



Review

Review of deep learning approaches in solving rock fragmentation problems

Mikhail V. Ronkin¹, Elena N. Akimova^{1,2,*} and Vladimir E. Misilov^{1,2}

¹ Department of Information Technologies and Control Systems, Institute of Radioelectronics and Information Technology, Ural Federal University, Mira Street 19, 620002 Ekaterinburg, Russia

² Krasovskii Institute of Mathematics and Mechanics, Ural Branch of RAS, S. Kovalevskaya Street 16, 620108 Ekaterinburg, Russia

* **Correspondence:** Email: aen15@yandex.ru.

Abstract: One of the most significant challenges of the mining industry is resource yield estimation from visual data. An example would be identification of the rock chunk distribution parameters in an open pit. Solution of this task allows one to estimate blasting quality and other parameters of open-pit mining. This task is of the utmost importance, as it is critical to achieving optimal operational efficiency, reducing costs and maximizing profits in the mining industry. The mentioned task is known as rock fragmentation estimation and is typically tackled using computer vision techniques like instance segmentation or semantic segmentation. These problems are often solved using deep learning convolutional neural networks. One of the key requirements for an industrial application is often the need for real-time operation. Fast computation and accurate results are required for practical tasks. Thus, the efficient utilization of computing power to process high-resolution images and large datasets is essential. Our survey is focused on the recent advancements in rock fragmentation, blast quality estimation, particle size distribution estimation and other related tasks. We consider most of the recent results in this field applied to open-pit, conveyor belts and other types of work conditions. Most of the reviewed papers cover the period of 2018-2023. However, the most significant of the older publications are also considered. A review of publications reveals their specificity, promising trends and best practices in this field. To place the rock fragmentation problems in a broader context and propose future research topics, we also discuss state-of-the-art achievements in real-time computer vision and parallel implementations of neural networks.

Keywords: computer vision; deep learning; convolutional neural networks; rock fragmentation; blast quality estimation; real-time performance; parallel computing

Mathematics Subject Classification: 68T07, 68W10

1. Introduction

The mining industry has undergone a rapid transformation in recent years, with the digitization of its operations. This transformation has often been referred to as the fourth industrial revolution or Industry 4.0. One of the most notable developments in this area has been the increasing use of deep learning-based approaches for automating a range of mining applications, including drilling, blasting, haulage and factory processing. Despite the widespread adoption of deep learning in the mining industry, its deployment has been relatively infrequent compared to other industries. Today in the mining industry and other fields, many jobs are carried out manually by highly qualified specialists, although some can be performed automatically. This is especially relevant when solving routine tasks, which require extensive specialist experience. Yet specialists often cannot explain (or formally describe) how they solve problems. Nevertheless, experts sometimes make mistakes (so-called human factors), and they require a lot of time and resources to become proficient. However, nowadays modern neural networks can automate such tasks, as shown by the general practice of deep learning (DL) implementation. The problem may be reduced to a few-times dataset collection for which expert markup is needed. The automated DL solution will reduce specialists' workload by handling routine tasks for them or providing guidance. This is done, for example, by suggesting a solution to a problem or giving information required for finding a solution.

In recent years, there has been an increasing trend in utilizing DL Computer Vision (CV) techniques for solving various applied problems in the mining industry. These problems include, but are not limited to, the classification of mineral types and their fraction relation [1–3], detection and analysis of the size distribution of mineral grains or particles [4, 5], analysis of boring and drilling results [6, 7], autonomous haulage vehicles driving [8], exploration and analysis of open-pit mines [9–12] and recognition of terrain types [13].

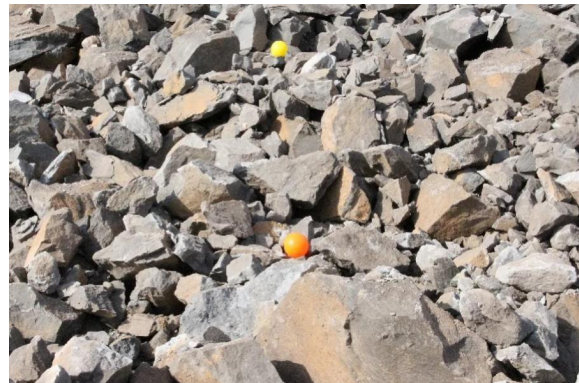
The mining industry is faced with numerous challenges, including the analysis and evaluation of the performance and productivity of valuable rock resources mining, particularly in open-pit mines. This task is of the utmost importance, as it is critical to achieving optimal operational efficiency, reducing costs and maximizing profits. To achieve this goal, it is necessary to accurately estimate rock fragmentation, blast quality and other related factors [12, 14–18]. Rock fragmentation estimation is an essential aspect of mining operations, as it affects a range of downstream processes, such as crushing and grinding. Accurate estimation of rock fragmentation can help optimize these processes, resulting in improved efficiency and productivity. Furthermore, it can also reduce the environmental impact of mining operations, as it can help minimize the amount of waste material generated. Similarly, blast quality estimation is also crucial for optimizing mining operations. This involves assessing the effectiveness of the blasting process, which is essential for achieving the desired rock fragmentation. Accurate blast quality estimation can help reduce costs by minimizing the amount of explosives required, while also improving safety by ensuring that the blasting process is carried out correctly.

In the context of mining operations, an open-pit mine can be described as the collection and processing of a large number of rock fragments resulting from the blasting of multiple smaller work areas. These rock fragments are then transported to a processing plant for further treatment, and the overall production yield for the entire open-pit mine is determined by the productivity estimates of all individual work areas. As a result, accurate assessment and control of productivity at each open-pit workplace are essential for effective management of the mine and processing plant. The examples of

such open-pit workplaces are illustrated in Figures 1 for several cases [12, 18–20]



(a) Bazhenovskoye asbestos field, Asbest City, Russia [18]



(b) Sungun Copper field, Iran [19]



(c) Tensa iron ore field, Sundargarh, Orissa, India [20]



(d) Nui Phao tungsten field, Vietnam [12]

Figure 1. Illustrations of Open Pit workplaces with rock chunks.

In recent years, there has been increasing interest in CV techniques for analyzing rock fragmentation and related tasks in the mining industry. These studies emphasize the need for fast and accurate DL methods in CV to tackle these challenges. This need is particularly crucial due to the use of low-performance mobile devices and the requirement for real-time measurements in mining operations.

DL-based approaches offer promising solutions by enabling efficient and accurate analysis of large amounts of visual data obtained from diverse mining operations. Real-time data processing is essential for making timely decisions and optimizing operations for maximum productivity and efficiency.

Furthermore, the development of DL-based techniques for mining applications is important because the data obtained in such operations is often complex and highly variable. Visual data from different rock fragments and work areas in an open pit, for example, can vary significantly in terms of size, shape and color. To effectively handle such variability and produce accurate results, DL methods must be trained on large and diverse datasets.

Unlike conventional DL CV problems, rock fragmentation tasks involve processing images of numerous relatively small and overlapping objects. The specifics and quality of the images can

vary depending on the work conditions (*e.g.*, open-pit mines, tunnels, conveyor belts) and weather conditions, such as lighting and air clarity. These factors can adversely affect CV algorithms, resulting in lower-quality results. Therefore, fine-tuning and modifying DL architectures and applying sophisticated pre- and post-processing techniques are often necessary. In addition, the CV systems for these tasks are often intended to work in real time, processing dozens of images per second.

The objective of our systematic literature review is to gather knowledge about recent advancements in DL-based CV systems for rock fragmentation and related tasks in the mining industry. We aim to identify the main methodologies and techniques used to solve rock fragmentation problems and correlate them with the current state-of-the-art in real-time CV and performance of neural networks. By doing so, we intend to place this topic in the context of DL tasks and neural network implementations. While we cannot suggest specific recipes for all cases, we provide recommendations for researchers in the fields of mining industry on promising fast approaches and architectures for real-time instance segmentation and performance optimization of training and inference on various devices.

The main contribution of our study is to draw attention to current methods for solving fragmentation problems, emphasize that problems can be solved in real time and demonstrate current trends in dealing with fragmentation problems and CV-related problems in general. The current state of solving fragmentation problems is several years behind the general CV, as we demonstrate in the review.

The paper is organized as follows. Section 2 describes the methodology of our survey, statistical results of the considered publications and their summary. Section 3 presents a brief review of the main publications. All papers were divided into DL tasks, namely semantic segmentation, instance segmentation, object detection and a classification/regression approach. Section 4 discusses the main findings from the reviewed papers and compares them with current state-of-the-art for DL CV algorithm architectures and optimization techniques for implementing neural networks. Section 5 concludes our survey and formulates our position on promising techniques and approaches to the discussed topics.

2. Methodology and preliminary results

To collect related papers, we formulated search strings by combining a pair of keywords from two groups:

- “rock fragmentation”, “blast quality estimation”, “open-pit mining”, “mining productivity estimation”, “mining industry problem”;
- “computer vision”, “deep learning”, “computer vision neural network architectures”, “feature extractors for computer vision”, “semantic segmentation”, “instance segmentation”, “real-time instance segmentation”.

Then, we used the resulting search strings in the Google Scholar and Scopus databases. We utilized the following criteria to filter out articles with insufficient content and assess paper quality. The paper must explain how CV is specifically used in the mining industry for tasks that include rock fragmentation or such relative tasks as: particle size distribution, blast quality estimation, rock size estimation or their detection. In addition to the DL-based methods, we also consider the traditional CV-based systems. Also during the review we pay attention to the topic of computation specificity and how parallel computing architectures with GPUs are used to implement neural network training

and inference. We also discard the papers not directly devoted to the image analysis, for example, the works on mathematical models of size distribution.

Table Appendix summarizes the publications that meet the criteria above. Let us note that most of the reviewed papers cover the period of 2018–2023. However, the most significant of the older publications are also considered. The studies in the table are sorted by publication year and grouped by practical field of the rock fragmentation problem.

The main features in Table Appendix, which were extracted during the review, were the following. In the review, we found that most of the papers utilized CV approaches in the traditional form of photo (or, in rare cases, in 3D by laser scanning). Thus, we did not consider other methods of solving the fragmentation problem.

- Year of publication.
- Practical Problem. This problem is the same as specificity of work conditions, we divide them into conveyor belt, open-pit, tunnel and other conditions. The term “other” types includes rare paper conditions or large scale images, which could be considered out of our topic.
- CV Task. Approach to CV problem solving. This is a taxonomy of the types of approaches common to DL in CV applications. The main types of problems are: classification/regression; object detection; semantic segmentation (or pixel/image segmentation) and instance segmentation. Semantic segmentation includes both segmentation and boundary segmentation, or their combination.
- DL application. All papers are categorized based on whether they use only classical CV approaches or approach with DL neural networks. A classical approach refers to using hand-crafted features and a variety of different decision making algorithms (such as the Watershed algorithm), in contrast to a DL approach that pulls features and makes decisions automatically using neural networks. Let us note here, that almost all papers that apply DL use convolutional neural networks (CNN). Thus, we did not divide DL and CNN techniques.
- Specific models or methods/algorithms applied in the paper.
- Primary CV Model. In this section, we highlight only the main method/algorithm considered by publication.

The review results in Table Appendix are shown in Figures 2–6. Figure 2 shows the visualization of the DL vs. classical CV approach for problem solution in the publications. Figure 3 shows the popularity of the fragmentation problem (practical problem) in publications over time. Figure 4 shows CV task vs. the practical problem type distribution. Figure 5 shows the primary CV model distribution by year starting from 2004. Figure 6 shows the visualization of CV task relative to the primary model distribution.

Figures 2–6 show that, starting from 2018, DL approach is applied in rock fragmentation and similar tasks. Moreover, since 2020, DL CNN is favored in rock fragmentation and similar tasks (see Figure 2). Also, the popularity of investigation of the fragmentation task in open pits is growing from 2018, probably with the general DL popularity growing (see Figure 3).

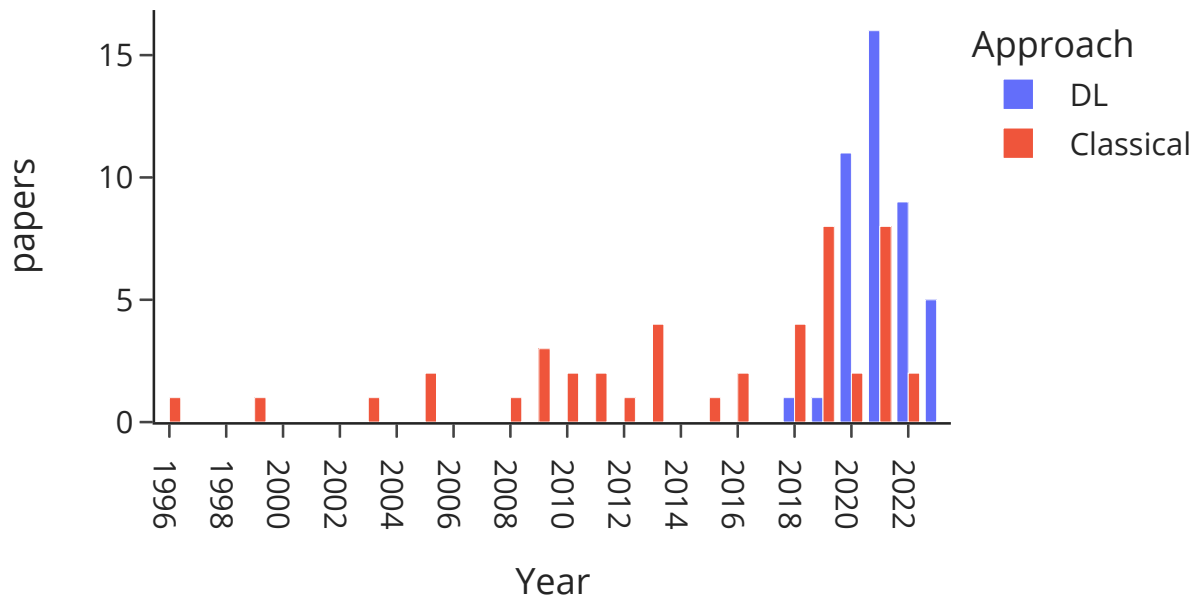


Figure 2. DL vs Classical CV approach application by year histogram.

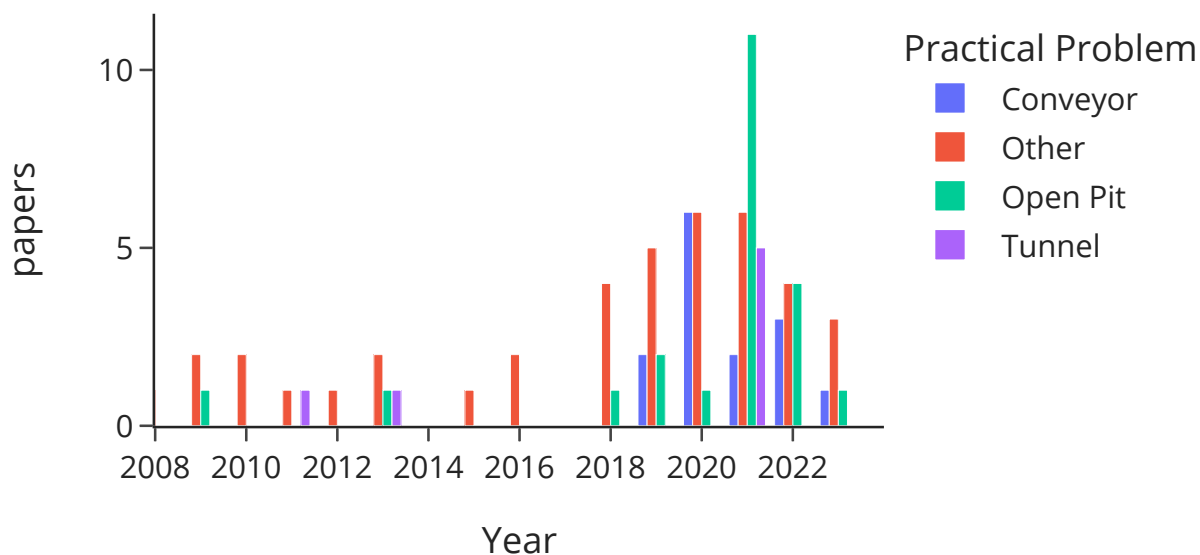


Figure 3. Practical Problem investigation popularity from 2008 year.

