NDT and E International 107 (2019) 102139

Contents lists available at ScienceDirect

NDT and E International

journal homepage: www.elsevier.com/locate/ndteint

Automated defect classification of SS304 TIG welding process using visible spectrum camera and machine learning

Daniel Bacioiu[a,b,*], Geoff Melton[b], Mayorkinos Papaelias[a], Rob Shaw[b]

[a] *Department of Metallurgy and Materials, University of Birmingham, UK*
[b] *TWI Ltd, Abington, Cambridge, UK*

ARTICLE INFO

Keywords:
Weld monitoring
High dynamic range camera
Machine learning
Vision
Automation

ABSTRACT

Tungsten Inert Gas welding is dependent on human supervision, it has an emphasis on visual assessment, and it is performed in a controlled environment, making it suitable for automation. This study designs a system for assessing the tungsten inert gas welding quality with the potential of application in real-time. The system uses images in the visible spectrum paired with the state-of-the-art approach for image classification. The welding images represent the weld pool in visible spectra balanced using high dynamic range technology to offset the powerful arc light. The study trains models on a new tungsten inert gas welding dataset, leveraging the state-of-the-art machine learning research, establishing a correlation between the aspect of the weld pool and surrounding area and the weld quality, similar to an operator's assessment.

1. Introduction

Tungsten Inert Gas (TIG) welding [1,2] was invented in the late 1930s for welding magnesium and aluminium, while in 1940 USA started the trials with inert gas for shielding [3]. Later, the process expanded to joining copper and steel alloys. Nowadays, the industry employs TIG welding for joining high-value components in nuclear, automotive and aerospace applications. The process decouples heat input from the deposition rate, offering a higher degree of control over the welding parameters, resulting in high-quality weldments. Although having higher flexibility, TIG remains predominantly a manual process dependent on a highly-skilled workforce mostly. TIG automation represents a clear focus for the industry considering the decrease in the number of human workforce. The very same process flexibility gives rise to complex weld pool aspects, which are difficult to assess and control.

Several authors tackled the problem of welding processes quality assessment [4–8]. Physically, the welding arc gives rise to a harsh environment and the welding generates a vast amount of information. The issues arising during process monitoring are complex since several aspects of metallurgy, image acquisition, image processing and software development need consideration.

Initial attempts to address this topic focused on the combination of laser illumination for offsetting the powerful arc light and video recording, by Lucas et al. [9]. The laser would shine light onto the weld pool in order to bridge the luminosity gap between the very bright arc and very dark surroundings. An additional filter eliminates ultraviolet (UV) radiation and most of the visible light spectrum that the arc predominantly emits, allowing only near-infrared (NIR) light to pass. The authors then devised a system for real-time measurement and control of the upper surface of the weld pool. The aim was to correlate the weld pool's geometric features while in the molten phase with weld depth penetration in order to achieve weld uniformity and repeatability.

Other techniques of assessment were employed using the spectral analysis to examine the optical emission of thermal plasma as in the case of Mirapeix et al. [10]. The study considered the temperature estimation and correlation of captured spectra with the occurrence of defects during the manufacturing of a large steam generator for nuclear power plants. The authors tried to identify defects through spectroscopic analysis of the background radiation, root mean square (RMS) spectral intensity and spectral information processing with principal component analysis (PCA) and Artificial Neural Networks (ANNs).

Song et al. [11] calculated the three-dimensional geometry of weld pool surfaces through structured light projection onto a weld pool and capturing the reflected light trajectory. In their study, two algorithms were used to test the viability of the system, including the edge-point algorithm (EPA) and one point algorithm (OPA), both assuming a convex shape of the weld pool.

[*] Corresponding author. Department of Metallurgy and Materials, University of Birmingham, UK.
E-mail addresses: dvb594@student.bham.ac.uk (D. Bacioiu), geoff.melton@twi.co.uk (G. Melton), m.papaelias@bham.ac.uk (M. Papaelias), robert.shaw@twi.co.uk (R. Shaw).

https://doi.org/10.1016/j.ndteint.2019.102139
Received 7 November 2018; Received in revised form 29 June 2019; Accepted 2 July 2019
Available online 06 July 2019

In another study, a different approach involving the assessment of spot welds with the help of computer vision, conducted by Yang et al. [12] reported a method of image registration based on geometric pattern matching, image segmentation, feature extraction and defect classification using an ANN.

Günther et al. [13] leveraged the power of machine learning in order to control the laser welding. In their study, the authors described an architecture composed of a deep auto-encoder neural network, used for data dimensionality reduction and synthesis of features invariant to the environment. The second stage is a prediction subsystem based on Reinforcement Learning (RL). The authors included in the overall system design, an additional third stage for controlling the laser power applied to the welding surface, by feeding the features resulting from the first auto-encoder subsystem, and the knowledge generated by the second part.

The state-of-the-art for image acquisition dedicated to arc weld monitoring are cameras with high dynamic range (HDR). The HDR is a property of the sensor which relates to its sensitivity to light. The sensor could offset the arc light without any additional filtering or laser illumination. The present study assesses the suitability of a vision-based monitoring technique based on HDR camera and machine learning for the analysis and categorisation of "good" vs. "defective" welds as well as specific defect identification.[1] The analysis adopts two neural networks architectures, Fully-connected Neural Networks (FCN) and Convolutional Neural Networks (CNN). This study considers two tests. The first test is discriminating between "good" vs. "defective" weldments (two-class test), while the second case is a six-class test for discriminating between 'good welds' vs. 'burn-through' vs. 'contamination' vs. 'lack of fusion' vs. 'lack of shielding gas' vs. 'high travel speed'.

