the algorithm was trained on the provided dataset described in section 5.1.1. (Box segmentation subsection). The results of these experiments are described in figure 42.

#	Original image	First setup CNN	Second setup CNN	Metrics
				w/o TLA	TLA

#	Metrics w/o TLA	Metrics TLA
1	IOU: 0,752 F1: 0,859 F2: 0,801 Recall: 0,976 Precision: 0,766	IOU: 0,855 F1: 0,922 F2: 0,903 Recall: 0,955 Precision: 0,891
2	IOU: 0,371 F1: 0,541 F2: 0,425 Recall: 0,993 Precision: 0,372	IOU: 0,672 F1: 0,804 F2: 0,721 Recall: 0,992 Precision: 0,675
3	IOU: 0,285 F1: 0,444 F2: 0,664 Recall: 0,286 Precision: 0,993	IOU: 0,960 F1: 0,980 F2: 0,973 Recall: 0,992 Precision: 0,968

Figure 42. The comparison of an algorithm trained with two data setups. The rows contain different types of core boxes. 1,2- taken by hand with a camera, 3 - taken with a camera placed on a core-imaging stand.

Unfortunately, when it was tested on new data from the other distribution, the results were not impressive (the detailed metrics can be found in figure 42, first setup column, the mean IOU on the provided examples is 0,469). An application of the developed augmentation technique called TLA (template-like augmentation) is described in section 3.4. was tested both on humans and during the training period. A set of images from the same template were generated to test the effectiveness of the TLA method (figure 43). A questionnaire has been made on different social networks using this image set. The results of the questionary are presented in figure 44.

Figure 43. A set of images was prepared with the TLA technique. A questionary has been made based on this picture. The questionary results are presented in figure 44.
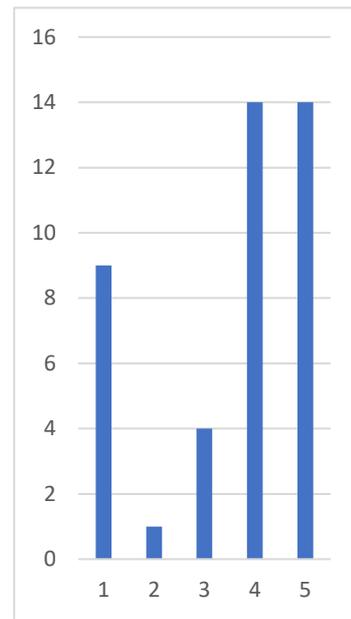
Figure 44. The results of a questionary on the generated images are described in figure 43; the x-axis is the image number, and the y–axis is the number of votes.

42 respondents took the survey. The questionary results demonstrate an overall confusion of humans within the images as the correct option (number 4, figure 43) was chosen the same number of times as the augmented version (rest of images, figure 43). It is important to note that 7 geologists also participated in this questionary and only 4 made the "correct" choice.

U-Net (Ronneberger et al., 2015), with ResNet-50 backbone available in the segmentation_models library, was used. Compared with other models (such as PSPNet (Zhao et al., 2017) and DeeplabV3 (Chen et al., 2017)), ResNet-50 shows the same metric values (discussed in the Results section). As the initial setup for neural network training, the hyperparameters were set to the following values: initial learning rate - 3e-4, batch size - 16, number of epochs - 15. The ReduceLROnPlateau was used as a scheduler to adjust the plateau's learning rate. The Early Stopping procedure was used to control the overfitting of the model. The complex loss function was used to train the model – binary cross-entropy and Jaccard loss with equal contribution to the optimization process.
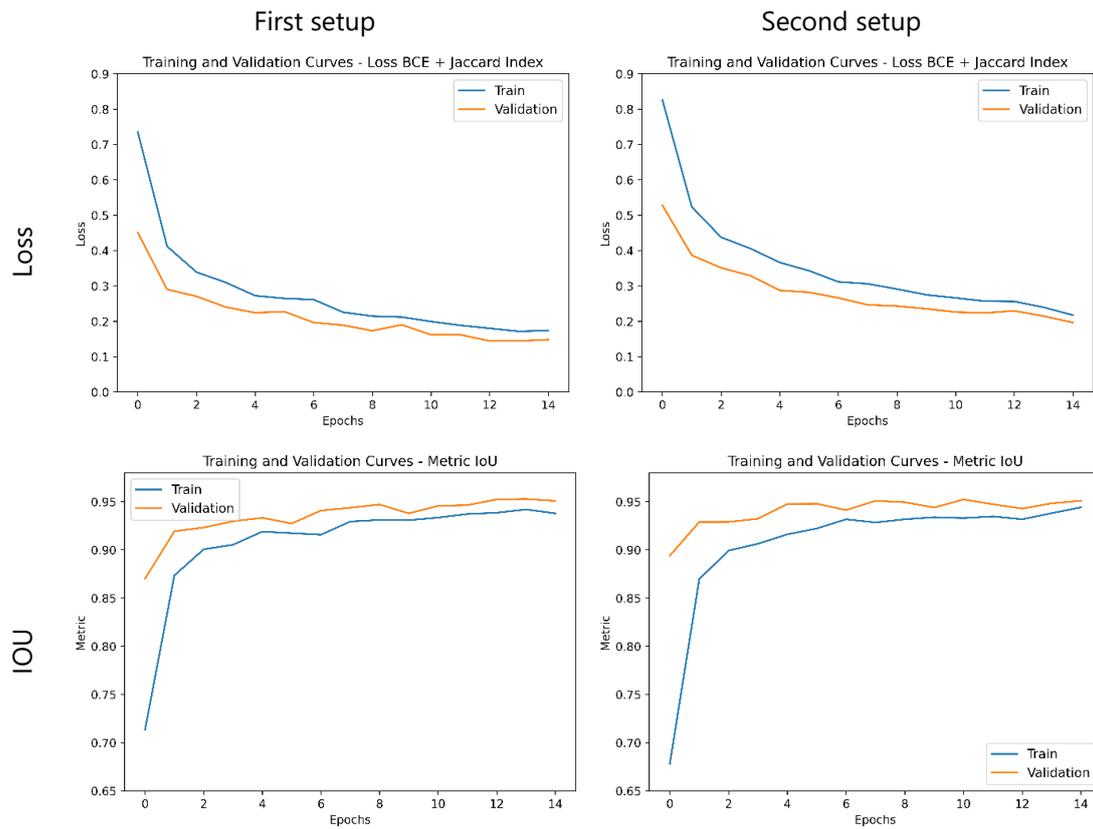
Figure 45. Loss and IOU curves for different data setups. First setup – the initial dataset separated by training and validation dataset, second setup – the training dataset expanded with TLA data.

Three groups of data conditions were analyzed during experiments: a dataset without TLA augmentation (first setup, which included an augmentation from the Albumentation library), a composed dataset with TLA and non-augmented images (second setup), and a dataset with TLA only (third setup).

In the experiments, it was noted that the CNNs segment the images differently. The first setup training tended to segment most new images as a target value. The second setup training provided the best results during the visual screening of any data, whether similar to the training set images or new images. The third setup worked properly only on a few images; thus, just two first settings are compared here. The application of the TLA algorithm increased the segmentation results on a new set of data (Figure 42, second setup, mean IOU on the provided examples is 0,829) in 1,6 times.