The instance segmentation approach is the most popular approach in open-pit conditions. Among all instance segmentation DL models, Mask R-CNN remains the most popular solution to date. However, since 2022 real-time object detection approaches (including YOLO-based models) have been considered as alternatives to the Mask R-CNN solutions (see Figures 4, 5, 6).

The peak of the investigation of the fragmentation task in the conveyor belt condition was in 2020. In these conditions, the semantic segmentation approach is the most popular one. Among all semantic segmentation DL models U-Net is the most popular solution. Additionally, DeepLab3+-based architectures have been utilized as some alternatives to U-Net-based models from 2022 (see Figures 4, 5, 6).

Among all classical models, the Watershed algorithm is the most popular solution (corresponds to instance segmentation). The peak of the Watershed popularity was from 2005 to 2019. This algorithm is mostly applied to conveyor belt problems, but also was tested for open-pit conditions (see Figure 6).

Let us note that by a classification/regression problem we address the estimation of coarse fragmentation sizes. This is done, for instance, by dividing the images into classes by the size of present fragments. Also, under semantic segmentation we assume either full object segmentation or (and) their boundaries segmentation. Frequently, such results can be divided into objects by any post-processing technique to extract instance parameters. As opposed to this, in instance segmentation problems we assume end-to-end routines to recognize instances. By estimating only the boundary box and central position of an instance, object detection is a part of instance segmentation. Figure 7 shows typical illustrations of the types of problems under consideration.

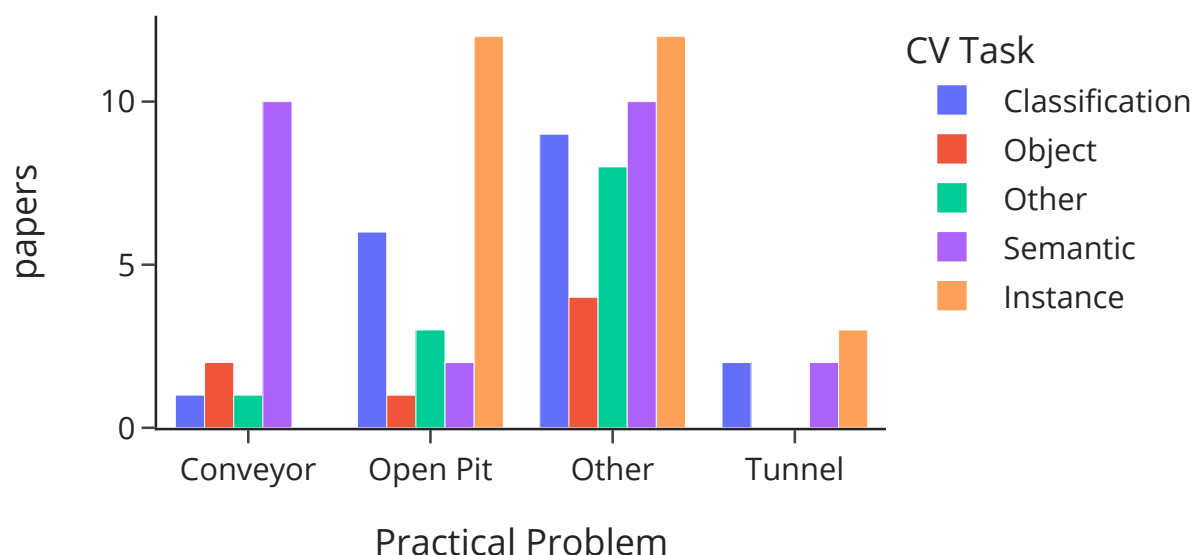


Figure 4. CV Task types to Practical Problem type histogram.

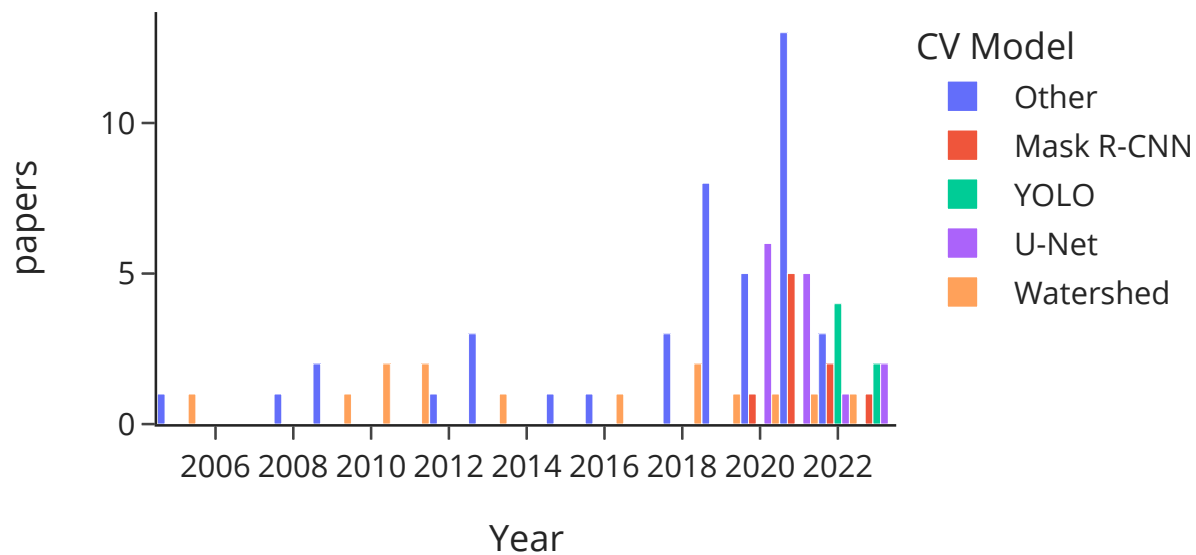


Figure 5. Primary CV Models/methods vs publication year histogram.

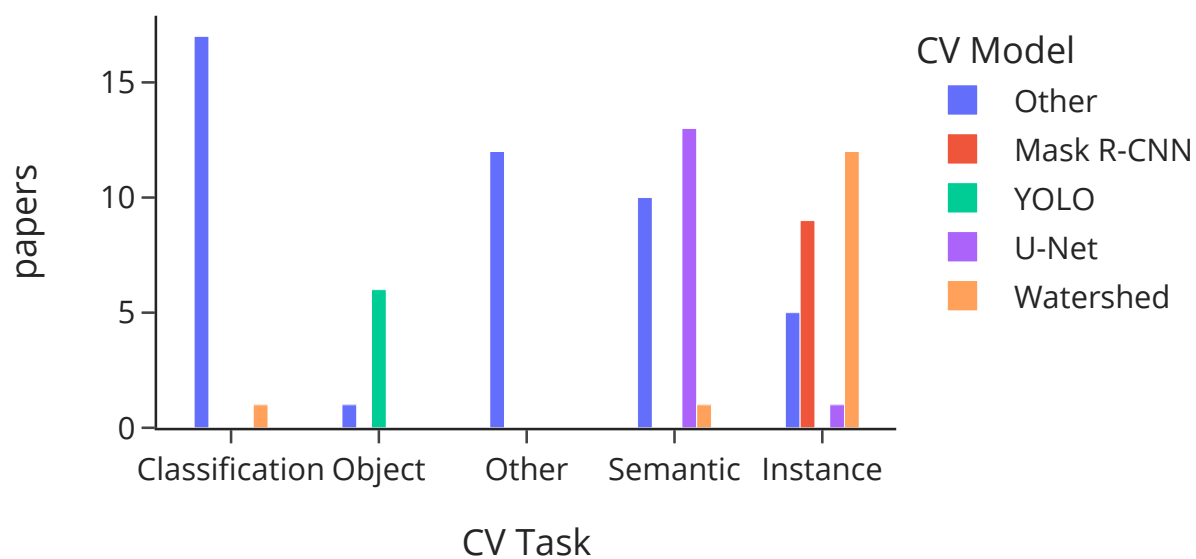


Figure 6. CV Models vs CV Task histogram.

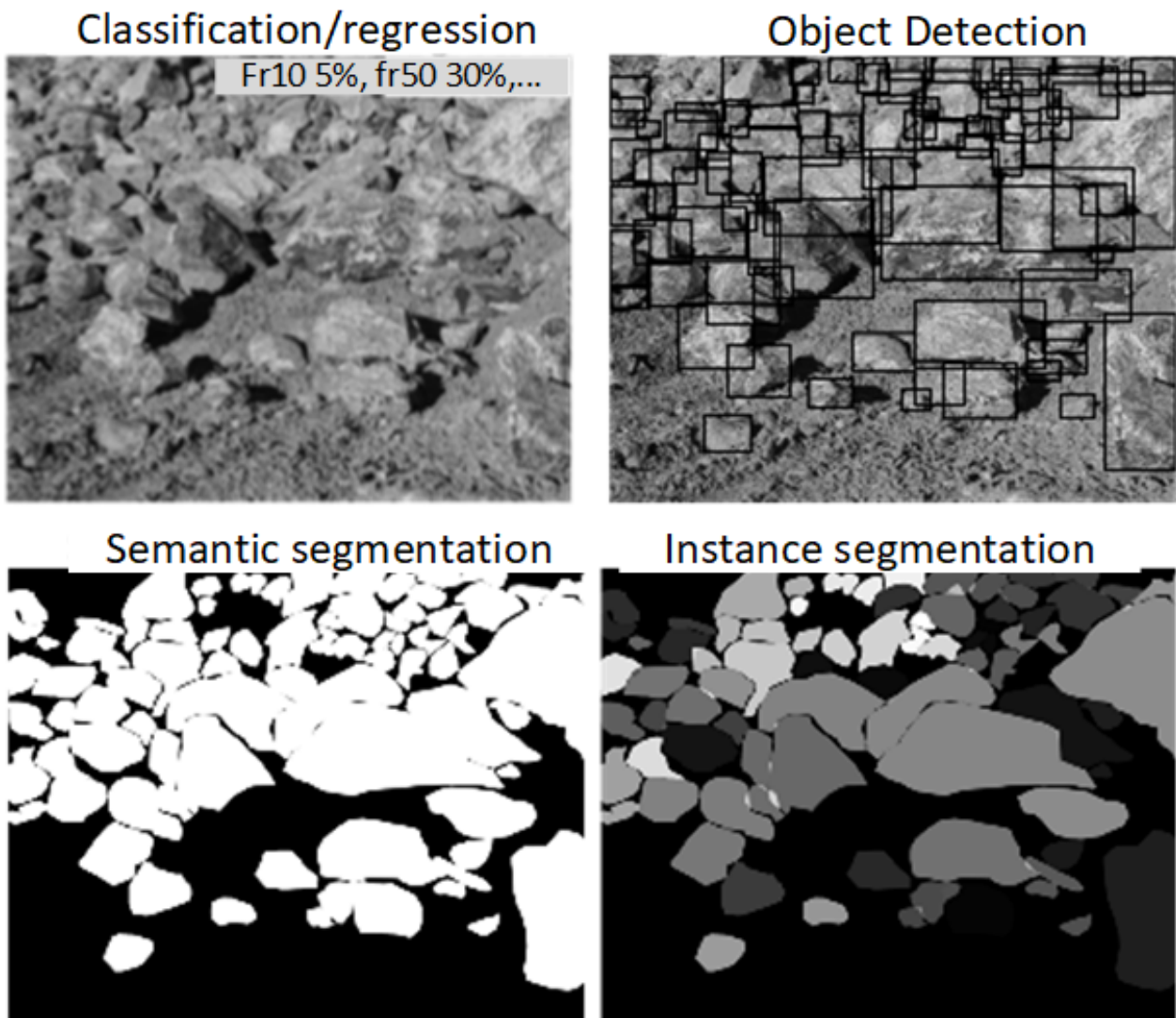


Figure 7. Typical illustration of the considered problems types visualization.

The next section will focus on the main papers on the considered topic from Table Appendix.

3. The main publication review

3.1. General position of applied DL architectures

A CV-based approach to rock fragmentation estimation is related to what is referred to as “indirect” methods [21]. Analysis of computer vision in the context of the considered and related tasks has shown that this approach is known as one of the most accurate in the literature. Most of the works here can be divided into two classes. The first one is based on applying classical CV techniques. The second one is built using contemporary DL techniques. The main difference lies in the way features are extracted and processed. In classical methods, features are extracted manually based on a synthetic assumption and using a variety of techniques. Afterwards, another method for making decisions (fragmentation in our

case) is applied to the results that were obtained. In the reviewed literature, the Watershed algorithm is the most popular among them. Alternatively, DL assumes an end-to-end approach, with automatic feature extraction as well as the decision making part within one neural network architecture. The classical-based approach suffers from variations in daytime, weather and season conditions [22–24].

Moreover, it is shown that the classical approach works well only for high resolution rock chunks in images and does not provide sufficiently accurate results for images of large open-pit places. However, such algorithms could be less resource-consuming than the DL-based approaches currently applied in this field [12, 21]. Most classical CV based solutions already exist in commercial software, like the Gold Size, Wipfrag and other [12, 21, 25–27]. In the preceding software, the fragment distributions are frequently estimated as values of the circle area with comparable sizes as rock chunks are expected to have [27]. This equivalent size value could be an incorrect measure and does not directly represent the maximum dimension size of a rock fragment [12, 21]. The current achievements of DL for instance segmentation tasks show the ability to achieve comparative performance with higher accuracy and robustness of the results, while allowing to overcome most of the classical CV approach drawbacks [22–24].

The basic idea of using convolutional architecture was proposed in 1989 for the recognition of hand-written ZIP Code numbers [28, 29]. The main application of this approach is processing data with known regular structure, such as images (two-dimensional) or time series (one-dimensional). Besides image processing, CNN models are also widely used in various applications in civil engineering, such as fault prediction [30, 31] and structural health monitoring [32–35].

The modern way of using CNN in DL CV was launched by the LeNet architecture in 1998 [36]. Automated feature extraction is the main benefit of using DL in CV. It is well known that most CV tasks are difficult to describe formally. However, it is well known that a well-suited set of features is necessary for a large generalization ability. Unlike shallow neural networks and other machine learning approaches, DL networks include two main parts: the feature extractor and the decision-making (so-called head) part. A CNN feature extractor consists of convolution and auxiliary layers. In the case of CV convolution operation, it has several inductive biases: providing scale, location, and rotation invariant feature extraction with a fewer number of parameters in comparison to other approaches [37, 38]. Currently, these advantages make CNN the main way of solving CV problems.

