Electronic Research Archive

ERA, 30(11): 4015−4037.
DOI: 10.3934/era.2022204
Received: 02 June 2022
Revised: 04 August 2022
Accepted: 23 August 2022
Published: 13 September 2022

http://www.aimspress.com/journal/ERA

Research article

Tapping stream tracking model using computer vision and deep learning to minimize slag carry-over in basic oxygen furnace

Dae-Geun Hong[1], Woong-Hee Han[2] and Chang-Hee Yim[1,*]

[1] Graduate Institute of Ferrous and Energy Materials Technology, Pohang University of Science and Technology, 77 Cheongam-Ro, Nam-Gu, Pohang, Gyeongbuk 37673, Republic of Korea
[2] Steel Making Research Group, POSCO Research Institute, 8 Pokposarang-gil, Gwangyang-si, Jeollanam-do 57807, Republic of Korea

* **Correspondence:** Email: chyim@postech.ac.kr; Tel: +82542799010; Fax: +82542799299.

Abstract: This paper describes a system that can automatically determine the result of the slag dart input to the converter during tapping of basic oxygen furnace (BOF), by directly observing and tracking the behavior of the pouring molten steel at the tapping hole after the dart is injected. First, we propose an algorithm that detects and tracks objects, then automatically calculates the width of the tapping stream from slag-detection system (SDS) images collected in real time. Second, we develop a time-series model that can determine whether the slag dart was properly seated on the tap hole; this model uses the sequential width and brightness data of the tapping stream. To test the model accuracy, an experiment was performed using SDS data collected in a real BOF. When the number of sequential images was 11 and oversampling was 2:1, the classification accuracy in the test data set was 99.61%. Cases of success and failure of dart injection were quantified in connection with operation data such as ladle weight and tilt angle. A pilot system was constructed; it increases the reliability of prevention of slag carry-over during tapping, and can reduce the operator's workload by as much as 30%. This system can reduce the secondary refining cost by reducing the dart-misclassification rate, and thereby increase the productivity of the steel mill. Finally, the system can contribute to real-time process control and management by automatically linking the task of determining the input of darts to the work of minimizing slag carry-over in a BOF.

Keywords: object detection; object tracking; recurrent neural network; steelmaking; tapping operation; slag detection system

1. Introduction

To manufacture high-purity steel, the steel-making process includes a refining operation to oxidize and remove silicon and phosphorus from molten iron by blowing oxygen and flux containing quicklime (CaO) into the converter [1,2]. After the refining operation, the converter is tilted to pour out the liquid steel. It is then tapped into a ladle, in which a secondary refining process is performed to adjust chemical composition and temperature of the molten steel before it is used in continuous casting. During the refining operation, silicon oxide, quicklime, magnesium oxide and iron oxide separate from molten steel and slag as a result of the difference in density. The amount of slag generated in this process is usually about 20−30 tons in a 250-ton converter [3−5]. This amount is about 80 to 120 kg per ton of molten steel, so the slag carry-over should be minimized as much as possible during tapping to ensure the cleanliness of the final molten steel and to maximize the grade of the steel.

The slag flows out with the molten steel when the converter is tilted after the blowing operation is completed. To minimize slag carry-over during tapping of molten steel made in a converter, the end point of tapping must be identified, but this is a difficult task [6−8]. When the completion of tapping is delayed, slag and other impurities enter the ladle with the molten steel, and its quality is degraded due to an increase in the slag carry-over amount; therefore, the time to the end point of the tapping process should be shortened. However, when the tapping operation is stopped too early, molten steel remains in the converter, so productivity is reduced. To balance this trade-off between slag carry-over and productivity, the duration of the tapping process must be optimized. Many converter processes use slag-inflow-prevention and slag-detection systems to minimize slag carry-over [6,9,10], but the identification of the end point of tapping still depends on the operator's expertise [11−13].

A Slag Detection System (SDS) is a convenient non-contact method to prevent and detect slag at the tapping hole of the basic oxygen furnace (BOF) [9,10]. The SDS exploits the difference in the emissivity of metal and slag [14,15]. SDS uses an infrared camera to monitor the flow of molten steel discharged from the tapping hole of the converter during tapping, then calculates the amount of slag contained in the molten steel. However, the existing SDS system does not enable checking whether the slag dart has properly seated in the tapping hole after middle of the tapping operation, and does not enable tracking and analysis of the pouring behavior of the liquid steel. In addition, the SDS is mainly used to detect slag carry-over at the end of tapping operation, and cannot support a worker's proactive response. Therefore, directly observation of whether or not the slag dart is seated in the tapping hole should be provided to help the operator in decision-making [11,16]. To determine the completion time of tapping operation, the workers need information about the tapping stream after the slag dart is inserted, rather than about the amount of slag at a specific point.

Therefore, this study presents a system that uses artificial intelligence (AI) to automatically determine whether a slag dart has been seated in the tapping hole of the BOF, by analyzing the shape of the tapping stream from SDS video collected in real time, and that can proactively respond to slag carry-over. The proposed model uses computer vision and deep learning to enable direct observation and analysis of the behavior of poured molten steel at the tapping hole after the dart is injected. The dart-input judgment using the behavior tracking of the tapping stream in the SDS image data can minimize the amount of slag carry-over by identifying situations in which the dart behavior changes on the surface of the molten steel in the converter, i.e., in an area that cannot be observed. The method proposed in this study has a lower misclassification rate than the method that uses the existing dart

image, and provides information on the tapping flow after inserting the slag dart, and therefore can reduce the workload of the operator.

Our goal is to minimize slag carry-over by automatically analyzing the slag dart input result during tapping, First, we develop an object-detection algorithm that automatically calculates the width of a tapping stream from SDS images collected in real time. Second, we suggest that changes in the width and brightness of the pouring stream can signal changes in the rate at which molten steel falls from the converter into the ladle. Third, we develop a time-series model that can trace the behavior of the pouring stream from SDS images by calculating the sequential width and brightness data. Finally, we develop an AI model that uses computer vision and is trained using data about the behavior of molten steel over time as an independent variable, and data about success or failure of the dart input as a dependent variable. To check the effectiveness of the proposed model, we performed an experiment to test the accuracy by using SDS data collected in a real foundry. We analyzed cases of success and failure of dart injection in connection with operation data such as ladle weight and tilt angle. We also quantify the effectiveness of the proposed method by comparing SDS data to actual darting success/failure results collected during the same period. Finally, to minimize slag carry-over during tapping, we constructed a pilot system that uses SDS images to automatically classify whether the slag dart has hit.

2. Literature review

2.1. Technology to detect and stop slag during tapping

When molten steel made from a converter is tapped, the slag outflow is largely divided into early, middle, and end of tapping. During the early stage, the slag mixes in the exit hole due to the tilt of the converter, and during the middle stage the slag mixes due to vortex formation [7,12,17]. At the end of tapping, the slag flows out with the small amount of remaining molten steel [15,18,19]; this process is called ‘slag carry-over’. The slag includes many oxidized compounds such as P2O5 and FeO, so when Al is added during the subsequent secondary refining process, most of the slag is reduced to molten steel, and this process degrades its quality. Representative methods to reduce slag carry-over include using a floating material such as a slag-check ball [20,21], inserting a slag dart during tapping to stop slag leakage [11,21], blasting gas over on the upper surface of molten steel during tapping [22], and using a sliding gate or stopper to control tapping [10,23].

During the steelmaking process, a slag dart is often injected after the middle of the tapping process [11,21]. A slag darts is a spindle-shaped plug composed of a refractory material, and is mechanically inserted into the exit hole of the converter, to stop slag inflow caused by vortex formation, and thereby minimize slag carry-over into the ladle. Whether the slag dart is properly seated at the exit hole of the BOF provides information that allows the operator to anticipate when the tapping operation will be completed [21,24]. The injection of the slag dart into the BOF is monitored and judged by an operator who watches video from a CCTV. The operator’s decision of whether the injected dart has or has not inserted into the hole is dependent on his/her expertise, and the timing varies among operators.

A deep neural network (DNN) model has been used to quantify the operator’s qualitative judgment of whether the injected slag dart had entered the hole; the model was trained on real-time videos collected from CCTV [11]. Compared with the existing system that relied on the visual

judgment of the operator, the accuracy of dart judgement increased by 10%, and the human workload decreased by 30%. However, the physical change of the molten steel cannot be observed directly at the tapping hole because the study discriminates the success and failure of the dart injection by considering the continuous shape of the dart dropped into the molten steel from the lance.

At the end of the tapping process, the levels of slag and molten steel reverse quickly, so a slag-stopper technology [10] linked with an SDS is generally used. It applies a high-resolution thermal imaging camera to monitor the flow of molten steel pouring from the converter during the tapping process, and detects in real time the amount of slag contained in the molten steel [9,15]. Infrared cameras can accurately distinguish between these two materials because they have very different radiative properties. The amount of slag carry-over can be calculated effectively in real time by evaluating the thermal image information. This information is provided to operators in real time to enable quick termination of tapping to prevent excessive slag inflow into the ladle from the BOF. However, the information does not help the operator to respond proactively to minimize slag carry-over.

2.2. Artificial neural networks that use computer vision and object tracking

In Computer Vision (CV), the key is to understand the image [25]. Research in the field is evolving from "understanding images" to "understanding video". A video is a series of images and provides more context than a single image. Object detection is a computer vision technology to identify or classify different types of objects in digital images and images [26,27]. Object-detection technology is developing very rapidly with the development of convolutional neural networks (CNNs) and of deep learning algorithms, and the increase in the computing power of GPUs [28].