2. Methodology

The camera mounts on the robot carrying the welding torch, trailing the weld, and at 45° relative to the welding direction, as depicted schematically in Fig. 1, or the real-life setup, in Fig. 2. The positioning facilitates the capture of the weld pool and the area immediately in front of the welding arc.

2.1. Welding process

The trials generated images of six defects starting with "good weld" and base parameters described in Table 1. The other defects emerged as the parameters deviated from base welding parameters as described in Table 1. An example is an increase in the heat input, rendering "burn through", while a decrease produced a "lack of fusion" defect.

Heat input is calculated as follow:

$$HI = \frac{V \times I}{TS} \times k$$

where V is the voltage, I is the current intensity, TS is travel speed and k is process thermal efficiency [14] (0.6 for TIG).

The production of other defects involved increasing the travel speed, introducing contaminant or turning off the supply of shielding gas.

The material used for welding was stainless steel grade 304 (SS304) plates of thickness 5 mm and 10 mm. The composition of the stainless steel plates used for the experimental work is detailed in Table 2. SS304 is a versatile and widely utilised material in a broad range of applications due to its good formability and weldability characteristics. In general, austenitic stainless steel is considered weldable by conventional fusion techniques although special attention is required towards ensuring the formation of ferrite in the weld deposit, otherwise "hot cracking" might occur.

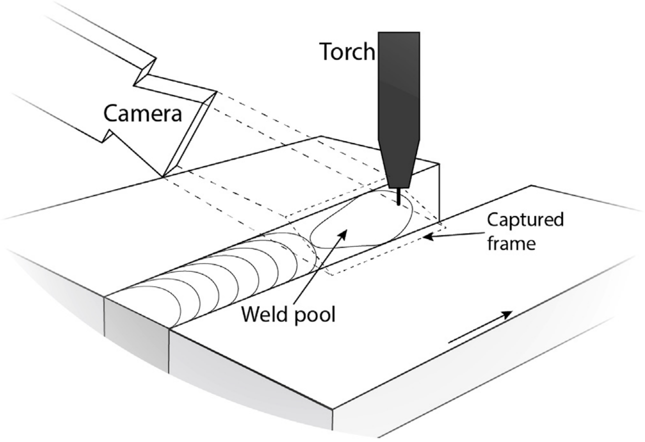

Fig. 1. TIG welding process layout.

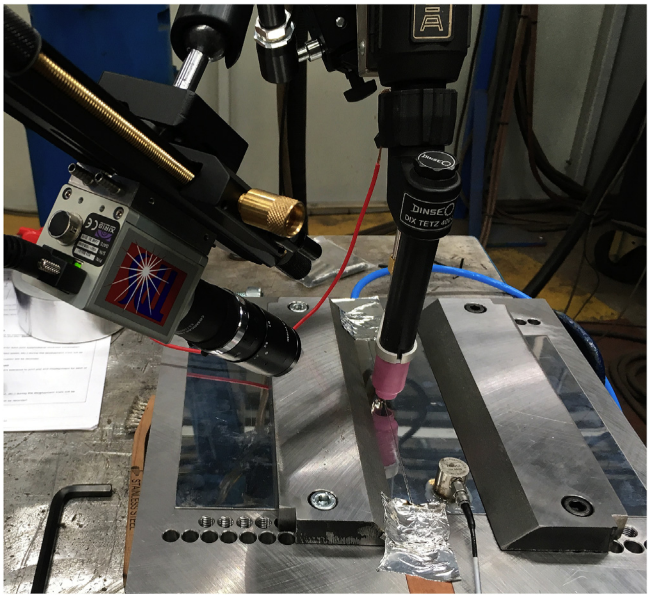

Fig. 2. Camera and robot setup.

Table 1
TIG welding process parameters.

Parameter	Baseline	Deviation from baseline
Gas flow rate (l/min)	30	[10, 15, 35, 40]
Traveling speed (cm/min)	19	[10, 10.5, 16, 23.2, 24.8, 26.4, 33.4, 50]
Voltage (V)	17.2	[12, 22]
Current (A)	200	[100, 150, 220, 235, 250, 270, 275, 300]

Table 2
Stainless Steel 304 composition.

Element	%
Fe	Balance
Cr	18.00–20.00
Ni	8.00–12.00
Mg	2.00 max
Si	0.75 max
N	0.10 max
C	0.08 max
P	0.045 max
S	0.03 max

[1] https://www.kaggle.com/danielbacioiu/tig-stainless-steel-304/.

