ISIJ International, Vol. 59 (2019), No. 3, pp. 481–488

Improving Blast Furnace Raceway Blockage Detection. Part 3: Visual Detection Based on Tuyere Camera Images

Stefan PUTTINGER[1)]* and Hugo STOCKER[2)]

1) Department of Particulate Flow Modelling, Johannes Kepler University, Altenbergerstrasse 69, Linz, 4040 Austria.
2) voestalpine Stahl Donawitz GmbH, Kerpelystrasse 199, Leoben, 8700 Austria.

(*Received on July 26, 2018; accepted on October 18, 2018; J-STAGE Advance published date: December 19, 2018*)

The present paper is part 3 of a paper series discussing raceway blockages and various approaches for automated detection based on blast furnace (BF) plant data. While part 1[1)] gave an overview on the different appearances of raceway blockages and part 2[2)] focussed on signal processing of hot blast data, this third part discusses various approaches for image processing of tuyere camera data. The visual impression of raceway blockages strongly varies between different events. This makes automated detection based on digital image processing of tuyere cameras a difficult task. On one hand the image processing algorithm should be robust and easy to tune for different tuyeres or different blast furnaces, on the other hand it should be fast enough, so that all tuyeres of a blast furnace can be processed on-line with a sufficiently high image frame rate. While algorithms optimized for motion detection fail due to the lack of a homogeneous background, adaptive thresholding of the grey-level histograms delivers useful results. Due to the nature of chaotic motion of coke particles inside the raceway also line based processing methods can extract the information from tuyere images in a sufficient manner and are very fast with regards to online implementation in a process control system. However, image processing of tuyere camera data has some disadvantages compared to the signal processing of hot blast data as discussed in part 2 of this paper.

KEY WORDS: blast furnace; tuyere monitoring; raceway; blockage detection; image processing.

1. Introduction

The monitoring of blast furnace raceways can be tackled by either analyzing available plant data like pressure or flow rate data or by attaching cameras to the inspection glasses at the end of the blow pipes. Visual raceway monitoring is not new, but the rapid development of camera technology over the past two decades has opened new possibilities and the constantly decreasing prices of camera hardware now allow a complete camera equipment for all tuyeres of a blast furnace at moderate costs. In addition, increasing CPU power enables sophisticated online processing of tuyere video data at reasonable frame rates.

While most studies published in literature focus on characterising the coal plume to obtain information about constant coal discharge and dispersion,[3–7)] this study is dedicated on the question how we can detect raceway blockages in a fully automated manner by digital image processing of tuyere camera images. However, this paper is not meant to be an extensive overview on image processing algorithms, nor will we present an ultimate solution for the purpose of blockage detection. It should be seen as a discussion of various aspects and difficulties to extract information from tuyere camera images. Even though the task of blockage detection might not be ideally solved by processing images from tuyere cameras, they can also be used for other purposes like erosion monitoring of the tuyeres or checking the correct positioning and burn-out of the PCI lances. Thus, blockage detection might only be one image processing task amongst others and the benefits of using tuyere cameras should be seen in a wider context that cannot be entirely covered here.

* Corresponding author: E-mail: stefan.puttinger@jku.at
DOI: https://doi.org/10.2355/isijinternational.ISIJINT-2018-532

2. Recording of Images

The current market of industry cameras offers a huge selection of mid-range priced cameras of high quality. For the purpose of raceway blockage detection, the main features a camera must have are a sensitive chip with high dynamic range and a flexible exposure time setting. The former feature is needed to adjust the camera to the radiation level of a specific tuyere and to avoid pixel saturation, the latter is necessary to freeze the fast motion of coke particles in the raceway and avoid blurred images. Exposure times in the range of t_{ex} = 0.1 up to 1 ms have shown to deliver high quality images, depending on the type of additional filters mounted with the tuyere spyglass or camera lens. If no IR filter is applied t_{ex} must be reduced to 0.1...0.25 ms to avoid chip saturation. When using an IR filter t_{ex} can be increased by roughly one order of magnitude. The image data discussed in the following sections were recorded at BF1 at voestalpine Stahl Donawitz GmbH with a Photon Focus MV1 CMOS camera which was temporarily installed on different tuyeres around the blast furnace and ran for several hours. In this way more than 100.000 images were recorded covering about 30 hours of BF operation. The images have been recorded at one frame per second. In parallel the hot blast data was stored from the process control system for comparison.

3. Classification of Blockage Events and Image Processing Requirements

A detailed discussion of various blockage events and raceway appearances has been presented in part 1 of this paper. **Figures 1**–**6** show examples of the most important cases and their grey level histograms. Figure 1 shows an example of ordinary raceway behaviour. While Fig. 2 shows a complete blockage, Fig. 3 is a typical case for an only partly blocked tuyere. Figure 4 is quite similar to Fig. 2 but the blocking structure is located deeper inside the raceway and there is some remaining space for coke particles to move around. The same is the case in Fig. 5 but the blocking structure is actually not visible. It can be assumed that a low porosity zone is also located deeper inside the raceway but covered by the coal plume. For completeness Fig. 6 shows a situation where the PCI lance has been switched off and the raceway is clear and free of coal particles.

Based on the presented examples we can define three classes of blockage conditions which should be identified by an image processing algorithm and distinguished from ordinary operation or no coal injection periods:

Type 1: Visually obvious blockage of more than 50% of the tuyere area. Figures 2 and 4 would correspond to this type of blockage.

Type 2: Visually obvious blockage of less than 50% of the tuyere area (c.f. Fig. 3).

