Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 50-2 (2017) 13–18

Performance of Convolutional Neural Networks for Feature Extraction in Froth Flotation Sensing

Z.C. Horn * **L. Auret** * **J.T. McCoy** * **C. Aldrich** *,** **B.M. Herbst** ***

* *Department of Process Engineering, University of Stellenbosch, Stellenbosch, South Africa (e-mail: lauret@sun.ac.za).*
** *Department of Mining Engineering and Metallurgical Engineering, Western Australian School of Mines, Curtin University, Perth, Australia (e-mail: chris.aldrich@curtin.edu.au)*
*** *Department of Applied Mathematics, Stellenbosch, South Africa (e-mail: herbst@sun.ac.za)*

Abstract: Image-based soft sensors are of interest in process industries due to their cost-effective and non-intrusive properties. Unlike most multivariate inputs, images are highly dimensional, requiring the use of feature extractors to produce lower dimension representations. These extractors have a large impact on final sensor performance. Traditional texture feature extraction methods consider limited feature types, requiring expert knowledge to select and may be sensitive to changing imaging conditions. Deep learning methods are an alternative which does not suffer these drawbacks. A specific deep learning method, Convolutional Neural Networks (CNNs), mitigates the curse of dimensionality inherent in fully connected networks but must be trained, unlike other feature extractors. This allows both textural and spectral features to be discovered and utilised. A case study consisting of platinum flotation froth images at four distinct platinum-grades was used. Extracted feature sets were used to train linear and non-linear soft sensor models. The quality of CNN features was compared to those from traditional texture feature extraction methods. Performance of CNNs as feature extractors was found to be competitive, showing similar performance to the other texture feature extractors. However, the dataset also exhibits strong spectral features, complicating comparison between texture feature extractors. The results gathered do not provide sufficient information to distinguish between the types of features detected by the CNN and further investigation is required.

Keywords: machine learning, soft sensing, computer vision, neural networks, data reduction

1. INTRODUCTION

Use of shallow neural networks in process engineering has proven promising. However, applications are generally limited to using neural networks as non-linear modelling tools for existing numeric process data. In the case of image processing, other machine learning methods are applied first as feature extractors due to the curse of dimensionality inherent in fully-connected networks as shown in Ali et al. (2015).

The emergence of deep learning has allowed the use of neural networks in applications which have previously been considered unsuitable. Specific hierarchies such Convolutional Neural Networks (CNNs) have been especially successful in image processing tasks and are of interest for extracting features from images and creating reduced representations as shown in Arel et al. (2010) and Ciresan et al. (2011). Such a feature extractor can be used in inferential sensing applications in the mineral processing industry.

In platinum extraction from mineral ores, froth flotation is a key process step in which froth appearance has been shown to correlate strongly with process conditions in Moolman et al. (1995). Amongst these conditions is platinum grade (fraction of total mass in a process stream or unit which is platinum). Platinum grade is difficult to infer from generally measured process conditions and cannot be measured directly in near real-time. This makes flotation froths an ideal case study for soft sensor design, allowing investigation of deep learning methods within soft sensors. Existing froth soft sensors are used primarily for measuring and controlling flotation cell conditions such as froth level, airflow, and mass-pull.

This article has three primary sections: Background, Methodology, and Results. In Background, relevant technical information of platinum processing, neural networks, and image-based soft sensors will be discussed. The Methodology section details parameters and methods used for feature extraction and the soft sensor; the outcomes of which will be discussed in the Results.

2405-8963
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2017.12.003

2. BACKGROUND

Vision systems for flotation froths have been investigated for extracting bubble criteria such as size, shape, and velocity. Sadr-Kazemi and Cilliers (1997) show that these factors may be correlated to process conditions but are more commonly used to determine operating parameters for the froth flotation equipment rather than overall process performance.

Research in Moolman et al. (1995) has shown that the appearance of flotation froths is indicative of process conditions. Additionally, experienced plant operators have been able to determine process conditions based on visual examination of froths and past experience.