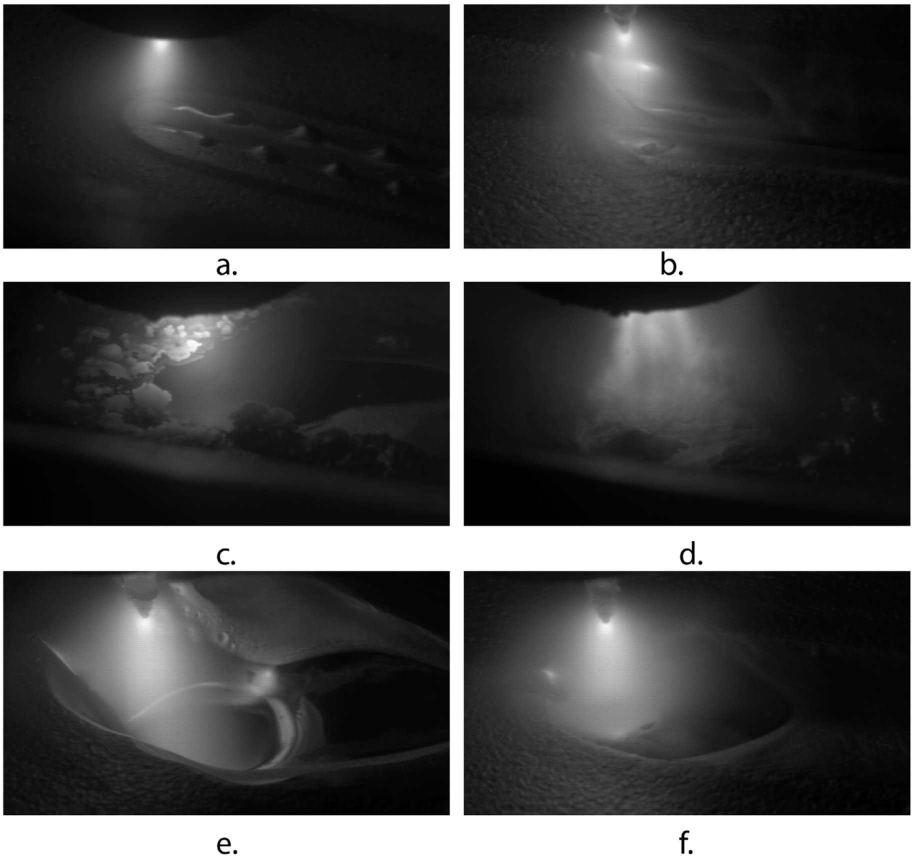

Fig. 3. Training samples. a) high travel speed, b) lack of fusion, c) contamination, d) lack of shielding gas, e) burn through, f) good weld.

2.2. Camera capabilities

Fig. 3 shows images extracted from the dataset, representative of the six categories considered for analysis in this study.

The camera used for the recording the images is Xiris XVC-1000. It has a dynamic range of 140 dB+, see Table 3, being able to absorb sufficient light to increase the luminosity the area surrounding the arc, while at the same time avoid the overexposure of the arc.

2.3. Data

The images recorded during the trials aim at capturing the weld pool and the area immediately ahead of the welding arc. The trials targeted particular defective and non-defective welds which were representative of the weld conditions. The dataset consists of 30008 images originating from 56 welding runs. The number of runs and images differs because the camera records at 55 frames per second, generating desired outcome representations at different stages during the weld. Each run is distributed to either train, validation or test subset. The assurance is necessary to reduce the correlation between subsets and assess the capacity of the networks for defect adaption and representation. Table 4 and Table 5 describe the dataset composition used for the two scenarios studied.

The six-class test (5 defects + good weld) is trained on a reduced dataset due to the discrepancy in the number of samples in each class. There still is an imbalance in the dataset but it is greatly reduced by eliminating samples from the classes with abundant data. Tables 4 and 5 describe dataset originating from the same 56 runs trials. The "defect" row in Table 4 incorporates all defects observed under one single category.

Table 3
Xiris XVC-1000 specifications [15].

Image Sensor	2/3″ Mono HDR CMOS
Speed/Resolution	Up to 55 FPS at 1280 (H) x 1024 (V) pixels
Dynamic Range	140 + dB
Shutter Range	1 μs - 53s Exposure

Table 4
Dataset split between training, validation and test for 2-class test.

Category	Number of samples		
	Train	Validation	Test
good weld	7871	3347	3812
defect	8452	2994	3532
Total	16323	6341	7344

Table 5
Dataset split between training, validation and test for 6-class test.

Category	Number of samples		
	Train	Validation	Test
good weld	954	875	769
burn through	977	646	731
contamination	967	339	576
lack of fusion	1005	780	744
lack of shielding gas	196	102	102
high travel speed	630	346	249
Total	4729	3088	3171