Type 3: Blockage event which is not clearly identifiable due to a larger distance from the tuyere. The blocking

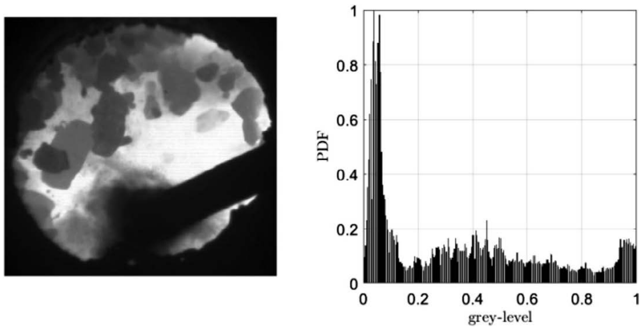


Fig. 1. Example of normal raceway operation and its corresponding grey-level histogram.

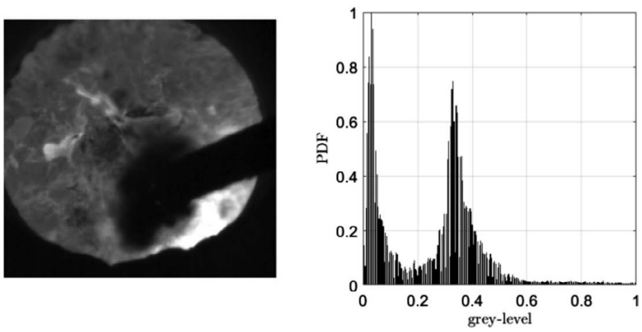


Fig. 2. Example of complete blockage close to the tuyere and its grey-level histogram.

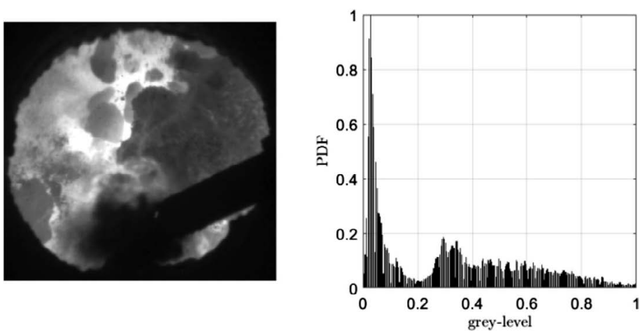


Fig. 3. Example of partly blocked tuyere and its grey-level histogram.

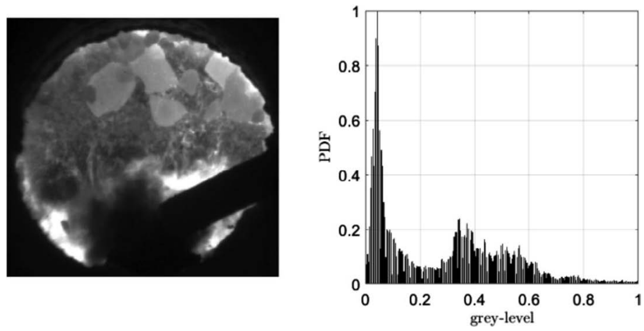


Fig. 4. Large blockage deeper inside the raceway with free moving coke particles in front of the blocking structure.

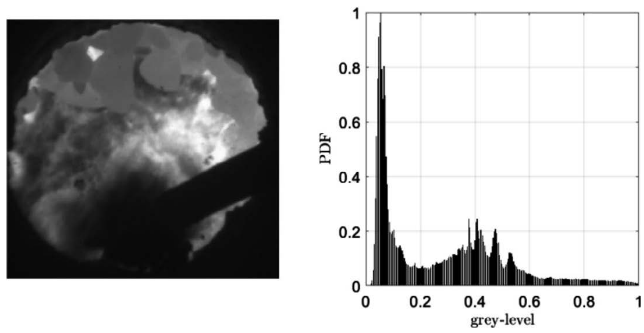


Fig. 5. Tuyere blocked located further inside the raceway. The low permeability zone does not show clear textures like in Figs. 2, 3 and 4.

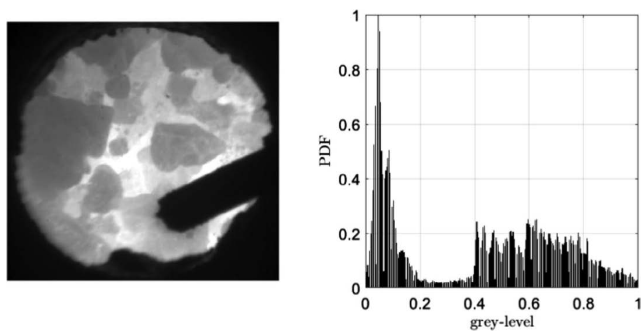


Fig. 6. Example of raceway with deactivated PCI branch and its corresponding grey-level histogram.

structure can be partly hidden by coke particles or the coal plume (Fig. 5).

The discussion of various blockage events can be summarized to the following requirements for a fully automated tuyere image processing:

- The algorithm must be able to distinguish normal raceway operation (with floating coke particles) from full or partly blockages close to the tuyere as well as deeper inside the raceway.
- The algorithm must be either independent or self-adaptive to the mean luminosity level at a specific tuyere (which is equivalent to the radiation level). These levels may vary due to inhomogeneous operation of the blast furnace as well as different conditions of the spy glasses and filters because of material ageing or dust deposition.
- ROI based processing must be independent of absolute positions and pixel resolutions. Hence, all spatial calculations must be done on a relative basis.
- Due to blast furnace maintenance issues the orientation and lateral and transversal positions of the camera relative to the tuyere will inevitable change during long term operation. Hence, this must be corrected automatically on a regular level.