A statistical approach was designed to filter out the "negative" and "positive" examples during image postprocessing to improve the algorithm's effectiveness. The

bounding box approach implemented in the OpenCV library was used to extract the core from the box. It produced bounding boxes in the following way: the starting coordinates of a box – x and y and the bounding box's width and height. There are several characteristics of bounding boxes to look at. Firstly, the number of boxes in images varies from 1 to 6. This information was used to roughly evaluate the performance of an algorithm with simple bounding box counting. Secondly, the core can be found at the upper part of an image, so we can use this information to filter some boxes by the y coordinate. In combination with bounding box width and height, we can have several easy-to-estimate statistics that can increase the core extraction results for the user and further increase the recall (figure 46, figure 47).
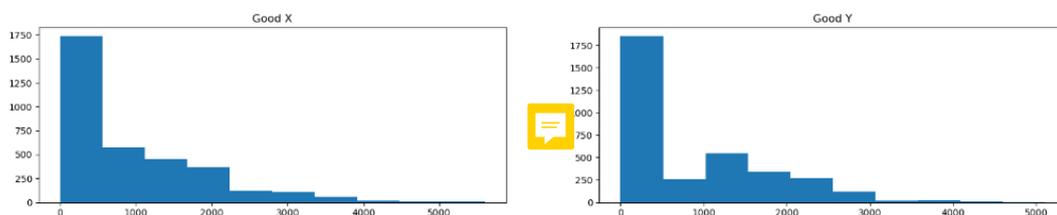
Figure 46. Distribution of x and y coordinates for "good" core bounding boxes detection examples.
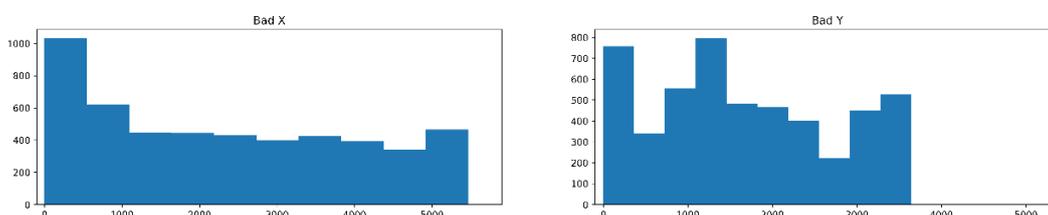
Figure 47. Distribution of x and y coordinates for "bad" core bounding boxes detection examples.

Based on statistical analysis, different cleaning-up approaches were developed:

1) Based on a median width and height of a core.

   After mask generation, the bounding boxes detected with the OpenCV library and their respective median height and width are calculated. After that, the median can be used to calculate boundary conditions for filtering.

   $xt\_up = x * n; xt\_bot = y/n,$

   where n – is a manually set coefficient (varies from 1.2 to 1.5), x – median width, y - median height, *xt_up* - is the threshold for the upper boundary of a core mask width, and *xt_bot* – is the bottom boundary of width.

   The same algorithm can be applied to height.

99

As a matter of demonstration – there are 10 bounding boxes on an image were detected. The calculated median width was 430, so after filtering, 5 core columns were left as the width of the others was either lower or higher than the threshold set (from 358 (*xt_bot*) to 516 (*xt_up*), since coefficient n was equal to 1.2).

2) Based on a width of an image threshold ratio (the core width cannot be less than this ratio).

gt = xi/m,

where *gt* – is the global image width ratio, *xi* – is the image width, and *m* – is the manual set coefficient (varies from 50 to 200).

For example, if the image width is 4000, the *m* coefficient is set to 100, so the minimum core width is 400. Followed by the previous demonstration, the same columns can be filtered out and receive 6 bounding boxes as all of them coincide with the minimum core width threshold (*gt*).

Both methods were used after experiments to increase the results for prediction and inform a user about the current segmentation status. The core image extraction procedure was shortened by 20 times with the developed algorithm. Also, as the bounding box is connected to the core's physical length, the applied algorithm allowed ease and improved the process of depth referencing.

As this method showed promising results, a small experiment was done on the rock segmentation dataset described below.

5.1.3. Rock type segmentation

As a small experiment to test a hypothesis – an algorithm for segmentation was tested for the ability to segment the rock types. The results were published (Baraboshkin et al., 2019). Several approaches were tested: unsupervised segmentation based on super-pixels construction and supervised approach based on deep learning.

Super-pixel-based segmentation

Superpixel segmentation allows the extraction of information in an unsupervised way. An expert should further decide which kind of information each segment belongs to. It could be either a rock type or some other characteristic.
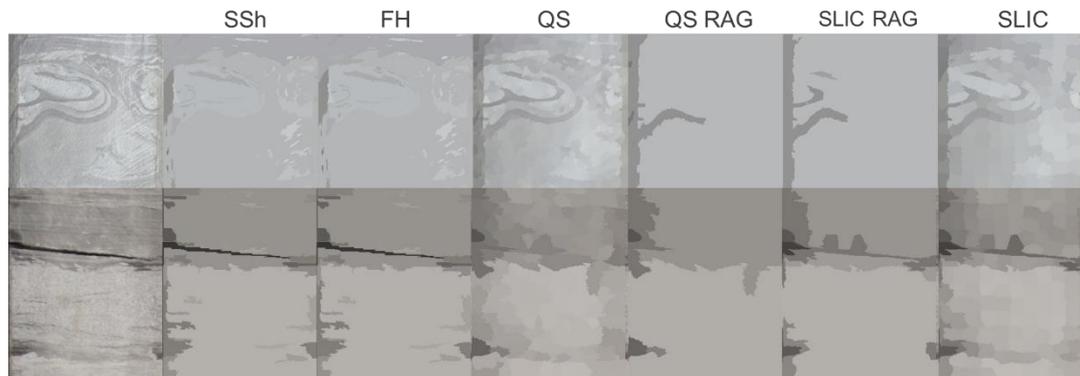
Figure 48. Comparison of different segmentation algorithms. AC – agglomerative clustering. FH - Felzenszwalb-Huttenlocher. QS - Quick Shift. SLIC - Simple Linear Iterative Clustering. RAG - Region Adjacency Graph. SSh - Selective Search.

As seen in figure 48, each tested algorithm shows different results except the SLIC algorithm, which is produced due to its nature of equally-sized regions, except those much different from others which can be used for further interpretation. Unfortunately, as these regions should be further processed, the methods are not suitable for fully automated description as the system based on them will be too complex.

Deep-learning-based segmentation

In the case of rock segmentation, the deep learning approach allowed the classification of the images on a pixel-by-pixel basis in a fully automated way (figure 49). Even with a small dataset, the classification results allowed us to calculate the NTG ratio without much human involvement.
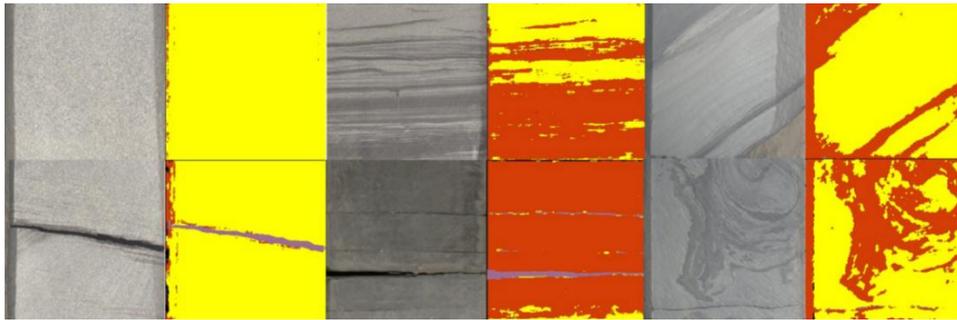
Figure 49. An example of CNN-based segmentation. The results allow calculating the NTG ratio after training on a small dataset (only 400 images)
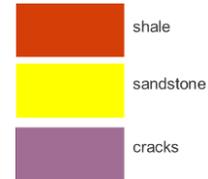
As can be seen in figure 49 – most of the structures are separated correctly, but an algorithm may misinterpret some shadows as shales. Also, thin layers are not classified correctly in most cases.

