



Detection and segmentation of iron ore green pellets in images using lightweight U-net deep learning network

Jiaxu Duan¹ · Xiaoyan Liu^{1,2} · Xin Wu¹ · Chuangang Mao¹

Received: 28 August 2018 / Accepted: 16 January 2019 / Published online: 24 January 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract

In steel manufacturing industry, powdered iron ore is agglomerated in a pelletizing disk to form iron ore green pellets. The agglomeration process is usually monitored using a camera. As pellet size distribution is one of the major measures of product quality monitoring, pellets detection and segmentation from the image are the key steps to determine the pellet size. Traditional image processing algorithms are not only challenged by the complicated constitution of pellets, sediment and residuals in the image, but also by the harsh and unbalanced light reflection on the pellet centrum area and the background which results in tedious parameter adjustment work and poor performance. To solve these problems, we design a lightweight U-net deep learning network to automatically detect pellets from images and to obtain the probability maps of pellet contours. Compared to classic U-net, the proposed network has fewer parameters and introduces batch normalization layers, which greatly reduces the computing time and improves generalization ability of the network. A concentric circle model is then used to separate clumped contours of the pellets, and the pellets shapes are detected via ellipse fitting. The proposed method is verified using images captured from an industrial pelletizing disk, and its performance is compared with traditional methods and the classic U-net. Results show that the proposed method achieves better segmentation performance in DICE and ROC indexes and shows good robustness to uneven illumination. Tests on temporal image sequences demonstrate that the proposed method is effective in monitoring the pellet size distribution and the pellet shape as well. Results of this work have potential usage in online detection of iron ore green pellets and other types of particles.

Keywords Iron ore pellets · U-net deep neural network · Image segmentation · Particle size distribution

1 Introduction

Agglomeration is a common process in steel manufacturing industry. Fine iron ore is fed into a disk pelletizer and sprayed by water. With continuous rotation of the disk, fine particles are gathered and agglomerated into larger pellets, falling out of the disk as green pellets. The pellet size distribution (PSD) should be monitored to guarantee that it is in a desired size range of 9–14 mm [1].

Manual measurement of PSD has many limitations. For instance, the pellet samples are limited and the sampling

and sieving process are rather time consuming. Human error also has a significant impact on the measuring results [2]. Optical imaging is a good method for online monitoring of pellet size [3–5]. It has advantages in simple hardware configuration, wide view field and abundant choice of image processing algorithms. A typical optical imaging system is shown in Fig. 1. The images of the pellets are usually taken by a camera and then, processed to detect individual pellets and to calculate the pellet size.

Traditional image processing framework for pellet size measurement includes filtering, thresholding, clustering and segmentation. Pellet detection and segmentation are crucial prerequisites in image processing procedures, the performance of which greatly affects the measuring accuracy of pellet size [6–8]. However, the noisy background, uneven illumination, harsh light reflectance at the pellet centers and pellet overlapping are the key problems in image segmentation. Traditional algorithms, like

✉ Xiaoyan Liu
xiaoyan.liu@hnu.edu.cn

¹ College of Electrical and Information Engineering, Hunan University, Changsha 410082, China

² Hunan Key Laboratory of Intelligent Robot Technology in Electronic Manufacturing, Changsha 410082, China

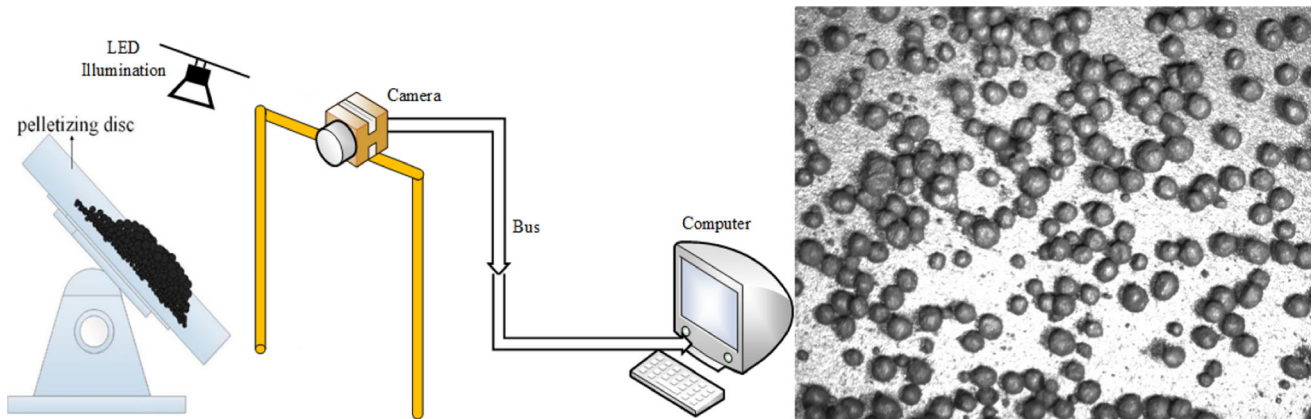


Fig. 1 Illustration of the optical imaging system (left). An image taken at the discharge of the disk pelletizer (right)

watershed-based algorithms [9, 10] and multi-threshold segmentation algorithm [11], are limited in solving such problems. For instance, watershed may result in oversegmentation, meanwhile, multi-threshold algorithms require redundant setting of thresholds to achieve satisfactory results and it may be turned out to be tedious works to adapt to the local pixel value distribution [12–14]. Furthermore, the overlapping problem also plays a key role that influences the segmentation performance. Lu et al. [15] developed a technique based on combining wavelet transform and Fuzzy C-means clustering (FCM) for particle image segmentation. This method was feasible for separating overlapping particles. Zhang et al. [16] proposed an effective approach for particle segmentation based on combining the background difference method and the graph cut-based local threshold method. Building the background model and performing subtraction between particle image and background model are performed to eliminate the droplets, while the local threshold method is performed further to eliminate the influences of particle shadow. FogBank [17] is also a marker-controlled watershed-based algorithm for pellet segmentation, which combines the features of pixel value histogram and distance transform. However, these algorithms are more or less limited in processing images with uneven illumination.

Deep neural network could be a powerful solution to solve the above-mentioned problems in image segmentation [18], because it has been successfully applied in biological and medical field for segmentation of cells [19], which has many similarities with segmentation of iron ore pellets. In particular, U-net architecture is characterized with its relatively lower requirement of the number of training data, while some other deep learning architecture has more or less limitations related to the training. The segmentation of granule-shaped edges is easy to be realized by using such networks under partially or heavily overlapping conditions. Plissiti and Nikou [20] presented a

method for the segmentation of overlapping nuclei, which combines local characteristics of the nuclei boundary and a priori knowledge about the expected shape of the nuclei. Fabijanska [21] used a U-net-based convolutional neural network for cell segmentation. Particularly, the network was trained to discriminate pixels located at the borders between cells. The edge probability map outputted by the network was next binarized and skeletonized in order to obtain one-pixel wide edges. Park et al. [22] also presented a U-net-based autoencoder deep neural network for the analysis of particle interaction images, which was an innovative attempt to adopt deep learning into industrial applications. This work was boosted by higher parameter efficiency and being able to get rid of the harsh background influences. Artificial neural networks such as backpropagation networks (BPNN) are also applied in the segmentation of shot-peened areas which are presented in round-like shape objects segmentation in industrial computer vision [23] to overcome the inaccurate and incorrect object segmentation problems in traditional techniques.

Inspired by the successful application of deep learning algorithm in biological and medical field for cell segmentation, we propose a lightweight U-net deep neural network in the present work for detection and segmentation of iron ore green pellets in order to overcome the limitations of traditional image segmentation algorithms. The features of the proposed method are listed as follows:

- (1) The proposed deep learning framework is a kind of lightweight U-net, with much less parameters than the classic U-net. Its requirement for GPU memory is dramatically reduced and therefore is more suitable for practical usage in industries where online processing is required;
- (2) Batch normalization (BN) layers are introduced after the deconvolution layer in each unit in the decoder part, so that the generalization ability of the proposed network can be improved.

- (3) A concentric circle model is proposed to automatically detect the centers and the respective surrounded contours of the pellets. With this model, it is convenient to perform ellipse fitting and statistical analysis of physical properties of the pellets.

This paper is organized as follows: The basic idea and the pipeline of the proposed method are given in Sect. 2; the experimental results and performance analysis of the proposed method are given in Sect. 3. The main conclusions are summarized in Sect. 4.

2 Method

The flowchart of the proposed method is shown in Fig. 2. It contains three main steps: (1) pellets detection using a U-net-based deep neural network; (2) separation of clumped pellet contours by a concentric circle model; (3) ellipse fitting of the pellet contour and statistical analysis of the pellet size distribution. Each step will be described in detail in the following sections.

2.1 Architecture of the U-net-based deep neural network

The U-net exhibits the encoder–decoder architecture where the encoder gradually reduces the input data spatial dimension, while the decoder gradually recovers it [24]. Here, we design an architecture which contains nine layers

(Fig. 3). A rectified linear (ReLU) function S is selected as the activation function after each convolutional layer:

$$A(x) = \max(0, x) \tag{1}$$

where x denotes the input of the ReLU function.

The size of convolutional kernel is 3×3 , meanwhile, the maximum pooling layers with the window size of 2×2 with 2-pixel stride for downsampling are deployed after each even convolutional layer. Additionally, the number of feature channels is doubled after pooling layers and reduced by half in the procedure of encoder and decoder, respectively. In the process of image propagation through this pipeline, each pixel is classified separately as edge or non-edge with a certain probability by using the Softmax classifier.