Existing studies for detecting objects in camera images mainly applied modeling based on background removal, image segmentation techniques, and object detection algorithms using deep learning [29]. Random Sample Consensus (RANSAC) [30], Bayesian filtering [31], and optical flow [32] have been developed as the most common the background extraction techniques to detect moving objects in images captured by a camera. Object detection using image segmentation is a technique that detects objects by dividing the image into perceptually-similar regions [33].

Object detection using deep learning uses a CNN in a method to detect and recognize objects. The methods can use a two-stage detector or a one stage detector [34]. The two-stage detector method performs candidate region extraction and classification sequentially; it has high accuracy but has the disadvantage of slow speed. Representative two-stage algorithms include R-CNN [35], Fast R-CNN [36], and Faster R-CNN [35]. The one-stage detector method performs candidate region extraction and classification simultaneously; compared to the two-stage detector method, the one-stage method has slightly lower accuracy, but is faster, so it is more suitable for real-time detection. Representative one-stage algorithms include YOLOv3 [37], SSD [35], RetinaNet [38], YOLOv4 [39], and YOLOv5 [40].

Artificial neural networks (ANNs) that process computer vision data are widely used in the steel industry. They apply deep learning (DL), which is a machine-learning technique that uses multiple layers of neural networks (NNs); it can solve problems by using pattern recognition [35−37]. DL that uses CNNs has been used for many visual detection tasks including recognition of slab identification number (SIN) [41,42], determination of dart shape in BOF [11], diagnosis of surface defect in steel products [43,44], and segmentation of phase in complex microstructures [45]. The

efficient use of graphic processing units and the development of new algorithms such as Rectified Linear Unit (ReLU) [46] and deep convolutional [37,39,40] have yielded excellent results for image classification and object detection.

However, traditional NNs function on the assumption that all inputs are independent, so the networks cannot easily process time-series data. Therefore, CNNs are not sufficient for object tracking, which recognizes a moving object in sequential images, and tracks its motion by calculating its position in them [47]. This task requires customized algorithms for such subtasks as image preprocessing [47], point detection [48], and background subtraction [49], so implementation of an object-tracking system is a difficult task.

2.3. Time-series model that can analyze object behavior

Object tracking is a technique the image processing field to detect and track a region of interest and an object of interest [50]. The technique recognizes an object in one frame of video and compares this information with the next frame to match object information. The movement paths of objects detected during the object-tracking process are time-series data that change over time.

Recurrent neural networks (RNNs) are effective for time series data analysis because RNNs include a memory that stores past shapes in the model [51]. The RNN has an output determined according to the previous state, and input stored in each cell, and learns using backpropagation. By using the trained data, the RNN can learn the movement path of an object and predict its next position. However, RNNs have a problem called gradient vanishing, which may cause the gradient near the input layer to gradually decrease during the backpropagation process as the network gets deeper. To solve this problem, the attention mechanism was originally proposed in the field of natural language processing and machine translation to overcome the limitations of the Seq2Seq model implemented on RNNs [52,53]. The technique does not refer to the entire input at the same rate, but by giving weight to a specific part and proceeding with learning, it solves the problem and improves the model's accuracy at the same time [52−54]. This process can be equally applied to general image-recognition or object-detection problems.

Several other NN architectures have been developed to increase the accuracy of identifying object behavior in successive images. Residual attention networks (RANs) [55] modularize the NN to which attention is applied and stack it in several layers. This architecture increased image-recognition accuracy but its complex structure increased the amount of computation. Subsequently, Squeeze and excitation network (SENet) [56] reduced model error considerably by using compression and readjustment to extract only important features from each channel of the image, and by learning by weighting them. Convolutional block attention module (CBAM) [57] achieved increased accuracy and shortened learning time by adding attention-focused computation for spatial information as well as channel information. Efficient channel attention networks (ECA-Nets) [58] increased the attention-focusing calculation for the channel by excluding dimensionality reduction. Most of the research is progressing in the direction of improving accuracy while minimizing overhead and model complexity by modularizing attention-intensive operations and embedding them in general NNs.

Long short-term memory (LSTM) is a model that compensates for the shortcomings of RNNs [51], and is a kind of RNN architecture that can remember old or recent data by using memory cells and three gates. The memory cells are responsible for storing information, and the three gates are responsible for controlling the addition and removal of information. Memory cells should be

configured to forget what is to be forgotten and to store what is to be stored. They have limited capacity, so the most efficient approach is to remember only important information. The memory of LSTM consists of a cell state and a hidden state. The cell state is activated before the input vector (processed information) is stored in the current state (hidden state) along with the old hidden state according to the timestep. In this process, the three gates (input, output, forget) determine whether the cell state is activated.

SDS detects and monitors transported slag from the tapping process in BOF [9,14]. The infrared image detection method [59−64] converts the infrared radiation temperature of molten steel to a grayscale to show the flow of molten steel as an image. Methods have been developed to correct the temperature threshold used to distinguish molten steel from slag in a specific environment [59−61], and to design and construct experimental equipment [62−64], but very few studies have considered object tracking in the field of infrared slag monitoring and detection. In this study, the LSTM model is used instead of the attention-focusing technique, because the image collected from the SDS is not used as an input, but the necessary features (left/right position, brightness) are extracted and analyzed from the image.

3. The proposed system

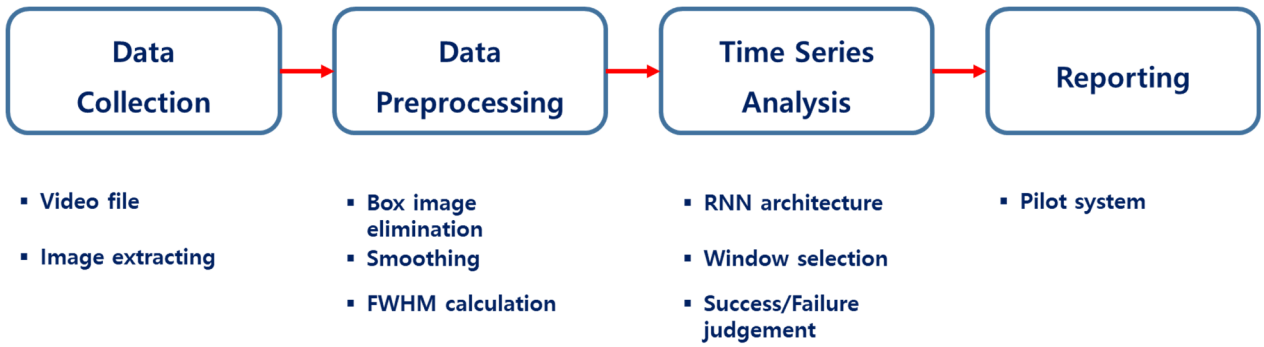

Figure 1. Proposed tapping-stream tacking system to minimize slag carry-over in BOF.

This research proposes a system that uses a model that integrates object detection/tracking and an RNN to recognize automatically whether the slag dart has seated in the tapping hole of the BOF. The research was composed of four steps conducted sequentially (Figure 1).

The first step is to collect image data from SDS video. The SDS is a device used to monitor and reduce slag carry-over in a steel-production facility. In real time, it monitors and stores the flow of molten steel discharged through the tap of the BOF, and calculates the amount of slag contained in the molten steel. Molten steel and slag can be distinguished according a large difference of emissivity in the far-infrared band. A correction method for the temperature threshold can be designed to distinguish molten steel from slag in a specific environment, and to calculate the amount of slag in real time. This SDS stores a video recording of all tapping operations. Images of the operation are extracted from the SDS video and stored for later use to recognize and analyze the progress of individual tapping operations.

The second step is to pre-process the collected data. For analysis of the behavior of the tapping stream in the BOF, unnecessary information is removed from the collected images to facilitate effective

time-series learning. In this study, the features required for time-series learning were generated in three steps: box removal, smoothing, and full width at half maximum (FWHM) calculation. To measure the thickness of tapping stream in the image collected from SDS, a region of interest (ROI) is selected from the original image, then visible noise (box removal) included in the ROI is removed, and the noise or edge with a large change rate of the brightness value is eliminated from the image to soften the entire image. Finally, the FWHM is used in the brightness graph to calculate the width of the tapping stream for object detection and tracking in the smoothed image. FWHM quantifies the width of a function, and indicates the difference between the values of two independent variables when each of their values is half of its maximum value [65].

The third step is to decide whether the slag dart has successfully seated in the tapping hole of the converter. For this purpose, we developed a technique to detect and track the behavior of molten steel in SDS images as it pours from the converter. The width and brightness of the tapping stream are extracted from the SDS images collected in real time, then input to LSTM in the form of sequential data (Figure 2). The sequential information to be input to the LSTM consists of images from the time at which the slag dart is inserted. When the slag darts properly occupy the tapping hole, the width of the tapping stream is reduced. The dart's movement is sequentially calculated from the moment the dart is inserted, so the judgment about whether it has hit the hole is continuously affected by the characteristics of the first image. This method can detect changes in the width and brightness of the consecutive tapping stream, and is therefore more reliable than using a single image. Finally, the final output is converted to a probability to recognize success or failure of slag dart insertion.