In this paper we will focus on the first task. The last three points will not be covered in detail in the further discussion as they are strongly depended on the chosen type of camera and lenses and the mechanical construction used for mounting the cameras on the tuyere blowpipes.

It should be emphasized that for the purpose of testing and comparing different image processing methods, the event detection algorithms should be as sensitive as possible, regardless if a blockage is relevant for blast furnace operation or not. Only if most of the suspicious raceway conditions are detected one can built a sound statistics of blockage events which can later be used to find correlations with other data from blast furnace operation (*e.g.* burden charging, coke quality *etc.*). For a final online implementation of the blockage detection in the process control system, the sensitivity of the algorithms can be reduced as needed according to the frequency and significance of the events on a specific tuyere based on the existing knowledge for a specific blast furnace.

4. Data Processing Framework, Test Data and Quality Assessment

To allow an efficient testing of various algorithms for digital image processing, the test-bench presented in part 1 was extended. Due to the modular concept of the test-bench the image processing algorithms could be implemented as a second group of plugin functions beside the signal processing algorithms discussed in part 2. To avoid confusion with the signal processing algorithms, the image processing algorithms are labelled ‘B1’ to ‘B5’. All algorithms were tested on the same series of 10 000 tuyere images that was used to validate the signal processing strategies. For completeness, **Fig. 7** shows the hot blast signal of the tuyere f_T, the overall BF pressure signal f_P that indicates the switching events of the hot blast stoves and the manually defined reference signal f_R. The marked events are the same as discussed in part 1 and 2. Events #3 and #6 are major blockages and for event #3 the currently implemented flow rate thresholding has deactivated the PCI lance at this tuyere (in fact Figs. 5 and 6 were taken from event #3 before and after the shutdown of the PCI branch).

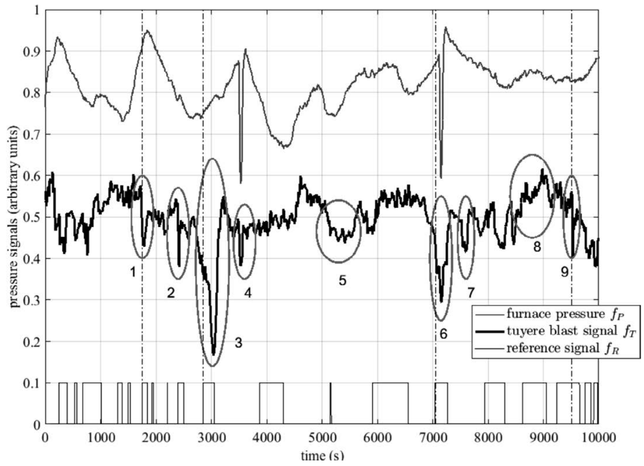


Fig. 7. Hot blast flow rate signal f_T for the time span covered by the images of the test sequence. Some blockage events have been marked for discussion and the reference signal f_R obtained by manually checking the tuyere camera images. In addition, the overall BF pressure signal f_p is plotted indicating the switching events of the hot blast stoves.

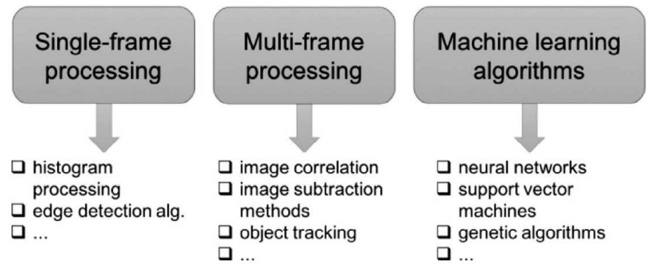


Fig. 8. Classification of image processing algorithms.

The task of blockage detection always boils down on obtaining an analog result signal f_S giving appropriate information on the current raceway condition. From this analog signal periods with blockages are then extracted via thresholding to convert f_S to a digital signal f_B which can be used to shut down certain PCI branches. Hence, the resulting data is finally the same as for the signal processing algorithms of part 2 and we can also apply the same quality assessment for the image processing results.

5. Discussion of Different Image Processing Algorithms and Blockage Detection Results

In this section we will discuss different algorithms for image processing and if they can fulfil the requirements previously defined. Generally, the algorithms can be divided in three families (**Fig. 8**). The simpler approaches process each image frame individually to extract certain features. This class of algorithms is mainly based on so-called morphological image segmentation. The second class is based on finding differences or correlations in two or more consecutive image frames. Such algorithms are commonly used in the field of motion detection. As a third class one could see the family of machine learning algorithms. The field of self-optimizing algorithms definitely offers some opportunities for applications like the one discussed herein. However, from an operator's perspective things should be kept as simple as possible and computationally efficient. Thus, we focus on the first two classes of algorithms in this project phase. At a later stage a machine learning tool could be wrapped around one of the image processing strategies discussed in the following sections to improve long term stability and compensate *e.g.* ageing effects on the inspection glasses due to dust deposition.

The presentation of the results is done in the same way as in part 2. The following figures show the tuyere pressure signal f_T as previously used for comparison, the analog result signal f_S of the processing algorithm and the digital result f_B after thresholding in comparison to the reference signal f_R. Likewise, **Table 1** gives an overview of the numeric results of the quality assessment. The runtime comparison given in Table 1 shall only be seen as a relative indicator for computational demand of the algorithms, as all computations have been done on a standard desktop PC without any kinds of runtime optimizations. An improved software implementation of the algorithms or the use of GPU based processing could certainly reduce the given numbers of Table 1 significantly.