Compared with classic U-net, the proposed network introduces the batch normalization layers at each end of the fully connected convolutional units. This feature could improve the generalization ability of network in processing images, meanwhile, batch processing improves the speed in adapting the weights model to all images in the training dataset. Moreover, the proposed network is a lightweight deep neural network, with much less parameters than the classic U-net. Its requirement for GPU memory is dramatically reduced and therefore is more suitable for practical usage in industries where online processing is required.

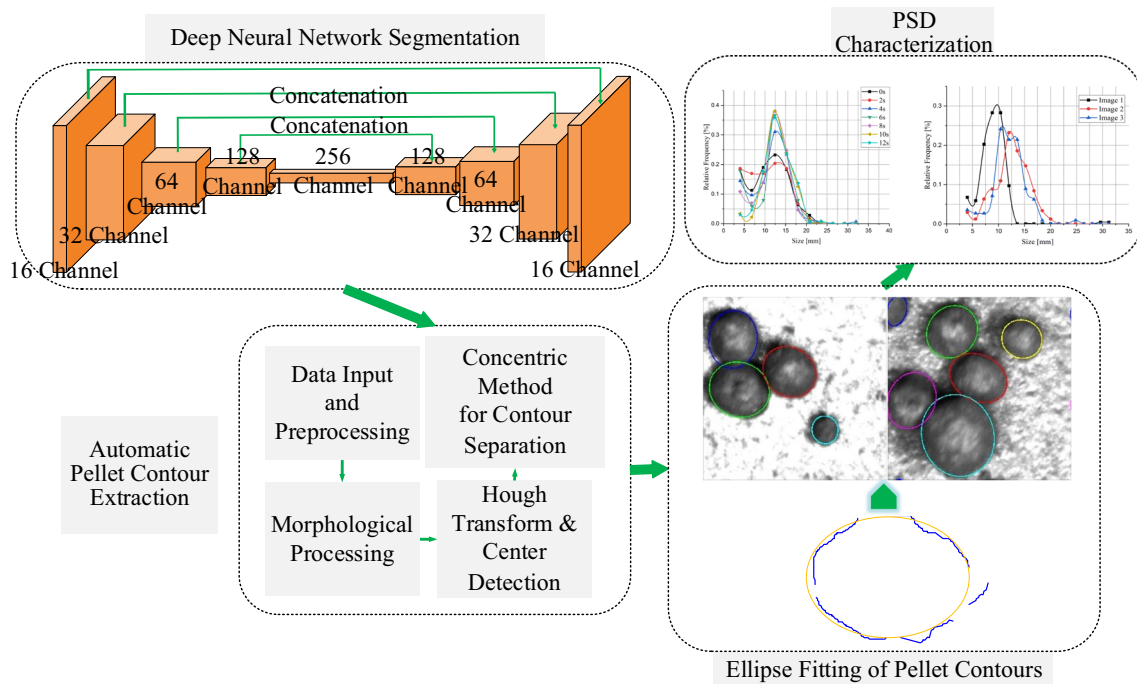


Fig. 2 Flowchart of the proposed method

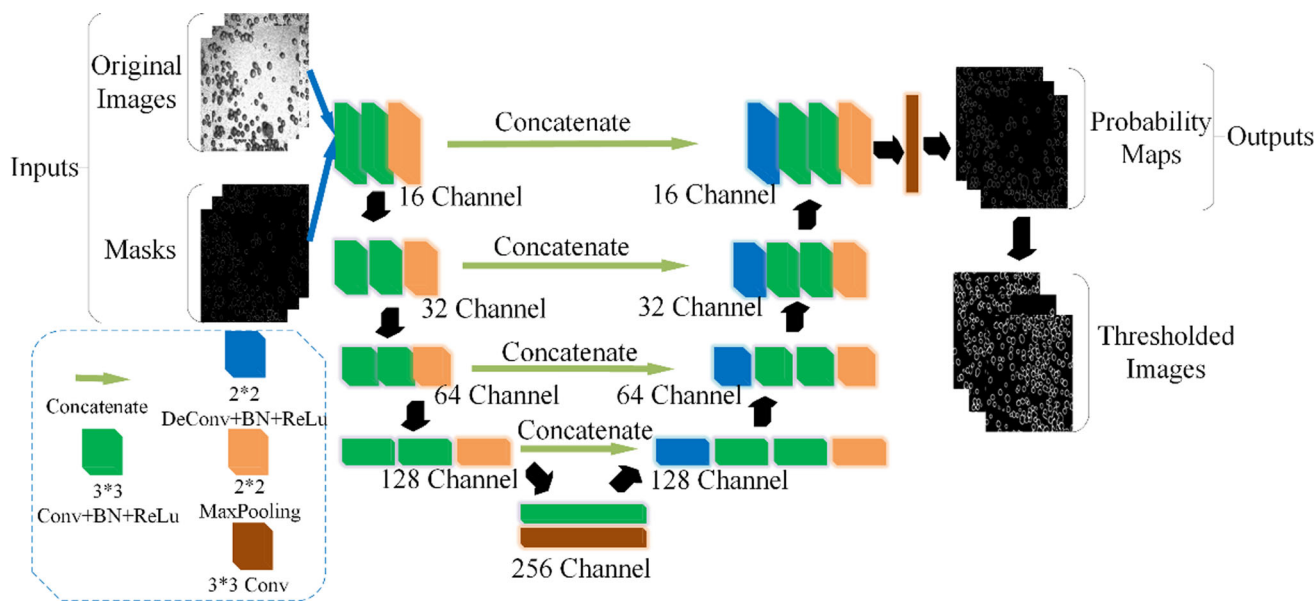


Fig. 3 Architecture of the U-net-based neural network

2.2 Network training

After the procedures as mentioned above, the probability map of the pellet edges can be obtained. More specifically, the U-net architecture is trained with its input of the original pellet image and the corresponding mask image. For the labeling work, the masks of the raw image dataset are labeled as ground truth by manual operations, while the objects and the background are marked with different colors to stand for the respective classes.

We captured 500 images with the size of 1280×1024 using the optical imaging system shown in Fig. 1. The number of images for training is 352, and the remaining images are used as testing dataset. The original image data with full size and the image patches cropped from the full-size images are used as the input data with respect to the size of 1280×1024 and 256×256 , respectively. The full-size images are used for pellet size distribution (PSD) statistics, while the cropped patches are used to test the segmentation performance of our deep neural network. The network was trained for 400 epochs with the batch size of 8 to achieve the minimum loss between the probability map and its ground truth. In particular, to train the networks, we employed weighted cross-entropy function as the loss function in training procedure.

$$\text{Loss}(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^2 w_i^c y_i^c \log \hat{y}_i^c \tag{2}$$

where \hat{y}_i^c denotes the probability of pixel i belonging to class c (background), w_i^c denotes the weight and y_i^c indicates the ground truth label for pixel i .

The U-net-based architecture used here is characterized by achieving a good segmentation performance after training with a small number of training data. For consideration of time efficiency, thereafter, this network is trained with 20 full-size images and 400 epochs. Then, it comes to a reasonable edge probability map of all the images with full size in the test dataset.

2.3 Prediction of the pellet edge

The prediction of the edge is shown in the form of the probability map. The transposed convolution of stride 2 is performed in two phases. On one hand, it is able to provide the probability map of each convolutional computation followed the convolutional layers. On the other hand, adopting transposed convolution ensures the dimension balance in the propagation through the network by addressing two-dimensional upsampling. Finally, the probability map of the binary segmentation task is generated from the output layer consisting of two units, and the activation values are fed into a binary Softmax function that is converted into probability distributions over the class labels. Suppose that o_k is the k -th output of the network for a given input, the probability p_k assigned to the k -th class can be calculated as the output of the Softmax function

$$p_k = \exp(o_k) / \sum_{h \in \{0,1\}} \exp(o_h) \tag{3}$$

where $k = 0$ and $k = 1$ represent non-nodule and nodule pixels, respectively.

By referring to overlapping instances problem [25], weighted cross entropy shown in Eq. (4) is used to emphasize learning the edges of pellets, and to force the network to learn the small separated edges among the touching pellets. This method helps to distinguish overlapping instances. The basic idea is to weight edges more and to push network toward learning gaps between close pellets. The separation edge is computed using morphological operations. The weight map is then computed as in Eq. (5).

$$E = \sum_{x \in \Omega} w(x) \log(p_{\ell(x)}(x)) \tag{4}$$

$$w(x) = w_c(x) + w_0 \cdot \exp\left(-\frac{(d_1(x) + d_2(x))^2}{2\sigma^2}\right) \tag{5}$$

where $\Omega \Rightarrow \mathbb{R}$ is the weight map to balance the class frequencies, $d_1 : \Omega \Rightarrow \mathbb{R}$ denotes the distance to the edge of the nearest pellet and $d_2 : \Omega \Rightarrow \mathbb{R}$ denotes the distance to the edge of the second nearest pellet. $w(x)$ denotes the weights function with the input x ; $w_c(x)$ denotes the class-weights function which belongs to the class c ; and w_0 is a constant value but not a function.

2.4 Contour detection using ellipse fitting

By use of the U-net-based network described above, the probability maps of the pellet edges can be obtained, based on which the pellet contour can be extracted using ellipse fitting and the pellet size can be analyzed.

Hereby, an ellipse observed in an image is described in terms of a quadratic polynomial equation shown in Eq. (6) [26].

$$Ax^2 + 2Bxy + Cy^2 + 2f_0(Dx + Ey) + f_0^2F = 0 \tag{6}$$

where f_0 is the scale factor and a constant number for adjusting the scale. Theoretically, we can let it be 1, however, for finite-length numerical computation, f_0 should be so chosen that x/f_0 and y/f_0 have approximately the order of 1, so that the numerical accuracy can be increased and loss of significant numbers be avoided. In view of this, we take the origin of the image X – Y coordinate system at the center of the image, rather than the upper-left corner as it is conventionally done, and then take f_0 as the square error to denote the scale factor of the ellipses needed to be fitted.