In process engineering, feature extraction methods used for vision systems include:

(1) GLCM - Grey-Level Co-occurrence Matrices. See Van Deventer et al. (1997)
(2) LBP - Linear Binary Patterns. See Steger et al. (2008)
(3) Wavelet transforms. See Peng and Chu (2004)

This work explores the usage of deep learning systems as feature extractors in comparison to these methods.

2.1 Platinum Flotation Process

Refining of platinum from ore is a complex multi-step process with a large number of interactions. Cramer (2001) indicates this is especially true for the flotation section; multiple flotation tanks cascade into each other with conservatively sized recycle loops and reagent dosages to ensure maximum recovery of platinum in the process.

Processed, finely ground platinum is suspended in water, forming a mineral rich slurry. This is fed into the froth flotation cells in which chemical conditions induce separation of platinum from waste material. The inefficiency of separation in a single cell is what requires multiple froth flotation cells to be used in the process.

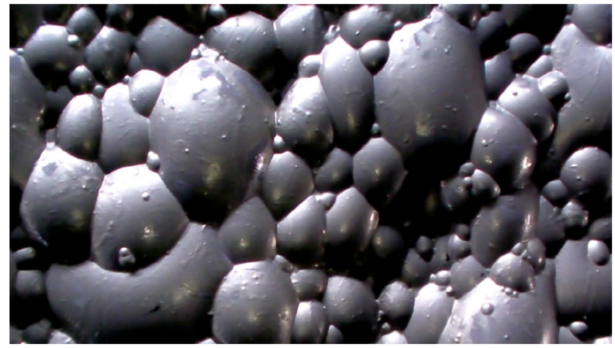
Plant operators aim to minimise inventory of platinum in concentrators (which is capital intensive on operations) while also maximising platinum grade in the final concentrate before smelting (lower final concentration results in more energy intensive smelting). Flotation cells are the last solid waste reduction stage and thus their performance is key to achieving these goals.

In Figure 1, example images of platinum froths can be observed. Both images are of the same flotation cell taken at different times, and thus under different process conditions. Some changes are apparent between the images, however, the meaning of these changes, and additional details not apparent, can only be determined by expert knowledge or through machine learnt correlations.

In Kistner (2013) it is shown that a suitably developed machine learning algorithm not only detects features apparent to human vision, but also hidden properties.

2.2 Inferential Soft Sensing

Kadlec et al. (2009) state that online and non-intrusive measurement are the prime advantages of inferential soft sensors. They are generally implemented to exploit standard process measurements such as temperature, pressure, pH, and more, to estimate process conditions that cannot be measured directly.



(a) Froth I

(b) Froth II

Fig. 1. Examples of flotation froth images exhibiting differing structures in the same process at different times and conditions

In Figure 2 the structure of a vision-based soft sensor is summarised. Images from the process are captured such as the flotation froth as shown in Figure 1. These are pre-processed to ensure uniform resolution and feature extraction is performed to reduce dimensionality. A model is generated to produce process information, in this work grade, from the extracted feature sets. This model is trained via machine learning using images correlated with previously collected and offline measured grade data.

Data-driven sensors (as is the case in image-based sensors) utilise empirical correlations between feature sets and process data. Conversely, fundamental sensors attempt to establish measured and inferred variable correlation through underlying physical and chemical relationships. This is not always possible for complex systems (such as image-based sensors) or where no fundamental relationships have been developed. Fortuna et al. (2007) indicate data-driven sensors are preferred in industry due to the difficulty in accurately modelling many chemical processes.

Feature Extraction Several feature extraction methods are compared in this work; in Figure 2, the feature extraction step is highlighted in green. Unlike other extractors tested, CNNs must be trained to generate a feature extractor before they can produce features. This training will occur separately from the soft sensor and is not represented in Figure 2. The feature extractors considered