Generally, Table Appendix shows that the rock fragmentation estimation task as a CV problem could be solved as a contour segmentation task utilizing a semantic segmentation approach (see, for instance, [15, 16, 39]) or as an instance segmentation task (see, for instance [12, 20, 40]). However, some authors also propose object detection-based solutions (see, for instance [41–43]) or classification/regression-based methods of strict fractions forecasting. The following sections are organized in accordance with this division.

The specifics of CV could be explained as follows. Typical CV problems require a lot of extracted features which cannot be formally described as usual. DL allows one to automatically extract complex data representations even from unprocessed datasets, providing meaningful information. On the other hand, this area requires rather large databases with all the ensuing difficulties of a typical big data application. Big data presents many challenges, but the most critical are its computational performance and data storage space requirements. Another particularly difficult aspect is the high diversity of valuable feature distributions that we often have during the training of a decision algorithm. Nearly any modern neural network architecture can be subject to these challenges [44].

In addition to the mentioned above DL-based methods, let us note some classical-based solutions to the fragmentation problem. The principle of the watershed algorithm [45] is to represent an image as a 3D structure. In the obtained structure the points of the so-called basins cross-sections are considered as instance boundaries. In the basic form, the Watershed algorithm suffers from noise influence and blurring of boards in the low-contrast cases. Both of these problems can lead to oversegmentation or undersegmentation issues [46]. Thus, an additional pre-processing step is usually required. This image processing can be done either by DL semantic segmentation or as a classical pipeline. The first case leads to the considered DL application. The classical approach seems to be challenging for segmentation in the case of multi-scale and multi-shape objects. It suffers from unstable outer conditions such as light, daytime, *etc.* which affect the texture, colors and other image parameters [20]. Despite the above problems, paper [47] suggests the use of the modified Watershed alone algorithm for open-pit segmentation.

3.2. Semantic segmentation

The semantic segmentation problem assumes the solution of the so-called pixelwise classification (in the sense of DL). The output of the neural network should have the same spatial dimensions as the input, and the number of channels should equal the number of classes. Then, each pixel for all channels is soft-reweighted (with softmax, as a vector) to determine the class it belongs to.

Among all DL semantic segmentation architectures, the most popular choice is the U-Net [48]. Even though it was introduced in 2015, this architecture remains one of the most popular today. Figure 8 illustrates the U-Net architecture and the pixelwise softmax. The U-Net architecture is related to encoder-decoder segmentation networks. Its main characteristic is that it uses feature maps from the encoder in the decoder. Since 2015 many architectural modifications have been proposed. Most of them were proposed for the medical applications [49–53] and some of them for the microscopic images analysis applications [54].

Paper [15] considers the ore size determination problem as the contour segmentation problem (as a part of the semantic segmentation task). The authors propose slightly modified Res-U-Net architectures for ore size determination in both conveyor belt and open pit conditions, with accuracy of about 90% in both cases. In addition, the authors have tested the classical Watershed algorithm, which shows comparable accuracy for high-scale images on the conveyor belt, but much less accurate for the open pit. For open-pit conditions, authors combined the U-Net segmentation with the Watershed algorithm to group results into objects.

A U-Net-based model with an Efficient-net-B3 backbone for conveyor belt conditions was proposed in paper [18]. The system allows one to use the same approach for rock chunks detection as well as for valuable resources inside it. The images are taken in the near-infrared range of light. The model achieved 0.97 of Dice coefficient for stone segmentation and 0.6 for valuable veins material segmentation.

An instance segmentation model is proposed for recognition of rock fragments during tunnel boring in work [55]. It is based on two subnetworks: object detection and semantic segmentation. The object detection network is based on a modified Single-Shot Detector (SSD) with multilevel feature fusion, prior anchors and self-attention. The semantic segmentation is using the U-Net architecture. In the downsampling stage, the data from the object detection network was used to share features of rock fragments. In the upsampling stage, the skip connection and self-attention were utilized. For the test

data, 0.87 average recall and 0.76 average IoU were achieved. On a workstation with the Intel i9-9900k and the NVIDIA GeForce RTX 2080 Ti GPU, the inference for a full size image of 4096×3072 resolution takes 7.5 seconds, while the 512×512 resolution requires 0.15 seconds. Post-processing and statistical analysis takes 0.18 seconds. Therefore, the proposed method can be used in real time.

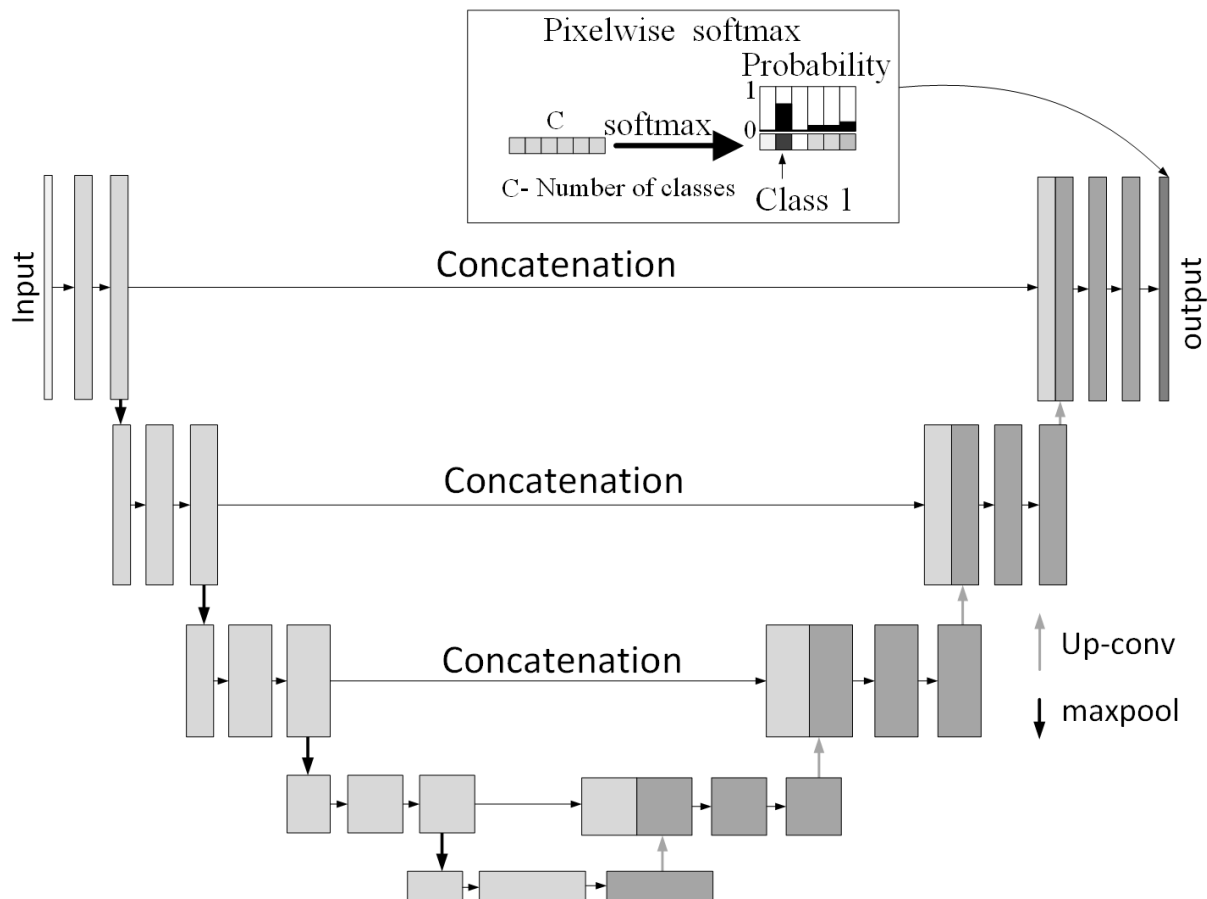


Figure 8. Illustration of the U-Net architecture and pixel-wise soft max.

Work [56] is aimed at analyzing the rock segmentation distribution on images of rocks produced by a tunnel boring machine, namely, collect statistical data on size distribution. The core algorithm is based on the semantic segmentation. Thus, the task of detecting the rock fragment is considered as a pixel-wise classification problem. To distinguish between adjacent rocks, additionally, the rock contours are considered. The labeling for contours is obtained from instance segmentation labels by the Canny edge detection algorithm. The dataset contains instance labels, semantic labels and contour labels. For semantic segmentation, five networks were trained: FCN8s, GCN, DFN, DeepLab v3, PSPNet. For contour segmentation, different models (based on the same architectures) were trained on the contour labels. For predicting rock object masks and contour masks, the mask fusion algorithm was applied to split adjacent rocks during the inference stage. Post-processing is used to reduce noise in the final prediction masks. The authors compare their approach with alternative methods, such as instance segmentation and traditional contour detection (Sobel operator) and show that their approach

has better accuracy, achieving the best visual results. In terms of segmentation accuracy, the mean IoU is 68.3%. To optimize the inference of two separate networks, the authors used the shared lightweight Y-shaped encoder network. The authors performed the inference speed test for the image resolution of 800×600 on the NVIDIA Titan XP GPU. By using the shared encoder, the performance went from 6.8 images per second to 33.3 without significant loss of accuracy. When authors replaced the ResNet101 by the lightweight MobileNetv2, the segmentation IoU score went from 75 to 68, but the size statistic accuracy was not lost. The performance was boosted to 57.2 images per second. For the larger resolution of 1600×1200 , the achieved inference speed was 19.4 images per second, which satisfies the system design requirement of 10.

A similar approach was used in work [57] for automatic segmentation of rock chips produced by the tunnel boring machine. The authors used a dual U-Net architecture with multi-scale inputs and side-output (MSD-UNet). Dual U-Net was used for multi-task semantic segmentation. Two decoders work to process the extracted features and generate segmentation maps separately for the regions and boundaries of the rocks. The post-processing was performed by smoothing the region masks, subtracting the boundary masks, using multi-radius erosion to separate the regions and using the seed filling algorithm to estimate the size and shape of the individual rocks. The authors compare their approach with U-Net and conventional segmentation methods, and show that it achieves a higher F1 score (0.867 and 0.64 for regions and boundaries respectively). Using the NVIDIA TITAN RTX GPU, their method can process a 2048×2048 image in 4 seconds. They note that only 10% of this time is spent on the inference, while the rest is spent on post-processing, namely, fusing the predictions and estimating size and shape. The boring system collects 30 images every 3 minutes. Thus, the method has sufficient performance to analyze the data in real-time.

Another application of the idea of separate region and boundary segmentation is presented in work [58]. It proposes the modification of U-Net for semantic segmentation of ore images. The model decoder includes two subnetworks, one for boundary detection and one for mask segmentation. The results of two networks is fused in boundary mask fusion blocks. As shown in experiments, the proposed architecture outperforms the original U-Net as well as the other known architectures, in terms of pixel accuracy, IoU for masks and error in ore particle size statistics. The models were trained using two NVIDIA GeForce GTX 1080 GPUs. The inference speed on a single GPU is 105 ms for the proposed method, in comparison with 44 ms for the original U-Net for the resolution of 256×256 .

Work [59] describes the segmentation method for processing images of ore fragments on a conveyor belt. The model is based on the RDU-Net model. It consists of DUNet (a modified U-Net segmentation network) and the residual connection structure of ResNet. This approach reduces the loss of information that occurs during information transfer between the deep convolution layers and improves the accuracy of model detection. The authors compare their model with traditional segmentation methods as well as with the DL methods, such as U-Net and DUNet. The RDU-Net model shows better accuracy metrics than alternative methods. The experiments were performed using the Intel i5-7500 CPU and the NVIDIA GeForce GTX 1050 Ti. The model achieves an inference speed of 20 seconds per image for an unspecified resolution which is a multiple of 48×48 .

A DL method for detecting and characterizing the rock fractures in tunnel face images was proposed in work [60]. The authors propose the FraSegNet model for pixel-wise semantic segmentation. It is based on a modified VGG19 CNN including five convolutional and four pooling layers, atrous spatial pyramid pooling module for sampling the feature maps and a decoder module that up-samples, fuses

the resulting maps and provides a final prediction using a guided filter. By qualitative comparison, the proposed model performs better than the traditional edge detection algorithms (namely, Canny and Laplacian) as well as the general-purpose semantic segmentation models (such as FCN and DeepLabV3+) in terms of boundary recognition and noise avoidance at the pixel level. On the NVIDIA GeForce GTX 1080 Ti GPU, the inference takes 0.44 seconds per image of size 1000×1000 , while the post-processing and statistical evaluation takes 7.2 seconds per image.

A model for segmenting the coal and gangue in the mining waste is presented in paper [14]. It is based on the U-Net architecture for pixel-level segmentation. The authors compare their model with several other gangue classification methods, namely, based on LeNet, AlexNet and CG-RPN. Their method provides comparable accuracy, and provides more information, not only identifying the samples, but also background pixels. The experiments were performed on an NVIDIA GeForce GTX 1080 Ti GPU. Training took 5.8 hours. The segmentation of a single image takes 48 ms for an image size of 512×512 which is sufficient for real-time processing.

In work [39], an ore segmentation method was proposed on the basis of U-Net architecture and the Watershed algorithm. To speed up the processing, the authors have modified the structure of U-Net without reducing accuracy. The modification consists of halving the number of convolution kernels in each layer of the network. To compensate for this reduction, the number of convolution operations at each stage is increased, making the network deeper. The loss function is modified to add the influence of uniform distribution and increase the weight of the background. The marked Watershed algorithm is used to post-process and optimize the segmentation results. The experiments were performed on the NVIDIA GeForce GTX 1080 Ti GPU. The inference speed for the resolution of 256×256 is 46 ms per image for the unmodified U-Net and 34 ms per image for the modified one. The post-processing takes 16 ms per image.

3.3. Instance segmentation

The instance segmentation task in DL is generally thought of as an object detection task with semantic segmentation of instances in each bounding box [61]. Besides there could be several other approaches, for instance, based on the idea of grouping semantic segmentation results into objects [62].

The most common architecture for instance segmentation is Mask R-CNN [63], which has a multistage architecture.. The Mask R-CNN consists of a common feature extractor (such as ResNet [64]), small region proposal network (for extracting a preliminary set of class-agnostic regions of interest called anchors), then the head part consists of classification, bounding box parameter regression and semantic segmentation for each anchor. After obtaining the network results, the Non-Maximum-Suppression (NMS) algorithm is applied to final object proposals. The illustration of the Mask R-CNN architecture is shown in Figure 9 [63]. Mask R-CNN was first proposed in 2017, but it remains one of the most popular solutions for segmentation [65].

The Mask R-CNN based model was proposed in [12] where the authors achieved accuracy of about 90% for an open-pit workplace. Furthermore, it has been shown that DL algorithms perform better than conventional CV approaches for high-scale images of workplaces. Also, this work notes that the accuracy of classical algorithms depends on daytime, sufficient light, dry weather and other external conditions.