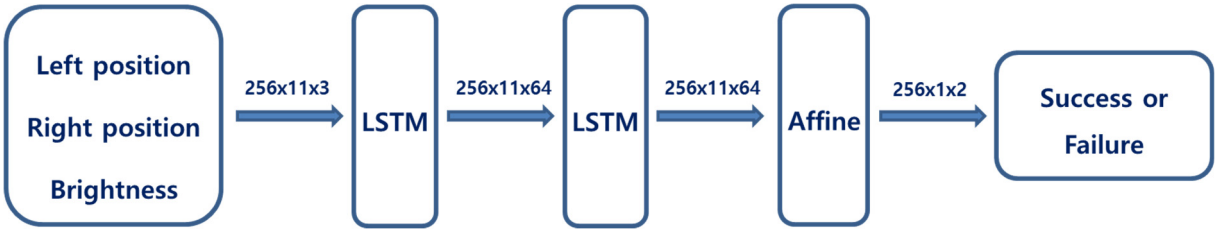

Figure 2. Proposed RNN Architecture for judgement of slag dart injection from SDS images.

The fourth step is to check the effectiveness of the computer vision and artificial NN tapping-stream tracking model. For this purpose, a prototype system was implemented; it uses the sampled SDS video data to monitor the thickness of the tapping stream, and uses the thickness to judge the success or failure of slag-dart injection. In this study, we used the Keras library [66] with Python and TensorFlow to train and test the proposed model. To develop the prototype system Python 3.5.0 and Python-related libraries were used to develop components such as data preprocessing calculation module, graphical user interface (GUI), and graphing the output. To identify and correct possible causes of problems, small-scale testing was performed before real-world implementation; i.e., the field applicability of the proposed model is tested by analyzing actual foundry data, and the system is upgraded in response to relevant feedback.

4. Materials and methods

4.1. Data collection

Each converter is equipped with a high-resolution video camera to monitor the input of darts, and an infrared camera to monitor the tapping stream, and uses these two cameras to record and save the entire tapping process as real-time video (Figure 3). These collected video data are stored in the system in mp4 format for up to one month. The duration of the total tapping operation is variable depending on the amount of filling per charge, so the playing time of the video data is between 4 and 7 min. The image size of the collected SDS video is 1000 × 510 × 3, and it is saved at 20 frames per second (fps).

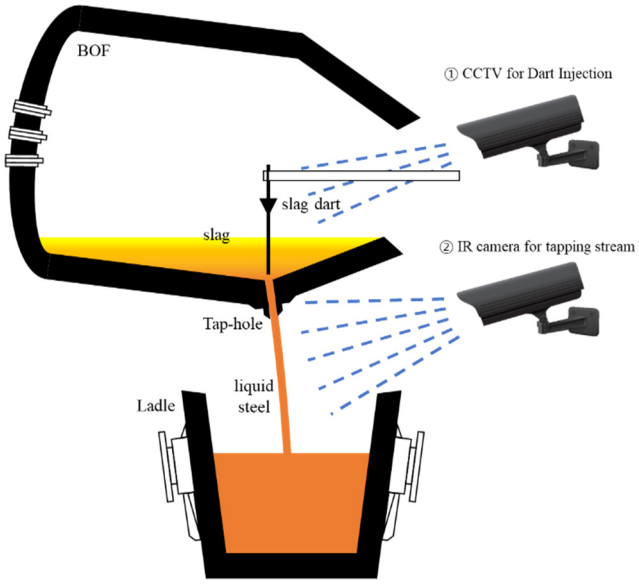

Figure 3. Schematic diagram of converter with a CCTV for darting and an IR camera for tapping.

This study used data from 1900 tapping operations conducted between February 1 and March 30, 2020. Data were divided into two classes: success (1697) and failure (203). Images were extracted at 1 fps from each SDS video file. The maximum number of images included in 1900 SDS videos is 1283 and the minimum number is 346, with a total of 940,198. Operational data such as tilting angle and ladle weight were also collected. The tilting angle indicates the angle at which the converter is tilted to discharge the molten steel from the tapping hole of the converter to the ladle. The ladle weight is obtained by calculating the mass of the molten steel taken from the converter to the ladle. These two types of data were used to determine the end point of the analysis on the width and brightness of the tapping stream.

4.2. Data preprocessing

To detect and track the tapping stream as an object in the images collected from SDS, a rectangular area of 830 × 100 pixels in each image (Figure 4, red line) was selected as the ROI. To identify and calculate accurately the width of the tapping stream in the ROI, the brightness values must be integrated in the vertical direction after detecting and excluding the image region (or noise) in which

the brightness changes rapidly. Therefore, in this study, sequential data for time series analysis were generated using three steps: box removal, smoothing, and calculation of FWHM.

First, the brightness is sharply high at about 280 and 500 pixels in the graph where the brightness values are integrated, because the image includes a white dotted line (Figure 4), which is the ROI selected to analyze the slag content contained the molten in the SDS system. Therefore, vertical and horizontal white box lines were detected using the find_peaks function [67] supported by the Python library for scientific and technological calculations, and excluded from the vertical brightness calculation. Second, the image was smoothed to remove high-frequency components that correspond to noise [68]. In the image, regions in which the brightness changes rapidly are regarded as edges or contours. Blocking these components can minimize the variance of the image. Low-pass frequency filtering removes high-frequency components that represent edges or contours are removed, so the entire image is smoothed [68,69]. Finally, FWHM was used to calculate the width of the tapping stream in the brightness graph of the smoothed images (Figure 4). In the brightness graph, the maximum value is 3300 and the minimum value is 200, so the FWHM is 1550. Objects were recognized and tracked in the ROI of each image collected by using this value as a reference point for the width of the tapping stream.

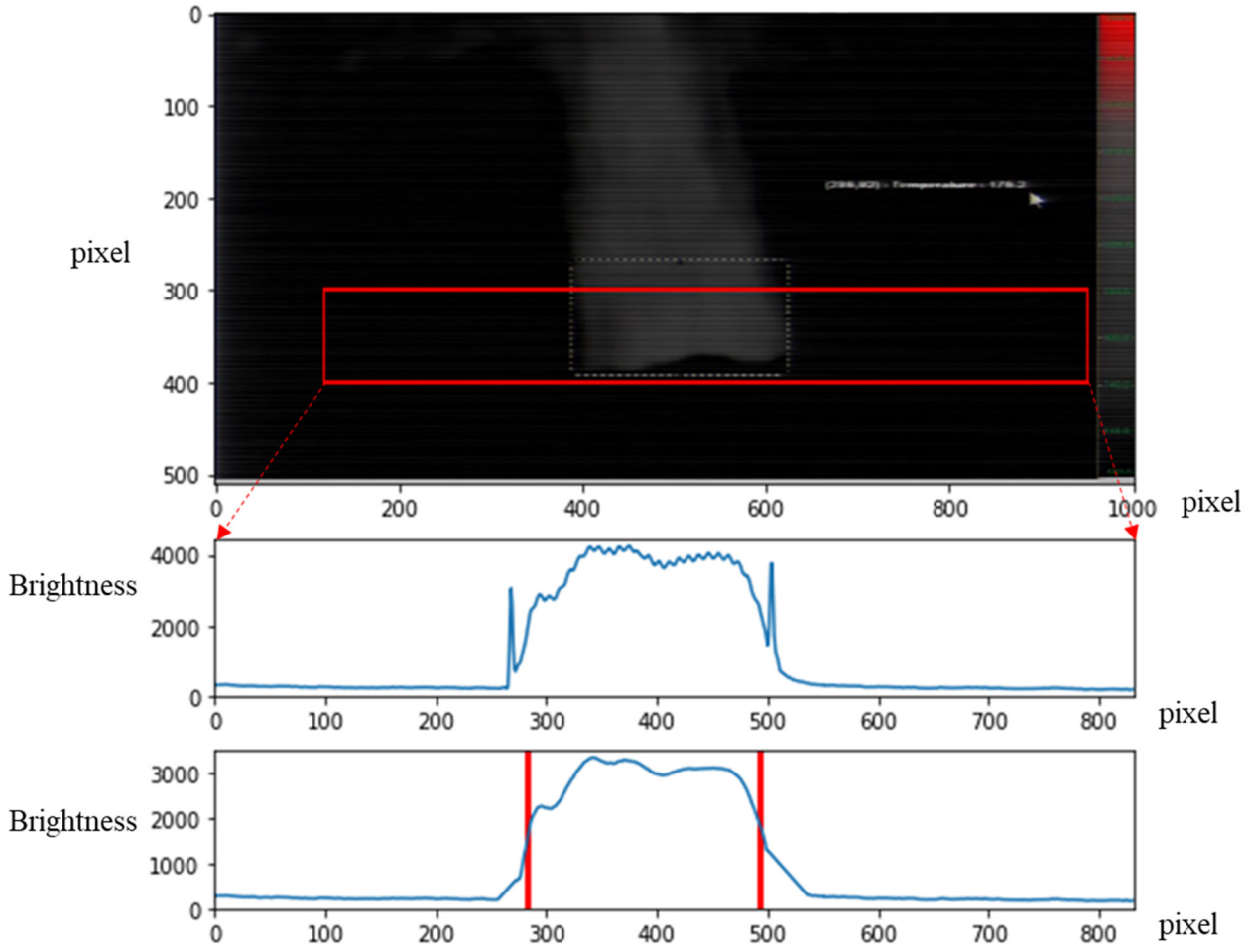

Figure 4. Schematic diagram of the process of detecting the tapping stream in SDS images.