5.1. Single Frame Based Processing

5.1.1. Algorithm B1: Adaptive Thresholding of Grey Level Histograms

The most intuitive approach is to test each single frame for its average grey level, as blockages of type 1 usually appear much darker than a normal raceway situation as can be seen in the histograms of Figs. 1 and 2. However, blockages of type 2 and 3 do not strictly follow this logic as they can also contain brighter areas shifting the average grey level to values similar of normal raceway behaviour (the histograms of Figs. 3 and 5 are quite similar to the histogram of Fig. 1). Thus, simply analysing the original grey level distribution will not deliver proper results. The grey level histogram must therefore be manipulated in a way that the features of interest are emphasized. This can be achieved by cutting off the dark grey level values by calculating a dynamic threshold value based on Otsu's method.[8] This method assumes two groups of pixel intensities and calculates a threshold level to separate these two groups in a way that the variance between these classes is maximized. Due to the dark surroundings of the tuyere as well as the PCI lance and coal plume, there is always one peak in the histogram at very low grey-levels. If a blockage is present a second peak at low grey-levels is formed (Fig. 2). Otsu's method therefore delivers a value separating these two peaks and delivering a value closer to 0 (on a normalized grey-level scale from 0 to 1). The second peak might be not so dominant for partial blockages (Fig. 3) or blockages superimposed by coke particles (Fig. 4) but still there is a lower number of bright pixels than in Figs. 1 and 6. **Figure 9** shows the processing results for Figs. 1 to 6 after masking with the threshold level calculated with Otsu's method and the remaining grey-level histograms. One can see that there is a significant difference in the cut-off level for cases with blockages except for the partially blockage, which is obviously more difficult to detect. Analysing the averaged grey level of the remaining histogram still gives no reliable information, as the overall brightness of the raceway may drift over several hours of BF operation. However, the number of remaining pixels after thresholding delivers a useful indicator for blockage detection and 11 out of 19 events can be captured by this signal (**Fig. 10**).

5.1.2. Algorithm B2: Pixel Gradient Evaluation

Calculating the pixel gradients in x and y direction for a single image is rather fast and delivers morphological information which is commonly used for edge detection. Precise edge detection of coke particles is hardly possible due to the high overlap of particles but during normal raceway operation the boundaries of the coke particles show strong luminosity gradients to the surrounding void area (c.f. Fig. 1). Inside the coke particles and in the gas phase the gradients are very small. In contrast, blockage structures only have a dark grey-level texture which comes along with small pixel gradients but covering a larger area of the image as can be seen from **Fig. 11**. The processing steps are as follows

1. mask out the PCI lance, coal plume and dark surrounding of the tuyere by cutting off all pixel values below 0.2
2. generate a second mask to cut off all pixels >0.65. This will eliminate the bright spots dominated by heat radiation and containing no gradient information

Table 1. Summary of quality measures for the tested blockage detection algorithms after processing the testcase.

Algorithm	Events calculated [a]	Events matched	Events missed	False positive events	Cum. time offset t_D (s)	$t_{s,ma}$ [b] (%)	$t_{s,mi}$ [c] (%)	$t_{s,fp}$ [d] (%)	Threshold level 'on' [e]	Threshold level 'off' [f]	CPU time per sample (ms)
B1	22	11	8	10	716	16.7	83.3	3.8	0.55	0.95	7.3
B2	25	7	12	12	352	25.1	74.9	5.7	0.35	0.75	105
B3	25	12	7	7	835	16.0	84.0	6.8	0.35	0.80	0.25
B4	15	7	12	6	758	14.1	85.9	4.8	0.65	0.90	2.9
B5	22	8	11	8	700	12.7	87.3	4.3	0.35	0.75	2.6

a) The reference signal contains 19 blockage events.
b) Number of time stamps with active blockage signal relative to the manually identified blockage time.
c) Relative number of missing active blockage signal based on the manually identified blockage time.
d) False active blockage signal relative to the total number of time stamps in the test data.
e) The blockage detection signal will be 'on' if the image processing result is smaller than the given value.
f) The blockage detection signal will be 'off' if the image processing result is bigger than the given value.