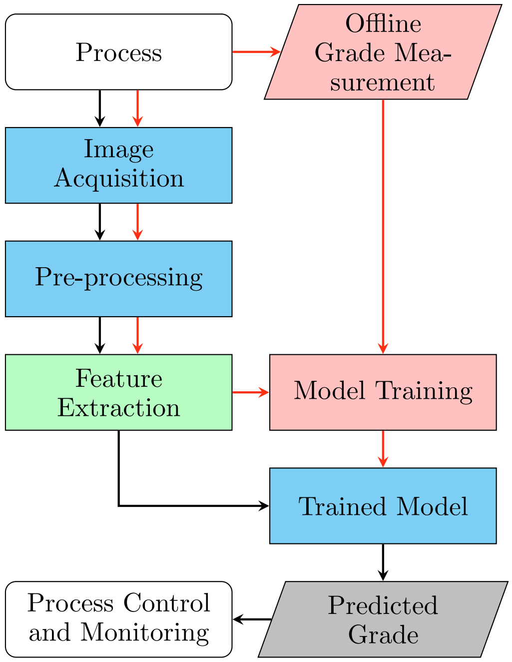


Fig. 2. Vision-based Inferential Sensor Framework. Grade is inferred from process images which are collected in real time to be used in process control and monitoring. The red portions of the framework are utilised during training of the sensor and flow of information during training is shown by red arrows. The normal flow of information during operation is represented by black arrows.

in this work use textural features which utilise spatial relationships between pixels. A spectral feature method (which ignores spatial relationships between pixels) is used as a control method. The goal of the feature extraction module is to produce robust feature sets.

Soft Sensor The soft sensor model (as represented by the blue "Trained Model" block in Figure 2) is trained to correlate feature sets with the requisite inferred variables (offline grade). This is performed using supervised learning methods. If the feature extractor requires training as with CNNs, it is possible to do so simultaneously with training of the soft sensor model. However, in order to provide fair comparison between various feature extractors, the soft sensor will only be trained on feature sets without optimising the extractors simultaneously.

2.3 Deep Learning

Deep neural networks, unlike shallow neural networks, allow specific network hierarchies to be developed for the task at hand. These hierarchies sacrifice generality of fully connected networks for advantages in specific applications. In the case of image processing, Convolutional Neural Networks (CNNs) have been shown as effective in He et al. (2015) in classifying large resolution images.

CNNs consist of multiple locally connected layers, the most important being convolutional layers. Originally developed in Fukushima (1980), convolutional layers convolve kernels with a small span over the entire area of the input image. This mimics biological vision research in Hubel and Wiesel (1968), which showed that locally interesting features are combined across the entire image field to build up complex patterns and images.

Following each convolution, max-pooling is performed where the strongest response in a pixel neighbourhood is taken to reduce dimensionality of the data. Boureau et al. (2010) shows this step has a large impact on the effectiveness of CNNs and the omission of pooling layers has a negative impact on performance. The combined convolution-pooling layers can be repeated as necessary to achieve a suitable final convolutional feature set with low dimensionality.

Finally, fully connected layers perform final processing to correlate the reduced feature sets with the classifier/regressor outputs.

Most applications of deep networks, including CNNs, focus on supervised learning. Regularisation methods (summarised in Goodfellow et al. (2016)) such as drop-out, early stopping, batch normalisation, and transfer learning from generative models, all contribute to the effectiveness of supervised learning from large to limited labelled datasets. For very limited labelled datasets other solutions must be found or network performance will suffer from overfitting as shown in LeCun et al. (2015).

3. METHODOLOGY

The dataset used to train the soft sensor and CNN feature extractor consists of 544 images with four classes, subsampled from a previous dataset to enhance variability in the data. Each class represents a distinct platinum grade, being highest in the first class, and lowest in the last. 55 images were set aside as a final testing set and the remaining 489 images were supplemented with 13 variants generated from each base image. Variants are not generated for the testing dataset, as they do not provide any benefit for performance measurement. Variants are images created from the base image by scaling, rotating, adding noise, and colour space transformation, in Dosovitskiy et al. (2014) these improve generality in the neural network and help to limit overfitting during training.

The final training set consists of 6846 images, stemming from the original 489 images and their variants. All images consist of three colour channels (red, green, and blue) which were cropped to a final size of 64x64 pixels each and stored with their platinum grade classification giving 12288 input values from each image. 64x64 pixel size was chosen as a balance between feature clarity and training time with available computer resources.