In addition, sequential data about changes in the width and brightness of tapping stream were generated using one frame per second in all video files. Direct comparison of the width and brightness of the stream is difficult because the scales differ. In modeling, parameter distortion might occur due to different scales between variables [70]. In this study, MinMaxScaler [71] was used to standardize the scale between 0 and 1 for the input variable and the dependent variable. First, the initial data of the vector $x = (x_1, x_2, \ldots, x_n)$ were normalized as

$$x_i' = \frac{x_i - Min(x_i)}{Max(x_i) - Min(x_i)} \tag{1}$$

where x_i' represents the normalized value of x_i, $Max(x_i)$ is its highest value and $Min(x_i)$ is its lowest value.

5. Case study

5.1. Time series analysis

The program was built using the Keras library that uses Python and TensorFlow to train and evaluate the proposed model. The model was trained in a Windows 10 64-bit environment with two GTX 2080 Ti graphics cards and an Intel i7-6700K processor and. Experimental data were partitioned as 60% training, 20% validation, and 20% testing, and were not used redundantly in any stage of training, validation, or testing. The goal is not simply to find model that fits the training data, but to find one that fits the test data, which were not used for training [11,72]. The training data were used use to train the model; this process ultimately entails estimating coefficients such as weights and biases. The validation data were used to fine-tune the hyper-parameters of the model, and the test data were used to evaluate the accuracy of the trained and validated model.

Input included two data that indicated the left and right sides of the tapping stream and one datum that represented its brightness in the form of [Batch_size, n_seq, 3]. Two inputs were used to train the recognition of tapping stream from SDS images, and three inputs including one brightness datum were used to learn the behavior of the pouring stream over time. A hidden layer was composed of two LSTM layers with 64 units each, and one affine layer (Figure 1). The hyper-parameters were set as batch size = 256, epoch number = 500, and learning rate = 0.0001 after the value of the cost function was minimized by learning the proposed model iteratively. The Adam optimizer was applied to make the learning fast and stable. In the output layer, the logistic sigmoid function [73] was applied to transform the output value to the probability of success or failure of slag dart input. Drop-out and early stopping techniques were also used to prevent overfitting.

Another requirement is to determine the input length (number of sequence) of the RNN for optimal time series analysis. The number of sequences was repeatedly learned and evaluated while changing values of length from five to 21 (Figure 5). The classification accuracy of success and failure was highest when the number of sequences = 11, so this number of sequences used. When this number of sequences was applied, the classification accuracy in the test data set was 99.61%.

The number of failure data was limited, because successes (1697) were much more common than failures (203). Therefore, we evaluated the effect of resampling on unbalanced data, either by under-sampling or by oversampling. Under-sampling solves data imbalance by reducing the number of data in the most abundant class in an unbalanced data set. This method dramatically reduces the total

number of data available for training, and this reduction can degrade prediction accuracy. When one datum is selected as a sampled from a database with ratio 3:1 (success:fail), the probability of success is 0.75 and of failure is 0.25. If the size of the batch is 16, the probability that all 16 are successful is $(3/4)^{16} = 1\%$; i.e., the probability of at least one failure is 99%. A ratio of 3:1 was adopted so that at least one failure case was almost always included in a batch. To verify this choice, the accuracy of dart-insertion identification was evaluated using three data sets that had success: failure ratios of 3:1 (609:203), 4:1 (812:203), and 5:1 (1015:203). The 3:1 first data set had the highest accuracy (Table 1).

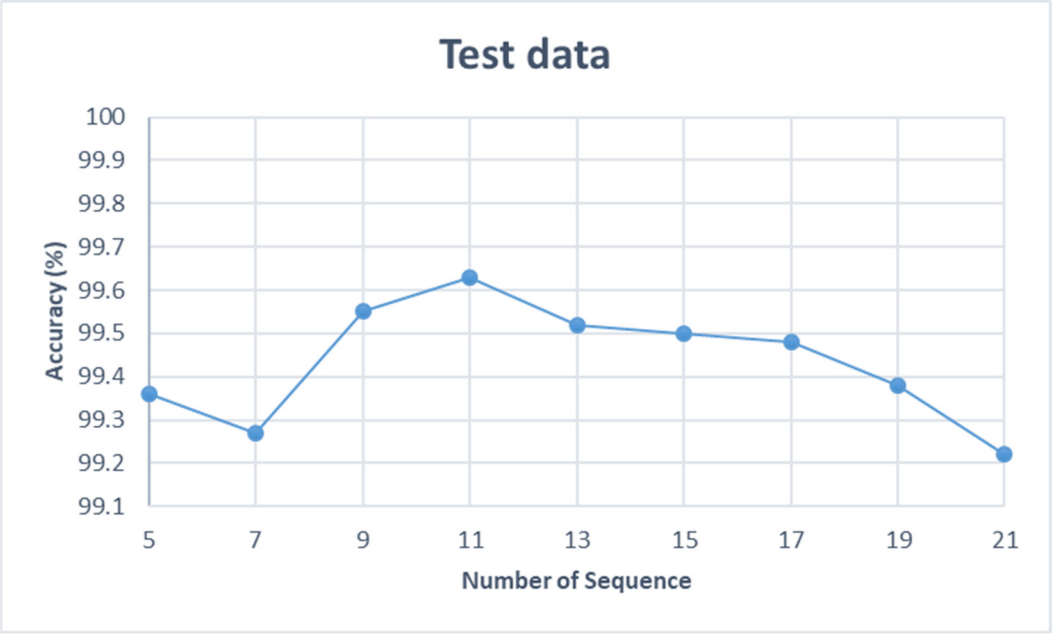

Figure 5. Length optimization of the input data for time-series analysis.

In contrast, oversampling is a way to resolve data imbalances by increasing the amount of data in low-rate classes. In this study, new data were generated using the nearest neighbors of low-ratio class data, by Synthetic Minority Over-sampling Technique (SMOTE). In the case of SDS where the failure data are scarce compared to success data, oversampling reduces precision because the rate of predicting failures increases. Conversely, the proportion of errors that are identified increases. Therefore, SMOTE was used, paying attention to the method of lowering the decrease in precision and increasing the increase in recall. In oversampling, three conditions were evaluated: 1:1 (1697:1697), 2:1 (1697:849), and 3:1 (1697:566). The second sampling method with a 2:1 ratio had the highest accuracy (Table 2). Over-sampling was considered to be a better solution than under-sampling. Therefore, a second dataset oversampled at a 2:1 ratio was used for training and evaluation to recognize and track the tapping stream in SDS images.

To quantify the reliability of the model's prediction, we used five evaluation indicators: confusion matrix, accuracy, recall, precision, and F1 score. The predicted class and the observed class are classified into negative (0) and positive (1) values, and divided into True Negative (TN; i.e., correct identification of a failure of the dart to insert), False Negative (FN; incorrect identification of failure), True Positive (TP; correct identification of insertion), and False Positive (FP; incorrect identification of insertion). The confusion matrix (error matrix) classifies the types of prediction error (Table 2).

Table 1. Comparison of classification analysis after over- and under-sampling.

	Over-sampling			Under-sampling		
Ratio	1:1	2:1	3:1	3:1	4:1	5:1
No. of test	679	509	453	163	203	244
No. of error	4	2	3	5	7	9
Accuracy	99.41%	99.61%	99.34%	96.93%	96.55%	96.31%

Accuracy [74] indicates the similarity of predicted data to real data; i.e., the total number of correct identifications as a proportion of all observations:

$$\text{Accuracy} = \frac{TP+TN}{TN+FN+TP+FN} \tag{2}$$

Recall [74], also called sensitivity, refers to the ratio of number of images correctly identified as positive, to the true total number of positives:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3}$$

Precision [74], also called positive predictive value, refers to the ratio of images correctly identified as positive to the total number identified as positive:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{4}$$

The F1 score [74] is the harmonic mean of recall and precision. The F1 score is highest when precision and recall are not biased to either side:

$$\text{F1} - \text{score} = \frac{2(Precision \times Recall)}{Precision+Recall} \tag{5}$$

The proposed time-series analysis model got good results on all indicators: Accuracy = 99.61%, recall = 99.70 %, precision = 99.70%, and F1 = 99.7%. Recall and Precision were the same in this case because, FN = FP = 1, so the denominators of (8) and (9) are identical.

Table 2. Performance evaluation of classification model using confusion matrix.

Confusion Matrix		Real values	
		Positive	Negative
Predictive values	Positive	335	1
	Negative	1	172

5.2. Reporting

To check the applicability of the proposed model, a prototype system was developed to judge the success or failure of slag dart-injection while detecting and tracking the tapping stream in video collected from the SDS. First, this system automatically calculates the width and brightness of the pouring stream from SDS images provided in real time. These two types of information were input

into the LSTM module in the form of sequential data to recognize whether the slag dart occupies the tap hole of the converter. This software was implemented in Python language. The sci py library was used for data preprocessing, and PyQt-5.9.2, matplotlib-3.0.0, and Numpy with the mkl library were applied to GUI.