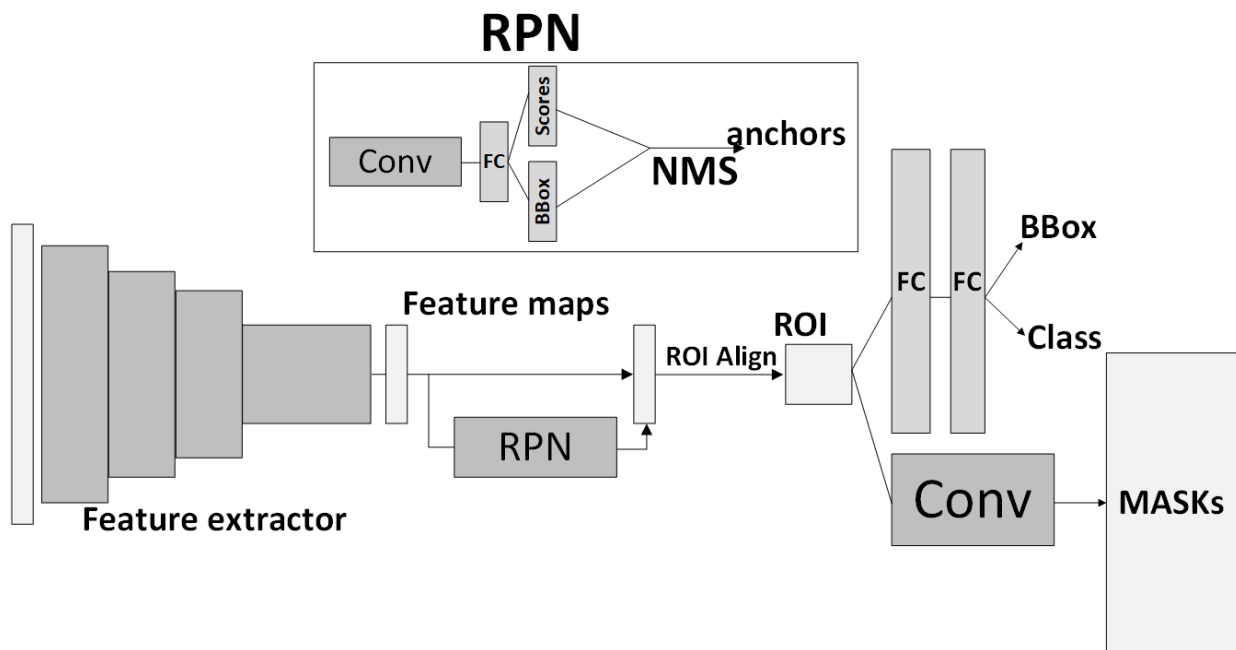


Figure 9. Illustration of the Mask R-CNN architecture with FPN neck part.

Almost similar results were shown in paper [66]. The paper compares conventional and DL approaches to large-scale images of open-pit workplaces. The authors stated the advantages of the Mask R-CNN for large scale images. Many contemporary authors state that traditional approaches suffer from wide dispersion of rock chunk sizes, overlapping instances and image quality for large-scale images, *e.g.*, see [5, 12, 23, 66].

Automatic CV system for estimating asbestos productivity in open pit workplace is constructed in work [67]. Its overall operation consists of 4 stages, namely, detecting rock chunks on the whole open-pit image using Mask R-CNN, fetching the images of individual rock chunk, segmenting the asbestos veins for the chunk and estimating average asbestos content in the workplace. The obtained results show high enough overall accuracy (absolute error is 0.4%), but less than geological service (manual estimation by an expert) do in the same condition.

The paper [68] proposed utilizing a Mask R-CNN based approach to the estimation of rock chunk size distribution on soil-rock mixture images. The authors proposed using simple Mask R-CNN model with ResNet-50 encoder. To correctly estimate the chunk size, the images contain a size standard. For results of the instance segmentation, equivalent size was calculated and distribution is compared to the sieve size distribution. The method of comparison is not pointed in the paper. The training was performed on an NVIDIA Tesla K80 GPU using the Detectron2 implementation of Mask R-CNN.

A DL model for analyzing the fragmentation (size distribution of particles) of rocks for producing rock-filled concrete is presented in [69]. Mask R-CNN based on the FPN and Resnet-101 is used for instance segmentation. The training and inference procedures run on a computer with an Intel Core i7-10700K CPU and NVIDIA GeForce GTX 1660 Ti GPU.

Another instance segmentation model for estimating the size distribution of rock-filled materials is constructed in work [70]. It is based on a modified Mask R-CNN network with ResNeXt101 as a backbone and Squeeze-and-Excitation block attention mechanism for enhanced feature extraction.

The core of SE-block is to automatically obtain the importance of each feature channel through loss learning. Thus, the network can use global information to selectively enhance beneficial feature channels, so as to achieve adaptive calibration of feature channels. The results of comparative experiments show that the authors' model achieves better results than other models (based on various Mask R-CNN backbones with or without SE-block), namely, 0.934 average precision and 0.879 average IoU values on the test images. The experiments were performed on an NVIDIA GeForce 3070 GPU and Intel i7-10700k CPU. The segmentation model takes 57 seconds to process the original resolution of 4000×3200 and 2.3 seconds for the size of 512×512 . The authors state that such performance is sufficient for practical use in real time.

A CV approach was proposed in work [71] for estimating the characteristics of riprap (rocks used to armor technical structures from erosion) from visual data. At its core, it has an instance segmentation model based on the Mask R-CNN and FCN architectures. This model provides 80% average completeness (ratio of segmented particles to ground-truth labeled particles) and 87% average IoU for the validation data set.

A DL approach for analyzing the rock sizes in the blast-induced rock fragmentation images was proposed in work [23]. It is based on a Mask R-CNN architecture, namely, ResNet50, which is used for instance segmentation, and global average pooling followed by dense fully connected layers as the top layer to predict fragmentation parameters, such as size distribution. The experiments were performed on an NVIDIA GeForce RTX 2080 Ti GPU. The trained model takes 5 seconds to process 153 image samples (30 images per second).

3.4. Object detection approach

The object detection approach is based on the idea of coarse bounding box determination for each object. This box could include the center position, width and height of each box. The difference with instance segmentation is that the object detection task does not incorporate segmentation of the objects inside each box. Some baseline of the object detection approach could be Faster-R-CNN which is acting similar to Mask R-CNN (see in Figure 9 but without a segmentation branch in the head part).

The “You Look Only Once” (YOLO) family of architectures was designed as some attempt at the trade-off between inference speed and accuracy. The main idea is to divide an image into uniform cells. Each object is detected as belonging to one of these cells. Each cell corresponds to one vector on the model output. The vector may include one or several anchors with sizes, shift of center coordinates predictions, class and objectness prediction. The term objectness means object or background determination in the cell. The result of objects is determined using a non-maximum suppression algorithm [72]. The YOLOv3 architecture utilizes a convolutional backbone based on the darknet architecture with a feature pyramid-based neck part and three scales of the output combined together. Each scale produces objects of the different sizes with a different trade-off between receptive field and sensitivity to small details [73]. The architecture family after YOLOv3 was designed using a lot of experiments with architecture and training techniques called “Bag of Specials” and “Bag of Fribies” correspondingly [74–76]. The illustration of the YOLOv3 architecture is shown in Figure 10.

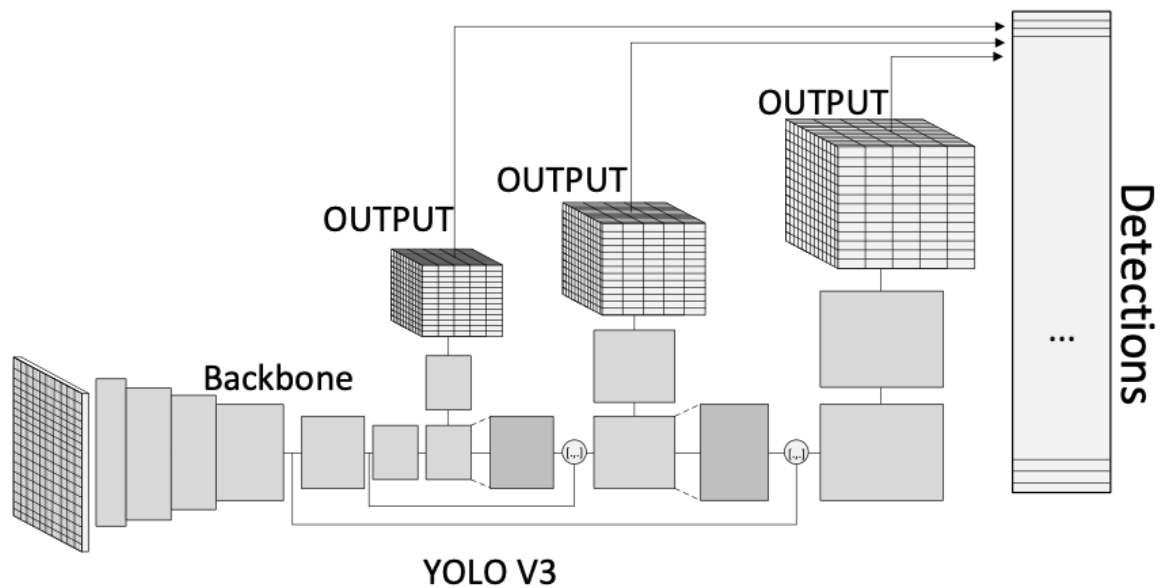


Figure 10. Illustration of the YOLOv3 architecture.

Paper [41] describes coke particle size distribution on a conveyor belt using the object detection approach. The CV approach in the task is proposed as a rapid and precise alternative to the sieving-based old-fashioned technique. The paper notices that the most common features for distribution estimation are area, circumference and diameter of estimated rock chunks. This set of features is obtained using a segmentation approach to the problem solution. The authors of [41] compare the common parameters of the bounding boxes: length, width, average of length and width, diagonal length and equivalent circle diameter to the real particle size making the conclusion that the width of boxes is the most relevant measure. As an additional benefit of using such measures selection, authors notice the reliability for the problem of over-lapping rock-chunks.

The authors of [41,77] compare several object detection models for the described problem solution. Among the compared models are: Single Shot MultiBox Detector (SSD), YOLOv3, YOLOv4, YOLOv5 and deformable convolution YOLO3 (DCN-YOLOv3), and the results indicated that DCN-YOLOv3 showed the highest accuracy and speed. It is noticed that as an additional benefit of using such measure selection, authors notice the reliability for the problem of overlapping rock chunks [41]. The algorithm of the DCN-YOLOv3 implementation to the coal gangue detection and recognition on the conveyor belt was also proposed in [41]. The authors achieved an inference speed of 2.7 to 2.9 images per second for YOLOv3 using the NVIDIA Tesla V100 16GB GPU for the resolution of 7360×4912 . Training took 8 hours. In [77], using the DCN-YOLOv3, authors achieved a speed of 33 images per second on the NVIDIA GeForce RTX 2070 Super GPU for a resolution of 416×416 .

Also, the paper [78] compares several popular approaches to the instance sizes distribution estimation. The following models were considered: Faster-R-CNN, SVM+HOG, Watershed and Yolo3. The comparison was done in the laboratory conditions. The Yolo3 shows the highest accuracy of 93% by mean average precision measures. The obtained accuracy for Faster-R-CNN is almost the same at 91%, but the inference speed is 5 times higher. As the authors state, the accuracy is enough for industry applications.

A system for detecting and sorting coal and gangue on the conveyor belt is described in paper [79].

The authors proposed their own YOLO3-based tiny architecture with a squeeze-and-excitation module and SPP-block. The achieved accuracy of 94% by mAP is comparable with the Faster-R-CNN based system but allowing it to work on a real-time scale. The authors trained the model on the Intel i7-8550U CPU in mixed precision using the NVIDIA APEX and Tensorflow frameworks. The inference speed is 4.28 seconds per image with full precision, and 1.56 seconds with mixed precision for 416×416 resolution.

Paper [42] suggests the YOLOv5 [80] for rock chunk determination and their size estimation, proving that this task may be considered as a problem of object detection. Via the use of the YOLOv5, open pit image fragmentation problems can be solved 10 times faster than the “traditional” Mask R-CNN approach without a significant loss of accuracy.

3.5. Classification/Regression approach

The problem of classifying the rock fragments produced by a tunnel boring machine is considered in paper [81]. The goal is to determine the category of image by the maximal size of rock fragments presented in it. For recognizing the fragments, a modified AlexNet was utilized. The modification consists of changing the output layer to contain 3 nodes (for 3 image classes) and integrating the network structure into a single GPU (because the original AlexNet is intended for multiple GPUs). The results of experiments show the competitive accuracy of the proposed methodology. The experiments were performed on a workstation with the NVIDIA GeForce RTX 2080 Ti GPU. For the input size of 512×512 the inference time is 28 ms per image (36 images per second).

A system for coal and empty rocks (gangue) classification on the conveyor belt using a custom CNN was proposed in paper [16]. The authors collected a 300-image dataset of images with only coal, mix and rocks. This was enough to achieve about 90% accuracy with both predefined and random augmentations.

Authors of [82] proposed to reduce the task of distribution estimation to the regression task. As a baseline, the VGG-16 network was proposed. The distribution was proposed to divide into 10 fractions parts. The novel loss function was also proposed to take into account the original regression problem as well as the relation of true fraction class predictions. The accuracy by R2-score of 80% was achieved in the on-belt system conditions. Training was conducted using Google Colab on an NVIDIA Tesla K80 GPU.

Paper [83] is devoted to classifying the rock structures on the images of a tunnel face. The authors use the CNN architecture Inception-ResNet-V2 combining GoogLeNet and ResNet backbone. The authors compare this architecture and other backbones, such as ResNet-50, ResNet-101 and Inception-v4. Their model achieves best performance in terms of precision, recall and F1 score. It also achieves the best inference time, namely, 0.325 seconds per image on the NVIDIA GeForce GTX 1080Ti GPU and 13 seconds per image on the Intel i7-8700K CPU. This means that the GPU is 40 times faster than the CPU.

A method for classifying the rock samples on the basis of CNN is proposed in paper [1]. Five types of CNN architectures were used for testing, namely, AlexNet, VGG16, ResNet50, InceptionV3 and MobileNet. The proposed models additionally use attention-based Squeeze-and-Excitation module (SENet) to improve the sensitivity of the model to channel features. The comparative experiments demonstrated the use of transfer learning to be effective against improving classification performance, such as reaching 94% with the MobileNet model.

4. Discussion

Let us briefly recite the main findings from the literature. The trends are divided in accordance with the methodology of the survey.

- The most results were obtained after 2000. Before this, only early attempts at fragmentation systems based on the classical CV approach were done.
- The main methods utilized in the range 2001-2017 are the Watershed and other classical algorithms, such as Otsu's thresholding and morphological reconstruction. Performance of these methods decreases under various outer conditions (light, fog, *etc*) and for very large-scale images.
- The classification/regression-based DL systems were popular in 2015–2019. The models are based on the prediction of each fraction's content. The results of such methods are very uninterpretable.
- The semantic segmentation systems was the main way in 2017–2021. Such systems require either an additional post-processing for boundaries detection or a separate semantic segmentation model for boundaries. The systems are mainly applied in conveyor conditions. The most popular solutions here are the U-Net-based architectures.
- The Mask R-CNN based instance segmentation based systems are applied starting from 2020. Such models allow one to work on large-scale images of the open-pit in different daytime and other conditions. But the Mask R-CNN models family require too much computing resources and are not capable of working in real time on poor-performance terminal devices.
- The YOLO-based object detection models are applied from 2021. These models provide quite accurate results at high speed of work.