2.4.1 Automatic pellet contour extraction

Before edge fitting step, some essential preprocessing steps in edge fitting by ellipse polynomial are only effective for the points distributed as single spherical shape. Thus, we design a framework to extract the pellet contour one by one

from a cluster of pellet contours in one image. The flowchart shown in Fig. 4 represents such a pipeline.

2.4.2 Detection of the pellet center

Firstly, we mark the background and the pellets with white and black color, which indicates the two targets to be classified. Blurring is optional, but useful to reduce high frequency noise to make our contour detection process more accurate. Moreover, morphological processing such as image dilation and erosion is also a strategy used as a preprocess to improve the performance for the next steps. Secondly, distance transform and complementary distance transform [27] are applied to highlight the centrum region of the pellet area in the binarized image. Let \mathfrak{A} be a regular grid, and $f : \mathfrak{A} \rightarrow \mathbb{R}$ is a function on the grid. The mirror function D_f of distance transform is

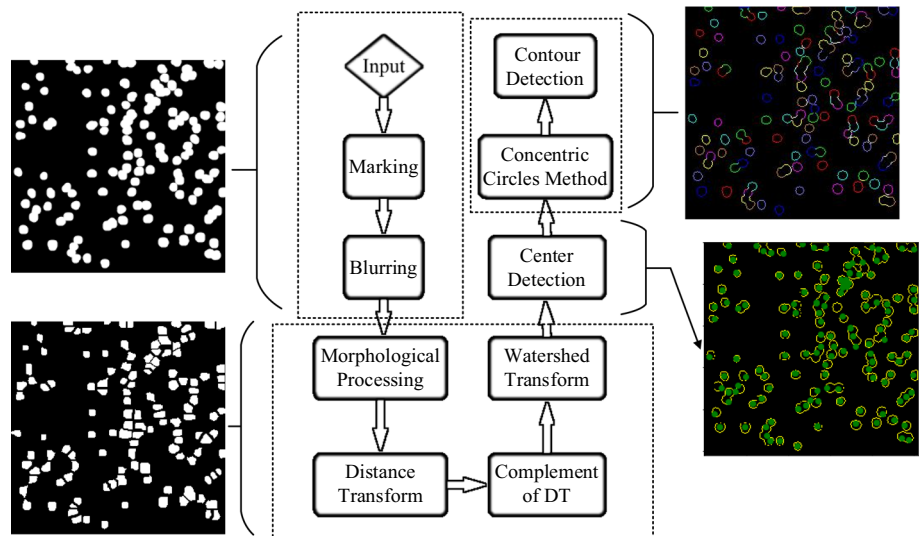
$$D_f(p) = \min_{q \in \mathfrak{A}} (d(p, q) + f(q)) \tag{7}$$

where $d(p, q)$ is some measure of the distance between p and q . Intuitively, for each point p , we find a point q that is close to p and for which $f(q)$ is small. Note that if the distance transform f has a small value at some location, D_f will have small value at that location and any nearby point, where nearness is measured by the distance $d(p, q)$.

In this way, we can make sure whatever regions of pellets are identified as really pellets. The remaining regions are pellets or background that can be determined by use of watershed algorithm [28]. An algorithmic definition of the watershed transform by simulated immersion was given by Vincent and Soille [29, 30]. The watershed transform is the method for image segmentation in the field of mathematical morphology. Imagine that the landscape is immersed in a lake with holes pierced in local minima. Basins will fill-up with water starting at these local minima, and, at points where water coming from different basins would meet, dams are built. When the water level has reached the highest peak in the landscape, the process is stopped. As a result, the landscape is partitioned into regions or basins separated by dams, called watershed lines or simply watersheds. In the present work, the discrete watershed transform proposed by Roerdink and Meijster [28] is applied to detect the pellet center, as described in Eqs. (8) and (9).

Let $f : D \rightarrow \mathbb{N}$ be a digital gray value image, and h_{\min} and h_{\max} is the minimum and maximum value of f , respectively. Define a recursion with the gray level h increasing from h_{\min} to h_{\max} , in which the basins associated with the minima of f are successively expanded. Let X_h denote the union of the set of basins computed at level h and MIN_h denotes the union of all regional minima at altitude level h . A connected component of the threshold

Fig. 4 Flowchart of automatic contour detection



set T_{h+1} at level $h + 1$ can be either a new minimum, or an extension of a basin in X_h , resulting in an updated value X_{h+1}

$$\begin{cases} X_{h_{\min}} = \{p \in D | f(p) = h_{\min}\} = T_{h_{\min}} \\ X_{h+1} = \text{MIN}_{h+1} \cup \text{IZ}_{T_{h+1}}(X_h), \quad h \in [h_{\min}, h_{\max}] \end{cases} \quad (8)$$

The watershed $W_{\text{shed}}(f)$ of f is the component of $X_{h_{\max}}$ in D :

$$W_{\text{shed}}(f) = D / X_{h_{\max}} \quad (9)$$

After the above processing, the contour circled area is split up. A novel contour-based object detector using Hough transform at each separated area is then proposed in the present work, in which each local part casts a vote for the possible locations of the object center. To be specific, the image I is described by use of polar parametrization (θ, ρ) , where ρ is the distance between the line and the origin and θ is the angle made by the normal to the line with the x -axis. If there is a line passing through point p , then we have

$$\theta_p = \arg \nabla I \quad (10)$$

$$\rho_p = \frac{|p \cdot \nabla I|}{\|\nabla I\|} \quad (11)$$

where θ_p denotes the direction of the gradient vector and ρ_p is the distance between the origin and the line passing through p and perpendicular to the gradient vector. The magnitude of gradient is $\|\nabla I\| = \sqrt{I_x^2 + I_y^2}$.

Assume that $C \in \mathbb{R}^2$ is the center and r is the radius of the circle. If there is a circle passing through point p , then we have

$$r_p = \frac{1}{|\kappa_p|} \quad (12)$$

$$pC_p = \frac{\nabla I}{\kappa_p \|\nabla I\|} \quad (13)$$

$$\kappa = -\frac{I_{xx}I_y^2 - 2I_{xy}I_xI_y + I_{yy}I_x^2}{\|\nabla I\|^3} \quad (14)$$

where the radius r_p is the inverse of the absolute curvature κ_p calculated at point p using Eq. (12). The center C_p is obtained by tracing from the vector p . The magnitude of p corresponds to r_p , and its direction depends on the sign of curvature and is the same as the gradient.

Under the concept of topological hierarchy-contour tracing, the separate contours of each pellet area are extracted successfully by detecting the outermost contour of the spherical pellets area and establishing the complete hierarchical tree where all the contours contain a list of contours surrounded by the contour directly [31–33]. 3D histogram is used in order to vote the matched feature in parameter space. As shown in Fig. 5, the vector $[x, y, \alpha]$ represents the position and the pose angle in the spatial coordinate. The higher the degree of matching, the higher vote at some bins. Consequently, the extreme voting value represents the position and pose of the center.

2.4.3 Contour exploring

The contours usually surround the respective centers, therefore, we design a concentric circle model to explore the position of the points that form the contour (see Fig. 6). In this figure, the black curve is the contour of one iron pellet, the white arrows indicate the radius to cover the circular area that is labeled by different colors. The white arrow grows with a certain length after each traversing of its covered area. During the whole exploration procedure, according to the shape and size of the detected pellets, we

Fig. 5 Voting scheme

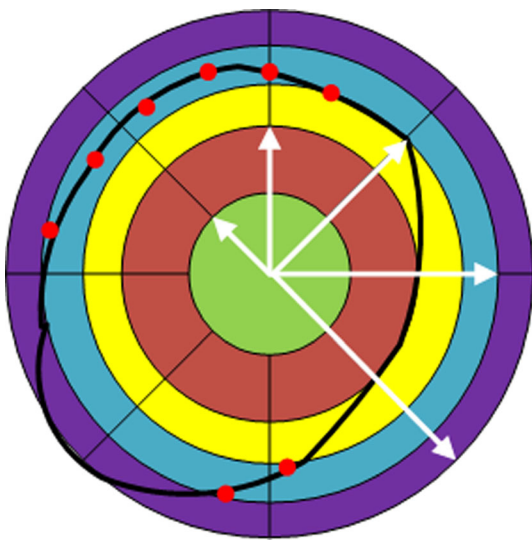
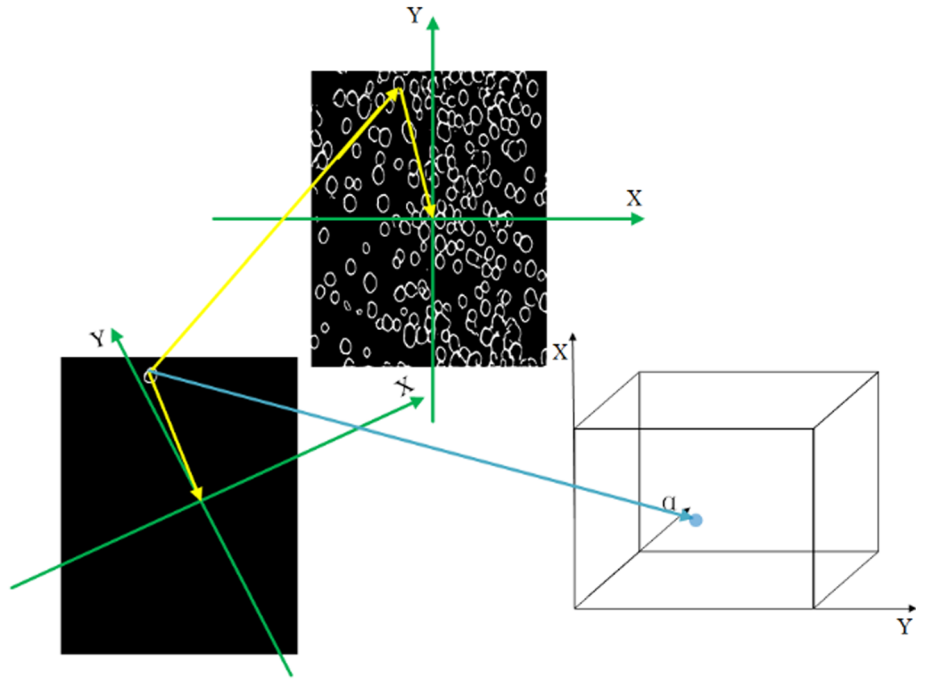


Fig. 6 Schematic diagram of concentric circle model

set the explore radius to increase from 1 to 20 gradually. The distance between two nearby pellet centers varies from 1 to 11, and the threshold value of the number of detected points on one pellet contour is set to 50. The exploration will be terminated when the number of detected points reaches this threshold value.