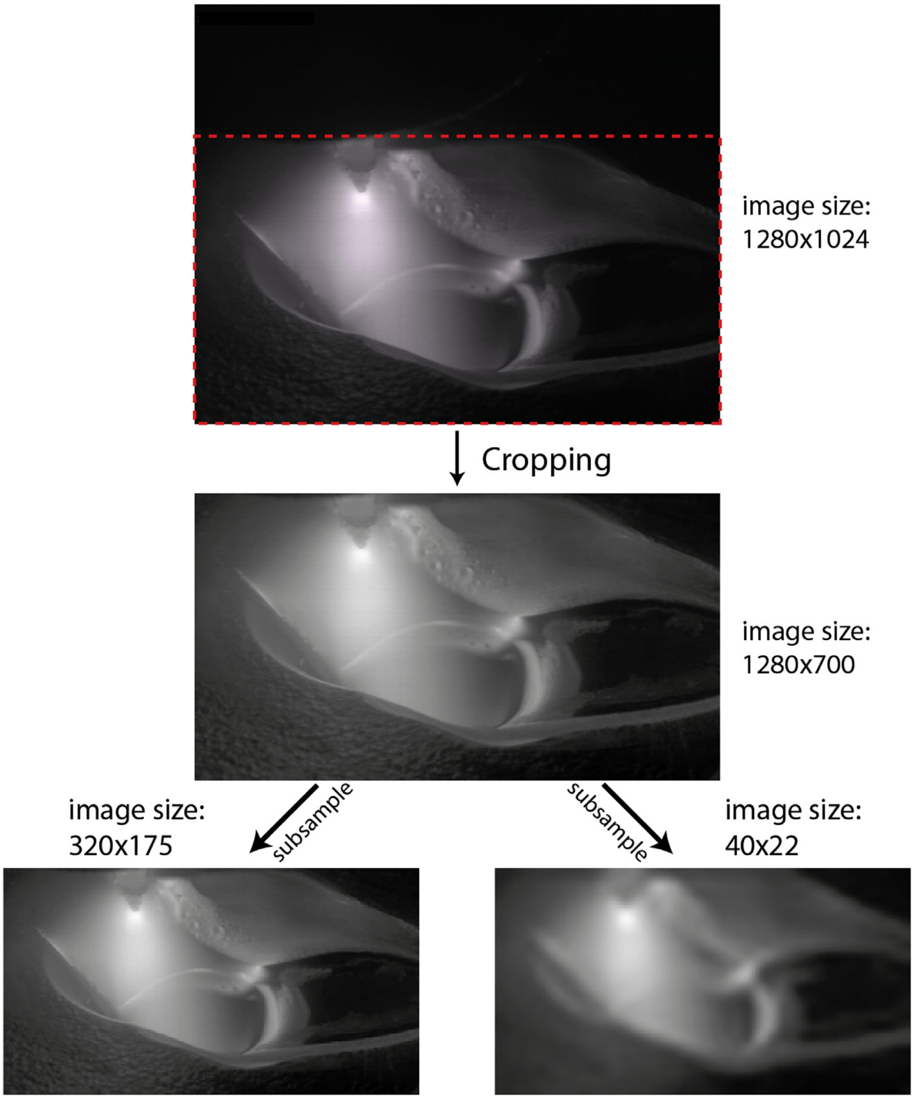

Fig. 4. Input image preprocessing.

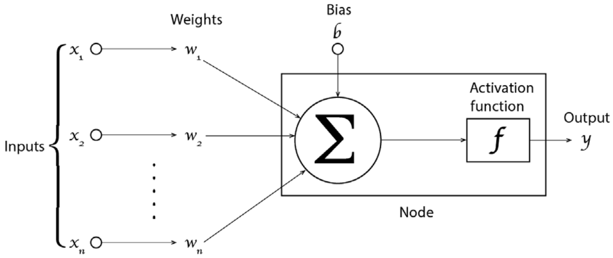

Fig. 5. Neural network node [21].

2.4. Preprocessing

Before processing the image, cropping is applied to the image to remove the area covered by black pixels bringing little information related to the weld quality. An example of the operation is shown in Fig. 4. Although the camera could deliver frames at a resolution of 1280 × 1024 pixels, the image dimension reduces to 1280 × 700 pixels after cropping.

The size at which neural networks processes the image is 320 × 175 due to processing unit hardware limitation. The neural network models remain small, hardware constraints are relaxed and processing time reduces.

The image subsampling has the potential of affecting the networks' accuracy performance adversely. The hypothesis is tested by subsampling the images from 1280 × 700 to 40 × 22 pixels then upsampling to 320 × 175. The results of the training are compared against the baseline results of training with images subsampled to 320 × 175 directly.

2.5. Processing

The definition of good welding conditions is challenging to pin down and contain within hard rules. One processing paradigm that adapts to this undefined nature and shows state-of-the-art performance for identifying the categories for images is neural network [16–19].

2.6. Fully-connected neural networks (FCN)

The fundamental component of a neural network is called a nodes (or sometimes neurone, borrowing the term from biology) [20] as in Fig. 5.

The mathematical equation governing every node inner operations is formulated as [21]:

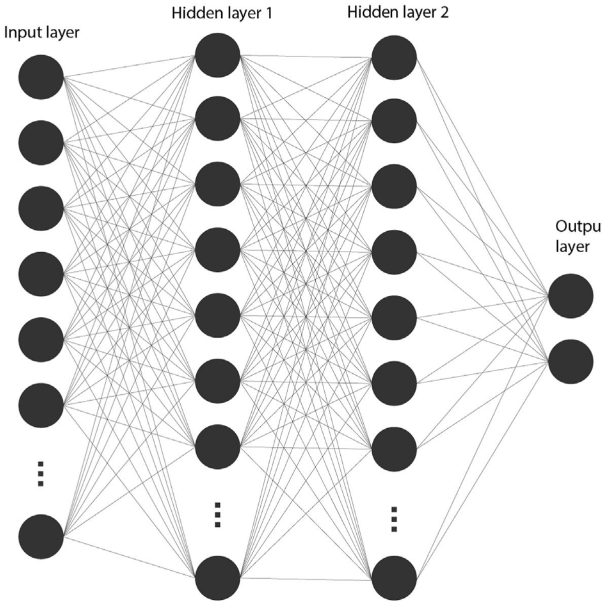

Fig. 6. Fully-connected Neural Network layout [20].

$$y = f\left(b + \sum_{i=1}^{n} x_i w_i\right)$$

where y is the output of the node, b is the bias, x_i is the incoming value, w_i is the weight, while f is a non-linear function (ex: sigmoid, tank, ReLu [22], etc.). The node are disposed in a lattice as shown in Fig. 6.

The first layer on the left is the input layer while the last layer to the right is the output layer. The values propagate from left to right (feed-forwarding) in cascade, each node collecting the outputs of previous layer's nodes, computing the node equation described above and passing the value to every node in the subsequent layer. The last layer is composed of a number of nodes equal to the number of classifying categories (in the present study, it is either two or six) and the node with the highest value gives the predicted class. At this stage, the error for logistic regression between the correct label and the predicted label is computed, the gradient of the loss calculated and propagated backwards through the network (also called back-propagation). The network's *weights* update to minimise the loss. In order to describe the calculations compactly and speed up the computation the inputs and weights are represented in a matrix format. The input to the network is a vector obtained by flattening the image (unrolling) as in Fig. 7.

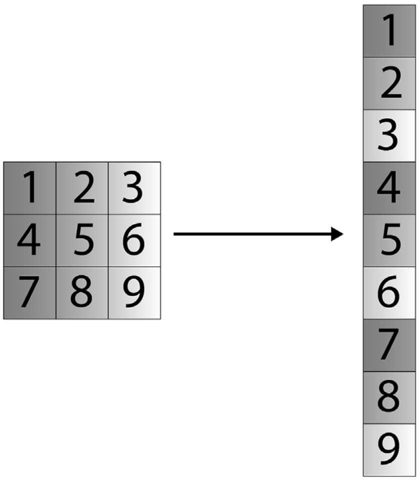

Fig. 7. Image (3 × 3 grid of pixels) flattening example.