According to the reviewed studies, the main solutions for the CV problems in the mining industry are U-Net [48] as an auxiliary semantic segmentation problem and Mask R-CNN [63] for prime instance segmentation tasks. Both networks contain feature extractors based primarily on the basic ResNet block and specific to head parts. However, in the original form these architectures do not provide high performance and accuracy compared to modern models (see for instance [84–86]).

Training and inference of neural network models for mining tasks is predominantly conducted on GPU-equipped workstations. However, there is significant variability in the reported performance data due to variations in GPU models and input image resolutions employed across different studies. Some authors provide specifications of devices without specific data on training or inference speeds. A number of papers correlate performance data with the throughput of associated equipment (*e.g.*, imaging cameras). Authors can draw conclusions about achieving real-time performance or the performance required to solve a practical problem. Notably, certain studies (*e.g.*, [60], [57]) demonstrate inference times that are considerably lower than post-processing times, diminishing the significance of neural network implementation performance. Certain authors [39,56] propose structural optimizations for neural network models, resulting in significant speed improvements in both training and inference. However, the performance and accuracy of these optimizations are typically evaluated only for specific tasks. In study [79], the NVIDIA APEX AMP framework is utilized to convert the model to mixed precision, leading to up to three-fold inference speedup.

The majority of the reviewed studies use different architectures and datasets, making direct comparisons difficult. Regardless of the obtained results, the task requires fast computation while

maintaining high accuracy. These requirements can be satisfied by combining modern lightweight neural network architectures and computation optimization techniques for DL from the other side.

The survey shows that researchers and organizations in the field of rock fragmentation continue to use the basic (and rather outdated) methods of CV both with and without DL neural networks. The majority of the related architectures refer to original papers published before 2017 (such architectures as U-Net and Mask R-CNN). These architectures are the most popular in the corresponding areas and can be considered as baseline models. In general, authors often provide limited discussions on the performance aspects of their developed models. Nevertheless, the intended industrial applications of the research results often involve lightweight, low-performance, energy-efficient computational devices, suggesting that the use of neural network optimization techniques would be beneficial.

In our opinion, the most promising approaches to rock fragmentation problems in open-pit mining appear to be so-called real-time semantic segmentation, instance segmentation and object detection tasks. The efficient models for these problems can be constructed using modifications in both the feature extractor part for more valid feature set extraction and head part for more time-efficient solution of the respective CV task. To achieve real-time performance, we recommend utilizing the modern high-performance implementation and optimization techniques for neural networks.

Sections 4.1 and 4.2 summarize the state-of-the-art achievements and trends both in models for real-time CV tasks and optimizing the implementations of neural networks on parallel architectures.

4.1. Computer vision architectures

Different feature extractor (or its block) architectures could be tested as a way of reducing computational complexity while keeping the trade-off with accuracy. Such feature extractors can be based on the EfficientNet V1/V2 [87, 88]; MobileNet [89–91] or Mobile ViT [92] principles of work rather than base-line ResNet [64, 93]. Also some other modifications of ResNet block could be tested [64, 94]. Also, a more advanced network training and regularization strategy could include such techniques as cross-data augmentation [95], batch normalization analogs [96] and meta-learning [97] to reduce computational costs while keeping high accuracy.

A survey of real-time semantic segmentation up to 2021 is presented in study [98]. The current (2023) best architectures in are PIDNet [99] and PP-LiteSeg [100]. PIDNet is based on the combination of CNN and Proportional-Integration Derivative (PID) controller to parse the information on details, context and boundary. The boundary information allows it to balance between the context and details and provide the precise annotation around boundary. PP-LiteSeg is a real-time semantic segmentation model based on the lightweight decoder, attention fusion module that utilizes spatial and channel attention to enhance the input feature, and the pyramid pooling module that aggregates global context at low computation cost.

Among approaches for fast object detection that can be tested, we want to note YOLOv5 – YOLOv8 [80, 86, 101, 102] and other models of the YOLO family. All of these models are based on the ideas cited above (see Section 3.4). However, today most of these architectures can be modified in the feature extractor part for different goals [75, 76]. Additionally, some of the YOLO family architectures provide real-time instance segmentation mode.

Among the other state-of-the-art results in real-time instance segmentation we can name YOLACT method [84] and their modifications, such as YOLACT++ [103] and SOLOv2 [104]. The discussed methods employ center-based segmentation as opposed to region-based segmentation in Mask R-

CNN. Here, the model is trained to distinguish pixels belonging to the same or different objects. For instance, YOLACT [84] generates prototype masks over the entire image, predicting a set of coefficients per instance by two parallel heads. The final masks are constructed after Non-Maximum suppression of predicted instances. The YOLACT++ architecture [103] improves previous results [84] by improving the feature extractor and adding deformable convolutions to the head part. The SOLO architecture [105] assumed that instances can be divided by their center position and sizes. The center positions are calculated by dividing the image into cells and calculating the center position inside each of them. The sizes of instances are determined by a pyramidal feature extractor. The SOLOv2 [104] improves the previous results by generating kernels for each mask. This concept is called a dynamic kernel.

Another model for real-time instance segmentation is proposed in work [106]. It uses the instance activation-guided queries to dynamically pick the pixel embeddings with rich semantic information for the initial queries for the Transformer decoder. A dual-path architecture is used for updating query and pixel features alternately. The ground truth mask-guided learning is used to replace the standard masked attention mechanism. As a result, masked attention can be directed to more appropriate regions.

An efficient fully convolutional framework for real-time instance segmentation is proposed in work [85]. Instead of using object detection and generating mask predictions on the basis of bounding boxes or dense centers, this model uses a new object representation. A sparse set of instance activation maps is used to highlight informative regions for objects. By aggregating features for the highlighted regions, the instance features for recognition and segmentation are obtained. This approach also allows one to avoid using non-maximum suppression in the post-processing phase.

To date (2023), the best performance in real-time segmentation is achieved by RTMDet model [107]. It is based on large-kernel depth-wise convolutions in a basic building block of the backbone and neck in the model. This allows the model to better capture the global context. To reduce the model depth and computational complexity, the number of building block is reduced, and model width is increased.

We should also note that the YOLOv5 and RTMDet models provide the oriented bounding box (or rotated object detection) mode, which extends the object detection task by predicting the slope of a bounding box at a little cost increase. Using this approach instead of traditional object detection may significantly increase the accuracy of some rock fragmentation problems such as particle size distribution estimation.

Table 1 presents the comparison of state-of-the-art real-time instance segmentation models. It contains the values of mask average precision obtained for the COCO dataset, as well as data on inference speed. It also contains the total number of parameters of the model, allowing one to estimate their complexity. Note that modern architectures, such as YOLO and RTMDet, provide models of varying sizes to allow users to achieve the best size-accuracy trade-off for specific application scenarios. The table contains data on the smallest and largest variants of these models. We also show the number of tunable parameters (such as score thresholds and loss functions) estimated from the article text or demonstration code for each model. We do not account for the standard hyper-parameters such as learning rate, batch size, etc. Note that more quantitative comparisons between the state-of-the-art architectures for various CV (and other machine learning) tasks are presented on the “Papers with Code” website*. Of course, achievements in benchmarks and synthetic tests do not

*<https://paperswithcode.com/sota>

guarantee performance in real practical applications.

Understanding of how tunable parameters influence model performance is a separate research. For example, the work [108] describes the collection of empirical techniques and refinements used to improve the accuracy of CNN models, in particular, the ResNet-50 feature extractor. Similarly, study [86] presents the model re-parameterization and model scaling methods for improving the YOLOv7 object detection model. Work [102] describes the model enhancement and training techniques used to boost the performance of YOLOv6 architecture so that in real-time object detection the new YOLOv6 v3.0 can compete with the latest YOLO models of similar size. A survey on hyper-parameter optimization methods is presented in study [109].

Table 1. Comparison of real-time instance segmentation models

Model	Year	Tunable parameters	Total parameters (M)	AP (%)	Inference Speed (FPS)
Mask R-CNN [63]	2017	10	63.3	35.7	5 (Titan XP)
YOLACT-550 [84]	2019	6	50	29.8	33.3 (Titan XP)
YOLACT-550++ [103]	2020	6	No data	34.6	27.3 (Titan XP)
SOLOv2-512 [105]	2020	7	46.4	37.1	31.3 (V100)
SparseInst [85]	2022	4	33.3	37.9	40 (RTX 2080 Ti)
FastInst [106]	2023	20	53	39.9	28 FPS (V100)
YOLOv5n-seg [80]	2020	20+	2	23.4	833 (A100)
YOLOv5x-seg	2020	20+	88.8	41.4	222 (A100)
YOLOv8n-seg	2022	20+	3.4	36.7	826 (A100)
YOLOv8x-seg	2022	20+	71.8	43.4	248 (A100)
RTMDet-Ins-s [107]	2022	6	5.6	38.7	518 (RTX 3090)
RTMDet-Ins-x	2022	6	102.7	44.6	188 (RTX 3090)

We would also like to acknowledge the latest achievements in the field of instance and semantic segmentation based on the Segment Anything [110] and Diffusion [111–113] approaches, although these methods have yet to prove their effectiveness in real-time applications in general and in rock fragmentation problems in particular.

4.2. Performance aspects

Key trends in optimizing DL neural network implementations include quantization, pruning, hardware and architecture co-design, knowledge distillation and parallelization techniques for high-performance systems.

Hardware manufacturers and software developers introduce capabilities for mixed- and reduced-precision calculations, as well as sparse computations, to support quantization and pruning techniques [114, 115]. These techniques enable model size reduction and computation time reduction, making trained models applicable to lightweight devices. The comprehensive survey of quantization

techniques and specialized frameworks for mixed-precision neural networks is presented in work [116]. Automatic tools for adaptive application of mixed-precision techniques are available in modern neural network frameworks [117–120].

Various hardware architectures, including CPUs, GPUs and specialized accelerators based on FPGA and ASIC technologies are employed for neural network implementations. While GPUs have significant potential for training and inference, their price and energy efficiency are often suboptimal. Lightweight specialized hardware accelerators offer better for energy efficiency, but their optimal performance requires full utilization of the hardware architecture. Hardware/software co-design strategies for implementing convolutional neural networks are extensively surveyed in work [121].

Knowledge distillation involves training a smaller student model from a pre-trained larger teacher model. A comprehensive study covering training schemes, architectures, distillation algorithms, performance comparison and applications of knowledge distillation is available in the survey [122].

To efficiently utilize the power of massive parallel systems, such as multi-GPU ones, for training DL networks, specialized parallelism strategies like data parallelism, model parallelism, pipelining or hybrid approaches are employed. Challenges and recent achievements in adopting these strategies are explored in survey [123].

5. Conclusions

This study presents a comprehensive survey of the application of DL-based CV systems for rock fragmentation estimation and related tasks in the mining industry, focusing on recent advancements and current research trends. Additionally, a brief summary is provided of state-of-the-art achievements in real-time semantic segmentation, instance segmentation, object detection and performance optimization of DL neural networks.

The problem of rock fragmentation consists of identification and differentiation of multiple rock chunks of various sizes from visual data. This review reveals a lack of consensus on both the choice of general CV approaches and individual machine learning models for solving these problems. Most studies in the field concentrate on size distribution estimation of rock fragments using either semantic segmentation or instance segmentation, while object detection has gained more recent attention.

Through the review, it was observed that contemporary methods in rock fragmentation lag behind the overall progress in CV over the past 2-3 years. Many studies primarily focus on baseline DL architectures such as U-Net, Mask R-CNN and YOLO for semantic segmentation, instance segmentation and object detection, respectively. Promising approaches for the task involve real-time DL instance segmentation or object detection (including oriented object detection) architectures utilizing various feature encoders and head parts. This could include considering new versions of the YOLO family (YOLO v7, v8) and RTMDet models, as well as optimizing models with feature extractors such as EfficientNet, modern ResNet-based or mobile ViT-based blocks. Improving model accuracy can also be achieved through meta-learning or other regularization techniques. However, the specific architecture choice, pros and cons of different approaches, and recommendations for usage require separate research.

The majority of studies in this field employ parallel computing architectures for training and inference, typically utilizing workstations with graphics processors, without performance optimization. Modern techniques for optimizing the performance and size of DL models include mixed-precision

arithmetic and sparse matrix computations. Lightweight DL models can be trained from larger pretrained models using knowledge distillation methods. Several strategies for parallelizing deep neural networks on massive parallel systems can be identified, including data parallelism, model parallelism, pipelining and hybridization. Implementing these techniques would enable researchers in the field of rock fragmentation to achieve higher performance, both in training their models on high-performance systems in data centers and in inference on specialized low-performance computing devices in industrial installations.

In conclusion, progress in the field of rock fragmentation can be achieved by striking a balance between state-of-the-art deep learning computer vision architectures and optimizing work performance using modern techniques.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research was supported by the Russian Science Foundation and Government of Sverdlovsk region, Joint Grant No 22-21-20051, <https://rscf.ru/en/project/22-21-20051/>.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix: Summary of reviewed papers on fragmentation topic

The following table presents the works on the survey topic. Important works summarized in Section 3 are marked in bold. The studies in the table are sorted by publication year and grouped by practical field of rock fragmentation problem.

Table Appendix. Summary of papers on the review topic.