2.4.4 Ellipse fitting

The basic concept of curve fitting is to fit the pellet discrete edge points by using ellipse function. Given the coordinate

value (x_i, y_i) of each point, Eq. (6) can be converted into matrix form:

$$f(x, y) = Ax^2 + 2Bxy + Cy^2 + 2f_0(Dx + Ey) + f_0^2F = X^T CX$$

$$= \begin{bmatrix} x^2 & 2xy & y^2 & 2x & 2y & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ Df_0 \\ Ef_0 \\ f_0^2F \end{bmatrix} = 0 \tag{15}$$

$$X = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad P = \begin{bmatrix} A & B & Df_0 \\ B & C & Ef_0 \\ Df_0 & Ef_0 & f_0^2F \end{bmatrix} \tag{16}$$

where X represents the homogeneous coordinate of points and P represents the coefficient matrix of ellipse.

If there are N points to be fitted, we can obtain ellipse function similar to Eqs. (15)–(16)

$$D = \|A\|P = A^T P A P^T \tag{17}$$

$$\|P\| = P^T P, \quad P = \begin{bmatrix} A \\ B \\ C \\ Df_0 \\ Ef_0 \\ f_0^2F \end{bmatrix} \tag{18}$$

where P represents the ellipse coefficients. The parameters vector P can be estimated by using singular value

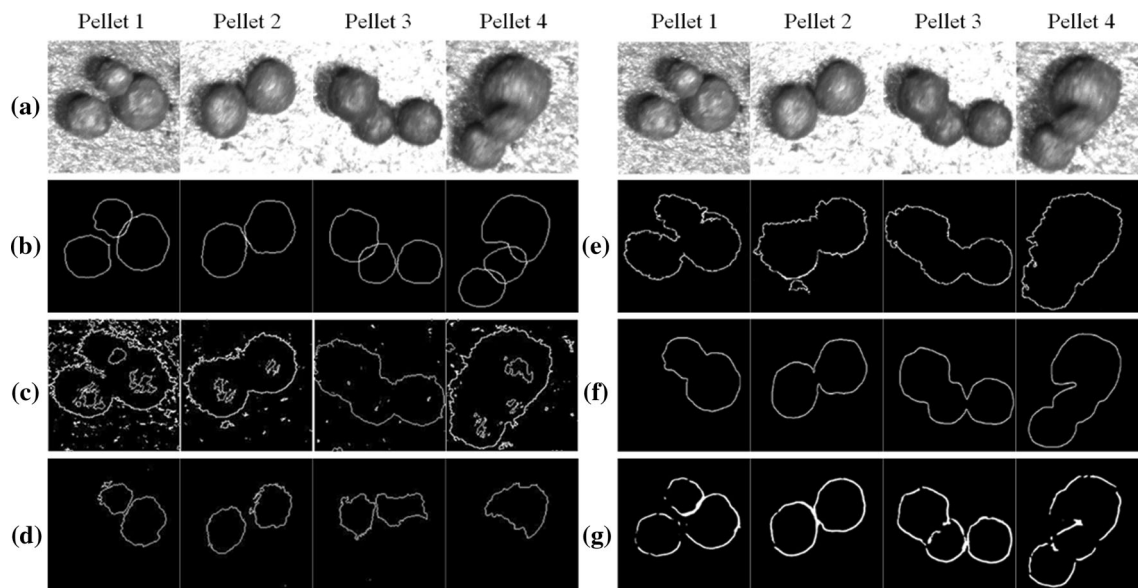


Fig. 7 Segmentation results of test pellets. **a** Original image of the pellet patches. **b** Ground Truth. **c** Segmentation results using mark-controlled watershed. **d** Seed-point segmentation results. **e** FogBank

segmentation results. **f** Segmentation results using classic U-net. **g** Segmentation results using the proposed network

Table 1 DICE index of the pellet patches from Pellets 1 to Pellets 4 using different segmentation methods

	Pellets 1	Pellets 2	Pellets 3	Pellets 4
Mark-controlled watershed	0.0795	0.3811	0.4697	0.1304
Seed-point segmentation	0.1313	0.4672	0.5008	0.0787
FogBank segmentation	0.5105	0.47	0.5202	0.4709
Classic U-net	0.6035	0.9397	0.9277	0.9396
Proposed method	0.8507	0.8785	0.9016	0.9260

decomposition or Eigenvalue decomposition. But, this estimation algorithm does not take use of the ellipse constraint condition

$$\Delta = \det \begin{pmatrix} A & B \\ B & C \end{pmatrix} > 0 \quad (19)$$

Consequently, the fitting result may be hyperbola or parabola when the noise exists, and the ellipse is unstable. To solve this problem, the direct least square fitting of ellipses algorithm proposed by Takashimizu and Iiyoshi [34] is applied in the present work, with which the fitting results can be forced to be ellipses when the noise exists.

3 Experiment and analysis

3.1 Data acquisition

The equipment and imaging system used in the present work are shown schematically in Fig. 1. The disk pelletizer is located in a steel company. It has a diameter of 6 meters and an inclination angle of 45 degree, operated with a

rotation speed of 8 rpm. An industrial camera (Baumer VCXG-53M with a lens 50 mm, resolution 2048×2592) is used to capture image of the pellets that falling directly from the outlet of the disk. The images are sampled with an interval of 2 s. Although LED lamps are used to improve illumination, there still exists problem of uneven illumination in the captured images due to the inclination and fast rotation of the pelletizing disk. The light reflection near the pellet center is much stronger than other areas, and there exist shadows in overlapping pellets. Detection of pellets from such images is a big challenge in image processing. Those images will be used to test the proposed method described in Sect. 2.

3.2 Evaluation benchmarks

A quantitative assessment of the proposed method is performed twofold. Firstly, the segmentation performance of the trained CNN for recognizing the pellets contours was evaluated by means of a ROC curve and the corresponding area under the curve (AUC) [35]. Denote the mapping function of the position p in segmentation results and

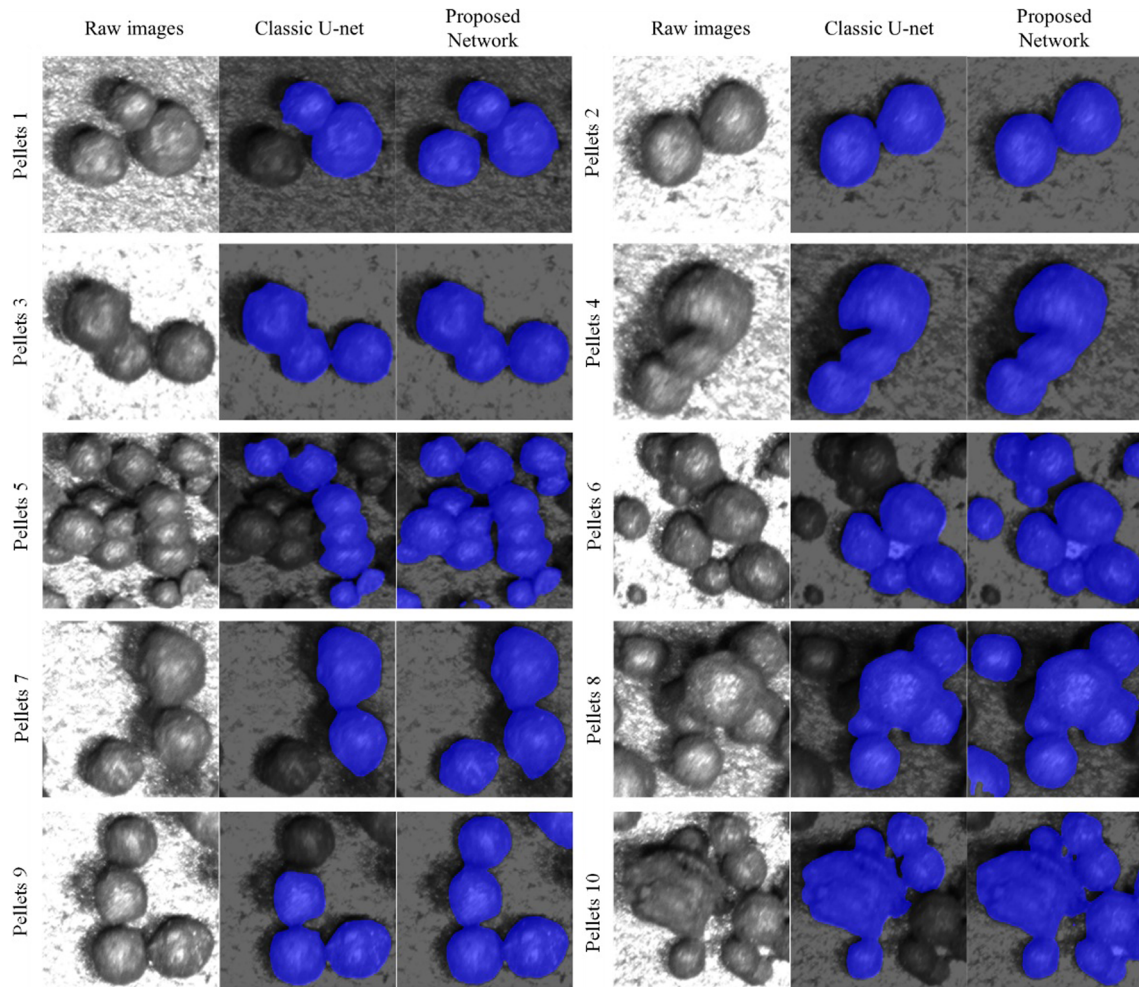


Fig. 8 Suspending mask of segmentation results generated by proposed network and classic U-net. The blue mask denotes the detected part and the remaining part denotes the area that are not detected (color figure online)