Testing on a new dataset

The trained and tested algorithm was applied to new data to see the potential of their application on the new data (figure 50). As can be seen, the algorithms can segment various features on an image. Unfortunately, for all the super-pixel-based methods, additional postprocessing is required. CNN's results are inspiring, but the manual labelling took time. The classification techniques (discussed in chapter 4) are much less time-consuming and more versatile in data preparation.
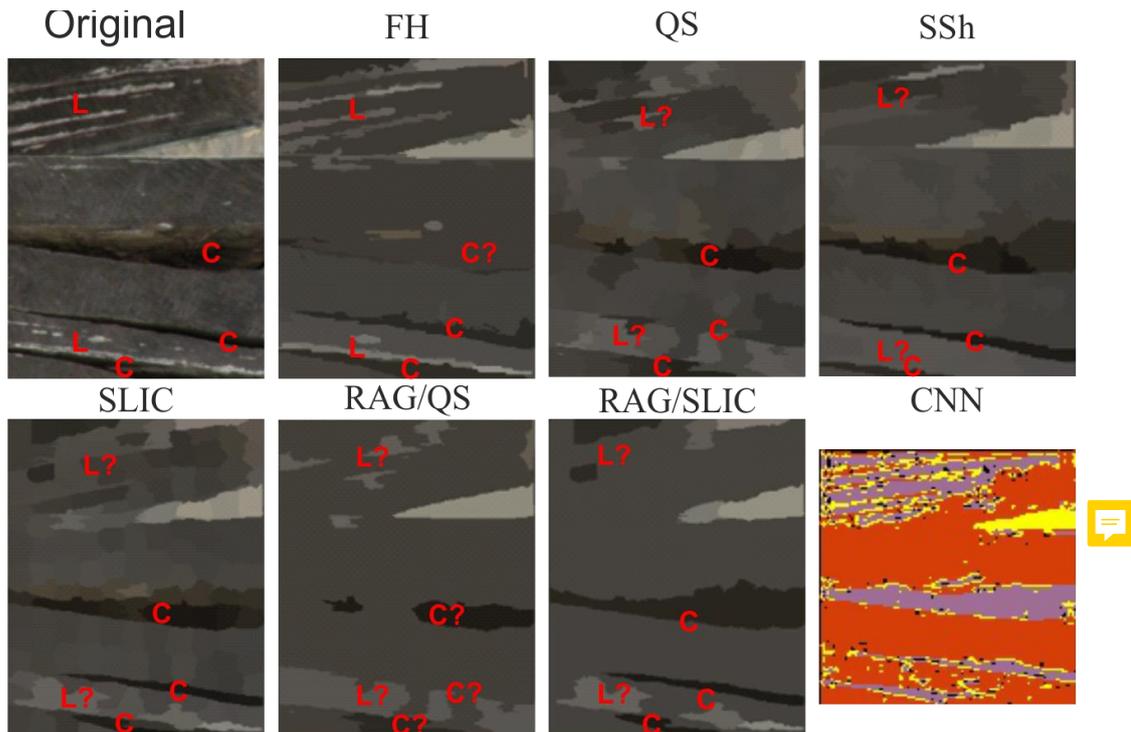
Figure 50. A segmentation example with different algorithms implementation. AC – agglomerative clustering. FH - Felzenszwalb-Huttenlocher. QS - Quick Shift. SLIC - Simple Linear Iterative Clustering. RAG - Region Adjacency Graph. SSh - Selective Search, CNN – convolutional neural networks. Various features can be found on an image: C – cracks, L – layers. Colors on the CNN image: yellow -sandstone, purple – cracks, orange – argillite.

5.2. Summary

Various segmentation algorithms were tested to solve different tasks: core box image understanding to ease the core column preparation process and different rock segmentation types.

The core box segmentation was very useful in data preparation combination, but as the available images didn't describe the variation of core boxes sufficiently – it failed on new data. A new approach for data augmentation was developed to improve the results - (E. E. Baraboshkin et al., 2022a), which helped to improve the core columns extractions on most new examples sufficiently.

Rock segmentation task may be useful, but due to the hard data preparation process (described in sections 5.1.1. and 3.2.). It is not suitable for production purposes in comparison to classification.

Chapter 6.   System construction

It is sometimes difficult to develop a good algorithm for research purposes, but it is even harder to turn this algorithm into a production-ready product.

6.1. Results and discussion

A set of system architectures were tested. Each of them has its benefits and drawbacks. The architecture unites both segmentation and classification algorithms. The segmentation is used to extract a core from core box images and do automated depth referencing, and classification is applied to classify the core into different classes.

6.1.1. A locally installed application

The first architecture was developed using ~~pure~~ python and a local launcher. Its GUI is presented in figure 51.

The main pitfall of such a system is that the whole process of algorithm calculations and data is stored and processed on a personal laptop with random characteristics.

The second problem is that such an implementation requires a lot of space to store the program and trained algorithms.

Finally, it is hard to check a license in case you have some commercial interests. Also, it is hard to update and maintain such a system due to variable system installations and the monolithic structure of the app (the whole program works in one way within one process).

Still, such an approach has its benefits, like it doesn't require a constant internet connection which can be a trouble for some users, and there is no need to spend a lot of money to maintain servers that will store all the information.

As most of the software currently made supports cloud solutions, the user could operate with the data from any point around the world and collaborate with his colleagues, a decision was made to prepare a prototype that will work as a web application.
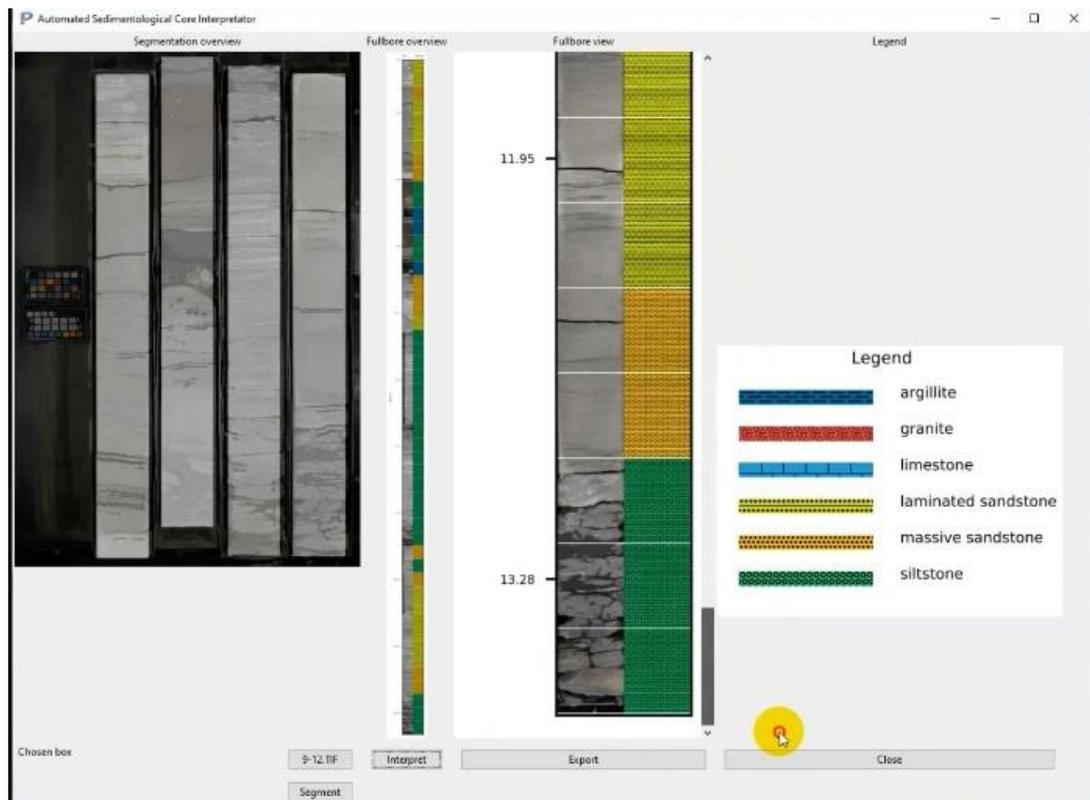
Figure 51. A pure python program with algorithms implementation.