Reference	Year	Practical Problem	CV Task	Approach	Specifics	Primary CV Model
[124]	2023	Conveyor	Semantic	DL	SLIC + U-Net	U-Net
[41]	2022	Conveyor	Object	DL	SSD, YOLOv3, YOLOv4, YOLOv5, DCN-YOLOv3	YOLO
[79]	2022	Conveyor	Object	DL	Yolo3, SPPNet, Yolo1,2	YOLO
[1]	2022	Conveyor	Classification	DL	AlexNet, VGG16, ResNet50, InceptionV3, MobileNet, SENet	MobileNet
[125]	2021	Conveyor	Semantic	DL	DexiNed Detection + Morphologic Watershed	Edge + Other
[18]	2021	Conveyor	Semantic	DL	U-Net	U-Net
[14]	2020	Conveyor	Semantic	DL	U-Net	U-Net
[15]	2020	Conveyor	Semantic	DL	U-Net, Res-U-Net + Watershed	U-Net
[59]	2020	Conveyor	Semantic	DL	U-Net, RDUNET, DUNET	U-Net
[126]	2020	Conveyor	Semantic	DL	U-Net + Watershed	U-Net
[127]	2020	Conveyor	Semantic	DL	CNN + Watershed	Watershed
[128]	2019	Conveyor	Semantic	Classical	Otsu + Histogram accumulation moment	Other
[129]	2019	Conveyor	Other	Classical	Super Voxel 3D	Other
[20]	2023	Open Pit	Instance	DL	Mask R-CNN	Mask R-CNN
[47]	2022	Open Pit	Instance	Classical	Modified Watershed	Watershed
[70]	2022	Open Pit	Instance	DL	ResNeXt101 + SE	Mask R-CNN
[68]	2022	Open Pit	Instance	DL	ResNet-50	Mask R-CNN

Continued

Reference	Year	Practical Problem	CV Task	Approach	Specifics	Primary CV Model
[42]	2022	Open Pit	Object	DL	YOLO	YOLO
[58]	2021	Open Pit	Semantic	DL	Dual boundary regions	U-Net, and
[46]	2021	Open Pit	Semantic	Classical	Wavelet Superpixel segmentation	+ Other
[12]	2021	Open Pit	Instance	DL	Mask R-CNN	Mask R-CNN
[130]	2021	Open Pit	Instance	Classical	Super-Voxel Segmentation	Other
[67]	2021	Open Pit	Instance	DL	Mask R-CNN	Mask R-CNN
[131]	2021	Open Pit	Instance	Classical	Watershed	Watershed
[40]	2021	Open Pit	Instance	DL	Mask R-CNN	Mask R-CNN
[7]	2021	Open Pit	Classification	Classical	ResNeXt-50	ResNet
[132]	2021	Open Pit	Classification	Classical	SM, GMB, SVM + Firefly evolutionary algorithm	Other
[133]	2021	Open Pit	Classification	Classical	ANN, Monte Carlo dropout	Other
[134]	2021	Open Pit	Other	Classical	Power software Sieve3	Other
[82]	2020	Open Pit	Classification	DL	VGG-16	VGG
[21]	2019	Open Pit	Instance	Classical	Split-Desktop Software	Other
[6]	2019	Open Pit	Other	Classical	Custom	Other
[135]	2018	Open Pit	Classification	Classical	SVM	Other
[136]	2013	Open Pit	Classification	Classical	ANN	Other
[137]	2009	Open Pit	Instance	Classical	Watershed	Watershed
[22]	2005	Open Pit	Instance	Classical	Watershed	Watershed
[138]	2003	Open Pit	Other	Classical	WipFrag software	Other

Continued

Reference	Year	Practical Problem	CV Task	Approach	Specifics	Primary CV Model
[56]	2021	Tunnel	Semantic	DL	FCN8s, GCN, DFN, DeepLabV3, PSPNet	DFN
[57]	2021	Tunnel	Semantic	DL	Dual boundary regions	U-Net, and
[55]	2021	Tunnel	Instance	DL	SSD + U-Net + post-processing	U-Net
[81]	2021	Tunnel	Classification	DL	AlexNet	Other
[83]	2021	Tunnel	Classification	DL	ResNet + GoogLeNet	ResNet
[139]	2013	Tunnel	Instance	Classical	Watershed	Watershed
[140]	2011	Tunnel	Instance	Classical	Watershed	Watershed
[141]	2023	Other	Semantic	DL	MSBA-Unet + CDD post-processing	U-Net
[43]	2023	Other	Object	DL	YOLO	YOLO
[78]	2023	Other	Object	DL	Watershed, FASTER-R-CNN, YOLO3	YOLO
[142]	2022	Other	Semantic	DL	Swin-Unet	U-Net
[143]	2022	Other	Semantic	DL	DeepLab3+	DeepLab
[77]	2022	Other	Object	DL	DCN-YOLOv3	YOLO
[144]	2022	Other	Classification	Classical	ANN + SVR	Other
[145]	2021	Other	Semantic	DL	GAN + U-Net	U-Net
[60]	2021	Other	Semantic	DL	FraSegNet: VGG19 + ASPP	Other
[23]	2021	Other	Instance	Other	ResNet50	Mask R-CNN
[69]	2021	Other	Instance	DL	Mask R-CNN	Mask R-CNN
[146]	2021	Other	Classification	DL	AlexNet, VGG	VGG
[2]	2021	Other	Other	Classical	Custom	Other

Continued

Reference	Year	Practical Problem	CV Task	Approach	Specifics	Primary CV Model
[147]	2020	Other	Semantic	DL	Lightweight U-Net + Morphologic + Watershed	U-Net
[71]	2020	Other	Instance	DL	Mask R-CNN + FCN	Mask R-CNN
[3]	2020	Other	Object	DL	Faster R-CNN	Faster R-CNN
[148]	2020	Other	Classification	Classical	Edge Detection	Other
[16]	2020	Other	Other	DL	Custom CNN	Other
[17]	2020	Other	Other	Classical	Custom	Other
[4]	2019	Other	Semantic	DL	SegNet	Other
[5]	2019	Other	Instance	Classical	SLIC Segmentation + Features + Classification	Other
[149]	2019	Other	Instance	Classical	Wavelet + manual features + ANN	Other
[150]	2019	Other	Instance	Classical	Morphological Reconstruction	Other
[151]	2019	Other	Instance	Classical	Otsu + Watershed	Watershed
[152]	2018	Other	Semantic	DL	Holistically-nested convolutional network	Other
[153]	2018	Other	Instance	Classical	Morphologic Watershed + Watershed	Watershed
[154]	2018	Other	Instance	Classical	Thresholding Watershed + Watershed	Watershed
[26]	2018	Other	Other	Classical	WipFrag software	Other
[155]	2016	Other	Classification	Classical	Morphologic Watershed + SVM for classification	Watershed
[156]	2016	Other	Other	Classical	WipFrag software	Other
[157]	2015	Other	Classification	Classical	ANN + post-processing	Other

Continued

Reference	Year	Practical Problem	CV Task	Approach	Specifics	Primary CV Model
[158]	2013	Other	Classification	Classical	ANN + Regression	Other
[159]	2013	Other	Other	Classical	GoldSize software	Other
[160]	2012	Other	Classification	Classical	SVM	Other
[161]	2011	Other	Instance	Classical	Watershed	Watershed
[163]	2010	Other	Instance	Classical	Watershed	Watershed
[164]	2009	Other	Semantic	Classical	Thresholding morphologic	+ Other
[165]	2009	Other	Classification	Classical	Fuzzy Logic	Other
[166]	2008	Other	Semantic	Classical	Edge Detection	Other
[167]	2005	Other	Other	Classical	Laser 3d Watershed	+ Other
[168]	1999	Other	Classification	Classical	Morphologic + ANN for classification	Other
[169]	1996	Other	Other	Classical	Image Preparation	Other

References

1. W. Zhou, H. Wang, Z. Wan, Ore image classification based on improved CNN, *Comput. Electr. Eng.*, **99** (2022), 107819. <https://doi.org/10.1016/j.compeleceng.2022.107819>
2. S. S. Amiripallia, G. N. Rao, J. Beharaa, K. Sanjay, Mineral rock classification using convolutional neural network, *Adv. Parallel Comput.*, **39** (2021), 499–505. <https://doi.org/10.3233/APC210235>
3. X. Liu, H. Wang, H. Jing, A. Shao, L. Wang, Research on intelligent identification of rock types based on Faster R-CNN method, *IEEE Access*, **8** (2020), 21804–21812. <https://doi.org/10.1109/ACCESS.2020.2968515>
4. S. Karimpouli, P. Tahmasebi, Segmentation of digital rock images using deep convolutional autoencoder networks, *Comput. Geosci.*, **126** (2019), 142–150. <https://doi.org/10.1016/j.cageo.2019.02.003>
5. J. Maitre, K. Bouchard, L. P. Bédard, Mineral grains recognition using computer vision and machine learning, *Comput. Geosci.*, **130** (2019), 84–93. <https://doi.org/10.1016/j.cageo.2019.05.009>
6. M. He, Z. Zhang, J. Ren, J. Huan, G. Li, Y. Chen, et al., Deep convolutional neural network for fast determination of the rock strength parameters using drilling data, *Int. J. Rock Mech. Min. Sci.*, **123** (2019), 104084. <https://doi.org/10.1016/j.ijrmms.2019.104084>

7. F. Alzubaidi, P. Mostaghimi, P. Swietojanski, S. R. Clark, R. T. Armstrong, Automated lithology classification from drill core images using convolutional neural networks, *J. Pet. Sci. Eng.*, **197** (2021), 107933. <https://doi.org/10.1016/j.petrol.2020.107933>
8. F. Zeng, A. Jacobson, D. Smith, N. Boswell, T. Peynot, M. Milford, Lookup: Vision-only real-time precise underground localisation for autonomous mining vehicles, In: *2019 International conference on robotics and automation (ICRA)*, IEEE, 2019, 1444–1450. <https://doi.org/10.1109/ICRA.2019.8794453>
9. T. Chen, N. Hu, R. Niu, N. Zhen, A. Plaza, Object-oriented open-pit mine mapping using Gaofen-2 satellite image and convolutional neural network, for the Yuzhou City, China, *Remote Sens.*, **12** (2020), 3895. <https://doi.org/10.3390/rs12233895>
10. J. Baek, Y. Choi, Deep neural network for predicting ore production by truck-haulage systems in open-pit mines, *Appl. Sci.*, **10** (2020), 1657. <https://doi.org/10.3390/app10051657>
11. J. Williams, J. Singh, M. Kumral, J. Ramirez Ruiseco, Exploring deep learning for dig-limit optimization in open-pit mines, *Nat. Resour. Res.*, **30** (2021), 2085–2101. <https://doi.org/10.1007/s11053-021-09864-y>
12. T. Vu, T. Bao, Q. V. Hoang, C. Drebenstedt, P. V. Hoa, H. H. Thang, Measuring blast fragmentation at Nui Phao open-pit mine, Vietnam using the Mask R-CNN deep learning model, *Min. Technol.*, **130** (2021), 232–243. <https://doi.org/10.1080/25726668.2021.1944458>
13. G. Somua-Gyimah, S. Frimpong, W. Nyaaba, E. Gbadam, A computer vision system for terrain recognition and object detection tasks in mining and construction environments, In: *SME Annual Conference*, 2019.
14. R. Gao, Z. Sun, W. Li, L. Pei, Y. Hu, L. Xiao, Automatic coal and gangue segmentation using U-Net based fully convolutional networks, *Energies*, **13** (2020), 829. <https://doi.org/10.3390/en13040829>
15. X. Liu, Y. Zhang, H. Jing, L. Wang, S. Zhao, Ore image segmentation method using U-Net and Res_Unet convolutional networks, *RSC Adv.*, **10** (2020), 9396–9406. <https://doi.org/10.1039/C9RA05877J>
16. L. Si, X. Xiong, Z. Wang, C. Tan, A deep convolutional neural network model for intelligent discrimination between coal and rocks in coal mining face, *Math. Probl. Eng.*, **2020**. <https://doi.org/10.1155/2020/2616510>
17. C. Su, S. Xu, K. Zhu, X. Zhang, Rock classification in petrographic thin section images based on concatenated convolutional neural networks, *Earth Sci. Inform.*, **13** (2020), 1477–1484. <https://doi.org/10.1007/s12145-020-00505-1>
18. V. Zyuzin, M. Ronkin, S. Porshnev, A. Kalmykov, Computer vision system for the automatic asbestos content control in stones, In: *Journal of Physics: Conference Series*, vol. 1727, IOP Publishing, 2021, 012014. [10.1088/1742-6596/1727/1/012014](https://doi.org/10.1088/1742-6596/1727/1/012014)
19. A. Azizi, H. Moomivand, A new approach to represent impact of discontinuity spacing and rock mass description on the median fragment size of blasted rocks using image analysis of rock mass, *Rock Mech. Rock Eng.*, **54** (2021), 2013–2038. <https://doi.org/10.1007/s00603-020-02360-4>

20. S. Shrivastava, S. Bhattacharjee, D. Deb, Segmentation of mine overburden dump particles from images using Mask R CNN, *Sci. Rep.*, **13** (2023), 2046. <https://doi.org/10.1038/s41598-023-28586-0>
21. M. Babaeian, M. Ataei, F. Sereshki, F. Sotoudeh, S. Mohammadi, A new framework for evaluation of rock fragmentation in open pit mines, *J. Rock Mech. Geotech. Eng.*, **11** (2019), 325–336. <https://doi.org/10.1016/j.jrmge.2018.11.006>
22. S. Mkwelo, V. Nicolls, G. De Jager, Watershed-based segmentation of rock scenes and proximity-based classification of watershed regions under uncontrolled lighting, *SAIEE Afr. Res. J.*, **96** (2005), 28–34. <https://doi.org/10.23919/SAIEE.2005.9488146>
23. T. Bamford, K. Esmaeili, A. P. Schoellig, A deep learning approach for rock fragmentation analysis, *Int. J. Rock Mech. Min. Sci.*, **145** (2021), 104839. <https://doi.org/10.1016/j.ijrmms.2021.104839>
24. D. Jung, Y. Choi, Systematic review of machine learning applications in mining: exploration, exploitation, and reclamation, *Minerals*, **11** (2021), 148. <https://doi.org/10.3390/min11020148>
25. J. A. Franklin, T. Katsabanis, *Measurement of blast fragmentation*, Rotterdam: CRC Press, 1996.
26. A. Tosun, A modified Wipfrag program for determining muckpile fragmentation, *J. S. Afr. Inst. Min. Metall.*, **118** (2018), 1113–1199. <https://doi.org/10.17159/2411-9717/2018/v118n10a13>
27. J. P. Latham, J. Kemeny, N. Maerz, M. Noy, J. Schleifer, S. Tose, A blind comparison between results of four image analysis systems using a photo-library of piles of sieved fragments, *Fragblast*, **7** (2003), 105–132. <https://doi.org/10.1076/frag.7.2.105.15899>
28. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, et al., Backpropagation applied to handwritten zip code recognition, *Neural Comput.*, **1** (1989), 541–551. <https://doi.org/10.1162/neco.1989.1.4.541>
29. Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, et al., Handwritten digit recognition with a back-propagation network, In: *Advances in Neural Information Processing Systems* (Ed. D. Touretzky), vol. 2, Morgan-Kaufmann, 1989.
30. C. Lu, Z. Wang, B. Zhou, Intelligent fault diagnosis of rolling bearing using hierarchical convolutional network based health state classification, *Mech. Syst. Signal Process.*, **32** (2017), 139–151. <https://doi.org/10.1016/j.aei.2017.02.005>
31. W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load, *Mech. Syst. Signal Process.*, **100** (2018), 439–453. <https://doi.org/10.1016/j.ymsp.2017.06.022>
32. Y. J. Cha, W. Choi, O. Büyüköztürk, Deep learning-based crack damage detection using convolutional neural networks, *Comput. Aided Civ. Infrastruct. Eng.*, **32** (2017), 361–378. <https://doi.org/10.1111/mice.12263>
33. H. V. Dang, M. Raza, T. V. Nguyen, T. Bui-Tien, H. X. Nguyen, Deep learning-based detection of structural damage using time-series data, *Struct. Infrastruct. Eng.*, **17** (2021), 1474–1493. <https://doi.org/10.1080/15732479.2020.1815225>