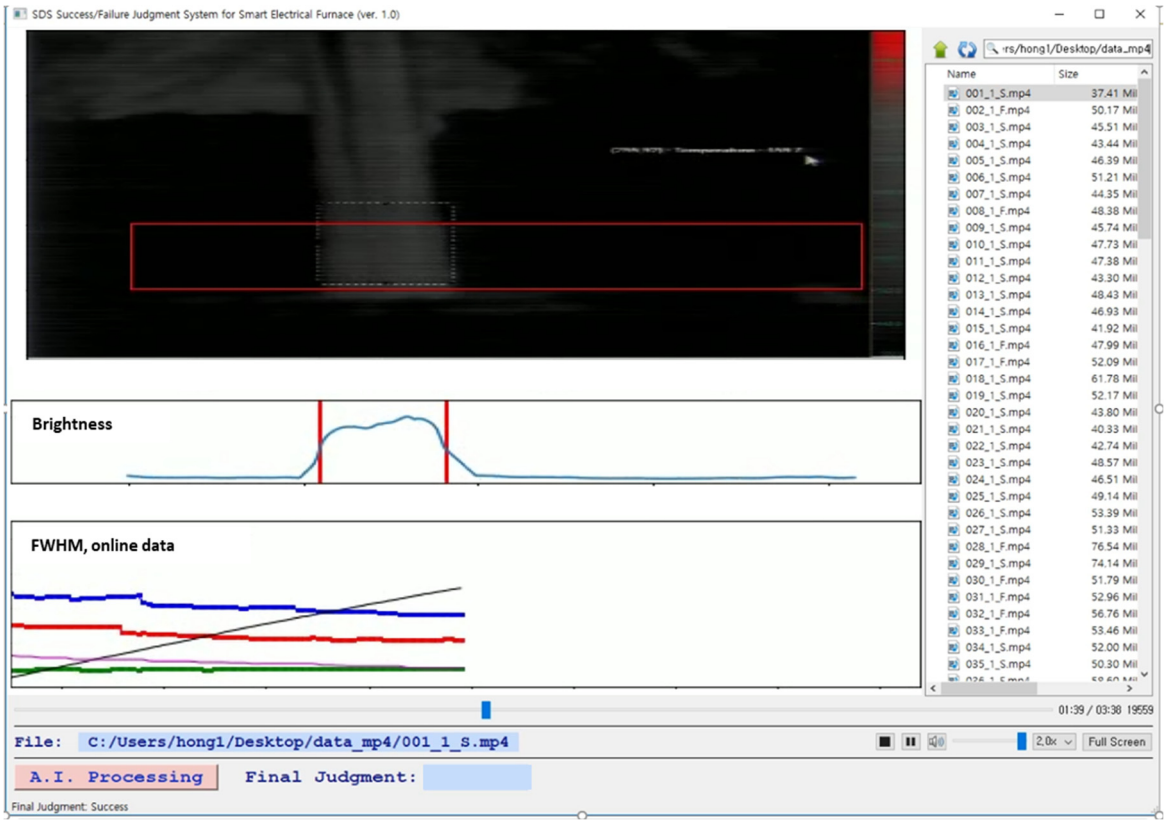

Figure 6. A pilot system for SDS success/failure judgement.

The GUI of the pilot system for SDS success/failure judgement is mainly composed of four panels: raw data, data preprocessing, behavior information and analysis tool (Figure 6). First, the raw data panel provides the user with the video of the pouring stream collected from the SDS system in real time. Second, the data preprocessing panel extracts SDS images from the video file, then detects a tapping stream by using brightness information from the extracted image. Third, the behavior information panel provides the user with operation data such as tilt angle and ladle weight as well as information on the thickness and brightness of the tapping stream generated during one tapping operation. The judgement of whether the slag dart insertion is hit is performed by calculating the width and brightness of the tapping stream in the section between the dart injection and the end of the tapping operation. Finally, the analysis tool panel displays the final judgment result. The SDS video monitored by this system is saved in the database. Users can drag playback button where you want it on the screen in the video, and then this system enables moving the screen to the desired location. In addition, this system enables the user to find the information of a desired date by providing directory information of the video stored. This system is a stand-alone type and was used to test the field applicability of the proposed model by using sampled video data. It is also being tested in connection with data transmitted from the SDS system in real time.

6. Results and discussion

This study automatically extracted the thickness and brightness of the tapping stream of molten steel pouring from BOF during tapping, and linked with operation data such as weight and tilt angle to determine in real time whether or not the slag dart properly occupy the tapping hole. This study results should be discussed from three perspectives to directly observe and effectively track slag-dart injection. First, the width and brightness information of tapping stream must be quantified and compared according to the success or failure of dart injection. Second, the ability to effectively support determination of the end time of tapping process in connection with operation data such as weight and tilt angle must be determined. Finally, the recognition of success or failure of slag dart input using the SDS image must be compared to the existing methods of determining the time of completion of tapping process.

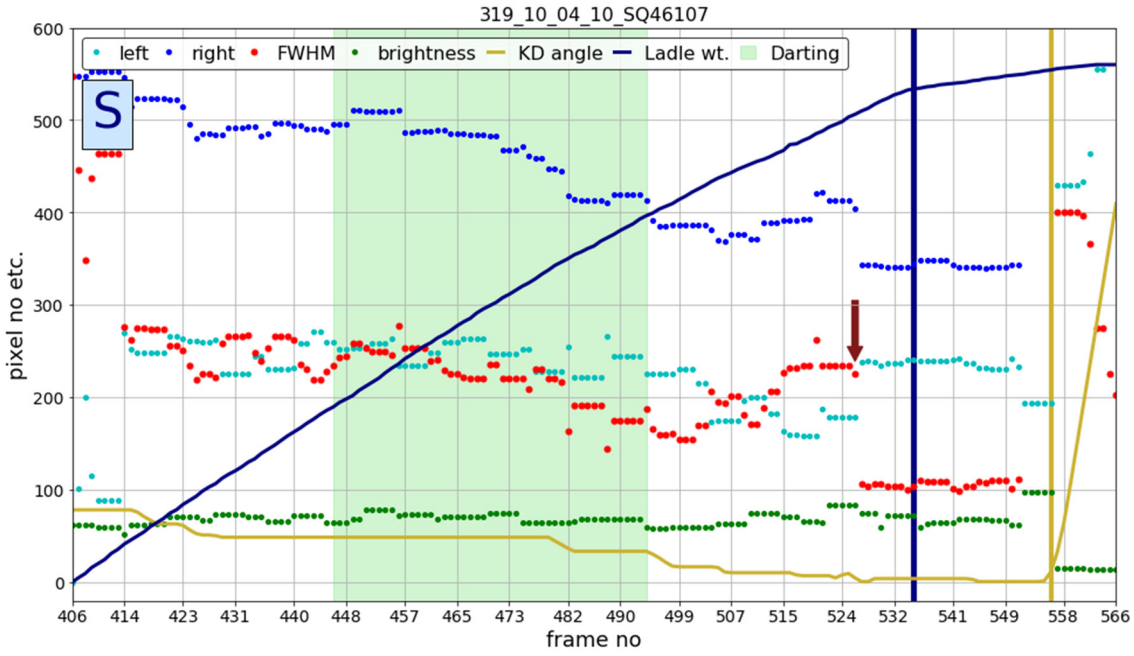

Figure 7. Summary and visualization of several information for the 319th tapping operation. Symbols: "left", light blue dots: left position of tapping stream; "right", dark blue dot: right position of tapping stream; "FWHM", red squares: calculated difference between "left" and "right"; "KD angle": angle of converter tilted during tapping. "Ladle wt.": mass of melt in ladle; "Darting": time when the slag dart was dropped from the lance to time of submergence under the surface of the molten steel. Verticals: yellow: start of increase in KD angle (i.e., end of tapping); Indigo: inflection point of Ladle wt. curve. Arrow: decrease in width of steel stream (= seating of dart).

We first consider a successful dart insertion. During the 319th tapping, the position and size of the FWHM were calculated for object detection and tracking using a total of 575 frames, taken at 1 fps in one SDS video file (Figure 6). During this tapping, the position of the converter started to return to its original position in the 556th frame to finish tapping (Figure 7, pink line); this frame was used as the end point of time series analysis of the SDS image. The darting was inserted during the 445th frame, so the analysis began at that point. Sequential data were sought by overlapping 11 frames

(i.e., 445−455, 446−456, 447−457, …, 545−555, 546−556). In the 527th frame, the width of the tapping stream sharply decreased (Figure 6, arrow) because the slag dart had become properly seated in the tapping hole after the middle of the tapping operation. All cases that earned an arrow were judged to have succeeded in inserting the slag dart and a 'true' label (capital letter S) was assigned. More than one of these arrows can exist in one case, and these cases were also judged as successful.

Then we considered a failed dart insertion. During the 69th (Figure 8) tapping stream, the thickness of the tapping stream initially widened, then became narrow toward the end, and the brightness of the pouring stream was not consistent. For example, when ferroalloy was added to adjust the alloy composition of the molten steel during the initial stage of tapping, the brightness of the tapping stream suddenly increased significantly. In contrast, during the tapping operation, the brightness of the molten steel suddenly decreased as the slag darts were inserted into the tapping hole. A decrease or increase in the brightness of molten steel in the SDS image provides behavioral information in tracking an object or the tapping stream. During training of an ANN that detects and tracks pouring stream in SDS images, both information about the width and information about the brightness are important. In tapping stream 69, the thickness of molten steel slightly decreased and then increased in the 448th to 455th frames (Figure 8). The absence of a pattern in which the width of the pouring stream sharply decreased means that the slag dart had not become properly seated in the tapping hole. This result for case 69 indicates that the algorithm can detect a failure of dart insertion by recognizing and tracking the molten steel in the SDS images.