Table 2 DICE index of the pellet patches from pellets 1 to pellets 10 using different deep neural networks

	Pellets 1	Pellets 2	Pellets 3	Pellets 4	Pellets 5	Pellets 6	Pellets 7	Pellets 8	Pellets 9	Pellets 10
Classic U-net	0.6035	0.9397	0.9277	0.9396	0.6441	0.7547	0.7667	0.8474	0.8031	0.8125
Proposed network	0.8507	0.8785	0.9016	0.9260	0.9391	0.9714	0.9537	0.9607	0.9691	0.9688

ground truth as $F(p)$ and $G(p)$, respectively. The ROC curve function and AUC are

$$ROC = G\{F^{-1}(p)\} \tag{20}$$

$$AUC = \int_{p \in \Omega} ROC(p) dp = \int_{p \in \Omega} G\{F^{-1}(p)\} dp \tag{21}$$

where $\Omega \Rightarrow \mathbb{R}^2$ stands for the two-dimensional space of the image data. Secondly, the segmentation performance of the proposed method is compared with traditional methods and

classic U-net in the aspects of Sørensen-DICE coefficient. The mathematical expression of DICE is

$$DICE = \frac{2|\varepsilon \cap \vartheta|}{|\varepsilon| + |\vartheta|} \tag{22}$$

where ε is the level of alignment between the pellet contours, and ϑ is the ground truth.

Additionally, training loss and Intersection of Union (IoU) are given to describe the training process of this deep neural network. Intersection over Union is an evaluation

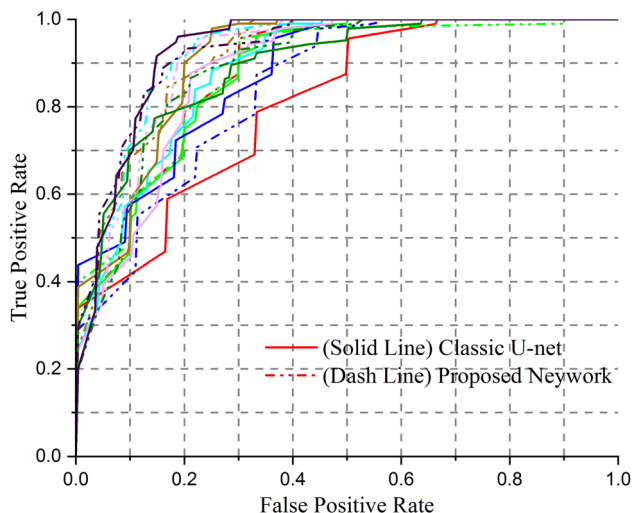


Fig. 9 ROC curve of the proposed network and the classic U-net for ten testing pellet patches (represented by different colors) (color figure online)

metric used to measure the accuracy of an object detector on a particular dataset, which is expressed as

$$IoU = \frac{A \cap B}{A \cup B} \tag{23}$$

where A and B denote the pellet contours in thresholded probability map and ground truth, respectively.

3.3 Experimental results and analysis

A number of patches cropped from the original full-size images are used to examine the performance of the proposed architecture. In Fig. 7, we select four pellet patches with uneven light illumination as example to illustrate the training procedure, and the segmentation results are compared with traditional methods (mark-controlled watershed segmentation, seed-point segmentation method, FogBank segmentation method) and classic U-net segmentation. DICE index (Table 1) is used to evaluate quantitatively the segmentation performance of different methods. It can be seen that the segmentation results generated by the traditional methods (see Fig. 7c–e) are strongly affected by the uneven illumination and harsh light reflection, resulting in lower values of DICE (lower than 0.55). In contrast, the performance of the classic U-net (Fig. 7f) and our proposed

network (Fig. 7g) is comparable for Pellets 2–Pellets 4 (DICE \approx 0.9) and both outperform the traditional methods to a great degree. For Pellets 1, the DICE of the proposed network is much higher than that of the classic U-net (0.8507 vs. 0.6035).

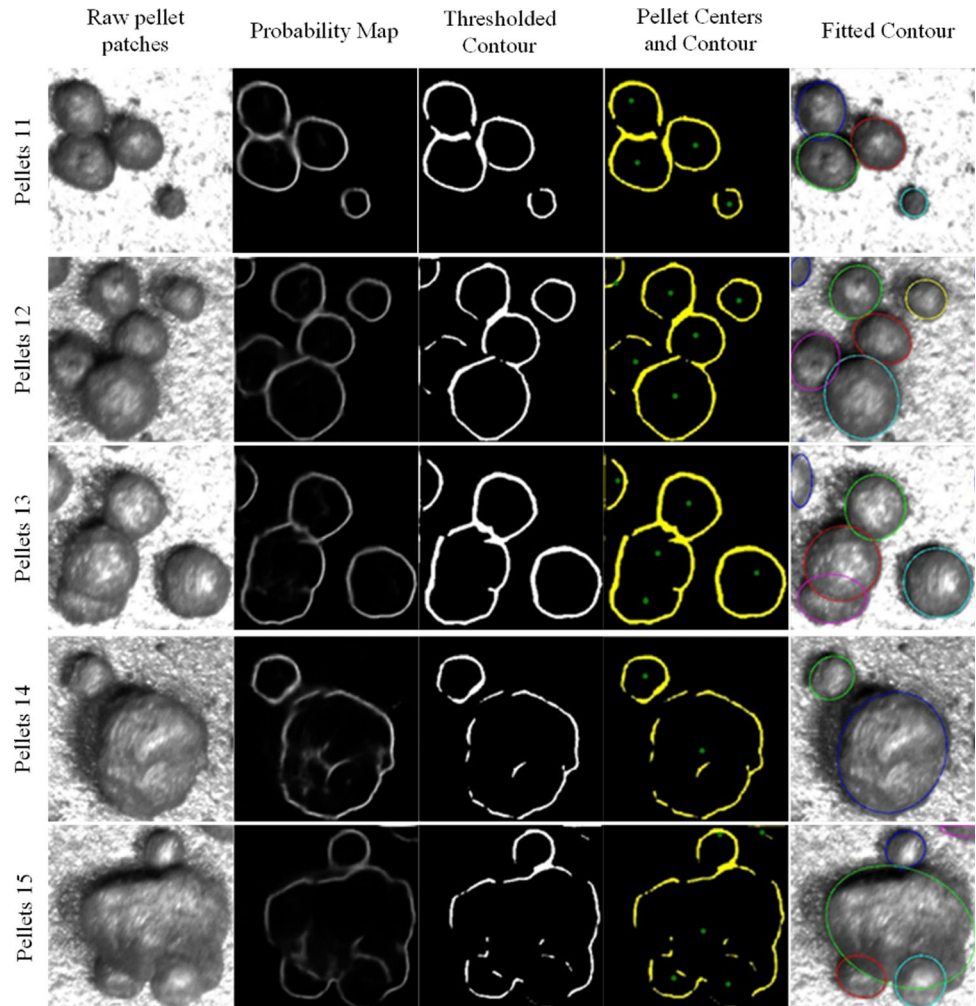
To illustrate the characteristics and advantages of our proposed network in segmenting iron ore pellets, some more pellet images are added for testing. In Fig. 8 and Table 2, the segmentation results are compared again with classic U-net. The blue suspending masks denote the successfully segmented pellet area. It can be seen that the proposed network is able to segment almost all the pellets from the background, while the classic U-net fails in the segmentation of some more pellets. The good performance of the proposed network is mainly due to the added BN layers which are able to improve the generalization ability of the network. As given in Table 2, for the case Pellets 2, Pellets 3 and Pellets 4, the proposed network has the similar DICE values to the classic U-net. However, the proposed network achieves much greater DICE values than classic U-net in the remaining seven cases.

For further comparison of the proposed network and the classic U-net, we sample the segmented image for multiple times. The sampled images have 80% of the original image size. For each sample image, the ROC curve function and AUC value of the two networks (see Fig. 9 and Table 3) are calculated using the mean value of respective index. Fig. 9 shows that the segmentation performances of the proposed network (plotted by dash lines) achieve higher convexity of ROC curves than the classic U-net does (plotted by solid lines). For the ten testing pellet patches, the corresponding AUC value of our proposed network is generally higher than that of the classic U-net, except the case Pellets 3 (see Table 3).