3.1 Feature Extraction and Training

Three textural feature extraction methods were applied to the testing and training datasets using parameters obtained from prior work on the dataset done by Kistner (2013). Standard implementations of each traditional method were used.

CNN The CNN feature extractor (developed in TensorFlow from Abadi et al. (2015)) was trained with the training image set, of which 10% was randomly selected as a validation set during training. The front-end of the neural network consisted of three sequential convolutional layers with 32 filters in each layer. The convolutional filter size for each layer was 11 pixels and 2x2 max-pooling was applied after each layer. Filter size in convolutional networks cannot easily be correlated with expected feature size in input images, rather, it is best determined through full factorial experimental analysis. Rectified linear units were used as the activation function for the convolutional neurons. The output of each convolutional layer was batch normalised to prevent neuron saturation in later layers.

The latter part of the network consisted of two fully connected layers with 512 and 16 neurons each with tanh activation functions. The final output layer consisted of four neurons (one for each class) with softmax activations to assign a prediction probability to each class. The network was trained using stochastic gradient descent to minimise the cross-entropy loss of classification, 64 images were used per training batch to increase training speed and avoid local minima. 10 training epochs were performed, images must be trained on multiple times to ensure that the network converges fully, especially when training data is limited.

The hyperparameters described in this section are those which gave the highest validation accuracy in an initial test batch of network parameters. While they are likely not the best values possible, they should be sufficient for proving the concept of CNNs as feature extractors.

The trained network was used to output intermediate layer values, giving a 16 variable feature set for each input image. Feature sets were generated for both the testing and training datasets to use in soft sensor training. The time taken to extract features for the training dataset was recorded and used to compare extraction speeds of the various methods.

GLCM The frequency at which certain pixel intensities occur within a specified neighbourhood are summarised and given. This provides information on local spatial relationships within an image. The key parameters are summarised in Table 1. They are somewhat rotationally invariant.

Table 1. GLCM Parameters

Parameter	Value
Levels	128
Distance	4 pixels
Type	Symmetric
Directions	0, 45, 90, and 135 degrees
Features	Contrast, Correlation, Energy, and Homogeneity
Output	8 features

LBP Each pixel intensity is compared to its local pixel neighbourhood giving a histogram of local binary patterns in the image. Specific mapping types are required to produce rotationally invariant features. Features were extracted with the parameters listed in Table 2.

Table 2. LBP Parameters

Parameter	Value
Radius	4 pixels
Samples in Radius	16
Map Type	uniform
Output	243 features

Wavelets Similar to Fourier transforms, wavelets perform frequency analysis but with an arbitrary range of prototype functions. This allows a prototype function to be selected specifically for the application under consideration. The parameters in Table 3 were used to extract features. The features produced are not rotationally or scaling invariant.

Table 3. Wavelet Parameters

Parameter	Value
Type	db4
Decomposition Level	6
Output	18 features

Spectral The control method; spectral features are overall image statistics which do not use textural information. For this work, the average value of all pixels in each colour channel (red, green, and blue) were taken to form the feature set. This control method allows for the efficacy of textural methods to be validated.

3.2 Soft Sensor Training

In the Model Training stage of Figure 2 the soft sensor was trained using k-Nearest Neighbour and Linear Discriminant Analysis methods. The standard classifier fitting methods of MATLAB R2016b were used to train both classifiers with the training data. Loss, the weighted misclassification rate, of each model and feature extractor pair was calculated as a key performance indicator alongside the confusion matrix for the testing data.

Linear Discriminant Analysis This linear classifier attempts to separate multiple classes with linear combinations of features. Using the standard implementation in MATLAB no hyperparameters were changed. The linear coefficient value was the default of 0 and no regularisation was specified. It should be noted that a minimal amount of regularisation may automatically be applied to the covariance matrix in order to invert it, though this will have an insignificant effect on classification performance.

k-Nearest Neighbours The method attempts to predict the class of a feature set by comparing to a number of nearest known sets in the feature space. Key hyperparameters for this method are the number of neighbours to compare with for classification and the weight assigned to each neighbour in the classification calculation. In the default MATLAB implementation (used in this work) all weights are equal and only the closest neighbour is used for classification.