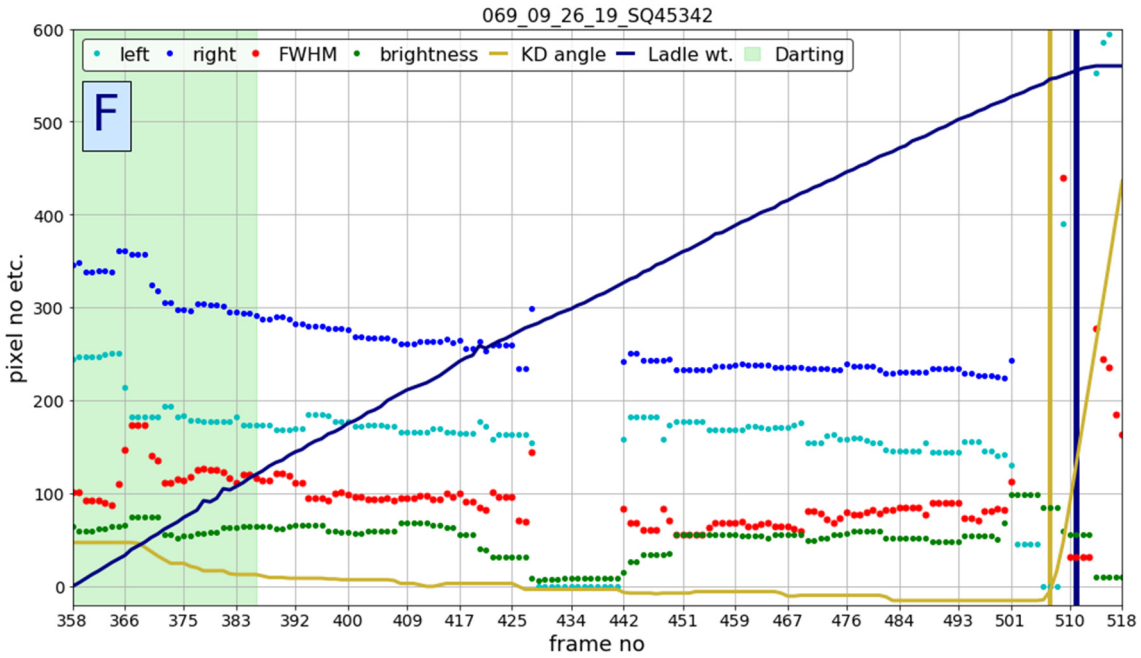

Figure 8. Summary and visualization of several information for the 69th tapping operation. Symbols: "left", light blue dots: left position of tapping stream; "right", dark blue dot: right position of tapping stream; "FWHM", red squares: calculated difference between "left" and "right"; "KD angle": angle of converter tilted during tapping. "Ladle wt.": mass of melt in ladle; "Darting": time when the slag dart was dropped from the lance to time of submergence under the surface of the molten steel. Verticals: yellow: start of increase in KD angle (i.e., end of tapping); Indigo: inflection point of Ladle wt. curve.

The ladle weight detected by the sensor was monitored in real time during tapping; for this purpose, a sensor was attached to the ladle car (Figures 7 and 8). Before using the SDS image, the ladle weight data were used as a reference to determine the success or failure of the slag dart injection. For example, first differentiated the slope in the graph of the ladle weight after slag dart input (Figures 7 and 8 indigo line), the identified point at which the slope rapidly increased. This change occurred in to 537th frame in Tapping 319 and the 512th frame in Tapping 69. In Tapping 319, the indigo vertical line was located in the frame ahead of the pink vertical line that indicates the end of the tapping, so dart injection judged to be successful (Figure 7). In Tapping 69, the indigo line followed the pink line during the tapping; this sequence indicates that the width of the molten steel did not change after input of slag dart, so dart injection was judged to have failed (Figure 8).

The success or failure of slag dart injection could be judged effectively using the weight of the ladle. However, this method requires detection of the inflection point of the time-curve of ladle weight, and therefore can only detect this point after it has happened, so the judgement must always be late. In the case of Tapping 319, the method that used the ladle weight to decide the success or failure of the slag dart injection was 10 s slower than the method that used the SDS image. A difference of 10 s in determining the completion time of tapping and moving a converter that contains approximately 250 tons of molten steel would seriously degrade operation of converter facilities at a production site. In addition, the converter is large and massive, so tilting work during tapping could not be done quickly. If the completion of tapping is delayed in the normal operation environment, the amount of slag that flows out during the tapping operation increases at least in proportion to the amount of delay. These results indicate that determining the success or failure of the slag dart input using the SDS image is effective in determining the time of completion of tapping. This quick identification of the completion of tapping can help to minimize the amount of slag that flows into the ladle from the converter.

Figure 9. Sample images of dart injection (left) and tapping stream from BOF (right).

Basically, the SDS is equipment to minimize slag carry-over while monitoring the amount of slag entrained in the molten steel during tapping. However, the SDS is intended for real-time response to carried-over slag, and not for proactive response to minimize slag carry-over. In contrast, injection of slag darts during tapping can help to minimize the slag by providing workers with information on the completion time of tapping in advance. The video of dart input provides behavior information in which the slag dart was dropped from the robot arm and was submerged under the molten steel surface, but

cannot indicate that the slag dart has inserted firmly in the tap hole, because this action occurs under the molten steel surface (Figure 9). For example, in one case, the slag dart was properly seated on the tap, but could not block the tap hole due to the low durability of the slag dart. Although the shape in which the slag dart was put into the molten steel was a success, failures often occurred, such as change in the behavior of the slag dart under the molten steel.

To compare the method that uses the SDS image to the method in which the operator makes a judgement by monitoring the dart image [11], the two judgement results during the same period were compared. About 6% of the judgment results using the dart image were misclassified. This misclassification rate means that 114 of the 1900 data collected over two months were incorrect, and that a total of 684 incorrect decisions occurred in one year. The failures and successes among the collected data were 6 and 94%, so assuming the same proportions, 41 were hits that were erroneously judged to be misses (i.e., FN) and 643 of the incorrect decisions were misses that were erroneously judged to be hits (i.e., FP). Each of the 41 FNs caused premature stopping of the tapping operation, and therefore a decrease in the amount of molten steel taken from the converter to the ladle. In contrast, the 643 FPs caused the tapping operation to last longer than it should, and therefore increased in the amount of molten steel taken from the converter to the ladle, and secondary refining was conducted unnecessarily. The additional cost caused by the secondary refining operation was approximately $US 0.5 million/year, as estimated from the average costs of labor, materials, and equipment operation. This direct cost and other indirect costs including reduced productivity were imposed on the steel mill by to the misclassification of dart hits. Considering that the SDS system made no errors, it is more effective than the method using the dart image to determine whether the slag dart is hit, and therefore can reduce the operating cost of this mill by at least $US 0.5 million.

7. Conclusions

When recognizing an object on a video screen, people do use all information including the background, but focus on the part they are interested in, and come to a conclusion by synthesizing the sequences of observation of the part. This paper proposed a method to automatically determine whether the slag dart was properly seated on the tap hole, by recognizing and tracking the tapping stream in the video collected in real time from the SDS. This method extracted quantitatively the width and brightness information of molten steel during tapping, and supported detection and tracking of the behavior of the tapping stream in the SDS images. Given operational data such as ladle weight and tilt angle, a new system was developed to determine results of the dart injection in real time for effective end of tapping according to various operation conditions.

This system can be used to minimize slag carry-over and determine the end time of tapping by discriminating slag dart input in the converter tapping process. First, the system is stable and accurate to prevent slag carry-over during tapping. The dart-input determination using the SDS can minimize the amount of carried-over slag by responding correctly to situations in which the dart behavior changes under the molten steel surface in the converter, i.e., in an area that cannot be observed. Second, the system can eliminate the need for human judgment of the slag dart input with dart images during tapping. The method that uses the SDS image has a lower misclassification rate than the method that uses the dart image, so the operator's work load can be reduced. In addition, the system can improve the productivity of the steelworks by reducing the cost of secondary refining caused by misjudgment of whether the dart entered the hole. Third, the decision on the end point of tapping is determined by

the amount of slag leakage detected by SDS, but the rate at which the tapping operation is terminated earlier than usual can be high depending on the operator's experience in the real field. A high-precision identification of the completion time of tapping can contribute to increased production of molten steel from converters by avoiding errors caused by the different operation patterns of different workers. Finally, object detection and tracking technology that use computer vision and ANN can also observe and analyze the behavior of tapping stream after the dart is inserted. This ability can contribute to an effective process control and management by linking the task of determining the dart input, to decision about the end point of tapping.

Some gaps remain between knowledge obtained from our findings and the field-adaptability of the proposed model; further research to extend and test the proposed model might reduce or eliminate these gaps. First, the developed system used field data obtained only over two months. Increase in the field-applicability of the proposed model requires additional training and parameter updates using data collected over a long period of time. Therefore, field operators have been collecting and storing data of slag dart and SDS every six months. For the stable field application of the proposed system, the model conformity must be tested using data from more than one year. In addition, the time of determining whether the slag dart input using images of SDS is hit is relatively later than that of the method using the dart image. A late judgment of the slag dart input might degrade the judgment on the completion of the tapping. Therefore, research must determine the latest time that field operators can agree on when judging the success or failure of dart input. The bottom line should be determined by analyzing the pattern of the failure point, in consultation with various experts who had more than 20 years of field experience. Finally, in this study, the behavior of the tapping stream, that is, the change in width, was directly calculated from SDS images and used as an input to the RNN model. An attention model could be included in the designed model for further improvement.