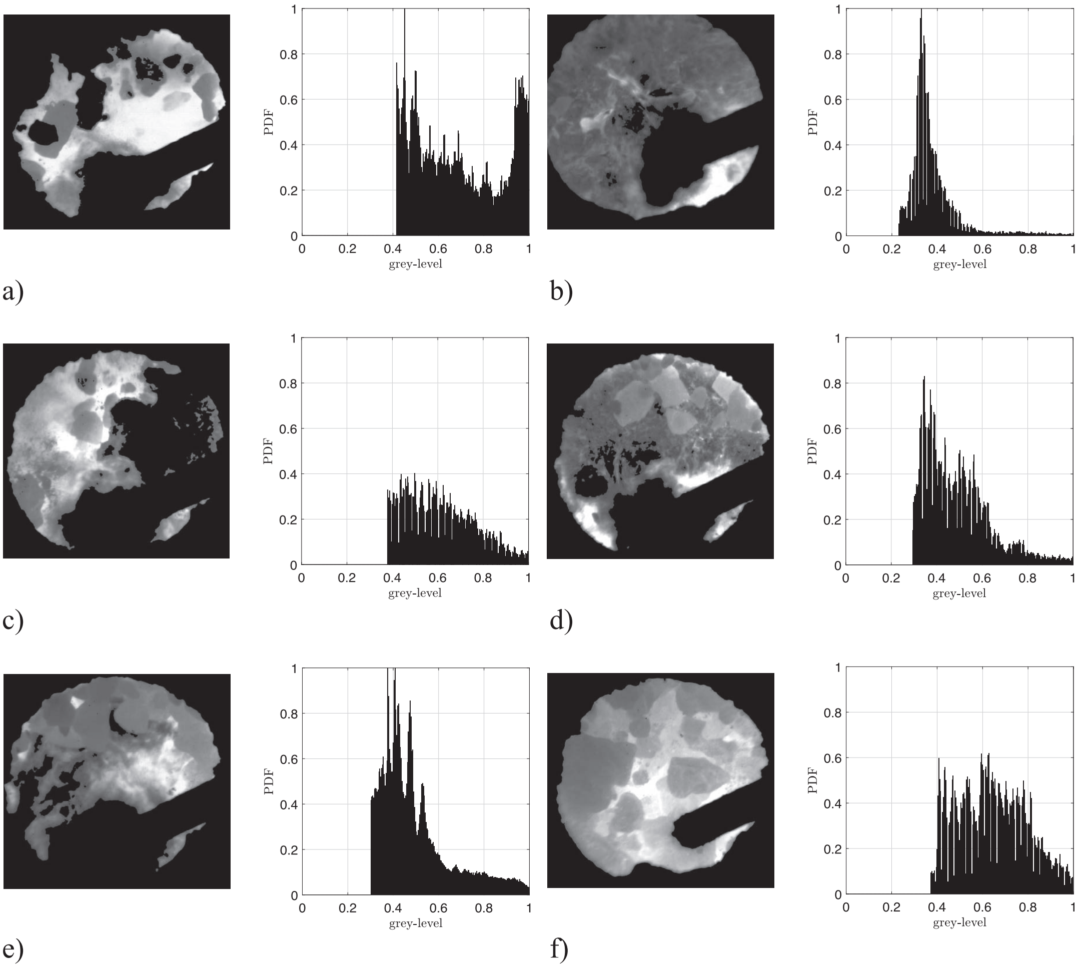


Fig. 9. Images after masking with dynamic threshold levels according to Otsu's method and the remaining grey-level histograms after masking.

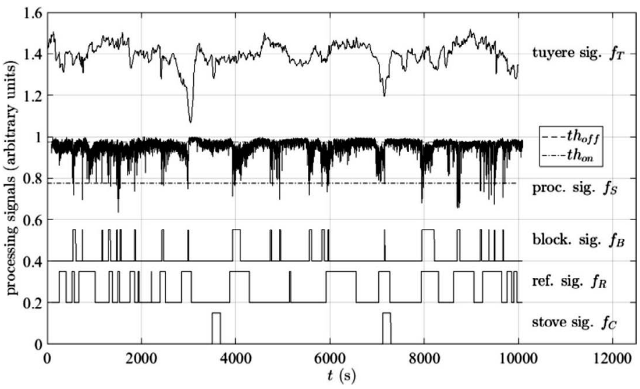


Fig. 10. Results of algorithm B1 (adaptive thresholding). f_T is the blast signal of the tuyere, and f_S is the result signal for the number of remaining pixels after adaptive thresholding. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B. Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

3. calculate the gradient magnitude of the image by a filter operation with a 3×3 Sobel kernel function.[9)]
4. apply the masks from step 1 and 2 to the gradient image
5. normalize the gradient image to the range 0..1.
6. cut-off all gradient levels below 0.3 (the low gradient levels are dominated by noise)
7. sum all remaining pixels

The results are shown in **Fig. 12** where f_S is the relative number of remaining pixels (related to the total pixel size of the image). The resulting signal of B2 appears more noisy compared to Fig. 10 and the gradient processing actually fails to detect the major blockage (event #3 in Fig. 7) around $t = 2\,900\ s$. The peak at $t = 3\,100\ s$ is actually caused by the brightening of the raceway after deactivation of the PCI lance and cannot be considered as a detection of the blockage.

5.1.3. Algorithm B3: Standard Deviation of Pixel Values on ROIs

It turns out that while the mean grey level is not able to cover type 2 and 3 blockages correctly, the standard deviation of the pixel grey levels contains a more precise information. In general, a normal raceway situation will have a high standard deviation due to the fact that coke particles (low grey levels) and the background (high grey levels) are both present and represent a highly alternating grey level signal. In contrast, a blockage of type 1 represents only small fluctuations in the lower range of the grey level histogram. This results in a smaller standard deviation of the grey levels. The same holds true for blockages of type 2 - a partially blocked and dark raceway will also significantly reduce the standard deviation of the grey levels for the entire image. Essentially, the information each pixel carries is a