Training and Prediction Once features have been stored for each of the selected methods, the soft sensor models were trained. Training was performed using features generated from the training dataset and models created. The resultant models were used to produce predictions using the originally separated testing dataset.

3.3 Soft Sensor Testing

The results were analysed by calculating the loss for each model. Loss is similar to average misclassification rate, but each misclassification is weighted according to the prevalence of each class in the testing dataset. This compensates for an uneven distribution of classes in the original dataset and provides a more representative measurement of classification performance for all classes.

Confusion matrices were produced for each model to allow further analysis of classification performance.

4. RESULTS

In Table 4 the loss for each method is summarised. k-NN classification gives the best result for each feature extractor, suggesting that the classification space is complex and classes are not easily separable with linear methods. As such LDA results will be disregarded in further discussion.

Table 4. Model Loss

Model	Classifier	
	LDA	k-NN
CNN	0.340	0.035
LBP	0.236	0.016
GLCM	0.411	0.263
Wavelet	0.440	0.017
Spectral	0.304	0.016

The textural methods return similar classification results, with the exception of GLCM features which perform poorly in comparison to other methods.

Table 5. LBP with k-NN Confusion Matrix

		Predicted Class			
		1	2	3	4
Actual Class	1	17	0	0	0
	2	0	8	0	0
	3	1	0	15	0
	4	0	0	0	14

However, spectral features provide better performance than textural features, indicating spectral information is strongly linked to grade class. They have a slight edge over LBPs when examining the confusion matrices in Tables 5 and 6. Both misclassify an image in class 3, but the spectral misclassification is closer to the true class due to grade increasing with class number.

Table 6. Spectral with k-NN Confusion Matrix

		Predicted Class			
		1	2	3	4
Actual Class	1	17	0	0	0
	2	0	8	0	0
	3	0	0	15	1
	4	0	0	0	14

In Table 7 only one image is misclassified for wavelet features. The marginally higher loss indicated in Table 4 is due to the misclassification occurring in a more common class. As previously stated, the loss function will assign weights to different classes based on their relative distributions within the datasets. More common classes will have higher weights as they should be easier for the model to predict.

Table 7. Wavelets with k-NN Confusion Matrix

		Predicted Class			
		1	2	3	4
Actual Class	1	16	1	0	0
	2	0	8	0	0
	3	0	0	16	0
	4	0	0	0	14

The CNN feature model only misclassified two images as shown in Table 8. This is sufficient as a proof of concept, with limited training data CNN features appear to be useful for classification of complex process images such as flotation froths. However, it is difficult to determine whether this classification accuracy is due to textural or spectral features.

CNN (and other deep neural network) architecture allows for spectral features to be extracted if they are found to be significant during training. It is possible that the performance achieved by CNNs are due to either spectral, textural, or a combination of both feature types being used for classification. While this hinders comparison between textural methods, this property may return more accurate results than purely textural or spectral methods under actual industrial conditions.

Table 8. CNN with k-NN Confusion Matrix

		Predicted Class			
		1	2	3	4
Actual Class	1	17	0	0	0
	2	0	8	0	0
	3	0	0	15	1
	4	0	1	0	13

The confusion matrices and loss results provide insufficient information to determine what features are being extracted by the CNN. Visualisation of the convolutional filters and using “deepdream” algorithms demonstrated in Mordvintsev et al. (2015) is required to determine the types of features detected by the network. It is also recommended that correlation studies are performed within feature sets, to determine if each feature in a set contributes uniquely to final classification.

The strong spectral features in the dataset complicate comparisons between the textural methods. While providing good performance in this specific instance, spectral features do not generalise well across differing process conditions and require controlled external lighting as shown in Vathavooran et al. (2006).

CNN training consumes some time, however, as can be seen in Table 9, features can be extracted rapidly from the trained convolutional network compared to other complex textural methods. The rate of feature set extraction is equivalent to the number of images processed in the same time.

Table 9. Feature Extraction Speed

Model	Feature Sets per Second
CNN	1881
LBP	8
GLCM	713
Wavelet	848
Spectral	8569

5. FUTURE WORK

Of immediate concern is further evaluation of features being extracted by CNNs in the dataset used in this work. While CNNs as texture feature extractors appear to be promising, conclusive results require further investigation into what features the neural network is detecting.