Acknowledgments

We would like to express our special thanks of gratitude to POSCO for allowing us to use SDS and dart image data.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. S. Xie, T. Chai, Prediction of BOF endpoint temperature and carbon content, in *Processing of 14th IFAC World Congress*, Academic Press, **32** (1999), 7039−7043. https://doi.org/10.1016/S1474-6670(17)57201-8
2. Z. Wang, Q. Liu, H. Liu, S. Wei, A review of end-point carbon prediction for BOF steelmaking process, *High Temp. Mater. Process.*, **39** (2020), 653−662. https://doi.org/10.1515/htmp-2020-0098
3. A. V. Luk'yanov, A. V. Protasov, B. A. Sivak, A. P. Shchegolev, Making BOF steelmaking more efficient based on the experience of the Cherepovets Metallurgical Combine, *Metallurgist*, **60** (2016), 248–255. https://doi.org/10.1007/s11015-016-0282-y

4. T. S. Naidu, C. M. Sheridan, L. D. Dyk, Basic oxygen furnace slag: review of current and potential uses, *Miner. Eng.*, **149** (2020), 106234. https://doi.org/10.1016/j.mineng.2020.106234
5. E. Belhadj, C. Diliberto, A. Lecomte, Characterization and activation of Basic Oxygen Furnace slag, *Cem. Concr. Compos.*, **34** (2012), 34−40. https://doi.org/10.1016/j.cemconcomp.2011.08.012
6. P. C. Pistorius, Slag carry-over and the production of clean steel, *J. S. Afr. Inst. Min. Metall.*, **119** (2019), 557−561. http://dx.doi.org/10.17159/2411-9717/kn01/2019
7. A. Kamaraj, G. K. Mandal, S. P. Shanmugam, G. G. Roy, Quantification and analysis of slag carryover during liquid steel tapping from BOF vessel, *Can. Metall. Q.*, **61** (2022), 202−215. https://doi.org/10.1080/00084433.2022.2044688
8. M. Brämming, B. Björkman, C. Samuelsson, BOF process control and slopping prediction based on multivariate data analysis, *Steel Res. Int.*, **87** (2016), 301−310. https://doi.org/10.1002/srin.201500040
9. Z. Zhang, L. Bin, Y. Jiang, Slag detection system based on infrared temperature measurement, *Optik*, **125 (**2014), 1412−1416. https://doi.org/10.1016/j.ijleo.2013.08.016
10. P. Patra, A. Sarkar, A. Tiwari, Infrared-based slag monitoring and detection system based on computer vision for basic oxygen furnace, *Ironmak. Steelmak.*, **46** (2019), 692−697. https://doi.org/10.1080/03019233.2018.1460909
11. D. G. Hong, W. H. Han, C. H. Yim, Convolutional recurrent neural network to determine whether dropping slag dart fills the exit hole during tapping in a basic oxygen furnace, *Metall. Mater. Trans. B*, **52** (2021), 3833–3845. https://doi.org/10.1007/s11663-021-02299-z
12. A. Kamaraj, G. K. Mandal, G. G. Roy, Control of slag carryover from the BOF vessel during tapping: BOF cold model studies, *Metall. Mater. Trans. B*, **50** (2019), 438–458. https://doi.org/10.1007/s11663-018-1432-3
13. W. S. Howanski, T. Kalep, T. Swift, Optimizing BOF slag control through the application of refractory darts, *Iron Steel Technol.*, **3** (2006), 36−43.
14. B. Chakraborty, B. K. Sinha, Development of caster slag detection system through imaging technique, *Int. J. Instrum. Technol.*, **1** (2011), 84−91. https://doi.org/10.1504/IJIT.2011.043599
15. Z. Zhang, Q. Li, L. Yan, Slag detection system based on infrared thermography in steel making industry, *Recent Pat. Signal Process.*, **5** (2015), 16−23. https://doi.org/10.2174/2210686305666150930230548
16. M. Tanaka, D. Mazumdar, R. I. L. Guthrie, Motions of alloying additions during furnace tapping in steelmaking processing operations, *Metall. Mater. Trans. B*, **24**, (1993), 639−648. https://doi.org/10.1007/BF02673179
17. P. Hammerschmid, K. H. Tacke, H. Popper, L. Weber, M. Bubke, K. Schwerdtfeger, Vortex formation during drainage of metallurgical vessels, *Ironmak. Steelmak.*, **11** (1984), 332−339.
18. D. You, C. Bernhard, P. Mayer, J. Fasching, G. Kloesch, R. Rössler, et al., Modeling of the BOF tapping process: the reactions in the ladle, *Metall. Mater. Trans. B*, **52** (2021), 1854−1865. https://doi.org/10.1007/s11663-021-02153-2
19. A. Dahlin, A. Tilliander, J. Eriksson, P. G. Jönsson, Influence of ladle slag additions on BOF process performance, *Ironmak. Steelmak.*, **39** (2012), 378−385. https://doi.org/10.1179/1743281211Y.0000000021

20. C. M. Lee, I. S. Choi, B. G. Bak, J. M. Lee, Production of high purity aluminium killed steel, *Metall. Res. Technol.*, **90** (1993), 501–506. https://doi.org/10.1051/METAL/199390040501
21. K. K. Lee, J. M. Park, J. Y. Chung, S. H. Choi, S. B. Ahn, The secondary refining technologies for improving the cleanliness of ultra-low carbon steel at Kwangyang Works, *Metall. Res. Technol.*, **93** (1996), 503–509. https://doi.org/10.1051/METAL/199693040503
22. J. M. Park, C. S. Ha, Recent improvement of BOF refining at Kwangyang Works, *Metall. Res. Technol.*, **97** (2000), 729–735. https://doi.org/10.1051/METAL/200097060729
23. R. Usamentiaga, J. Molleda, D. F. Garcia, J. C. Granda, J. L. Rendueles, Temperature measurement of molten pig iron with slag characterization and detection using infrared c omputer vision, *IEEE Trans. Instrum. Meas.*, **61** (2012), 1149−1159. https://doi.org/10.110 9/TIM.2011.2178675
24. S. C. Koria, U. Kanth, Model studies of slag carry-over during drainage of metallurgical vessels, *Steel Res. Int.*, **65** (1994), 8−14. https://doi.org/10.1002/srin.199400919
25. A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep learning for computer vision: a brief review, *Comput. Intell. Neurosci.*, **2018** (2018), 1−13. https://doi.org/10.115 5/2018/7068349
26. J. Suri, Computer vision, pattern recognition and image processing in left ventricle segmentation: the last 50 years, *Pattern Anal. Appl.*, **3** (2000), 209–242. https://doi.org/10.1007/s100440070008
27. V. H. Nguyen, V. H. Pham, X. Cui, M. Ma, H. Kim, Design and evaluation of features and classifiers for OLED panel defect recognition in machine vision, *J. Inf. Telecommun.*, **1** (2017), 334−350. https://doi.org/10.1080/24751839.2017.1355717
28. X. Guo, X. Liu, M. K. Gupta, Machine vision-based intelligent manufacturing using a novel dual-template matching: a case study for lithium battery positioning, *Int. J. Adv. Manuf. Technol.*, **116** (2021), 2531–2551. https://doi.org/10.1007/s00170-021-07649-4
29. M. Yazdi, B. Thierry, New trends on moving object detection in video images captured by a moving camera: a survey, *Comput. Sci. Rev.*, **28** (2018), 157−177. https://doi.org/10. 1016/j.cosrev.2018.03.001
30. R. Raguram, O. Chum, M. Pollefeys, J. Matas, J. Frahm, USAC: a universal framework for random sample consensus, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 2022−2038. https://doi.org/10.1109/TPAMI.2012.257
31. J. Ko, D. Fox, GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models, *Auton. Robot.*, **27** (2009), 75–90. https://doi.org/10.1007/s10514-009-9119-x
32. D. Sun, S. Roth, M. J. Black, Secrets of optical flow estimation and their principles, in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (2010), 2432−2439. https://doi.org/10.1109/CVPR.2010.5539939
33. T. Brox, J. Malik, Object segmentation by long term analysis of point trajectories, in *Computer Vision – ECCV 2010* (eds. K. Daniilidis, P. Maragos, N. Paragios), Springer, Berlin, Heidelberg, **6315** (2010), 282−295. https://doi.org/10.1007/978-3-642-15555-0_21
34. R. M. Fikri, B. Kim, M. Hwang, Waiting time estimation of hydrogen-fuel vehicles with YOLO real-time object detection, in *Information Science and Applications* (eds. K. Kim and H. Y. Kim), Springer, Singapore, **621** (2020), 229−237. https://doi.org/10.1007/978-981-15-1465-4_24
35. J. Kim, J. Y. Sung, S. Park, Comparison of faster-RCNN, YOLO, and SSD for real-time vehicle type recognition, in *2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia)*, **2020** (2020), 1−4. https://doi.org/10.1109/ICCE-Asia49877.2020.9277040