6.1.2. A web application

The web application was developed with the application of Rest-API protocols. These protocols are widely used in all browsers, and almost any modern web application has them. As mentioned before, the web app doesn't require installation and any user who visits the www-site will be able to use it.

The app was also built as a monolith, so a user should wait some time to process his images, and only one processor could do that. But, due to a server deploy, the number of simultaneous connections increased, which enables processing many more images within different users. No database was created, so the user could only see the result.

It was also easy to modify the app because all users got the required update. At the same time, if an update has some bug, all users would suffer, and if the monolithic system crushes, it has to recover for a long time.

Another pitfall was in the CNN architecture update – it wasn't possible to update the architecture "on the fly"; only reboot and immediate logout of all users were required.
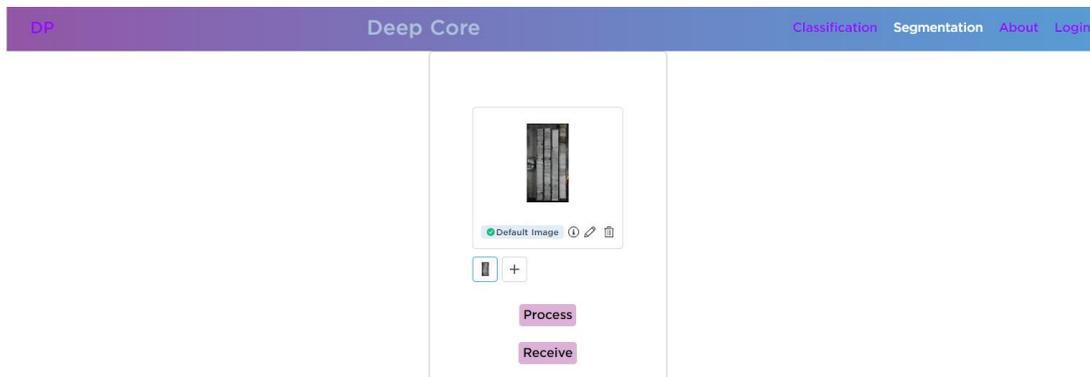
Figure 52. A web-app prototype.

6.1.3. Microservice web application

Another way to operate the system with much more flexibility is microservices-based software architecture. This time command of 6 people developed the system, and each member had a different function.

In such a case, many virtual or physical servers are required. Each service which will be created will have its server, so the most significant disadvantage here is the cost of such infrastructure.

Four services were created. Authentication service compares the information user entered with one in the database and registers the user if required. The classification service accepts the image path and returns the resulting classification. Segmentation takes the path of an image, creates a mask, and returns the path to the mask. The backend service operates the internal processes within and exchanges data between services.

A huge modification was also made to create a database, separate object storage vault (for images and models), and task-management system. Each of these must be deployed on different machines.

The benefit of such a system is that it can be easily scaled, and in case some services fall, they could be recovered much faster, and, in most cases, the user won't notice it.

The final GUI examples are shown in figures (figure 53, figure 54).
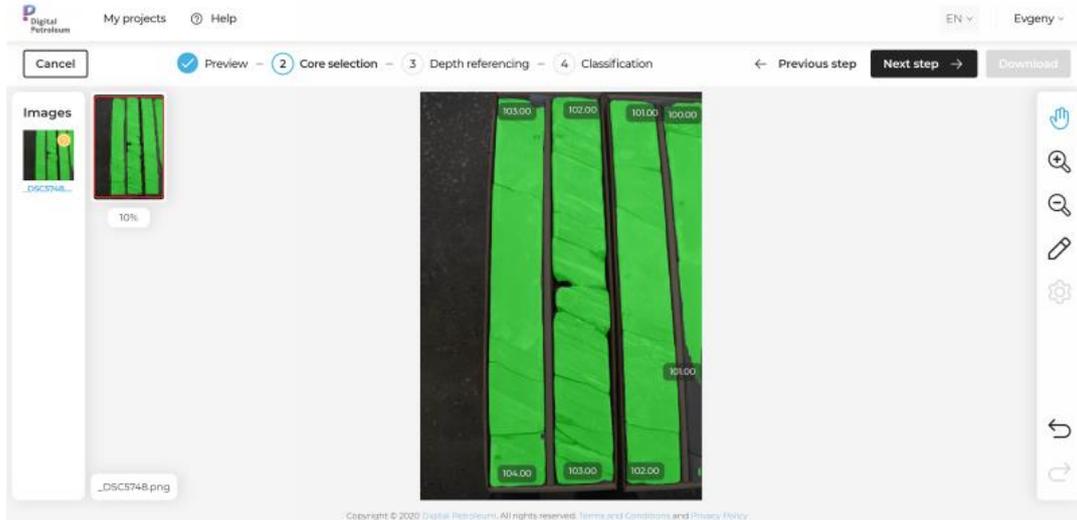
Figure 53. A mix of python and additional frameworks implementation. Core box extraction step.

A significant modification was made to the results editing. A user could modify segmentation and classification results which allow for checking the results and adding some specific classes if they were not used in the model.
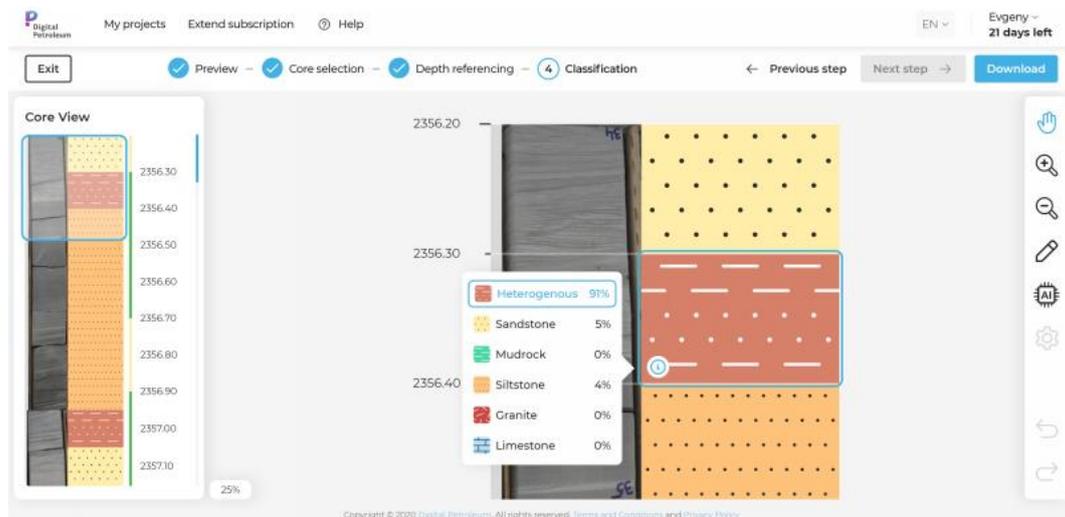
Figure 54. A mix of python and additional frameworks implementation. Core classification step.

The developed system increases the geologist performance by a factor of 7 which was proved by tests in production (E. Baraboshkin et al., 2022; Baraboshkin et al.,

2021a, 2021b; E. E. Baraboshkin et al., 2022b). A geologist spent only 40 minutes on depth referencing and description with editing instead of 5 hours.

Chapter 7.   Conclusions

The work introduces a new method for applying convolutional neural networks (CNN) for geological image understanding. The method was tested on different datasets taken under different conditions (sect. 4.1.3.). The accuracy of such a method could be up to 95% which may depend on a geologist's opinion. The developed algorithm could easily be applied or modified according to the new conditions. Also, a comparison of CNN with "traditional" algorithms was made (sect. 4.1.2.). Most "traditional" algorithms failed to perform an outstanding description which would be at least slightly comparable to the geologist's work. The applied CNN performed outstanding results in different tasks (4.1.3.).