Table 6
Fully-connected Neural Network Architecture considered in current study.

Type	Size
Unrolled image	56000
First fully-connected layer	64
Second fully-connected layer	64
Output layer	{6 or 2}

The 320 × 175 image flattens into a 56000 vector before being presented to the neural network. Each of 56000 input nodes connects with 64 nodes in the subsequent layer. The second layer connects each of its nodes to 64 nodes from the subsequent, third layer, and so on. The output layer has either two nodes (good weld vs. defective) or six nodes (6-class test) indicating which of the classes the image belongs.

Table 6 describes the number of internal nodes in the fully-connected network used in this study.

Typically, the images are compiled in a batch before the neural network to speed up the training stage. The test considered in this work uses an input of 56000 values (320 × 175 = 56000), reducing to 64 values in the second layer (hidden layer 1), 64 values in the third layer (hidden layer 2) and an output of two for the good weld vs. defective case and six outputs for the 6-class classification scenario.

2.7. Convolutional neural networks (CNN)

The concept of CNN is known for many years and was initially used to identify characters [23]. Their current widespread popularity and versatility became apparent when a deep CNN obtained improved results over the state-of-the-art in the ImageNet classification challenge [16]. Compared to fully-connected layers, convolutional layers have a less intuitive output, which depends on the shapes of the input, shapes of the kernel, the padding and stride, while the output from fully connected layers depends on the dimension of the weights matrix. Conceptually, the CNN preserves the feature-ordering (knows where a feature locates relative to another), takes advantage of multi-dimensional nature of input (red, green and blue channels represent data from different perspective), it is sparse (not all weights from the multiplication matrix contribute to the layer output) and share weights (same weights alter multiple locations from the input to the output) [24].

Fig. 8 shows the sliding dynamics of the kernel (brown) over an input image (yellow).

Each element (red) of the output (green) is the sum of multiplications between an input image, i.e. feature map, and kernel values at different locations. In practice several kernels are batched together, forming a convolutional layer, and all of them are applied over the

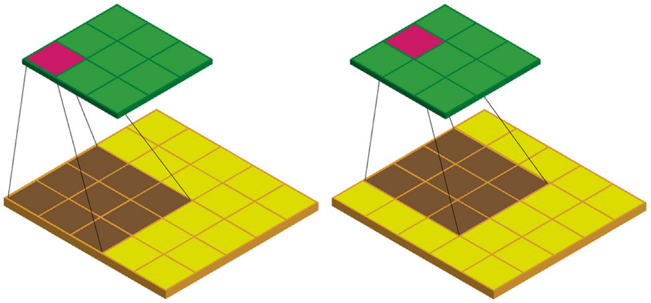

Fig. 8. Convolutions dynamics within CNN. Convolution between a feature map of dimension 5 × 5 (yellow) with a kernel of dimension 3 × 3 (brown), stride 1 and padding image 0. The result is another feature map of dimension 3 × 3 (green). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

Table 7
Convolutional Neural Network Architecture considered in the current study.

Type	Filters	Size
Image	1	320 × 175
First convolutional layer	16	3 × 3/2
Maximum pooling		3 × 3/2
Second convolutional layer	16	3 × 3/2
Maximum pooling		3 × 3/2
Third convolutional layer	16	3 × 3/2
Maximum pooling		3 × 3/2
Fully-connected layer		64
Output layer		{6 or 2 }

Table 8
Optimisation hyper-parameters. η = learning rate, β_1 = exponential decay rate for first moment estimate, β_2 = exponential decay rate for second moment estimate.

Hyper-parameter	Fully-con6	Fully-con2	Conv6	Conv2
η	10^{-5}	10^{-5}	5×10^{-5}	5×10^{-5}
β_1	0.99			
β_2	0.999			
epochs	5			
batch size	10			

image in the manner just described. Table 7 describes the number of internal nodes in the fully-connected network used in this study.

In the present study, the architectures investigated consist of three 3 × 3 convolution layers interlaced with three maximum pooling layers [25]. The maximum pool selects the maximum value from a 3 × 3 portion sliding across (striding) the image with the step size two (/2). The stride for the convolution layers is also two. Following the third maximum pooling, the resulting 16 × 1 × 4 output is unrolled into a vector and fed into a fully-connected layer of dimension 64, which connects to the final layer formed of either 6 or 2 nodes, depending on the number of categories within training set.

2.8. Convergence

The loss is calculated using the cross-entropy difference between the network assigned label for a specific image and the true label for the same image. Calculating the gradient (g_t) of the loss and back-propagating it through the network achieves the convergence towards the solution. During the back-propagation, to minimise the total loss, all the incoming weights for the nodes are altered. Successive forward-backward passes through the network successfully lead in most cases to the solution or very close approximation. Three parameters influence the back-propagation, controlling the steps size at which the network converges to the solution: learning rate (η), exponential decay rates for first moment estimate (β_1) and exponential decay rates for the second moment estimate (β_2). The optimisation algorithm used for computing the solution is Adaptive Moment Estimation (Adam) [26] and it works as follows:

Table 9
Testing accuracy.