36. J. Li, X. Liang, S. Shen, T. Xu, J. Feng, S. Yan, Scale-aware fast R-CNN for pedestrian detection, *IEEE Trans. Multimedia*, **20** (2018), 985−996. https://doi.org/10.1109/TMM.2017.2759508
37. Q. C. Mao, H. M. Sun, Y. B. Liu, R. S. Jia, Mini-YOLOv3: real-time object detector for embedded applications, *IEEE Access*, **7** (2019), 133529−133538. https://doi.org/10.1109/ACCESS.2019.2941547
38. X. Cheng, J. Yu, RetinaNet with difference channel attention and adaptively spatial feature fusion for steel surface defect detection, *IEEE Trans. Instrum. Meas.*, **70** (2021), 1−11. https://doi.org/10.1109/TIM.2020.3040485
39. R. Gai, N. Chen, H. Yuan, A detection algorithm for cherry fruits based on the improved YOLO-v4 model, *Neural Comput. Appl.*, **2021** (2021). https://doi.org/10.1007/s00521-021-06029-z
40. G. Yang, W. Feng, J. Jin, Q. Lei, X. Li, G. Gui, et al., Face mask recognition system with YOLOV5 based on image recognition, in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, **2020** (2020), 1398−1404. https://doi.org/10.1109/ICCC51575.2020.9345042
41. S. J. Lee, W. K. Kwon, G. G. Koo, H. E Choi, S. W. Kim, Recognition of slab identification numbers using a fully convolutional network, *ISIJ Int.*, **58** (2018), 696−703. https://doi.org/10.2355/isijinternational.ISIJINT-2017-695
42. H. B. Wang, S. Wei, R. Huang, S. Deng, F. Yuan, A. Xu, et al., Recognition of plate identification numbers using convolution neural network and character distribution rules, *ISIJ Int.*, **59** (2019), 2041−2051. https://doi.org/10.2355/isijinternational.ISIJINT-2019-128
43. M. Chu, R. Gong, Invariant feature extraction method based on smoothed local binary pattern for strip steel surface defect, *ISIJ Int.*, **55** (2015), 1956−1962. https://doi.org/10.2355/isijinternational.ISIJINT-2015-201
44. J. Yang, W. Wang, G. Lin, Q. Li, Y. Sun, Y. Sun, Infrared thermal imaging-based crack detection using deep learning, *IEEE Access*, **7** (2019), 182060−182077. https://doi.org/10.1109/ACCESS.2019.2958264
45. A. Choudhury, S. Pal, R. Naskar, A. Basumallick, Computer vision approach for phase identification from steel microstructure, *Eng. Comput.*, **36** (2019), 1913−1933. https://doi.org/10.1108/EC-11-2018-0498
46. D. Boob, S. S. Dey, G. Lan, Complexity of training ReLU neural network, *Discrete Optim.*, **2020** (2020), 100620. https://doi.org/10.1016/j.disopt.2020.100620
47. A. P. Shukla, M. Saini, Moving object tracking of vehicle detection: a concise review, *Int. J. Signal Process. Image Process. Pattern Recognit.*, **8** (2015), 169−176. https://doi.org/10.14257/IJSIP.2015.8.3.15
48. H. Goszczynska, A method for densitometric analysis of moving object tracking in medical images, *Mach. Graphics Vision Int. J.*, **17** (2008), 69−90. https://doi.org/10.5555/1534494.1534499
49. W. Budiharto, E. Irwansyah, J. S. Suroso, A. A. S. Gunawan, Design of object tracking for military robot using PID controller and computer vision, *ICIC Express Lett.*, **14** (2020), 289−294. https://doi.org/10.24507/icicel.14.03.289
50. J. F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernalized correlation filters, *IEEE Trans. Pattern Anal. Mach. Intell.*, **37** (2015), 583−596. https://doi.org/10.1109/TPAMI.2014.2345390

51. A. Sherstinsky, Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network, *Phys. D*, **404** (2020). https://doi.org/10.1016/j.physd.2019.132306
52. J. C. Lin, Y. Shao, Y. Djenouri, U. Yun, ASRNN: a recurrent neural network with an attention model for sequence labeling, *Knowledge-Based Syst.*, **212** (2021), 106548. https://doi.org/10.1016/j.knosys.2020.106548
53. Y. Shao, J. C. Lin, G. Srivastava, A. Jolfaei, D. Guo, Y. Hu, Self-attention-based conditional random fields latent variables model for sequence labeling, *Pattern Recognit. Lett.*, **145** (2021), 157−164. https://doi.org/10.1016/j.patrec.2021.02.008
54. J. C. Lin, Y. Shao, J. Zhang, U. Yun, Enhanced sequence labeling based on latent variable conditional random fields, *Neurocomputing*, **403** (2020), 431−440. https://doi.org/10.1016/j.neucom.2020.04.102
55. H. Ling, J. Wu, L. Wu, J. Huang, J. Chen, P. Li, Self residual attention network for deep face recognition, *IEEE Access*, **7** (2019), 55159−55168. http://doi.org/10.1109/ACCESS.2019.2913205
56. Y. Li, Y. Liu, W. G. Cui, Y. Z. Guo, H. Huang, Z. Y. Hu, Epileptic seizure detection in EEG signals using a unified temporal-spectral squeeze-and-excitation network, *IEEE Trans. Neural Syst. Rehabil. Eng.*, **28** (2020), 782−794. https://doi.org/10.1109/TNSRE.2020.2973434
57. J. Wang, X. Qiao, C. Liu, X. Wang, Y. Liu, L. Yao, et al., Automated ECG classification using a non-local convolutional block attention module, *Comput. Methods Programs Biomed.*, **203** (2021), 106006. https://doi.org/10.1016/j.cmpb.2021.106006
58. X. Lin, Q. Huang, W. Huang, X. Tan, M. Fang, L. Ma, Single image deraining via detail-guided efficient channel attention network, *Comput. Graphics*, **97** (2021), 117−125. https://doi.org/10.1016/j.cag.2021.04.014
59. F. Wu, Y. Wang, A method for detecting the slag transferring from ladle to tundish based on video system, *Ind. Control Comput.*, **18** (2005) 38−47.
60. P. Y. Li, T. Gan, G. Z. Shen, Embedded slag detection method based on infrared thermographic, *J. Iron Steel Res.*, **22** (2010), 59−63.
61. D. P. Tan, P. Y. Li, X. H. Pan, Application of improved HMM algorithm in slag detection system, *J. Iron Steel Res. Int.*, **16** (2009), 1–6. https://doi.org/10.1016/S1006-706X(09)60001-7
62. Z. Zhang, Q. Li, L. Yan, Slag detection system based on infrared thermography in steelmaking industry, *Recent Pat. Signal Process. (Discontinued)*, **5** (2015), 16−23. https://doi.org/10.2174/2210686305666150930230548
63. B. Chakraborty, B. K. Sinha, Development of caster slag detection system through imaging technique, *Int. J. Instrum. Technol.*, **1** (2011), 84−91. https://doi.org/10.1504/IJIT.2011.043599
64. P. C. Pistorius, Slag carry-over and the production of clean steel, *J. S. Afr. Inst. Min. Metall.*, **119** (2019), 557−561. http://dx.doi.org/10.17159/2411-9717/kn01/2019
65. M. A. Merkx, J. O. Bescós, L. Geerts, E. M. H. Bosboom, F. N. van de Vosse, M. Breeuwer, Accuracy and precision of vessel area assessment: manual versus automatic lumen delineation based on full-width at half-maximum, *J. Magn. Reson. Imaging*, **36** (2012), 1186−1193. https://doi.org/10.1002/jmri.23752
66. N. K. Manaswi, Understanding and working with keras, in *Deep Learning with Applications Using Python*, Apress, Berkeley, CA, **2018** (2018), 31−43. https://doi.org/10.1007/978-1-4842-3516-4_2

67. Z. Deng, D. Weng, X. Xie, J. Bao, Y. Zheng, M. Xu, et al., Compass: towards better causal analysis of urban time series, *IEEE Trans. Visual Comput. Graphics*, **28** (2022), 1051−1061. https://doi.org/10.1109/TVCG.2021.3114875
68. D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, M. N. Do, Fast global image smoothing based on weighted least squares, *IEEE Trans. Image Process.*, **23** (2014), 5638−5653. https://doi.org/10.1109/TIP.2014.2366600
69. F. Wang, H. Liu, J. Cheng, Visualizing deep neural network by alternately image blurring and deblurring, *Neural Networks*, **97** (2018), 162−172. https://doi.org/10.1016/j.neunet.2017.09.007
70. D. G. Hong, S. H. Kwon, C. H. Yim, Exploration of machine learning to predict hot ductility of cast steel from chemical composition and thermal conditions, *Met. Mater. Int.*, **27** (2020), 298−305. https://doi.org/10.1007/s12540-020-00713-w
71. S. Patro, K. Sahu, Normalization: a preprocessing stage, preprint, arXiv:1503.06462.
72. A. K. Dubey, V. Jain, Comparative study of convolution neural network's Relu and leaky-Relu activation functions, in *Applications of Computing, Automation and Wireless Systems in Electrical Engineering* (eds. S. Mishra, Y. Sood, A. Tomar), Springer, Singapore, **553** (2019), 873−880. https://doi.org/10.1007/978-981-13-6772-4_76
73. A. Menon, K. Mehrotra, C. K. Mohan, S. Ranka, Characterization of a class of sigmoid functions with applications to neural networks, *Neural Networks*, **9** (1996), 819−835. https://doi.org/10.1016/0893-6080(95)00107-7
74. J. J. Jijesh, Shivashankar, Keshavamurthy, A supervised learning based decision support system for multi-sensor healthcare data from wireless body sensor networks, *Wireless Pers. Commun.*, **116** (2021), 1795–1813. https://doi.org/10.1007/s11277-020-07762-9

AIMS Press