After segmentation of the pellet area from the background, the contour of each individual pellet should be extracted so that the pellet size can be measured in case of pellet overlapping. For this, the proposed centrum detection algorithm is used to extract the pellet centers and the points on the pellet contour in the thresholded probability map. With extracted contour of the pellet, ellipse polynomial fitting is then applied to approximate the pellet shape. Results of the above procedure are shown in Figs. 10 and 11, for example. It can be seen that the proposed method is

Table 3 Mean AUC value of the pellet patches from Pellets 1 to Pellets 10 processed by classic U-net and the proposed network

	Pellets 1	Pellets 2	Pellets 3	Pellets 4	Pellets 5	Pellets 6	Pellets 7	Pellets 8	Pellets 9	Pellets 10
Classic U-net	0.80	0.87	0.87	0.88	0.90	0.88	0.89	0.90	0.92	0.88
Proposed network	0.88	0.88	0.84	0.92	0.91	0.90	0.90	0.89	0.92	0.92

Fig. 10 Extraction of pellet contour for pellet patches

capable of extracting the contours of most pellets in the image, even for strong overlapping pellets (see Pellets 13 and Pellets 15 in Fig. 10).

In order to investigate the performance of the proposed method on processing full-size image, the ROC curves, AUC and DICE of the proposed network for the full-size images in Fig. 11 are given in Fig. 12 and Table 4. Each point on the ROC curve represents a sensitivity/specificity pair. In this section, we adopt ROC curve analysis in a cross-validation manner, consequently, segmentation results and the respective markers are matched randomly with 80% of their whole size, and the ROC curve together with AUC index are calculated ten times. Thereafter, we select three candidates with the highest AUC indexes to calculate the mean ROC curve and AUC. From Table 4, it can be seen that the AUC values of the full-size pellet images vary between 0.83 and 0.86, and the values of DICE indexes are very high (about 0.95). This indicates that the proposed method has good performance in segmenting pellets (including overlapping) and is advantageous to the subsequent processing for PSD statistics.

3.4 Discussion

3.4.1 Computing time

The proposed method is accomplished on an ordinary computing platform, which has an Intel Xeon E2520 v3 CPU with 16 GB memory and a NVIDIA GTX750i GPU with 2 GB memory. Python V3.5 is employed as programming language. In the study, 352 patch images with the size of 256×256 are used as training dataset. The batch size and training epochs are set to 8 and 400, respectively. Under such conditions, it needs about 3 days to complete the whole training procedure described above. After training, the weight model is obtained, and it needs only 1.5 s to 3 s to process one test patch image (256×256).

Considering the practical usage of the proposed network, we adopt 30 full-size raw images (1280×1024) for training and 148 full-size images for testing. The experiments show that it needs about 3 days for network training, but needs only about 4–6 s for segmentation of one full-

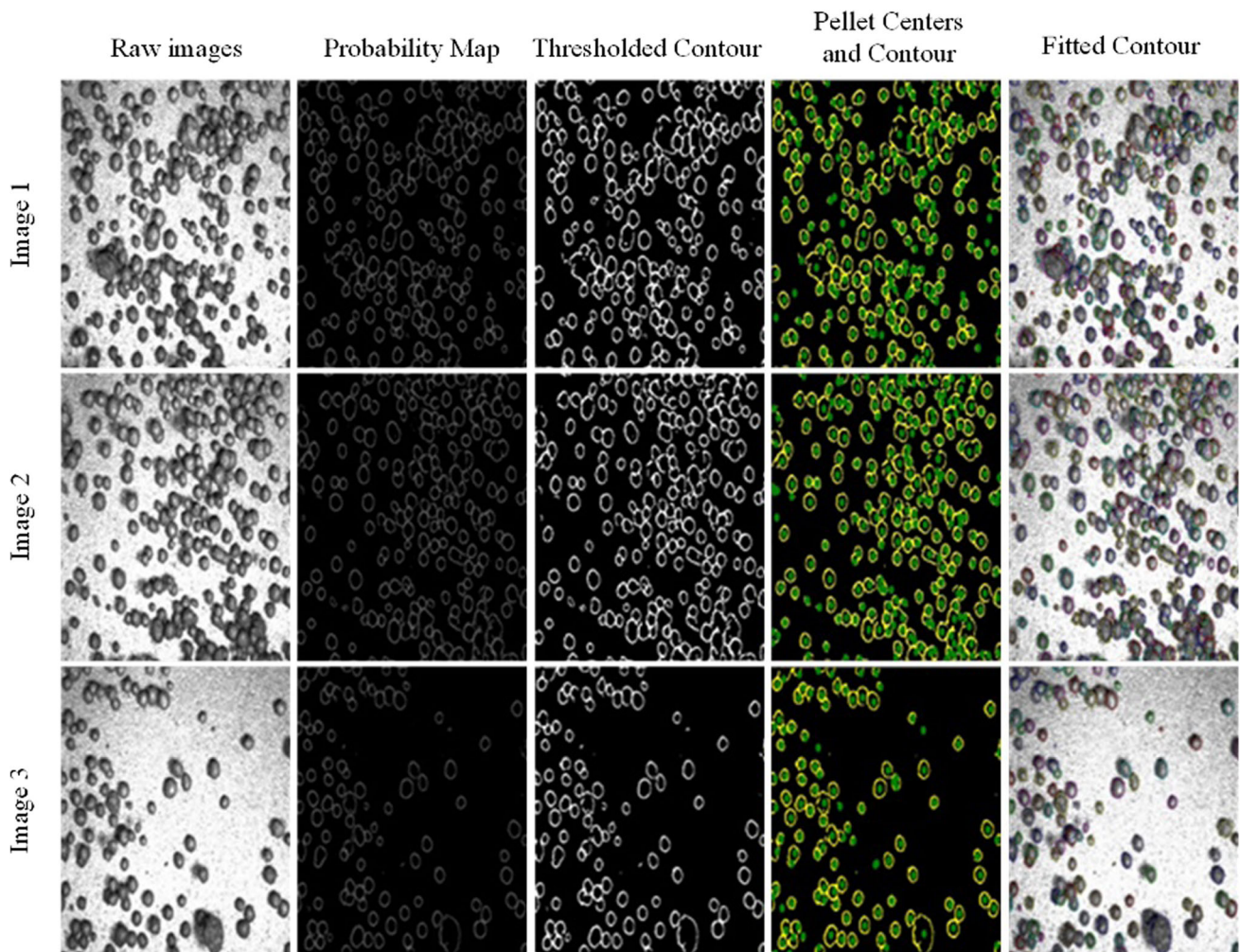


Fig. 11 Extraction of pellet contour for full-size pellet image

size testing image. Such computing time is acceptable for online usage in practice. The computing time of the proposed method can be further reduced by employing computer with higher configurations and is prosperous for real-time measurement.

3.4.2 Parametric study

Training epochs and batch size are two important parameters that may influence the segmentation performance of the proposed network and are therefore discussed in this section.

Figure 13 shows the influence of training epochs on segmentation performance (the loss and mean IoU) of the proposed network, using images of pellets patches and full-size images for testing. It is shown that, for the training progress of the pellet patches, the loss of the proposed U-net-based framework converges fast at the first 150 epochs and settles at 0.01 after 350 epochs. The intersection of union (IoU) is almost constant (0.7) when the

epochs are less than 100 and then increases rapidly to 0.85 when the epochs are 400. For the training progress of full-size images, the loss settles at about 0.01 after 300 epochs, while the mean IoU is approximately constant (0.96) after 400 epochs. Therefore, it is reasonable for us to 400 training epochs in the training of the proposed network.

Batch size in the training procedure is another important factor during training procedure. It affects both indexes of segmentation performance and memory occupation. To clarify the influence of batch size selection to these two indexes, the values of mean IoU and memory occupation ratio (the percentage of required computing memory in our GPU) with respect to six different batch size values (2, 4, 6, 8, 12 and 16) are given in Fig. 14. It can be seen that the memory occupation ratio is proportional to batch size values, and mean IoU generally increases with the increase in batch size. However, the improvement of mean IoU may be limited and memory occupation ratio will increase at the same time. The GPU memory will be exhausted if the batch size is set to 16 or greater. Therefore, we select batch