A lot of requirements should be met to make an automated core description with CNN:

1) There should be a lot of image data.
2) The dataset should be clean.
3) The labels should be strict.
4) The images should be of fine quality (at least 80 DPI).

In most cases, a geologist with a perfectly prepared database will have to rework the database several times before the above requirements are met (sect. 4.1.1.), as the databases are usually designed on a larger scale.

It was shown that multi-label classes decrease the overall CNN performance and may cause a misleading conclusion on the machine learning application in geology (4.1.3. , 5-class geotechnical description).

Every developed classification CNN was "unboxed" with modern visualization techniques (the description of such techniques in sect. 2.4.2.).

Each result is provided by a stable classification CNN model that was checked and approved by an expert on a new dataset.

CNN can be applied both for the classification and segmentation of an image. The latest is specifically useful in two cases: core column image detection and pixel-by-

pixel classification, but the latter is time-consuming in data preparation. The core column segmentation allowed 20 times the speedup of core extraction from an image (5.1.2.).

A set of unsupervised algorithms was applied to both segmentation and classification tasks. Unfortunately, they failed due to the unreasonable complexity of such algorithms in pre- and postprocessing.

The following classification models were developed: core condition description (3 classes), sedimentological description (14 classes), and geotechnical description (5 classes). These models were applied both in hydrocarbon recovery and mining industries.

Two models for segmentation were developed: core box segmentation (2 classes) and rock type segmentation (3 classes). The first model is universal for any data. A special augmentation technique (3.4.) was introduced to increase the model's generalisation ability. The second model was only tested on validation data and probably should be further developed in case of production needs.

The developed models were united in a system that helped geologists describe the core much faster. The segmentation model helps to detect the core on a core box image, and the classification model classifies the images according to the chosen model.

The developed algorithms are implemented inside a system as the software allows speeding up the description by a factor of 7.

Chapter 8. <mark>Recommendations for future research</mark>

A broad area of research in the field of automated core description is still uncovered.

It is a question of future research on how the performance of an algorithm changes if the multi-label class is separated into several classes, and several CNN would be trained on top of it. A special algorithm could be developed to use such multi-label trained CNN to create semi-automated labelling.

An interesting task is semi- and unsupervised labelling applied to rock-type classification. This work failed to create an unsupervised algorithm, but it was tested on a very small data quantity. The results may change if the data is sufficiently large (at least 10000 images per class).

Another way to improve the classification and create a better classification algorithm is to simultaneously use multimodal data such as well- and core logs, UV images, and daylight images. Currently, it is hard to reference these data to each other as it either missed or corrupted.

The developed core box segmentation algorithms may be further improved by adding more labels to the dataset. The labels could be additionally used for data extraction like depth referencing and coring direction.

Bibliography

Abashkin, V., Seleznev, I., Chertova, A., Samokhvalov, A., Istomin, S., Romanov, D., 2020. Digital analysis of the whole core photos, in: 1st EAGE Digitalization Conference and Exhibition. European Association of Geoscientists and Engineers, EAGE, pp. 1–5. https://doi.org/10.3997/2214-4609.202032058

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2010. SLIC Superpixels. EPFL Technical Report 149300.

Alférez, G.H., Vázquez, E.L., Martínez Ardila, A.M., Clausen, B.L., 2021. Automatic classification of plutonic rocks with deep learning. Applied Computing and Geosciences 10. https://doi.org/10.1016/J.ACAGS.2021.100061

Allaby, M., 2013. A Dictionary of Geology and Earth Sciences, A Dictionary of Geology and Earth Sciences. Oxford University Press. https://doi.org/10.1093/acref/9780199653065.001.0001

Alzubaidi, F., Mostaghimi, P., Si, G., Swietojanski, P., Armstrong, R.T., 2022. Automated Rock Quality Designation Using Convolutional Neural Networks. Rock Mechanics and Rock Engineering 2022 1, 1–16. https://doi.org/10.1007/S00603-022-02805-Y

Alzubaidi, F., Mostaghimi, P., Swietojanski, P., Clark, S.R., Armstrong, R.T., 2021. Automated lithology classification from drill core images using convolutional neural networks. J Pet Sci Eng 197, 107933. https://doi.org/10.1016/j.petrol.2020.107933

American Petroleum Institute, 1998. Recommended Practices for Core Analysis. Rp 40 236.

Appendix no 5. To the regulations on "Core investigation" (In Russian), 2017. . Rosneft, Moscow, p. 77.

Arthur, D., Vassilvitskii, S., 2007. k-means++: The Advantages of Careful Seeding, in: Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms. https://doi.org/10.1145/1283383.1283494

Arya, S., Mount, D.M., Netanyahu, N.S., Silverman, R., Wu, A.Y., 1998. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. Journal of the ACM 45, 891–923. https://doi.org/10.1145/293347.293348

Ayranci, K., Yildirim, I.E., Waheed, U. bin, Maceachern, J.A., 2021. Deep learning applications in geosciences: Insights into ichnological analysis. Applied Sciences (Switzerland) 11. https://doi.org/10.3390/APP11167736

Baraboshkin, E., Ismailova, L., Orlov, D., Koroteev, D., 2019. Machine Vision Methods in the Application for Core Image Segmentation, in: Progress'19. European Association of Geoscientists & Engineers, pp. 1–5. https://doi.org/10.3997/2214-4609.201953061

Baraboshkin, E.E., Baraboshkin, E.Yu., Ismailova, L.S., Orlov, D.M., Zhukovskaya, E.A., Kalmykov, G.A., Khotylev, O.V., Koroteev, D.A., 2020. Deep convolutions for in-depth automated rock typing. Comput Geosci 135. https://doi.org/10.1016/j.cageo.2019.104330

Baraboshkin, E.E., Demidov, A.E., Orlov, D.M., Koroteev, D.A., 2022. Core box image recognition and its improvement with a new augmentation technique. Comput Geosci 162. https://doi.org/10.1016/j.cageo.2022.105099

Bradski, G., 2000. The OpenCV Library. Dr. Dobb's Journal of Software Tools 120, 122–125.

Bradski, G., Kaehler, A., 2008. Learning OpenCV, Computer Vision with OpenCV Library. https://doi.org/10.1109/MRA.2009.933612

Bréhéret, A., 2017. Pixel Annotation Tool.

Breiman, L., 2001. Random Forests. Mach Learn 45, 5–32. https://doi.org/10.1023/A:1010933404324

British Geological Survey (BGS) Rock Classification Scheme [WWW Document], n.d. URL http://www.bgs.ac.uk/bgsrcs/

Budennyy, S., Pachezhertsev, A., Bukharev, A., Erofeev, A., Mitrushkin, D., Belozerov, B., 2017. Image processing and machine learning approaches for

petrographic thin section analysis. Society of Petroleum Engineers - SPE Russian Petroleum Technology Conference 2017.

Bukharev, A., Zhukovskaia, E., Stc, L.L.C.G., 2018. SPE-191591-18RPTC-MS Digitization of Petrographic Data for Standardization and Genetic Interpretation.

Buslaev, A., Parinov, A., Khvedchenya, E., Iglovikov, V.I., Kalinin, A.A., 2018. Albumentations: fast and flexible image augmentations. ArXiv e-prints 1–4.

Butterfield, A., Ngondi, G.E., Kerr, A., 2016. A Dictionary of Computer Science, Oxford quick reference. Oxford University Press.