5.1 Process Data and Images

An ongoing problem in evaluating image-based soft sensors is lack of known process conditions which can be correlated with captured images. This is especially apparent with deep learning methods and their higher risk of overfitting. The prohibitive costs of gathering the associated process data for deep learning methods limits the size of available datasets. Special care must be taken in ongoing work to ensure validity of results with smaller datasets in deep learning.

5.2 Performance of Feature Extraction Methods

The application of deep learning hierarchies other than CNNs must be evaluated in future as feature extractors. Additionally, while CNNs have proven to be effective in classification tasks for this case study, their long-term performance under industrial conditions remains untested.

5.3 Deep Learning as a Soft Sensor

This work is more concerned with determining the relative performance of CNNs as feature extractors rather than soft sensor performance as a whole. With the modelling capabilities of fully connected layers in a CNN it is possible to implement the entire soft sensor system within the neural network.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. URL `http://tensorflow.org/`. Software available from tensorflow.org.

Ali, J.M., Hussain, M., Tade, M.O., and Zhang, J. (2015). Artificial intelligence techniques applied as estimator in chemical process systems a literature survey. *Expert Systems with Applications*, 42(14), 5915 – 5931.

Arel, I., Rose, D.C., and Karnowski, T.P. (2010). Deep machine learning - a new frontier in artificial intelligence research [research frontier]. *IEEE Computational Intelligence Magazine*, 5(4), 13–18.

Boureau, Y.L., Ponce, J., and LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, 111–118.

Ciresan, D.C., Meier, U., Masci, J., Maria Gambardella, L., and Schmidhuber, J. (2011). Flexible, high performance convolutional neural networks for image classification. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, 1237.

Cramer, L.A. (2001). The extractive metallurgy of south africas platinum ores. *Jom*, 53(10), 14–18.

Dosovitskiy, A., Springenberg, J.T., Riedmiller, M.A., and Brox, T. (2014). Discriminative unsupervised feature learning with convolutional neural networks. *CoRR*, abs/1406.6909.

Fortuna, L., Graziani, S., Rizzo, A., and Xibilia, M.G. (2007). *Soft sensors for monitoring and control of industrial processes*. Springer Science & Business Media.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4), 193–202.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, 1026–1034.

Hubel, D.H. and Wiesel, T.N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195(1), 215–243.

Kadlec, P., Gabrys, B., and Strandt, S. (2009). Data-driven soft sensors in the process industry. *Computers & Chemical Engineering*, 33(4), 795–814.

Kistner, M. (2013). *Image Texture Analysis for Inferential Sensing in the Process Industries*. Master's thesis, University of Stellenbosch.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.

Moolman, D., Aldrich, C., Deventer, J.V., and Bradshaw, D. (1995). The interpretation of flotation froth surfaces by using digital image analysis and neural networks. *Chemical Engineering Science*, 50(22), 3501 – 3513.

Mordvintsev, A., Olah, C., and Tyka, M. (2015). Deepdream-a code example for visualizing neural networks. *Google Res.*

Peng, Z. and Chu, F. (2004). Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography. *Mechanical systems and signal processing*, 18(2), 199–221.

Sadr-Kazemi, N. and Cilliers, J. (1997). An image processing algorithm for measurement of flotation froth bubble size and shape distributions. *Minerals Engineering*, 10(10), 1075 – 1083.

Steger, C., Ulrich, M., and Wiedemann, C. (2008). Machine vision algorithms and applications. *Illumination*, 5(2.1), 1.

Van Deventer, J., Bezuidenhout, M., and Moolman, D. (1997). On-line visualisation of flotation performance using neural computer vision of the froth texture. In *Proceedings of the XX International mineral processing congress*, volume 1, 315–326.

Vathavooran, A., Batchelor, A., Miles, N.J., and Kingman, S.W. (2006). Applying froth imaging techniques to assess fine coal dewatering behavior. *Coal Preparation*, 26(2), 103–121.