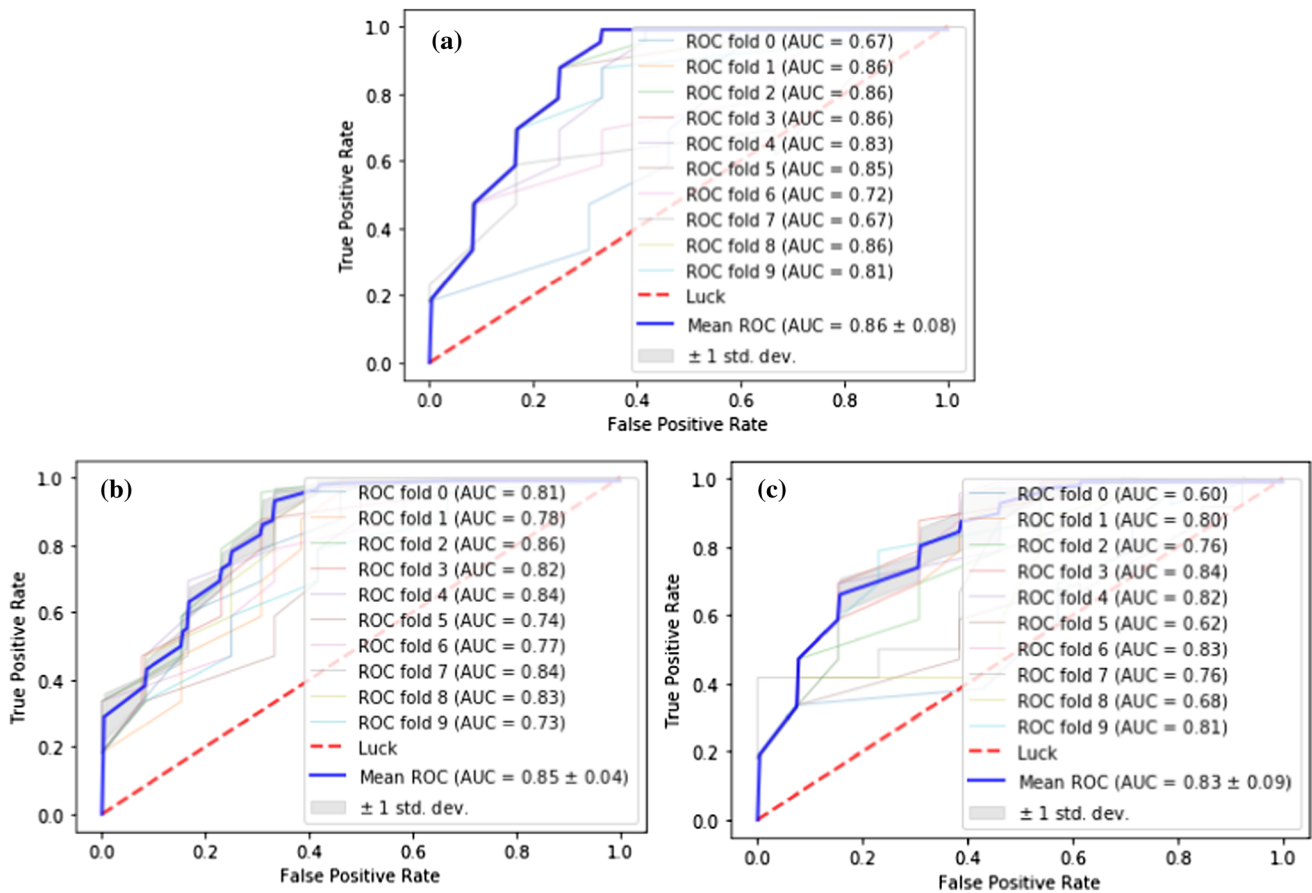


Fig. 12 ROC curve of the proposed network for **a** image 1 in Fig. 11, **b** image 2 in Fig. 11, **c** image 3 in Fig. 11

Table 4 Mean AUC and DICE index of the segmented full-size images in Fig. 11

	Image 1	Image 2	Image 3
Mean AUC	0.86	0.85	0.83
DICE	0.9545	0.9542	0.9568

size of 8 for our method to achieve good segmentation performance under reasonable memory occupation.

4 Pellet properties statistics

4.1 Pellet radius statistics

With the proposed method described above, the size of each pellet in the image can be described by a fitted ellipse with a short radius r_{short} and a long radius r_{long} , as shown in Fig. 15a for the pellets in the three full-size images (see Fig. 11). For simplicity, the pellet shape can be also described roughly by a circle with diameter d :

$$d = (r_{short} + r_{long}) \tag{24}$$

The pellet size distribution (PSD) defines the relative number of particles present according to pellet size d , as shown in Fig. 15b for example.

4.2 Roundness analysis

Roundness is a physical property of the pellets that may affect the discharge behavior [36]. It can be calculated by Eqs. (25–27):

$$\text{Circularity} = 4\pi \cdot \frac{\text{Area}}{\text{Perimeter}^2} \tag{25}$$

$$\text{AR} = \frac{r_{short}}{r_{long}} \tag{26}$$

$$\text{Roundness} = \text{Circularity} + (\text{Circularity}_{\text{Perfect_circle}} - \text{AR}) \tag{27}$$

where the aspect ratio AR is equal to one for a perfect circle, and $\text{Circularity}_{\text{Perfect_circle}}$ is the maximum of circularity.

The roundness distribution of the pellets detected in seven successively captured full-size images is shown in

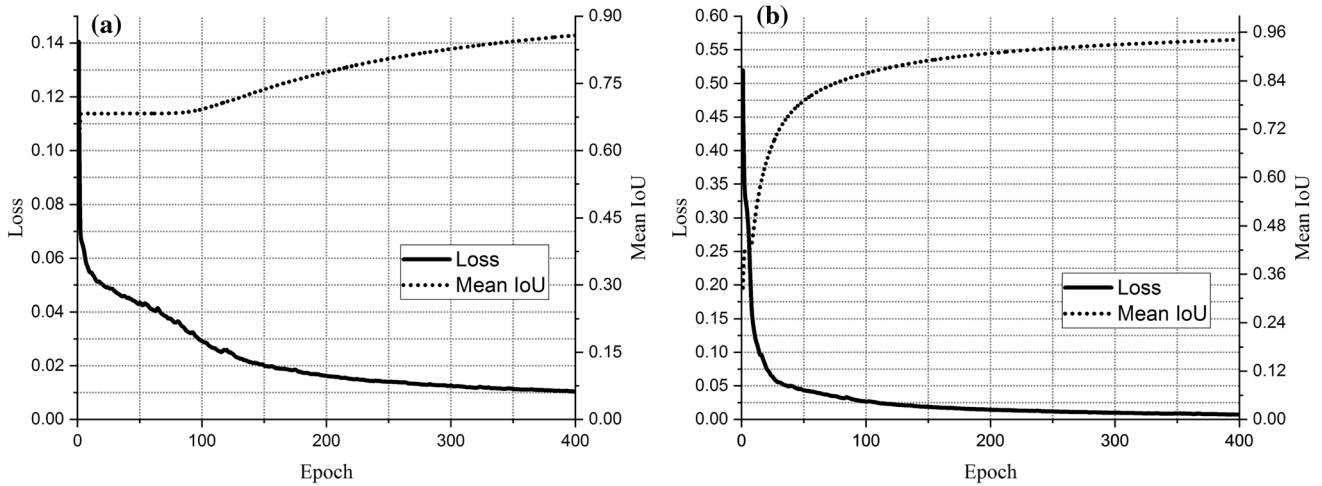


Fig. 13 Loss and mean IoU in the training process of the proposed network using images of **a** pellets patches and **b** full-size images for PSD statistics

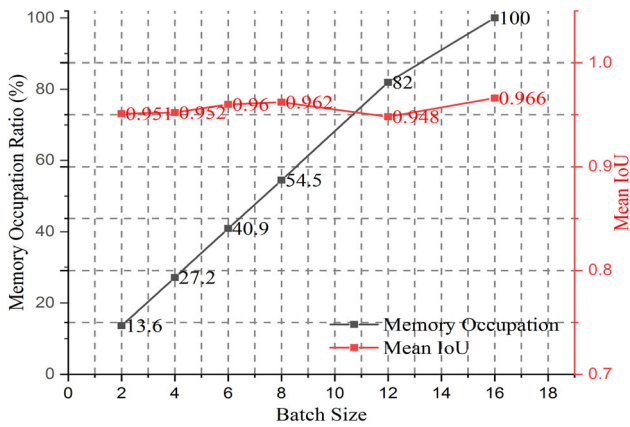


Fig. 14 Memory occupation and mean IoU w.r.t batch size after 400 epoch training

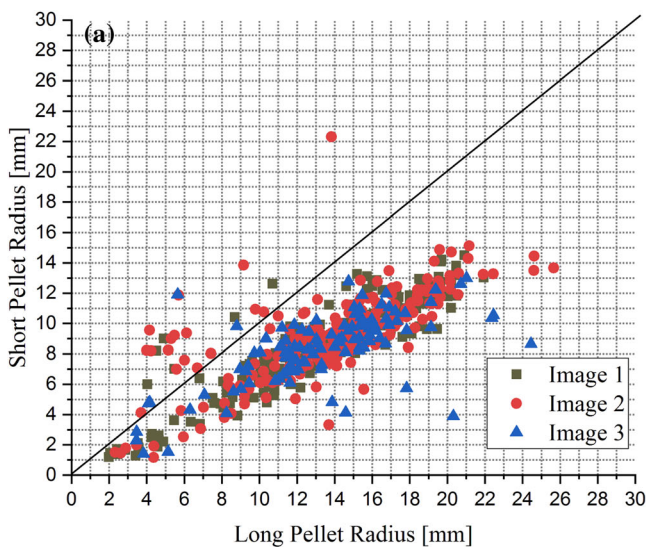


Fig. 16. The average number of pellets in each image is 70. It can be seen that there is a peak for each curve, and 30–50% of the pellets have a roundness of 1.33.

In the pelletizing process of the local steel company, pellets with roundness 1–1.3 and particle size 9–15 mm are considered as “good quality.” With our proposed method, the change of product quality with time can be monitored, as shown in Fig. 17 where percentage of pellets with desired roundness and size is calculated for seven full-size images captured at different times. It can be seen that at least 55% pellets have desired roundness and size (good quality), except at time $t = 0$ s and $t = 2$ s where only about 45% pellets have desired size.