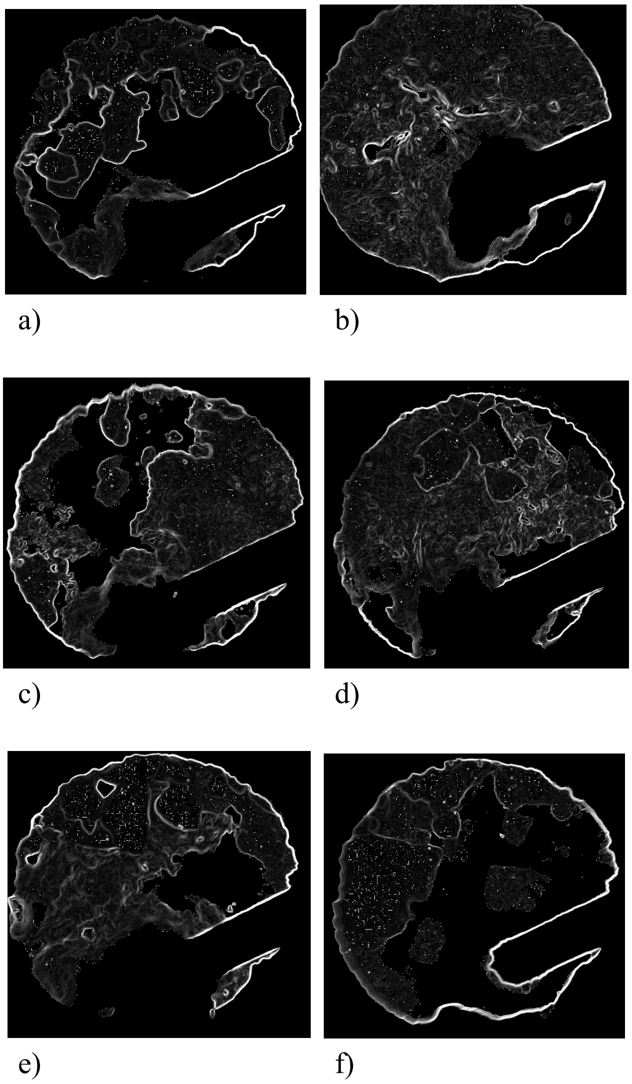


Fig. 11. Gradient maps of the example images in Figs. 1 to 6 after processing with algorithm B2.

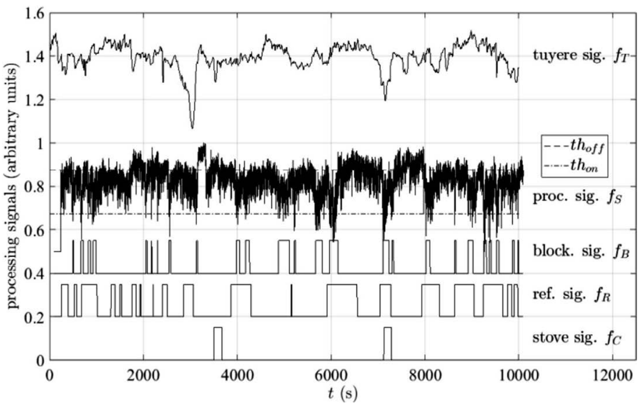


Fig. 12. Results of algorithm B2 (pixel gradients). f_T is the blast signal of the tuyere, and f_S is the result signal for the relative number of remaining pixels. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B. Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

fluctuating grey level with a certain mean value and standard deviation. So, a moving coke particle basically carries this information (the grey level texture) along its trajectory.

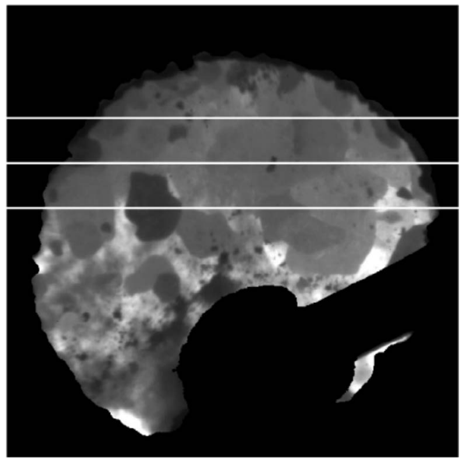


Fig. 13. Example image with the line positions marked that are used for processing in algorithms B3 and B4.

Considering the fast and highly turbulent (chaotic) motion of the coke particles in the raceway, we can assume that, from a statistical point of view, this information will be transported to any place in the raceway. Hence, a reduced region of interest (ROI) in the image will contain the same information as the entire image as a fast-moving coke particle will sooner or later pass a certain selected ROI position in the raceway. We shall also recall here that we only take snapshots of the raceway once a second (with sufficiently low exposure times to avoid motion blur), but that the actual movement of the coke particles is roughly three orders of magnitude faster. This leads to the conclusion that there is no need to process the entire image, and by processing only a small portion of the image we can significantly reduce the calculation times. For the present algorithm we put that principle to the extreme and defined only three horizontal lines for processing as is illustrated in **Fig. 13**. Instead of selecting two-dimensional ROIs (*e.g.* squares or circles), the line-based processing seems more appropriate for the task of blockage detection. As the lower part of the visible raceway is dominated by the PCI lance and the coal plume, the three lines are located in the upper half of the image. For each line the standard deviation S_i of the grey levels along this line is calculated. To combine the results from the three line locations, they are multiplied to obtain the result signal $f_S(k) = \Pi_{i=1}^{3} S_i(k)$ (**Fig. 14**). The quality of the results is much better than *e.g.* for algorithm B2 while the processing times of B3 are more than two orders of magnitude lower. Thus, processing of very small portions of the images can provide equal or even better results than processing the complete images.

5.2. Processing Based on Correlating Information of Consecutive Frames

Using the information of consecutive images is a more powerful approach as also the aspects of temporal changes are considered. There is plenty of literature from the field of motion detection which can be used as a starting point, but however, most motion detection algorithms are optimized for the goal of identifying moving objects in front of a more or less static background.[10–15] As the tuyere images do not contain something which could be considered as a static background these methods will not meet the requirements

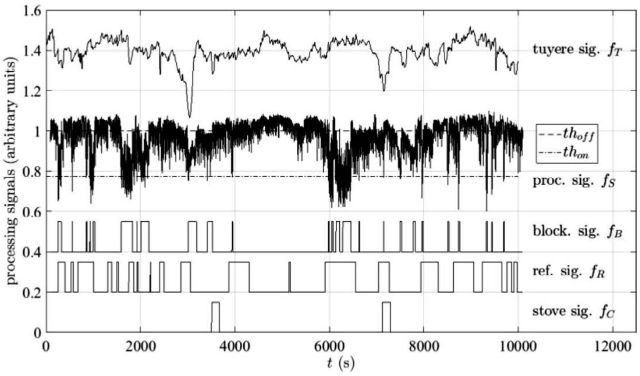


Fig. 14. Results of algorithm B3 (pixel standard deviation on three lines). f_T is the blast signal of the tuyere, and f_S is the product of the grey level standard deviations on the three selected lines. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B. Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