34. F. Yessoufou, J. Zhu, One-class convolutional neural network (OC-CNN) model for rapid bridge damage detection using bridge response data, *KSCE J. Civ. Eng.*, **27** (2023), 1640–1660. <https://doi.org/10.1007/s12205-023-0063-7>
35. D. Ai, J. Cheng, A deep learning approach for electromechanical impedance based concrete structural damage quantification using two-dimensional convolutional neural network, *Mech. Syst. Signal Pr.*, **183** (2023), 109634. <https://doi.org/10.1016/j.ymssp.2022.109634>
36. Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE*, **86** (1998), 2278–2324. <https://doi.org/10.1109/5.726791>
37. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, Cambridge, MA: MIT Press, 2016.
38. A. Zhang, Z. C. Lipton, M. Li, A. J. Smola, Dive into deep learning, (2023). <https://doi.org/10.48550/arXiv.2106.11342>
39. H. Li, C. Pan, A. Chen, Z. Wulamu, A. Yang, Ore image segmentation method based on U-Net and Watershed, *Comput. Mater. Contin.*, **65** (2020), 563–578. <https://doi.org/10.32604/cmc.2020.09806>
40. D. Yang, X. Wang, H. Zhang, Z. Yin, D. Su, J. Xu, A Mask R-CNN based particle identification for quantitative shape evaluation of granular materials, *Powder Technol.*, **392** (2021), 296–305. <https://doi.org/10.1016/j.powtec.2021.07.005>
41. M. Li, X. Wang, H. Yao, H. Saxén, Y. Yu, Analysis of particle size distribution of coke on blast furnace belt using object detection, *Processes*, **10** (2022), 1902. <https://doi.org/10.3390/pr10101902>
42. M. Ronkin, A. Kalmykov, K. Reshetnikov, V. Zyuzin, Investigation of object detection based method for open-pit blast quality estimation, In: *2022 Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBREIT)*, IEEE, 2022, 248–251. <https://doi.org/10.1109/USBREIT56278.2022.9923353>
43. G. Xue, S. Li, P. Hou, S. Gao, R. Tan, Research on lightweight Yolo coal gangue detection algorithm based on resnet18 backbone feature network, *Internet Things*, **22** (2023), 100762. <https://doi.org/10.1016/j.iot.2023.100762>
44. A. K. Sangaiah, *Deep learning and parallel computing environment for bioengineering systems*, St. Louis, MO: Academic Press, 2019.
45. S. Beucher, Use of watersheds in contour detection, In: *Proceedings of the International Workshop on Image Processing*, CCETT, 1979.
46. Z. Yang, H. Ding, L. Guo, M. Lian, Superpixel image segmentation-based particle size distribution analysis of fragmented rock, *IEEE Access*, **9** (2021), 59048–59058. <https://doi.org/10.1109/ACCESS.2021.3072998>
47. Q. Guo, Y. Wang, S. Yang, Z. Xiang, A method of blasted rock image segmentation based on improved watershed algorithm, *Sci. Rep.*, **12** (2022), 7143. <https://doi.org/10.1038/s41598-022-11351-0>
48. O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, In: *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

49. N. Siddique, S. Paheding, C. P. Elkin, V. Devabhaktuni, U-Net and its variants for medical image segmentation: A review of theory and applications, *IEEE Access*, **9** (2021), 82031–82057. <https://doi.org/10.1109/ACCESS.2021.3086020>
50. X. X. Yin, L. Sun, Y. Fu, R. Lu, Y. Zhang, U-Net-based medical image segmentation, *J. Healthc. Eng.*, **2022** (2022), 4189781. <https://doi.org/10.1155/2022/4189781>
51. J. Wu, Y. Guo, F. Gou, Z. Dai, A medical assistant segmentation method for MRI images of osteosarcoma based on DecoupleSegNet, *Int. J. Intell. Syst.*, **37** (2022), 8436–8461. <https://doi.org/10.1002/int.22949>
52. Z. Ling, S. Yang, F. Gou, Z. Dai, J. Wu, Intelligent assistant diagnosis system of osteosarcoma mri image based on transformer and convolution in developing countries, *IEEE J. Biomed. Health Inform.*, **26** (2022), 5563–5574. <https://doi.org/10.1109/JBHI.2022.3196043>
53. Z. Zhou, F. Gou, Y. Tan, J. Wu, A cascaded multi-stage framework for automatic detection and segmentation of pulmonary nodules in developing countries, *IEEE J. Biomed. Health Inform.*, **26** (2022), 5619–5630. <https://doi.org/10.1109/JBHI.2022.3198509>
54. J. Wu, W. Liu, C. Li, T. Jiang, I. M. Shariful, H. Sun, et al., A state-of-the-art survey of U-Net in microscopic image analysis: from simple usage to structure mortification, (2022). <https://doi.org/10.48550/arXiv.2202.06465>
55. W. Qiao, Y. Zhao, Y. Xu, Y. Lei, Y. Wang, S. Yu, et al., Deep learning-based pixel-level rock fragment recognition during tunnel excavation using instance segmentation model, *Tunn. Undergr. Space Technol.*, **115** (2021), 104072. <https://doi.org/10.1016/j.tust.2021.104072>
56. Z. Xue, L. Chen, Z. Liu, F. Lin, W. Mao, Rock segmentation visual system for assisting driving in tbn construction, *Mach. Vis. Appl.*, **32** (2021), 77. <https://doi.org/10.1007/s00138-021-01203-8>
57. X. Zhou, Q. Gong, Y. Liu, L. Yin, Automatic segmentation of tbn muck images via a deep-learning approach to estimate the size and shape of rock chips, *Autom. Constr.*, **126** (2021), 103685. <https://doi.org/10.1016/j.autcon.2021.103685>
58. W. Wang, Q. Li, C. Xiao, D. Zhang, L. Miao, L. Wang, An improved boundary-aware U-Net for ore image semantic segmentation, *Sensors*, **21** (2021), 2615. <https://doi.org/10.3390/s21082615>
59. D. Xiao, X. Liu, B. T. Le, Z. Ji, X. Sun, An ore image segmentation method based on RDU-Net model, *Sensors*, **20** (2020), 4979. <https://doi.org/10.3390/s20174979>
60. J. Chen, M. Zhou, H. Huang, D. Zhang, Z. Peng, Automated extraction and evaluation of fracture trace maps from rock tunnel face images via deep learning, *Int. J. Rock Mech. Min. Sci.*, **142** (2021), 104745. <https://doi.org/10.1016/j.ijrmms.2021.104745>
61. W. Gu, S. Bai, L. Kong, A review on 2D instance segmentation based on deep neural networks, *Image Vis. Comput.*, **120** (2022), 104401. <https://doi.org/10.1016/j.imavis.2022.104401>
62. A. M. Hafiz, G. M. Bhat, A survey on instance segmentation: state of the art, *Int. J. Multimed. Inf. Retr.*, **9** (2020), 171–189. <https://doi.org/10.1007/s13735-020-00195-x>
63. K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, In: *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, 2980–2988. <https://doi.org/10.1109/ICCV.2017.322>

-
64. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, 770–778. <https://doi.org/10.1109/CVPR.2016.90>
65. S. Ramesh C., A review on instance segmentation using mask R-CNN, In: *Proceedings of the International Conference on Systems, Energy & Environment (ICSEE)*, 2021.
66. F. Schenk, A. Tscharf, G. Mayer, F. Fraundorfer, Automatic muck pile characterization from UAV images, In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-2/W5, 2019, 163–170. <https://doi.org/10.5194/isprs-annals-IV-2-W5-163-2019>
67. V. Zyuzin, M. Ronkin, S. Porshnev, A. Kalmykov, Automatic asbestos control using deep learning based computer vision system, *Appl. Sci.*, **11** (2021), 10532. <https://doi.org/10.3390/app112210532>
68. S. Shrivastava, D. Deb, S. Bhattacharjee, Prediction of particle size distribution curves of dump materials using convolutional neural networks, *Rock Mech. Rock Eng.*, **55** (2022), 471–479. <https://doi.org/10.1007/s00603-021-02657-y>
69. L. Fu, X. Xu, F. Jin, H. Zhou, Evaluation of the particle size distribution of on-site rockfill using mask r-cnn deep learning model, In: *2021 7th International Conference on Hydraulic and Civil Engineering & Smart Water Conservancy and Intelligent Disaster Reduction Forum (ICHCE & SWIDR)*, 2021, 291–297. <https://doi.org/10.1109/ICHCESWIDR54323.2021.9656248>
70. H. Fan, Z. Tian, X. Xu, X. Sun, Y. Ma, H. Liu, et al., Rockfill material segmentation and gradation calculation based on deep learning, *Case Stud. Constr. Mater.*, **17** (2022), e01216. <https://doi.org/10.1016/j.cscm.2022.e01216>
71. H. Huang, J. Luo, E. Tutumluer, J. M. Hart, A. J. Stolba, Automated segmentation and morphological analyses of stockpile aggregate images using deep convolutional neural networks, *Transp. Res. Rec.*, **2674** (2020), 285–298. <https://doi.org/10.1177/0361198120943887>
72. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, 779–788. <https://doi.org/10.1109/CVPR.2016.91>
73. J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, (2018). <https://doi.org/10.48550/arXiv.1804.02767>
74. A. Bochkovskiy, C. Y. Wang, H. Y. M. Liao, YOLOv4: Optimal speed and accuracy of object detection, (2020). <https://doi.org/10.48550/arXiv.2004.10934>
75. T. Diwan, G. Anirudh, J. V. Tembhurne, Object detection using YOLO: Challenges, architectural successors, datasets and applications, *Multimed. Tools Appl.*, **82** (2023) 9243–9275. <https://doi.org/10.1007/s11042-022-13644-y>
76. P. Jiang, D. Ergu, F. Liu, Y. Cai, B. Ma, A review of Yolo algorithm developments, *Procedia Comput. Sci.*, **199** (2022), 1066–1073. <https://doi.org/10.1016/j.procs.2022.01.135>
77. D. Li, G. Wang, Y. Zhang, S. Wang, Coal gangue detection and recognition algorithm based on deformable convolution YOLOv3, *IET Image Process.*, **16** (2022), 134–144. <https://doi.org/10.1049/ipr2.12339>

-
78. X. Wang, Y. Guo, Y. Yu, An appropriate approach to recognize coke size distribution in a blast furnace, *Processes*, **11** (2023), 187. <https://doi.org/10.3390/pr11010187>
79. H. Pan, Y. Shi, X. Lei, Z. Wang, F. Xin, Fast identification model for coal and gangue based on the improved tiny YOLO v3, *J. Real-Time Image Proc.*, **19** (2022), 687–701. <https://doi.org/10.1007/s11554-022-01215-1>
80. J. Terven, D. Cordova-Esparza, A comprehensive review of YOLO: From YOLOv1 and beyond, (2023). <https://doi.org/10.48550/arXiv.2304.00501>
81. Z. Yang, B. He, Y. Liu, D. Wang, G. Zhu, Classification of rock fragments produced by tunnel boring machine using convolutional neural networks, *Autom. Constr.*, **125** (2021), 103612. <https://doi.org/10.1016/j.autcon.2021.103612>
82. L. E. Olivier, M. G. Maritz, I. K. Craig, Estimating ore particle size distribution using a deep convolutional neural network, *IFAC-PapersOnLine*, **53** (2020), 12038–12043. <https://doi.org/10.1016/j.ifacol.2020.12.740>
83. J. Chen, T. Yang, D. Zhang, H. Huang, Y. Tian, Deep learning based classification of rock structure of tunnel face, *Geosci. Front.*, **12** (2021), 395–404. <https://doi.org/10.1016/j.gsf.2020.04.003>
84. D. Bolya, C. Zhou, F. Xiao, Y. J. Lee, Yolact: Real-time instance segmentation, In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, 9156–9165. <https://doi.org/10.1109/ICCV.2019.00925>
85. T. Cheng, X. Wang, S. Chen, W. Zhang, Q. Zhang, C. Huang, et al., Sparse instance activation for real-time instance segmentation, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, 4433–4442. <https://doi.org/10.1109/CVPR52688.2022.00439>
86. C. Y. Wang, A. Bochkovskiy, H. Y. M. Liao, Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, 7464–7475.
87. M. Tan, Q. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, In: *International conference on machine learning*, PMLR, 2019, 6105–6114.
88. M. Tan, Q. Le, EfficientNetv2: Smaller models and faster training, In: *International Conference on Machine Learning*, PMLR, 2021, 10096–10106.
89. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, et al., MobileNets: Efficient convolutional neural networks for mobile vision applications, (2017). <https://doi.org/10.48550/arXiv.1704.04861>
90. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, 4510–4520. <https://doi.org/10.1109/CVPR.2018.00474>
91. A. Howard, M. Sandler, G. Chu, L. C. Chen, B. Chen, M. Tan, et al., Searching for MobileNetV3, In: *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, 1314–1324. <https://doi.org/10.1109/ICCV.2019.00140>
92. S. Mehta, M. Rastegari, MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer, (2022). <https://doi.org/10.48550/arXiv.2110.02178>

-
93. K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, In: *European conference on computer vision*, Springer, 2016, 630–645. https://doi.org/10.1007/978-3-319-46493-0_38
94. A. Supani, Y. Andriani, H. Indarto, Enhancing deeper layers with residual network on CNN architecture: A review, In: *Proceedings of the 6th FIRST 2022 International Conference (FIRST 2022)*, vol. 14, Springer Nature, 2023, 449. https://doi.org/10.2991/978-94-6463-118-0_46
95. H. Naveed, S. Anwar, M. Hayat, K. Javed, A. Mian, Survey: Image mixing and deleting for data augmentation, (2023). <https://doi.org/10.48550/arXiv.2106.07085> .
96. J. L. Ba, J. R. Kiros, G. E. Hinton, Layer normalization, (2016). <https://doi.org/10.48550/arXiv.1607.06450>
97. T. M. Hospedales, A. Antoniou, P. Micaelli, A. J. Storkey, Meta-learning in neural networks: A survey, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2022), 5149–5169. <https://doi.org/10.1109/TPAMI.2021.3079209>
98. C. J. Holder, M. Shafique, On efficient real-time semantic segmentation: A survey, (2022). <https://doi.org/10.48550/arXiv.2206.08605>
99. J. Xu, Z. Xiong, S. P. Bhattacharyya, PIDNet: A real-time semantic segmentation network inspired by PID controllers, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, 19529–19539.
100. J. Peng, Y. Liu, S. Tang, Y. Hao, L. Chu, G. Chen, et al., PP-LiteSeg: A superior real-time semantic segmentation model, (2022). <https://doi.org/10.48550/arXiv.2204.02681>
101. C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, et al., YOLOv6: A single-stage object detection framework for industrial applications, (2022). <https://doi.org/10.48550/arXiv.2209.02976>
102. C. Li, L. Li, Y. Geng, H. Jiang, M. Cheng, B. Zhang, et al., YOLOv6 v3.0: A full-scale reloading, (2023). <https://doi.org/10.48550/arXiv.2301.05586>
103. D. Bolya, C. Zhou, F. Xiao, Y. J. Lee, Yolact++ better real-time instance segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2022), 1108–1121. <https://doi.org/10.1109/TPAMI.2020.3014297> .
104. X. Wang, R. Zhang, T. Kong, L. Li, C. Shen, SOLOv2: Dynamic and fast instance segmentation, In: *Advances in Neural Information Processing Systems* (Eds. H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan and H. Lin), vol. 33, Curran Associates, Inc., 2020, 17721–17732.
105. X. Wang, T. Kong, C. Shen, Y. Jiang, L. Li, Solo: Segmenting objects by locations, In: *European Conference on Computer Vision*, Springer, 2020, 649–665. https://doi.org/10.1007/978-3-030-58523-5_38
106. J. He, P. Li, Y. Geng, X. Xie, FastInst: A simple query-based model for real-time instance segmentation, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, 23663–23672.
107. C. Lyu, W. Zhang, H. Huang, Y. Zhou, Y. Wang, Y. Liu, et al., RTMDet: An empirical study of designing real-time object detectors, (2022). <https://doi.org/10.48550/arXiv.2212.07784>