Architecture	Testing
	accuracy (%)
Fully-con6	69
Conv6	93.4
Fully-con2	89.5
Conv2	75.5

- calculate exponential decay average of past gradients

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \tag{1}$$

- calculate exponential decay average of past squared gradients

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \tag{2}$$

- compute bias corrected first moment estimate

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{3}$$

- compute bias corrected second moment estimate

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{4}$$

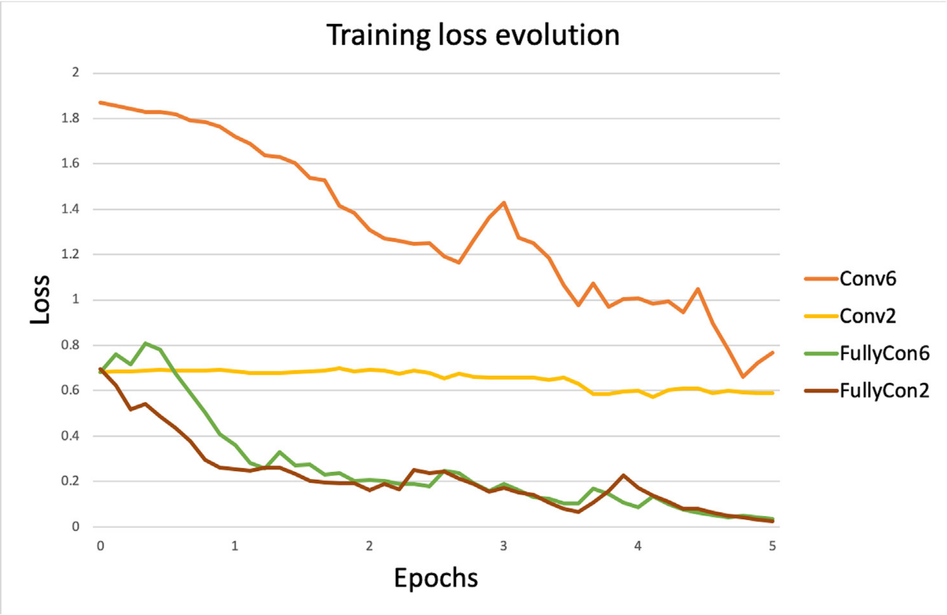

Fig. 9. Loss evolution during training stage.

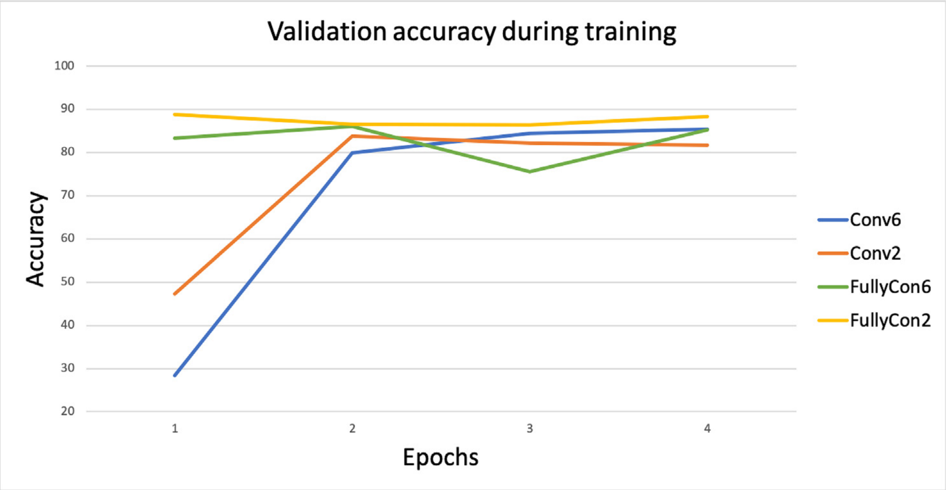

Fig. 10. Validation accuracy evolution during training stage.

Table 10
Metrics for 6-class test.

	Conv6			Fully-con6		
Metric	Precision	Recall	Macro F-score	Precision	Recall	Macro F-score
good weld	0.98	1	0.99	0.96	0.99	0.97
burn through	0.96	1	0.98	0.98	0.09	0.17
contamination	0.90	0.90	0.90	0.70	0.90	0.79
lack of fusion	0.91	0.96	0.94	0.89	0.99	0.94
lack of shielding gas	0	0	0	0.40	0.35	0.38
high travel speed	0.85	0.90	0.87	0.09	0.22	0.12
Average	0.77	0.79	0.78	0.67	0.59	0.56

Table 11
Metrics for 2-class test.

	Conv2			Fully-con2		
Metric	Precision	Recall	Macro F-score	Precision	Recall	Macro F-score
good weld	0.74	0.81	0.77	0.83	0.99	0.91
defective	0.77	0.69	0.73	0.99	0.78	0.87
Average	0.76	0.75	0.75	0.91	0.89	0.89

- update weights

$$\theta_t = \theta_{t-1} - \eta \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} \tag{5}$$

The optimisation is performed using the Adam algorithm with the hyper-parameters detailed in Table 8.

The neural network implementation used one of the many freely available frameworks for building and executing operations on graph structures, called Pytorch [27].

3. Results & discussion

The neural network output is the decision on which category an individual input image belongs. The metric for measuring the training time is *epoch* which represents one iteration through the entire training dataset, either 10988 for the six-class test or 30008 for the two-class test.

The current work analyses two architectures for each of the two tests taken into consideration, resulting in four sets of results. The two architectures taken into consideration, described above in Tables 6 and 7, are composed of approximately 3.6 million parameters and 5000 parameters, respectively.

Fig. 9 shows the training loss for each of the architecture analysed, during the training stage. Fully-con6 denotes an FCN with six output classes, Fully-con2 is an FCN with two output classes, Conv6 is a CNN with six output classes, and Conv2 is a CNN with two output classes.