Cai, W., Deng, J., Lu, Q., Lu, K., Luo, K., 2021. Automated Classification of High-resolution Rock Image Based on Residual Neural Network. J Phys Conf Ser 2095, 012051. https://doi.org/10.1088/1742-6596/2095/1/012051

Chatterjee, S., 2013. Vision-based rock-type classification of limestone using multi-class support vector machine. Applied Intelligence 39, 14–27. https://doi.org/10.1007/s10489-012-0391-7

Chatterjee, S., Bhattacherjee, A., Samanta, B., Pal, S.K., 2010. Image-based quality monitoring system of limestone ore grades. Comput Ind 61, 391–408. https://doi.org/10.1016/j.compind.2009.10.003

Chatterjee, S., Bhattacherjee, A., Samanta, B., Pal, S.K., 2008. Rock-type classification of an iron ore deposit using digital image analysis technique. Int J Min Miner Eng 1, 22. https://doi.org/10.1504/IJMME.2008.020455

Chauhan, S., Rühaak, W., Anbergen, H., Kabdenov, A., Freise, M., Wille, T., Sass, I., 2016. Phase segmentation of X-ray computer tomography rock images using machine learning techniques: An accuracy and performance study. Solid Earth 7, 1125–1139. https://doi.org/10.5194/se-7-1125-2016

Chawla, N. V, Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: Synthetic Minority Over-sampling Technique, Journal of Artificial Intelligence Research.

Cheng, G., Guo, W., 2017. Rock images classification by using deep convolution neural network, in: Journal of Physics: Conference Series. Institute of Physics Publishing. https://doi.org/10.1088/1742-6596/887/1/012089

Chollet, F., 2017. Deep learning with python. Manning Publications Company.

Chollet, F., Google, Microsoft, Others, 2015. Keras [WWW Document]. https://keras.io.

Collins COBUILD Advanced Learner`s Dictionary, Ninth edit. ed, 2018. . HarperCollins UK, Glasgow.

Comaniciu, D., Meer, P., 2002. Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell 24, 603–619. https://doi.org/10.1109/34.1000236

Cracknell, M.J., Reading, A.M., 2014. Geological mapping using remote sensing data: A comparison of five machine learning algorithms, their response to variations in the spatial distribution of training data and the use of explicit spatial information. Comput Geosci 63, 22–33. https://doi.org/10.1016/J.CAGEO.2013.10.008

Cui, Y., Jia, M., Lin, T.-Y., Song, Y., Belongie, S., n.d. Class-Balanced Loss Based on Effective Number of Samples.

Dalal, N., Triggs, B., 2005. Histograms of Oriented Gradients for Human Detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). IEEE, pp. 886–893. https://doi.org/10.1109/CVPR.2005.177

Denning, P.J., Comer, D.E., Gries, D., Mulder, M.C., Tucker, A., Turner, A.J., Young, P.R., 1989. Computing as a discipline. Computer (Long Beach Calif) 22, 63. https://doi.org/10.1109/2.19833

DeVries, T., Taylor, G.W., 2017. Improved Regularization of Convolutional Neural Networks with Cutout. https://doi.org/10.1016/j.neuron.2007.06.026

Dimitri, F., 2019. The Art of PostgreSQL 457.

Duchi JDUCHI, J., Singer, Y., 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization * Elad Hazan. Journal of Machine Learning Research 12, 2121–2159.

Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Classification.

Enge, H.D., Buckley, S.J., Rotevatn, A., Howell, J.A., 2007. From outcrop to reservoir simulation model: Workflow and procedures. Geosphere 3, 469–490. https://doi.org/10.1130/GES00099.1

Fan, G., Chen, F., Chen, D., Dong, Y., 2020. Recognizing Multiple Types of Rocks Quickly and Accurately Based on Lightweight CNNs Model. IEEE Access 8, 55269–55278. https://doi.org/10.1109/ACCESS.2020.2982017

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., Lin, C.-J., 2008. LIBLINEAR: A Library for Large Linear Classification, Journal of Machine Learning Research.

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recognit Lett 27, 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Felzenszwalb, P.F., Huttenlocher, D.P., 2004. Efficient Graph-Based Image Segmentation. Int J Comput Vis 59, 167–181. https://doi.org/10.1023/B:VISI.0000022288.19776.77

Ferreira, A., Giraldi, G., 2017. Convolutional Neural Network approaches to granite tiles classification. Expert Syst Appl 84, 1–11. https://doi.org/10.1016/j.eswa.2017.04.053

Fisher, R.B., Breckon, T.P., Dawson-Howe, K., Fitzgibbon, A., Robertson, C., Trucco, E., Williams, C.K.I., 2014. Dictionary of Computer Vision and Image Processing, 2nd ed, Dictionary of Computer Vision and Image Processing. https://doi.org/10.1002/9781119286462

Francis, R., Mcisaac, K., Thompson, D.R., Osinski, G.R., 2014. Autonomous rock outcrop segmentation as a tool for science and exploration tasks in surface operations, in: 13th International Conference on Space Operations, SpaceOps

2014. American Institute of Aeronautics and Astronautics Inc. https://doi.org/10.2514/6.2014-1798

Fu, D., Su, C., Wang, W., Yuan, R., 2022. Deep learning based lithology classification of drill core images. PLoS One 17, e0270826. https://doi.org/10.1371/journal.pone.0270826

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Gor, V., Castano, R., Manduchi, R., Anderson, R.C., Mjolsness, E., 2001. Autonomous rock detection for mars terrain, in: AIAA Space 2001 Conference and Exposition. https://doi.org/10.2514/6.2001-4597

GOST R 53375-2016. Oil and gas wells. Geological-technological logging. General requirements., 2016.

Gressly, A.F.L., 1838. Observations geologiques sur le Jura Soleurois. Neue Denkschr Allg Schweiz Ges Gesamten Naturwiss 2, 1–112.

Guo, B., 2019. Petroleum reservoir properties, in: Well Productivity Handbook. Elsevier, pp. 17–51. https://doi.org/10.1016/b978-0-12-818264-2.00002-6

Halko, N., Martinsson, P.G., Tropp, J.A., 2011. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. SIAM Review 53, 217–288. https://doi.org/10.1137/090771806

Hall, J., Ponzi, M., Gonfalini, M., Maletti, G., 1996. Automatic Extraction And Characterisation Of Geological Features And Textures From Borehole Images And Core Photographs, in: SPWLA 37th Annual Logging Symposium. New Orlean, Louisiana, pp. 1–13.

Harinie, T., Janani Chellam, I., Sathya Bama, S.B., Raju, S., Abhaikumar, V., 2012. Classification of rock textures, in: Advances in Intelligent and Soft Computing. Springer Verlag, pp. 887–895. https://doi.org/10.1007/978-3-642-27443-5_102

Harraden, C.L., Cracknell, M.J., Lett, J., Berry, R.F., Carey, R., Harris, A.C., 2019. Automated core logging technology for geotechnical assessment: A study on core

from the Cadia East porphyry deposit. Economic Geology 114, 1495–1511. https://doi.org/10.5382/ECONGEO.4649

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep Residual Learning for Image Recognition. ArXiv e-prints.

Hunter, J.D., 2007. Matplotlib: A 2D Graphics Environment. Comput Sci Eng 9, 90–95. https://doi.org/10.1109/MCSE.2007.55

Intrator, N., 1990. A Neural Network for Feature Extraction, in: Advances in Neural Information Processing Systems. pp. 719–728.

Jadon, S., 2020. A survey of loss functions for semantic segmentation. 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB 2020. https://doi.org/10.1109/cibcb48159.2020.9277638

Kawaguchi, K., Huang, J., Kaelbling, L.P., 2019. Every Local Minimum Value Is the Global Minimum Value of Induced Model in Nonconvex Machine Learning. Neural Comput 31, 2293–2323. https://doi.org/10.1162/NECO_A_01234

Ketkar, N., 2017. Introduction to PyTorch, in: Deep Learning with Python. Apress, Berkeley, CA, pp. 195–208. https://doi.org/10.1007/978-1-4842-2766-4_12

Khasanov, I.I., 2015. An application of digital images computer analyses for core material investigation (in Russian). Geomodel-2015 359–362.