- 108.T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, M. Li, Bag of tricks for image classification with convolutional neural networks, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, 558–567. <https://doi.org/10.1109/CVPR.2019.00065>
- 109.B. Bischl, M. Binder, M. Lang, T. Pielok, J. Richter, S. Coors, et al., Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges, *WIREs Data Min. Knowl. Discov.*, **13** (2023), e1484. <https://doi.org/10.1002/widm.1484>
- 110.A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, et al., Segment Anything, (2023). <https://doi.org/10.48550/arXiv.2304.02643>
- 111.Z. Gu, H. Chen, Z. Xu, J. Lan, C. Meng, W. Wang, DiffusionInst: Diffusion model for instance segmentation, (2022). <https://doi.org/10.48550/arXiv.2212.02773>
- 112.T. Amit, T. Shaharbany, E. Nachmani, L. Wolf, SegDiff: Image segmentation with diffusion probabilistic models, (2022). <https://doi.org/10.48550/arXiv.2112.00390>
- 113.W. Wu, Y. Zhao, M. Z. Shou, H. Zhou, C. Shen, DiffuMask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models, (2023). <https://doi.org/10.48550/arXiv.2303.11681>
- 114.A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, K. Keutzer, A survey of quantization methods for efficient neural network inference, In: *Low-Power Computer Vision*, Chapman and Hall/CRC, 2022, 291–326.
- 115.T. Liang, J. Glossner, L. Wang, S. Shi, X. Zhang, Pruning and quantization for deep neural network acceleration: A survey, *Neurocomputing*, **461** (2021), 370–403. <https://doi.org/10.1016/j.neucom.2021.07.045>
- 116.M. Rakka, M. E. Fouda, P. Khargonekar, F. Kurdahi, Mixed-precision neural networks: A survey, (2022). <https://doi.org/10.48550/arXiv.2208.06064>
- 117.R. Tian, Z. Zhao, W. Liu, H. Liu, W. Mao, Z. Zhao, SAMP: A toolkit for model inference with self-adaptive mixed-precision, (2022). <https://doi.org/10.48550/arXiv.2209.09130>
- 118.NVIDIA Corporation: A PyTorch Extension: Tools for easy mixed precision and distributed training in Pytorch, 2022. Available from: <https://github.com/nvidia/apex>.
- 119.Linux Foundation: Automatic Mixed Precision package - torch.amp, 2022. Available from: <https://pytorch.org/docs/stable/amp.html>.
- 120.T. Honka, *Automatic Mixed Precision Quantization of Neural Networks using Iterative Correlation Coefficient Adaptation*, PhD thesis, Tampere University, (2021).
- 121.C. Pham-Quoc, X. Q. Nguyen, T. N. Thinh, Hardware/software co-design for convolutional neural networks acceleration: A survey and open issues, In: *Context-Aware Systems and Applications* (Eds. P. Cong Vinh, A. Rakib), Springer International Publishing, Cham, 2021, 164–178. https://doi.org/10.1007/978-3-030-93179-7_13
- 122.J. Gou, B. Yu, S. J. Maybank, D. Tao, Knowledge distillation: A survey, *Int. J. Comput. Vis.*, **129** (2021), 1789–1819. <https://doi.org/10.1007/s11263-021-01453-z>
- 123.W. Xu, Y. Zhang, X. Tang, Parallelizing DNN training on GPUs: Challenges and opportunities, In: *Companion Proceedings of the Web Conference 2021, WWW '21*, Association for Computing Machinery, New York, NY, USA, 2021, 174–178. <https://doi.org/10.1145/3442442.3452055>

124. Y. Fu, C. Aldrich, Online particle size analysis on conveyor belts with dense convolutional neural networks, *Miner. Eng.*, **193** (2023), 108019. <https://doi.org/10.1016/j.mineng.2023.108019>
125. H. Li, G. Asbjörnsson, M. Lindqvist, Image process of rock size distribution using dexined-based neural network, *Minerals*, **11** (2021), 736. <https://doi.org/10.3390/min11070736>
126. V. Suprunenko, Ore particles segmentation using deep learning methods, In: *Journal of Physics: Conference Series*, vol. 1679, IOP Publishing, 2020, 042089. <https://doi.org/10.1088/1742-6596/1679/4/042089>
127. X. Ma, P. Zhang, X. Man, L. Ou, A new belt ore image segmentation method based on the convolutional neural network and the image-processing technology, *Minerals*, **10** (2020), 1115. <https://doi.org/10.3390/min10121115>
128. Y. Zhan, G. Zhang, An improved OTSU algorithm using histogram accumulation moment for ore segmentation, *Symmetry*, **11** (2019), 431. <https://doi.org/10.3390/sym11030431>
129. Z. Tungol, Y. Kawamura, I. Kitahara, H. D. Jang, Development of a remote rock fragmentation size distribution measurement system for surface mines using 3D photogrammetry, In: *The 10th International Conference on Explosives and Blasting*, 2019.
130. Y. Wang, W. Tu, H. Li, Fragmentation calculation method for blast muck piles in open-pit copper mines based on three-dimensional laser point cloud data, *Int. J. Appl. Earth Obs. Geoinf.*, **100** (2021), 102338. <https://doi.org/10.1016/j.jag.2021.102338>
131. F. Bai, M. Fan, H. Yang, L. Dong, Image segmentation method for coal particle size distribution analysis, *Particuology*, **56** (2021), 163–170. <https://doi.org/10.1016/j.partic.2020.10.002>
132. C. Xie, H. Nguyen, X. N. Bui, Y. Choi, J. Zhou, T. Nguyen-Trang, Predicting rock size distribution in mine blasting using various novel soft computing models based on meta-heuristics and machine learning algorithms, *Geosci. Front.*, **12** (2021), 101108. <https://doi.org/10.1016/j.gsf.2020.11.005>
133. T. Vu, T. Bao, Development of a predictive model of rock fragmentation for Nui Phao open-pit mine in vietnam using multiple-output neural networks and Monte Carlo dropout technique, *Research Square*, <https://doi.org/10.21203/rs.3.rs-171960/v1>
134. N. Koteleva, S. Khokhlov, I. Frenkel, Digitalization in open-pit mining: A new approach in monitoring and control of rock fragmentation, *Appl. Sci.*, **11** (2021), 10848. <https://doi.org/10.3390/app112210848>
135. W. Chen, X. Li, H. He, L. Wang, A review of fine-scale land use and land cover classification in open-pit mining areas by remote sensing techniques, *Remote Sens.*, **10** (2017), 15. <https://doi.org/10.3390/rs10010015>
136. A. Sayadi, M. Monjezi, N. Talebi, M. Khandelwal, A comparative study on the application of various artificial neural networks to simultaneous prediction of rock fragmentation and backbreak, *J. Rock Mech. Geotech. Eng.*, **5** (2013), 318–324. <https://doi.org/10.1016/j.jrmge.2013.05.007>
137. F. I. Siddiqui, Measurement of size distribution of blasted rock using digital image processing, *Eng. Sci.*, **20** (2009), 81–93.
138. S. H. Cho, M. Nishi, M. Yamamoto, K. Kaneko, Fragment size distribution in blasting, *Mater. Trans.*, **44** (2003), 951–956. <https://doi.org/10.2320/matertrans.44.951>

- 139.M. J. Thurley, Automated image segmentation and analysis of rock piles in an open-pit mine, In: *2013 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, 2013, 1–8. <https://doi.org/10.1109/DICTA.2013.6691484>
- 140.W. D. Lei, K. Li, X. P. Wang, Improved watershed segmentation method in rock fragmentation analysis on digital photos, *Adv. Mater. Res.*, **261** (2011), 1734–1737. <https://doi.org/10.4028/www.scientific.net/AMR.261-263.1734>
- 141.W. Wang, Q. Li, D. Zhang, J. Fu, Image segmentation of adhesive ores based on MSBA-Unet and convex-hull defect detection, *Eng. Appl. Artif. Intell.*, **123** (2023), 106185. <https://doi.org/10.1016/j.engappai.2023.106185>
- 142.X. Tang, X. Wang, N. Yan, S. Fu, W. Xiong, Q. Liao, A new ore image segmentation method based on Swin-Unet, In: *2022 China Automation Congress (CAC)*, IEEE, 2022, 1681–1686. <https://doi.org/10.1109/CAC57257.2022.10055952>
- 143.W. Wang, C. Su, H. Zhang, Automatic segmentation of concrete aggregate using convolutional neural network, *Autom. Constr.*, **134** (2022), 104106. <https://doi.org/10.1016/j.autcon.2021.104106>
- 144.R. Amoako, A. Jha, S. Zhong, Rock fragmentation prediction using an artificial neural network and support vector regression hybrid approach, *Mining*, **2** (2022), 233–247. <https://doi.org/10.3390/mining2020013>
- 145.H. Li, X. Wang, C. Yang, W. Xiong, Ore image segmentation method based on GAN-UNet, *Control Theory Appl.*
- 146.Y. Liu, Z. Zhang, X. Liu, L. Wang, X. Xia, Ore image classification based on small deep learning model: Evaluation and optimization of model depth, model structure and data size, *Miner. Eng.*, **172** (2021), 107020. <https://doi.org/10.1016/j.mineng.2021.107020>
- 147.J. Duan, X. Liu, X. Wu, C. Mao, Detection and segmentation of iron ore green pellets in images using lightweight u-net deep learning network, *Neural Comput. Appl.*, **32** (2020), 5775–5790. <https://doi.org/10.1007/s00521-019-04045-8>
- 148.F. Sereshki, S. Hoseini, M. Ataei, Blast fragmentation analysis using image processing, *Int. J. Min. Geo-Eng.*, **50** (2016), 211–218. <http://dx.doi.org/10.22059/ijmge.2016.59831>
- 149.H. Yaghoobi, H. Mansouri, M. A. E. Farsangi, H. Nezamabadi-Pour, Determining the fragmented rock size distribution using textural feature extraction of images, *Powder Technol.*, **342** (2019), 630–641. <https://doi.org/10.1016/j.powtec.2018.10.006>
- 150.M. Firla, P. Lipnicki, D. Lewandowski, Image processing algorithm for the assessment of the ore fragmentation size distribution, In: *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, 2019, 505–512. <https://doi.org/10.1109/ETFA.2019.8869462>
- 151.M. Liancheng, Y. ZHANG, S. Guoqing, M. Zhen, L. Tianqi, Ore granularity detection and analysis system based on image processing, In: *2019 Chinese Control And Decision Conference (CCDC)*, IEEE, 2019, 359–366. <https://doi.org/10.1109/CCDC.2019.8832862>

- 152.H. Chen, Y. Jin, G. Li, B. Chu, Automated cement fragment image segmentation and distribution estimation via a holistically-nested convolutional network and morphological analysis, *Powder Technol.*, **339** (2018), 306–313. <https://doi.org/10.1016/j.powtec.2018.08.015>
- 153.R. Wang, W. Zhang, L. Shao, Research of ore particle size detection based on image processing, In: *Proceedings of 2017 Chinese Intelligent Systems Conference: Volume II*, Springer, 2018, 505–514. https://doi.org/10.1007/978-981-10-6499-9_48
- 154.X. Liu, C. Mao, W. Sun, X. Wu, Image-based method for measuring pellet size distribution in the stable area of disc pelletizer, *ISIJ Int.*, **58** (2018), 2088–2094. <https://doi.org/10.2355/isijinternational.ISIJINT-2018-384>
- 155.M. Heydari, R. Amirfattahi, B. Nazari, P. Rahimi, An industrial image processing-based approach for estimation of iron ore green pellet size distribution, *Powder Technol.*, **303** (2016), 260–268. <https://doi.org/10.1016/j.powtec.2016.09.020>
- 156.M. Roy, R. Paswan, M. Sarim, S. Kumar, R. R. Jha, P. Singh, Rock fragmentation by blasting – a review, *J. Mines Met. Fuels*, **64** (2016), 424–431.
- 157.E. Ebrahimi, M. Monjezi, M. R. Khalesi, D. J. Armaghani, Prediction and optimization of back-break and rock fragmentation using an artificial neural network and a bee colony algorithm, *Bull. Eng. Geol. Environ.*, **75** (2016), 27–36. <https://doi.org/10.1007/s10064-015-0720-2>
- 158.I. Enayatollahi, A. Aghajani Bazzazi, A. Asadi, Comparison between neural networks and multiple regression analysis to predict rock fragmentation in open-pit mines, *Rock Mech. Rock Eng.*, **47** (2014), 799–807. <https://doi.org/10.1007/s00603-013-0415-6>
- 159.F. Faramarzi, H. Mansouri, M. E. Farsangi, A rock engineering systems based model to predict rock fragmentation by blasting, *Int. J. Rock Mech. Min. Sci.*, **60** (2013), 82–94. <https://doi.org/10.1016/j.ijrmms.2012.12.045>
- 160.X. Shi, Z. Jian, B. Wu, D. Huang, W. Wei, Support vector machines approach to mean particle size of rock fragmentation due to bench blasting prediction, *Trans. Nonferrous Met. Soc. China*, **22** (2012), 432–441. [https://doi.org/10.1016/S1003-6326\(11\)61195-3](https://doi.org/10.1016/S1003-6326(11)61195-3)
- 161.M. J. Thurley, Automated online measurement of limestone particle size distributions using 3D range data, *J. Process Control*, **21** (2011), 254–262. <https://doi.org/10.1016/j.jprocont.2010.11.011>
- 162.G. Zhang, G. Liu, H. Zhu, Segmentation algorithm of complex ore images based on templates transformation and reconstruction, *Int. J. Miner. Metall. Mater.*, **18** (2011), 385. <https://doi.org/10.1007/s12613-011-0451-8>
- 163.A. Amankwah, C. Aldrich, Rock image segmentation using watershed with shape markers, In: *2010 IEEE 39th Applied Imagery Pattern Recognition Workshop (AIPR)*, 2010, 1–7. <http://doi.org/10.1109/AIPR.2010.5759719>
- 164.D. P. Mukherjee, Y. Potapovich, I. Levner, H. Zhang, Ore image segmentation by learning image and shape features, *Pattern Recognit. Lett.*, **30** (2009), 615–622. <https://doi.org/10.1016/j.patrec.2008.12.015>
- 165.M. Monjezi, M. Rezaei, A. Y. Varjani, Prediction of rock fragmentation due to blasting in Gol-E-Gohar iron mine using fuzzy logic, *Int. J. Rock Mech. Min. Sci.*, **46** (2009), 1273–1280. <https://doi.org/10.1016/j.ijrmms.2009.05.005>

-
- 166.W. Wang, Rock particle image segmentation and systems, In: *Pattern recognition techniques, technology and applications*, I-Tech Vienna, Austria, 2008, 197–226.
- 167.M. J. Thurley, K. C. Ng, Identifying, visualizing, and comparing regions in irregularly spaced 3d surface data, *Comput. Vis. Image Underst.*, **98** (2005), 239–270. <https://doi.org/10.1016/j.cviu.2003.12.002>
- 168.C. Perez, A. Casali, G. Gonzalez, G. Vallebuona, R. Vargas, Lithological composition sensor based on digital image feature extraction, genetic selection of features and neural classification, In: *Proceedings 1999 International Conference on Information Intelligence and Systems (Cat. No. PR00446)*, IEEE, 1999, 236–241. <https://doi.org/10.1109/ICIIS.1999.810267>
- 169.N. H. Maerz, Image sampling techniques and requirements for automated image analysis of rock fragmentation, In: *Measurement of Blast Fragmentation*, Routledge, 2018, 115–120.



AIMS Press

© 2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)