The training stage loss evolution suggests the neural networks

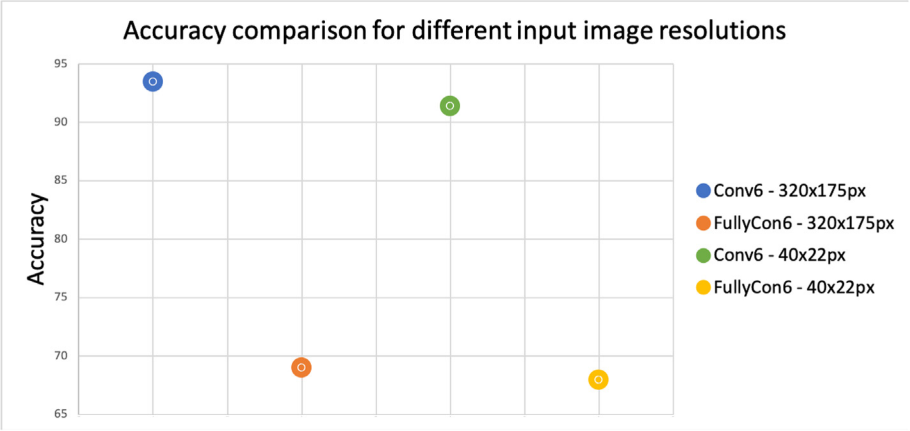

Fig. 11. Accuracy performance for different fidelities on 6-class test.

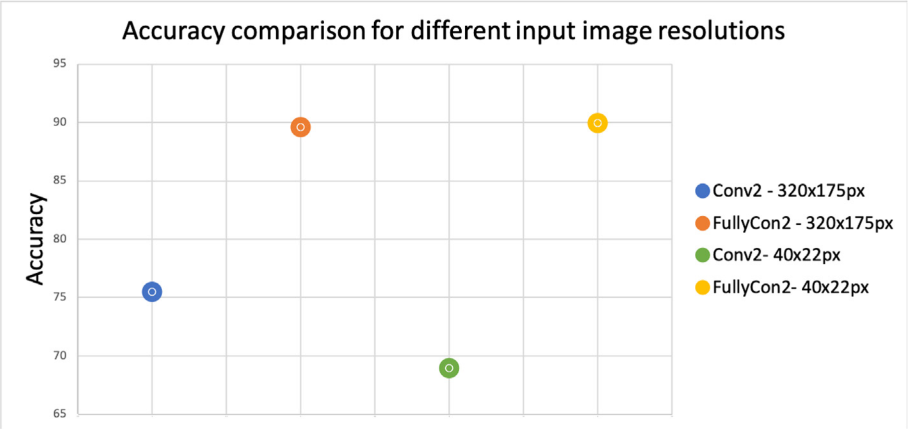

Fig. 12. Accuracy performance for different fidelities on 2-class test.

converge towards a solution. At the same time, the training loss for fully connected architectures is smaller that for convolutional architectures. However, comparing the accuracy after training, FCN architecture builds weaker representations of defects, conclusion arising from studying Table 9.

The principal metric for comparison of the two architectures in the present study is accuracy, calculated as:

$$accuracy = \frac{correct\ predictions}{total\ samples}$$

Accuracy has the benefits of reducing the entire complexity of model performance to a single number which could be easily compared. Overall, CNN performs better in terms on pure accuracy compared to FCN when applied to 6-class test, but worst on the 2-class test.

Fig. 10 shows the accuracy assessed on the validation subset of the models, at different points during the training stage. All models reach almost maximum performance and stable state very early in the training stage, suggesting the local minima found in the parameter space represent well the image categories.

This study analysis further the models accuracy, taking a closer look at different metrics, such as precision, recall and macro F-score, considering per-class errors. Per-class metrics, especially macro F-score, take into account false positives and false negatives. The metrics relating to the number of true positives while ignoring the true negatives, as the overall accuracy, convey less information regarding the multi-class classification. Values closer to 1 denotes models with higher capacity for discerning the defect, offsetting the effects of a skewed dataset and balancing the trade-off between high accuracy for one class against low accuracy for another.

Tables 10 and 11 show precision, recall and macro F-score for the models trained and tested on the 6-class and 2-class test, respectively.

Conv6 outperforms Fully-con6 with the most significant difference observed in the classification of ‘burn-though’ and ‘lack of shielding gas' defects. The Fully-con6 model does not recall a consistent part of ‘burn-though’ samples while the Conv6 completely misclassifying ‘lack of shielding gas' samples. Taking into account the ‘lack of shielding gas' representation in the dataset, 3.6%, the models weighted the class low during optimisation. Fully-con6 misclassifies ‘burn-though’, which could be seen as ‘lack of fusion’ or ‘high travel speed’ due to the presence of a gap while ‘lack of shielding gas' is a form of ‘contamination’.

The accuracy performance when tested on the 2-class dataset reverses, the convolutional network, Conv2, performs worst than a fully connected network, i.e. Fully-con2.

In case of a dataset where the samples have only one label, as in the current study, the accuracy is the same as micro F-score (not macro F-score). Micro F-score weights each sample (i.e. image), therefore classes with more samples tend to skew the score. Macro F-score weights each class, regardless of the number of samples within the class, therefore offsetting the imbalance. The divergence between the macro F-score and the accuracy is visible when applied on the 6-class test, and almost insignificant for the more balanced 2-class test.

3.1. Input image resolution impact

Figs. 11 and 12 show the effect of resolution reduction on the accuracy performance for all the neural networks assessed.

The image resolution after cropping is 1280 × 700. The hardware constraints would not allow assessing the networks at that resolution. Therefore the baseline resolution is 320 × 175. The assessment of fidelity loss reflected in the networks accuracy performance results from subsampling to 40 × 22, then upsampling to 320 × 175. The image losses details of the weld pool and surrounding area and the neural network architectures remain unchanged. The ablation analysis of the effect of fidelity loss shows that convolutional neural networks are affected more than fully connected neural network. We conclude that the pixels gradient loss and spatial distribution of features affect the convolutional filters more than the fully connected nodes. The nodes in fully connected networks are more agnostic to the value difference between pixels (gradient) or the features ordering and weight more the values of pixels.