Khasanov, I.I., 2014. Rock color distribution analysis of core images (in Russian). OIL AND GAS GEOLOGY 33–39.

Khasanov, I.I., 2013. A method for quantitative estimation of core images color distribution and application (in Russian), in: Materials of "The School for Petrophysical Modeling of Sedimentary Rocks." Baltic PetroModel.

Khasanov, I.I., Ponomarev, I.A., Postnikov, A.V., Osintseva, N.A., 2016. A method for capacity characteristics for pay rocks quantitative estimation with the application of digital core images processing (in Russian). Geomodel-2016.

Khorram, F., Morshedy, A.H., Memarian, H., Tokhmechi, B., Zadeh, H.S., 2017. Lithological classification and chemical component estimation based on the visual features of crushed rock samples. Arabian Journal of Geosciences 10, 324. https://doi.org/10.1007/s12517-017-3116-8

Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Optimization.

Koeshidayatullah, A., Morsilli, M., Lehrmann, D.J., Al-Ramadan, K., Payne, J.L., 2020. Fully automated carbonate petrography using deep convolutional neural networks. Mar Pet Geol 122, 104687. https://doi.org/10.1016/j.marpetgeo.2020.104687

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks Alex. Advances in Neural Information Processing Systems 25 2012.

Laurenge, P., Bonduà, S., 2004. The Virtual Tile Store and other digital applications for stone tile production, in: Přikryl, R. (Ed.), Dimension Stone 2004. Balkema, Leiden, pp. 133–137.

Leeder, M.R., 1982. Sedimentology, Sedimentary Geology. Springer Netherlands, Dordrecht. https://doi.org/10.1007/978-94-009-5986-6

Li, M., Han, S., 2018. Automatic identification and classification in lithology based on deep learning in rock images.

Lidwell, William., Holden, Kritina., Butler, J., Elam, K., 2010. Universal principles of design : 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design 272.

Liu, X., Wang, H., Jing, H., Shao, A., Wang, L., 2020. Research on intelligent identification of rock types based on faster R-CNN method. IEEE Access 8, 21804–21812. https://doi.org/10.1109/ACCESS.2020.2968515

Liu, X.-Y., Wu, J., Zhou, Z.-H., n.d. Exploratory Undersampling for Class-Imbalance Learning.

Lloyd, S.P., 1982. Least Squares Quantization in PCM, IEEE TRANSACTIONS ON INFORMATION THEORY.

Lowe, D.G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. Int J Comput Vis 60, 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

Ma, Y. de, Liu, Q., Qian, Z.B., 2004. Automated image segmentation using improved PCNN model based on cross-entropy. 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, ISIMP 2004 743–746. https://doi.org/10.1109/ISIMP.2004.1434171

Macqueen, J., 1967. Some methods for classification and analysis of multivariate observations. IN 5-TH BERKELEY SYMPOSIUM ON MATHEMATICAL STATISTICS AND PROBABILITY 281--297.

Malvadkar, S.B., Forbes, S., McGurl, G. V., 2004. Coal Resources, Formation of, in: Encyclopedia of Energy. Elsevier, pp. 529–550. https://doi.org/10.1016/b0-12-176480-x/00281-3

Marjoribanks, R., 2010. Geological Methods in Mineral Exploration and Mining. Springer Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-74375-0

Martín Abadi, Ashish Agarwal, Paul Barham, E.B., Zhifeng Chen, Craig Citro, Greg S. Corrado, A.D., Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, I.G., Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Y.J., Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, M.S., Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, J.S., Benoit Steiner, Ilya Sutskever, Kunal Talwar, P.T., Vincent Vanhoucke, Vijay Vasudevan, F.V., Oriol Vinyals, Pete Warden, Martin Wattenberg, M.W., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

McGuffee, J.W., 2000. Defining computer science. ACM SIGCSE Bulletin 32, 74–76. https://doi.org/10.1145/355354.355379

Metz, C.E., 1978. Basic Principles of ROC analysis. Semin Nucl Med 8, 283–298. https://doi.org/10.1016/S0001-2998(78)80014-2

Młynarczuk, M., Górszczyk, A., Ślipek, B., 2013. The application of pattern recognition in the automatic classification of microscopic rock images. Comput Geosci 60, 126–133. https://doi.org/10.1016/j.cageo.2013.07.015

Moradi, M., Tokhmechi, B., Masoudi, P., 2019. Inversion of well logs into rock types, lithofacies and environmental facies, using pattern recognition, a case study of carbonate Sarvak Formation. Carbonates Evaporites 34, 335–347. https://doi.org/10.1007/s13146-017-0388-8

Müller, A.C., Guido, S., 2016. Introduction to Machine Learning with Python: a guide for data scientists, Journal of Chemical Information and Modeling.

Müller, R., Kornblith, S., Google, G.H., Toronto, B., n.d. When Does Label Smoothing Help?

Murphy, K.P., 2012. Machine Learning. A Probabilistic Perspective, Chance Encounters: Probability in Education. Massachusetts Institute of Technology, Massachusetts.

Murtagh, F., Legendre, P., 2014. Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? J Classif 31, 274–295. https://doi.org/10.1007/s00357-014-9161-z

Nesterov, Y., 1983. A method for unconstrained convex minimization problem with the rate of convergence o(1/k^2). Doklady ANSSSR (translated as Soviet.Math.Docl.) 269.

Oja, E., 1982. A Simplified Neuron Model as a Principal Component Analyzer. Journal o[Mathematical Biology 15, 267–273.

Oxford Advanced Learner's Dictionary, Tenth Edit. ed, 2020. . Oxford University Press, Oxford.

Paclík, P., Verzakov, S., Duin, R.P.W., 2005. Improving the maximum-likelihood co-occurrence classifier: A study on classification of inhomogeneous rock images, in:

Lecture Notes in Computer Science. Springer Verlag, pp. 998–1008. https://doi.org/10.1007/11499145_101

Patel, A.K., Chatterjee, S., Gorai, A.K., 2017a. Development of machine vision-based ore classification model using support vector machine (SVM) algorithm. Arabian Journal of Geosciences 10. https://doi.org/10.1007/s12517-017-2909-0

Patel, A.K., Chatterjee, S., Gorai, A.K., 2017b. Development of online machine vision system using support vector regression (SVR) algorithm for grade prediction of iron ores, in: 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA). IEEE, pp. 149–152. https://doi.org/10.23919/MVA.2017.7986823

Patel, A.K., Gorai, A.K., Chatterjee, S., 2016. Development of Machine Vision-based System for Iron Ore Grade Prediction using Gaussian Process Regression (GPR). Pattern Recognition and Information Processing (Prip'2016) 45–48.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Petrov, O.V. (Ed.), 2011. Dictionary of Geology. Volume Two. K to P (in Russian). VSEGEI, Saint Petersburg.

Pickup, G.E., Hern, C.Y., 2002. The Development of Appropriate Upscaling Procedures. Transport in Porous Media 2002 46:2 46, 119–138. https://doi.org/10.1023/A:1015055515059

Postnikova, O.V., Postnikov, A.V., Khasanov, I.I., 2017. New approaches for modeling in oil and gas lithology (in Russian). New Ideas in geology of oil and gas 283–287.

Prince, C.M., Chitale, J., 2008. Core Image Analysis : Reliable Pay Estimation in Thin-Bedded Reservoir Units. Image Processing 1–6.