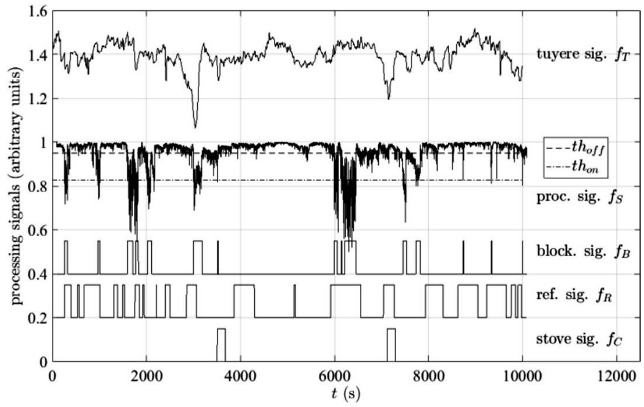
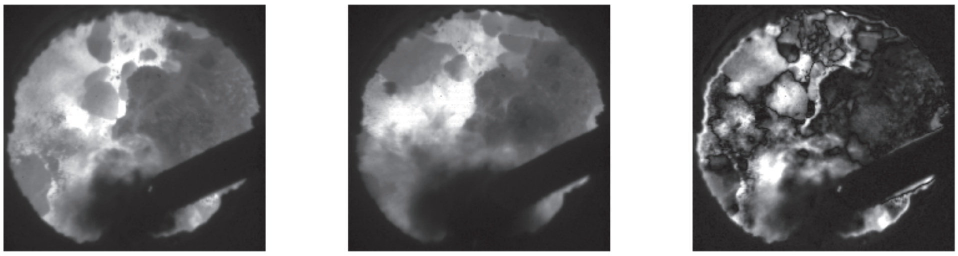


Fig. 15. Results of algorithm B4 (line based pixel correlation). f_T is the blast signal of the tuyere, and f_S is the product of the line correlations. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B. Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.



a) b) c)

Fig. 16. Example for image subtraction. c) shows the absolute difference of images a) and b). The left half of c) shows higher values due to the differing presence of coke particles in a) and b). The right side of the result in c) shows values close to 0 as the blocking structure does not move very much between images a) and b).

for raceway blockage detection.

Another aspect which has to be noted here is that for normal raceway operation there is no correlation between the coke particles visible in two consecutive frames due to the fast motion of the particles inside the raceway. Tests with a high-speed camera have shown that frame rates of approximately 2 000 Hz provide a sufficient temporal resolution of the coke particle motion. Thus, tuyere images recorded at 1 Hz do not contain correlated information unless a blocking structure appears that has a minimum lifetime of a few seconds. These slowly moving structures are therefore the only related information in consecutive images.

5.2.1. Algorithm B4: Line Based Pixel Correlation

Calculating the full two-dimensional correlation matrix of two images is rather slow (processing of the example images was in the range of 15 s per image pair on our test hardware) and hence not useful for an online implementation. However, for correlating information in consecutive frames the same statistical argumentation can be applied as discussed for algorithm B3, and processing of the full images is actually not necessary. We use the same three lines L_1, L_2 and L_3 at different y positions from Fig. 13 and calculate the correlation signal for each of these lines according to

$$R_{L,i}(l_x,k)=\sum_{x=1}^{n_x} I(x,L_i,k)I(x-l_x,L_i,k-1) \quad \text{....... (1)}$$
$$k=2..N,\ i\in\{1,2,3\}$$

where k is the index of the images, i is the line index and l_x is the horizontal coordinate in the images. To squeeze the information of the resulting signals $R_{L,i}(l_x,k)$ into one time depended signal, the correlation signals are averaged and the product of the three averages is calculated (Eq. (2)). This emphasizes correlating luminosity distributions appearing on more than one line and prevents false positive signals caused by local effects.

$$f_S(k)=\Pi_{i=1}^{3}\overline{R_{L,i}(l_x,k)} \quad \text{........................ (2)}$$

The resulting signal (**Fig. 15**) has a lower noise content than the results of B3 and a good separation of detected blockages from the base level. B4 shows the least number of false positive signals but on the other hand also detects a lower number of blockages and also misses the major event #6. It is interesting to note that the line-based algorithms B3 and B4 detect the suspicious period between t = 5 900 s and 6 600 s quite accurately. In the camera images from this period the raceway appears darker than usual and we can assume that the boundary conditions for coal combustion are not ideal. There are no major blocking structures visible though, and also the hot blast signal does not show any reduction of the flow rate (c.f. Fig. 7). This is one of the few examples where signal processing of the flow rate data does not deliver any result and image processing is the only possible way to detect this raceway condition. However, it is not yet clear how the raceway behaviour during this period correlates with other BF operating data and if it is necessary

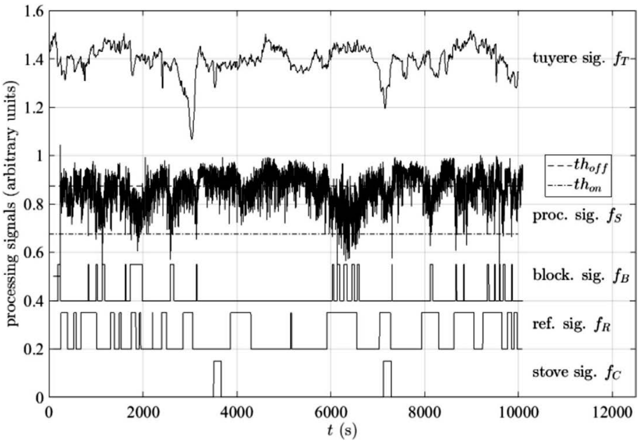


Fig. 17. Results of algorithm B5 (sum of absolute differences). f_T is the blast signal of the tuyere, and f_S is the resulting signal for the sum of absolute differences. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B. Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

to detect such phases.