4. Conclusion

In this study, we designed a new system for monitoring the TIG welding process using an HDR camera capturing the images of the weld pool and surrounding area, and a processing paradigm based on neural networks. Unlike previous studies that focused on the correlation of hand designed features to different weld properties, or spectral analysis to identify weld defects, while imposing welding process constraints through the use of laser illumination, our approach seeks to adapt better to the undefined nature of a good weld aspect. The camera successfully filters out the powerful light emitted by the arc while balancing the image and bringing up the details from the weld pool. The neural networks are capable of adapting to the process by learning the key differences between good and defective welds or recognising specific defects from weldments. The analysis focused on the performance assessment for fully-connected neural networks and convolutional neural networks in classifying weld defects.

CNN has the capability of learning more powerful representations of the defects and better balance the identification of one defect against misclassification of another, achieving an overall accuracy of 93.4%. Although superior on accuracy performance, the FCN proves to be more resilient across variations in input fidelity, as well as the capability of achieving 89.5% in classifying good vs. defective welds. The present study shows neural networks potential to adapt to industrial requirements contributing to increased productivity, quality and consistency for TIG welding processes.

Acknowledgements

The authors are indebted to the Engineering and Physical Sciences Council (EPSRC) and TWI Limited for providing the funding for this work. The authors also gratefully acknowledge the provision of facilities from TWI Limited, the National Structural Integrity Research Centre and the School of Metallurgy and Materials of the University of Birmingham.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ndteint.2019.102139.

References

[1] Jeffus LF. Welding: principles and applications. Thomson/Delmar Learning; 2004.
[2] O'Brien RL, American Welding Society. Welding handbook: welding processes. Welding handbook. American Welding Society; 2004.
[3] Weman K. 1 - introduction to welding. Welding processes handbook (second edition), woodhead publishing series in welding and other joining technologies. second ed.Woodhead Publishing; 2012. p. 1–12.
[4] Kovacevic R, Zhang Y, Li L. Monitoring of weld joint penetration based on weld pool geometrical appearance. Weld J - WELD J 1996;75:01.
[5] Lee SK, Na SJ. A study on automatic seam tracking in pulsed laser edge welding by using a vision sensor without an auxiliary light source. J Manuf Syst 2002;21(4):302–15.
[6] Vasudevan M, Chandrasekhar N, Maduraimuthu V, Bhaduri AK, Raj B. Real-time monitoring of weld pool during gtaw using infra-red thermography and analysis of infra-red thermal images. Weld World Jul 2011;55(7):83–9.
[7] Liu J, Fan Z, Olsen SI, Christensen KH, Kristensen JK. Boosting active contours for weld pool visual tracking in automatic arc welding. IEEE Trans Autom Sci Eng April 2017;14(2):1096–108.
[8] Li C, Shi Y, Gu Y, Yuan P. Monitoring weld pool oscillation using reflected laser pattern in gas tungsten arc welding. J Mater Process Technol 2018;255:876–85.
[9] Lucas W, Bertaso D, Melton G, Smith J, Balfour C. Real-time vision-based control of weld pool size. Weld Int 2012;26(4):243–50.
[10] Mirapeix J, Garca-Allende PB, Cobo A, Conde OM, López-Higuera JM. Real-time arc-welding defect detection and classification with principal component analysis and artificial neural networks. NDT E Int 2007;40(4):315–23.
[11] Song HS, Zhang YM. Three-dimensional reconstruction of specular surface for a gas tungsten arc weld pool. Meas Sci Technol 2007;18(12):3751.
[12] Ou Y, Li Y. Quality evaluation and automatic classification in resistance spot welding by analyzing the weld image on metal bands by computer vision. Int J Signal Process Image Process Pattern Recogn 2015;8:301–14. 05.
[13] Günther J, Pilarski PM, Helfrich G, Shen H, Diepold K. Intelligent laser welding through representation, prediction, and control learning: an architecture with deep neural networks and reinforcement learning. Mechatronics. 2015.
[14] What is the difference between heat input and arc energy? http://www.twi-global.com/technical-knowledge/faqs/material-faqs/faq-what-is-the-difference-between-heat-input-and-arc-energy/. Accessed: 2019-03-27.
[15] Xiris open-arc camera. https://www.xiris.com/products/open-arc-cameras/xvc-1000/. Accessed: 2019-03-27.
[16] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Pereira F, Burges CJC, Bottou L, Weinberger KQ, editors. Advances in neural information processing systems, vol 25. Curran Associates, Inc.; 2012. p. 1097–105.
[17] M.D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. CoRR, abs/1311.2901, 2013.
[18] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014;1556. CoRR, abs/1409.
[19] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. CoRR, abs/1512.03385, 2015.
[20] Rosenblatt F. Principles of neurodynamics. Spartan Books; 1959.
[21] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT Press; 2016http://www.deeplearningbook.org.
[22] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In Geoffrey Gordon, David Dunson, and Miroslav Dudk, editors, Proceedings of the fourteenth international conference on artificial intelligence and statistics, volume 15 of proceedings of machine learning research, pages 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.
[23] LeCun Y, Bottou L, Bengio Y. Reading checks with multilayer graph transformer networks. 1997 IEEE international conference on acoustics, speech, and signal processing. vol 1. Apr 1997. p. 151–4. vol. 1.
[24] Dumoulin V, Visin F. A guide to convolution arithmetic for deep learning. 2016. cite arxiv:1603.07285.
[25] LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE 1998;86(11):2278–324.
[26] D.P. Kingma and J. Ba. Adam: A method for stochastic optimization. CoRR, abs/1412.6980, 2014.
[27] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In NIPS-W, 2017.