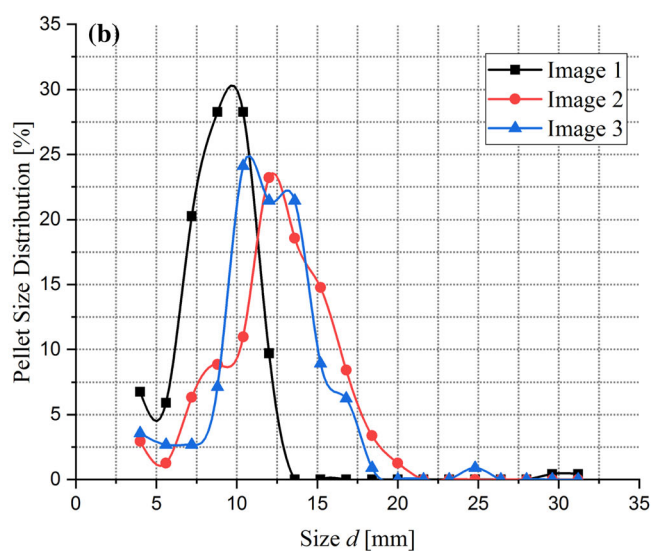


Fig. 15 Analysis of the pellet size for full-size images. **a** Pellet size measured by the proposed method, **b** pellet size distribution (PSD)

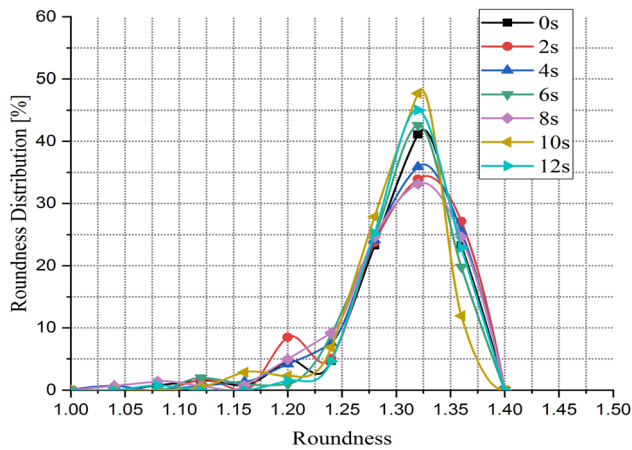


Fig. 16 Roundness distribution of pellets detected from seven successively captured full-size images

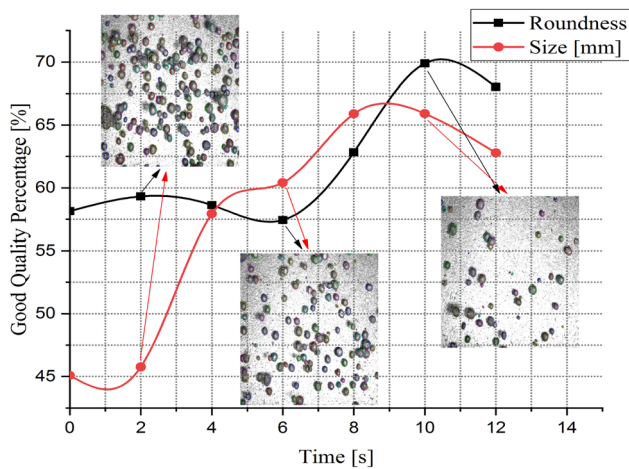


Fig. 17 Percentage of pellets with good quality and its change with time

5 Conclusion

The present work proposed a lightweight U-net for detection and segmentation of iron ore green pellets in images, with the aim to solve pellet overlapping problem and uneven illumination problem that are difficult for traditional methods. Compared to the classic U-net, the proposed deep learning framework has two advantages: (1) The network is a kind of lightweight U-net requiring less GPU memory and computing time. It is therefore more suitable for online image processing in practical usages; (2) by introducing batch normalization (BN) layers after the convolution and deconvolution layers in each unit in the encoder and decoder parts, respectively, the generalization ability of the proposed network can be improved.

The proposed method was tested by images captured from a local steel company. It shows good segmentation performance in terms of DICE and ROC evaluations.

Compared with traditional morphological algorithms and classic U-net, it has much better robustness to overlapping, uneven illumination and harsh light reflectance. Tests with static images and temporal image sequences demonstrate that the proposed method is effective in measuring the pellet size distribution and the shape evolution as well. The proposed method has potential usage in online detection of iron ore green pellets and other types of particles.

Acknowledgements Financial support from National Natural Science Foundation of China (No. 61374149) and Hunan Key Laboratory of Intelligent Robot Technology in Electronic Manufacturing (No. 2018001) is greatly appreciated.

References

1. e Silva BB, da Cunha ER, de Carvalho RM, Tavares LM (2018) Modeling and simulation of green iron ore pellet classification in a single deck roller screen using the discrete element method. *Powder Technol* 332:359–370
2. Liao CW, Tarnng YS (2009) On-line automatic optical system for coarse particle size distribution. *Powder Technol* 189:508–513
3. Facco P, Santomaso AC, Barolo M (2017) Artificial vision system for particle size characterization from bulk materials. *Chem Eng Sci* 164:246–257
4. Laitinen N, Antikainen O, Yliruusi J (2002) Does a powder surface contain all necessary information for particle size distribution analysis? *Eur J Pharm Sci* 17:217–227
5. Sandler N (2011) Photometric imaging in particle size measurement and surface visualization. *Int J Pharm* 417:227–234
6. Heydari M, Amirfattahi R, Nazari B, Rahimi P (2016) An industrial image processing-based approach for estimation of iron ore green pellet size distribution. *Powder Technol* 303:260–268
7. Hamzeloo E, Massinaei M, Mehrshad N (2014) Estimate of particle size distribution on an industrial conveyor belt using image analysis and neural networks. *Powder Technol* 261:185–190
8. Thurley M (2014) Measuring the visible particles for automated online particle size distribution estimation. In: *IMPC 2014*, 20–24 Oct 2014, Santiago, Chile
9. Heydari M, Amirfattahi R, Nazari B, Rahimi P (2016) An industrial image processing-based approach for estimation of iron ore green pellet size distribution. *Powder Technol* 303:260–268
10. Subramanyam V, Patra P, Singh MK (2017) Automatic image processing based size characterization of green pellets. *Int J Autom Smart* 7(3):85–91
11. Roozbahani MM, Borela R, Frost JD (2017) Pore size distribution in granular material microstructure. *Materials* 10(11):1237–1257
12. Nellros F, Thurley MJ (2011) Automated image analysis of iron-ore pellet structure using optical microscopy. *Miner Eng* 24(14):1525–1531
13. Budzan S, Pawelczyk M (2016) Grain size determination and classification using adaptive image segmentation with shape-context information for indirect mill faults detection. In: *International Congress on Technical Diagnosis, ICDT 2016*, 12–16 Sept 2016, Gliwice, Poland, pp 215–224
14. Budzan S, Pawelczyk M (2018) Grain size determination and classification using adaptive image segmentation with grain shape information for milling quality evaluation. *Diagnostyka* 19(1):41–48
15. Lu ZM, Zhu FC, Gao XY et al (2018) In-situ particle segmentation approach based on average background modeling and

- graph-cut for the monitoring of L-glutamic acid crystallization. *Chemom Intell Lab Syst* 178:11–23
16. Zhang B, Abbas A, Romagnoli JA (2011) Multi-resolution fuzzy clustering approach for image-based particle characterization for particle systems. *Chemom Intell Lab Syst* 107(1):155–164
 17. Chalfoun J, Majurski M, Dima A et al (2014) FogBank: a single cell segmentation across multiple cell lines and image modalities. *BMC Bioinform* 15:431
 18. Zhang H, Ji Y, Huang W, Liu L (2018) Sitcom-star-based clothing retrieval for video advertising: a deep learning framework. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-018-3579-x>
 19. Wang Y, Mao H, Yi Z (2017) Stem cell motion-tracking by using deep neural networks with multi-output. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-017-3291-2>
 20. Plissiti ME, Nikou C (2012) Overlapping cell nuclei segmentation using a spatially adaptive active physical model. *IEEE Trans Image Process* 21(11):4568–4580
 21. Fabijanska A (2018) Segmentation of corneal endothelium images using a U-Net-based convolutional neural network. *Artif Intell Med* 88:1–13
 22. Park JW, Carranza A, Jiang Z (2017) Semantic segmentation of 3D particle interaction data using fully convolutional DenseNet. In: 2017 IEEE conference on computer vision and pattern recognition workshops. Honolulu, Hawaii, USA, 21–26 July 2017
 23. Shahid L, Janabi-Sharifi F (2018) A neural network-based method for coverage measurement of shot-peened panels. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-017-3339-3>
 24. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: Medical image computing and computer-assisted intervention (MICCAI), LNCS, vol 9351, pp 234–241
 25. Ying X, Zhanyi H (2004) Catadioptric camera calibration using geometric invariants. *IEEE PAMI* 26(10):1260–1271
 26. Fitzgibbon A, Pilu M, Fisher RB (1999) Direct least square fitting of ellipses. *IEEE PAMI* 21(5):476–480
 27. Felzenszwalb PF, Huttenlocher DP (2004) Distance transforms of sampled functions. *Theory Comput* 8(19):415–428
 28. Roerdink JBTM, Meijster A (2001) The watershed transform: definitions, algorithms and parallelization strategies. *Fundam Inform* 41:187–228
 29. Vincent L (1990) Algorithmes Morphologiques a Base de Files d'Attente et de Lacets. Extension aux Graphes. PhD thesis, Ecole Nationale Supérieure des Mines de Paris, Fontainebleau
 30. Vincent L, Soille P (1991) Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans Pattern Anal Mach Intell* 13(6):583–598
 31. Koomsap P, Chansri N (2014) Topological hierarchy-contour tracing algorithm for nests of interconnected contours. *Int J Adv Manuf Technol* 70:1247–1266
 32. Liu Q, Hong X, Zou B et al (2017) Hierarchical contour closure-based holistic salient object detection. *IEEE Trans Image Process* 26(9):4537–4552
 33. Arbeláez P, Maire M, Fowlkes C, Malik J (2011) Contour detection and hierarchical image segmentation. *IEEE Trans Pattern Anal Mach Intell* 33(5):898–916
 34. Takashimizu Y, Iiyoshi M (2016) New parameter of roundness R: circularity corrected by aspect ratio. *Prog Earth Planet Sci* 3(1):1–16
 35. Kaushal B, Jain K, Sharma SK (2014) Estimation of area under receiver operating characteristic curve for bi-pareto and bi-two parameter exponential models. *Open J Stat* 4:1–10
 36. Gui N, Yang X, Jiyuan T, Jiang S (2017) Effect of roundness on the discharge flow of granular particles. *Powder Technol* 314:140–147

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.