5.2.2. Algorithm B5: Sum of Absolute Differences

Subtracting images emphasizes areas which differ significantly between two frames. An overall value for each pair of images in a sequence can be calculated by summing up the absolute values of all pixel differences (Eq. (3)).

$$D(k) = \sum_{y=1}^{n_y} \sum_{x=1}^{n_x} | I(x, y, k) - I(x, y, k-1) | \quad (3)$$

In image processing literature this approach is usually called 'sum of absolute differences'. **Figure 16** shows an illustrative example based on Fig. 3 and its immediate following image. The resulting signal in **Fig. 17**, though, cannot not provide an optimal blockage detection. Both major blockages (#3 and #6) in the dataset are missed and the signal to noise ratio of the result is not as good as for other algorithms.

6. Summary & Outlook

In this third part of the study on raceway blockage detection we presented various approaches for blockage detection based on the visual information of tuyere cameras. The results demonstrate that blockage detection via image processing is not trivial as the optical appearance of the raceway can change significantly over time and even for a human operator it needs some experience to distinguish between normal raceway behaviour and various types of blockages or periods with reduced permeability of the raceway. The very different nature of the highly dynamic flow situation inside the raceway makes it difficult to find proper strategies for automated analysis based on image processing. As an additional constraint the computational effort must remain feasible for an online implementation in the process control system. However, from a statistical point of view it is sufficient to process only reduced ROIs of the images and the line-based algorithms B3 and B4 actually perform better than other algorithms processing the entire images.

In direct comparison with the results of the signal processing algorithms from part 2 it is evident, that the image processing algorithms produce signals with a worse signal to noise ratio. To discuss the pros and cons of image processing the presented results are all unfiltered signals, though. Thus, applying a moving average filter with a window size of a few seconds will improve the results.

Finally, the analog result signals f_S need to be converted to a digital yes/no decision if a blockage is present or not. So, the ideal result for this final thresholding step shows significant peaks off a more or less constant base level. The only algorithms which fulfil this requirement are the adaptive thresholding method (B1) and the line-based correlation (B4). All other algorithms show higher noise levels and drifting mean values which makes thresholding more difficult.

In the end we can state that for the task of blockage detection signal processing of hot blast data delivers the best overall performance. So, even if a complete tuyere camera installation is available it does not necessarily mean that running an image-based blockage detection makes sense (in addition to other image-based monitoring tasks). A feasible strategy for the process control system could be to run a blast signal-based blockage detection and activate additional visual information from the tuyere cameras for the operators as soon as a blockage is detected from hot blast data. An optimal solution would therefore use a combination of both methods to obtain a maximum benefit from the available plant data. Such a combined method will also provide an ideal basis for further analysis of the reasons and frequency of occurrence of raceway blockages and their correlations with other blast furnace operating conditions.

Acknowledgements

The authors gratefully acknowledge the funding support of K1-MET GmbH, metallurgical competence center. The research program of the competence center K1-MET is supported by COMET, the Austrian program for competence centers.

REFERENCES

1) S. Puttinger and H. Stocker: *ISIJ Int.*, **59** (2019), 466.
2) S. Puttinger and H. Stocker: *ISIJ Int.*, **59** (2019), 474.
3) P. R. Austin, S. J. Chew, D. Maldonado, J. G. Mathieson, D. J. Pinson, H. Rogers, M. Walsh, B. D. Wright and P. Zulli: Proc. 6th European Coke and Ironmaking Cong., Stahl-Institut VDEh, Düsseldorf, (2011), FN1.
4) J. Daoud and I. Nipl: Master thesis, Lulea University of Technology, (2000), http://ltu.diva-portal.org/smash/record.jsf?pid=diva2%3A1029982&dswid=-7053, (accessed 2017-05-13).
5) W. Birk, O. Marklund and A. Medvedev: *IEEE Trans. Ind. Appl.*, **38** (2002), 571.
6) Y. Wang, Y. Hu, G. Yang, T. Qin and W. Yuan: *Procedia Eng.*, **102** (2015), 265.
7) R. Zhang, S. Cheng and C. Guo: *ISIJ Int.*, **58** (2018), 244.
8) N. Otsu: *IEEE Trans. Syst. Man Cybern.*, **9** (1979), 62.
9) R. C. Gonzales and R. E. Woods: Digital Image Processing, 3rd Ed., Pearson Education, London, (2010), 189.
10) J. B. Kim and H. J. Kim: *Pattern Recognit. Lett.*, **24** (2003), 113.
11) Q. Z. Wu, H. Y. Cheng and B. S. Jeng: *Pattern Recognit. Lett.*, **26** (2005), 555.
12) A. Manzanera and J. C. Richefeu: *Pattern Recognit. Lett.*, **28** (2007), 320.
13) Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent and C. Rosenberger: *J. Electron. Imaging*, **19** (2010), 033003.
14) M. A. Nasr, M. F. Al and A. S. Tolba: *J. King Saud Univ. - Comput. Inf. Sci.*, **29** (2017), 399.
15) C. Chen, S. Li, H. Qin and A. Hao: *Pattern Recognit.*, **52** (2015), 410.