122

Rahman, M.A., Wang, Y., 2016. Optimizing intersection-over-union in deep neural networks for image segmentation, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Verlag, pp. 234–244. https://doi.org/10.1007/978-3-319-50835-1_22

Ran, X., Xue, L., Zhang, Y., Liu, Z., Sang, X., He, J., 2019. Rock classification from field image patches analyzed using a deep convolutional neural network. Mathematics 7. https://doi.org/10.3390/math7080755

Ribeiro, M.T., Singh, S., Guestrin, C., 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. https://doi.org/10.1145/2939672.2939778

Robbins, H., Monro, S., 1951. A Stochastic Approximation Method. The Annals of Mathematical Statistics 22, 400–407. https://doi.org/10.1214/aoms/1177729586

Rosebrock, A., 2017. Deep Learning for Computer Vision with Python. PyImageSearch.

Ruder, S., 2016. An overview of gradient descent optimization algorithms. ArXiv e-prints.

Rudin, C., 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat Mach Intell 1, 206–215. https://doi.org/10.1038/S42256-019-0048-X

Sasaki, Y., Fellow, R., 2007. The truth of the F-measure.

Seleznev, I.A., Abashkin, V., Chertova, A., Istomin, S.B., Samokhvalov, A.F., 2019. Quantitative Analysis of Whole Core Images, in: Geomodel 2019. pp. 9–13. https://doi.org/10.3997/2214-4609.201950103

Seleznev, I.A., Abashkin, V.V., Chertova, A.A., Makienko, D.O., Istomin, S.B., Romanov, D.V., Samokhvalov, A.F., 2020. Joint Usage of Whole Core Images Obtained in Different Frequency Ranges for the Tasks of Automatic Lithotype Description and Modeling of Rocks' Petrophysics Properties 2020, 1–5. https://doi.org/10.3997/2214-4609.202050090

Shang, C., Barnes, D., 2012. Support vector machine-based classification of rock texture images aided by efficient feature selection, in: Proceedings of the International Joint Conference on Neural Networks. https://doi.org/10.1109/IJCNN.2012.6252634

Sharif, H., Ralchenko, M., Samson, C., Ellery, A., 2015. Autonomous rock classification using Bayesian image analysis for Rover-based planetary exploration. Comput Geosci 83, 153–167. https://doi.org/10.1016/j.cageo.2015.05.011

Shu, L., McIsaac, K., Osinski, G.R., Francis, R., 2017. Unsupervised feature learning for autonomous rock image classification. Comput Geosci 106, 10–17. https://doi.org/10.1016/j.cageo.2017.05.010

Simonyan, K., Zisserman, A., 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition, in: International Conference on Learning Representations.

Singh, M., Javadi, A., Singh, S., 2004. A comparison of texture teatures for the classification of rock images. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/978-3-540-28651-6_26

Sirotin, V.I., Shatrov, V.A., Buneev, V.N., Voytsekhovsky, G.V., 2004a. The doctrine of Facies (in Russian). VGU, Voronezh.

Sirotin, V.I., Shatrov, V.A., Buneev, V.N., Voytsekhovsky, G.V., 2004b. The doctrine of Facies (in Russian). VGU, Voronezh.

Smith, L.N., 2018. A disciplined approach to neural network hyper-parameters: Part 1 -- learning rate, batch size, momentum, and weight decay.

Sun, S., Cao, Z., Zhu, H., Zhao, J., 2020. A Survey of Optimization Methods from a Machine Learning Perspective. IEEE Trans Cybern 50, 3668–3681. https://doi.org/10.1109/TCYB.2019.2950779

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going Deeper with Convolutions, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

The GIMP Development Team, n.d. GIMP.

Thomas, A., Rider, M., Curtis, A., MacArthur, A., 2011. Automated lithology extraction from core photographs. First Break 29, 103–109.

Timmer, E., Knudson, C., Gingras, M., 2021. Applying deep learning for identifying bioturbation from core photographs. Am Assoc Pet Geol Bull 105, 631–638. https://doi.org/10.1306/08192019051

Tipping, M.E., Bishop, C.M., 1999. Mixtures of probabilistic principal component analysers, Neural Computation. MIT Press.

Travis, O.E., 2006. A guide to NumPy. Trelgol Publishing, USA.

Tremeau, A., Colantoni, P., 2000. Regions adjacency graph applied to color image segmentation. IEEE Transactions on Image Processing 9, 735–744. https://doi.org/10.1109/83.841950

Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M., 2013. Selective Search for Object Recognition. Int J Comput Vis 104, 154–171. https://doi.org/10.1007/s11263-013-0620-5

Valentín, M., Bom, C., Albuquerque, Marcio, Albuquerque, Marcelo, Faria, E., Correia, M., Surmas, R., 2017. On a method for Rock Classification using Textural Features and Genetic Optimization. Notas Técnicas 7, 18–30. https://doi.org/10.7437/NT2236-7640/2017.01.003

Van Rossum, Guido., Drake, F.L., 2011. The Python language reference manual : for Python version 3.2. Network Theory Ltd.

Vedaldi, A., Soatto, S., 2008. Quick Shift and Kernel Methods for Mode Seeking, in: Computer Vision – ECCV 2008. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 705–718. https://doi.org/10.1007/978-3-540-88693-8_52

Viola, P.;, Jones, M.;, Snow, D., 2003. Detecting Pedestrians Using Patterns of Motion and Appearance.

Wang, Y. da, Shabaninejad, M., Armstrong, R.T., Mostaghimi, P., 2021. Deep neural networks for improving physical accuracy of 2D and 3D multi-mineral segmentation of rock micro-CT images. Appl Soft Comput 104, 107185. https://doi.org/10.1016/J.ASOC.2021.107185

Wang, Q., Ma, Y., Zhao, K., Tian, Y., 2020. A Comprehensive Survey of Loss Functions in Machine Learning. Annals of Data Science 2020 1–26. https://doi.org/10.1007/S40745-020-00253-5

Wei, X.S., Qin, X.H., Rong, C.L., Nan, J.X., Cheng, G.J., 2014. Image Classification Recognition for Rock Micro-Thin Section Based on Probabilistic Neural Networks. Applied Mechanics and Materials 602–605, 2147–2152. https://doi.org/10.4028/www.scientific.net/AMM.602-605.2147

Wiatowski, T., Bolcskei, H., 2018. A Mathematical Theory of Deep Convolutional Neural Networks for Feature Extraction. IEEE Trans Inf Theory 64, 1845–1866. https://doi.org/10.1109/TIT.2017.2776228

Wieling, I.S., 2013. Facies and permeability prediction based on analysis of core images.

Wold, S., Esbensen, K., Geladi, P., 1987. Principal component analysis. Chemometrics and Intelligent Laboratory Systems 2, 37–52. https://doi.org/10.1016/0169-7439(87)80084-9

Wu, C.H., Lee, J.S., Shie, C.Y., Su, M.Y., 2012. Rock classification via a mobile device, in: 2012 7th International ICST Conference on Communications and Networking in China, CHINACOM 2012 - Proceedings. pp. 663–666. https://doi.org/10.1109/ChinaCom.2012.6417566

Xu, Z., Shi, H., Lin, P., Liu, T., 2021. Integrated lithology identification based on images and elemental data from rocks. J Pet Sci Eng 205, 108853. https://doi.org/10.1016/j.petrol.2021.108853

Yakubovskiy, P., 2019. Segmentation Models. GitHub repository.

Yapaskurt, O.V., 2016. Lithology (in Russian), 2nd ed. Infra-M, Moscow.

Zhang, Y., Li, M., Han, S., 2018a. Automatic identification and classification in lithology based on deep learning in rock images. Acta Petrologica Sinica 34.

Zhang, Y., Wang, G., Li, M., Han, S., 2018b. Automated Classification Analysis of Geological Structures Based on Images Data and Deep Learning Model. Applied Sciences 2018, Vol. 8, Page 2493 8, 2493. https://doi.org/10.3390/APP8122493