Myocardial Infarction Detection and Quantification Based on a Convolution Neural Network with Online Error Correction Capabilities

Shui-Hua Wang
School of Mathematics and Actuarial Science
University of Leicester
Leicester, United Kingdom
sw546@le.ac.uk

Gerry McCann
Department of Cardiovascular Sciences
University of Leicester
Cardiovascular Research Centre Glenfield
General Hospital
Leicester, United Kingdom
gpm12@le.ac.uk

Ivan Tyukin
School of Mathematics and Actuarial Science
University of Leicester and Norwegian University of Science and Technology and Saint-Petersburg State Electrotechnical University
Leicester, United Kingdom and Trondheim, Norway and Saint-Petersburg, Russia
i.tyukin@le.ac.uk

Abstract—Myocardial infarction (MI), more commonly known as heart attack, occurs when the blood flow to the heart decreases or stops. Over 100,000 people each year in the UK suffer from an MI according to the report by British Heart Foundation. Following an MI, there is irreversible heart muscle damage that will become scar. The amount of scar following larger heart attacks, ST segment elevation myocardial infarction, drives enlargement of the heart and is associated with worse prognosis (increased risk of death and subsequent heart failure). Cardiac Magnetic Resonance Imaging (MRI) late gadolinium enhancement (LGE) has become the "gold standard" for the visualization of MI. However, to date, no "gold standard" fully automated methods exist for the quantification of MI from MRI.

In this work, we propose an approach to construct such methods using Artificial Intelligence (AI) and Machine Learning (ML) technologies, in particular, Convolutional Neural Networks (CNN). Uncertainties, variability, and a possibility of bias inherent to any data imply that data-driven systems which are intended for use in clinical research and practice must be capable of learning from mistakes on-the-job. Here we develop and test a first deep learning CNN system *with error correction capabilities* (CNNEC) for the detection and quantification of MI. The system could be viewed as a proof-of-principle for the technology.

Keywords—Convolution neural network, Error correction, Myocardial infarction, Automatic detection.

I. INTRODUCTION

Myocardial infarction (MI), more commonly known as heart attack, occurs when the blood flow to the heart decreases or stops, which is usually caused by arteriosclerosis with narrowing of the coronary arteries and eventually leading to a thrombosis [1, 2] Following an MI, there is irreversible heart muscle damage that will become scars. The amount of scar following larger heart attacks, ST segment elevation myocardial infarction, drives enlargement of the heart and is associated with worse prognosis (increased risk of death and subsequent heart failure). Each year, over 100,000 people in the UK suffer from an MI according to the report by British Heart Foundation [3].

Late gadolinium enhancement with cardiac magnetic resonance has become the non-invasive "gold standard" technique for visualization of MI, which will predict the adverse cardiovascular outcomes in patients with post myocardial infarction, heart failure and in cardiomyopathies [4]. However, to date, there is no "gold standard" or fully automated method to detect or quantify the volume of MI.

Typical methods for the quantification of infarct size include manual planimetry (Manual), visual scoring.

In recent years, several groups have been developing tools for semiautomatic/automatic detection and quantification of MI. For example, Eitel, et al. [5] Kim, et al. [6] and Amado, et al. [7] proposed standard deviation (SD)-based thresholding techniques, and compared manual contouring, different thresholding SD techniques and full width at half maximum method. Flett, et al. [8] proposed an automated feature analysis and combined thresholding method. Hsu, et al. [9] proposed another automatic method based on computer-aided quantification by taking as input pairs of raw TIR images and corresponding global Left Ventricular (LV) contours, and a mass of edematous tissue as output. Tong, et al. [10] proposed the recurrent interleaved attention network for cardiac MRI segmentation: LV, right ventricular and myocardium. Xu, et al. [11] proposed the segmentation method based on spatiotemporal generative adversarial learning. Chenchu Xu [12] proposed long short-term memory recurrent neural network (LSTM-RNN) for MI detection. Héloïse Bleton [13] proposed MI localization based on Neighbourhood Approximation Forests (NAF) and compared with the stack autoencoder method. Fahmy, et al. [14] developed a CNN model for automatic cardiac MR scar quantification.

Although great progress has been achieved in the task of cardiac MRI segmentation and quantification, several fundamental issues remain. The problem of manual/semi-automated methods is that they mainly rely upon personal experience of clinical experts involved in the segmentation task, and, as a result, these methods are prone to low reproducibility due to the intra- and inter-observer variability. Automatic methods aim to overcome the shortcomings. However, most automatic MI detection methods, and especially those based on data-driven AI and ML approaches [14], are not fully free from the variability and bias issues associated with the manual tracing of the LV myocardial contours. Such manual labelling, and hence the variability and bias, become embedded into the model development pipelines, through training and testing datasets.

To overcome these fundamental issues, we propose a novel fully automatic MI detection and quantification model based on deep learning CNNs with error correction capabilities (CNNEC). The error correction part of the proposed CNN model is one-shot trainable and is based on prior theoretical work [15-18] on the concentration of measure
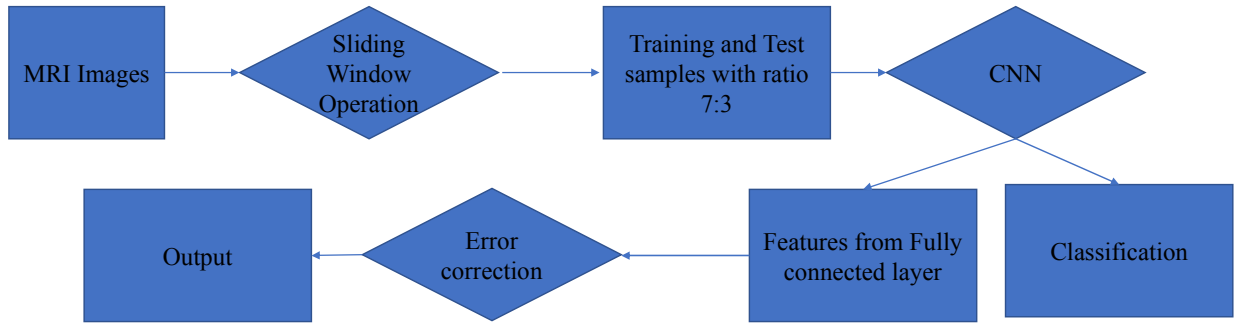
Fig. 1. Flowchart of the proposed model

in high dimension. Equipping the backbone CNN with an efficient one-shot learning compartment enables continuous improvements of the scar detection accuracy and opens a way to address the challenge of data bias and incompleteness.

The rest of the paper is organized as follows. Section I describes our methodology for automatic detection and quantification of MI; it also details the proposed approach for continuous AI learning on-the-job. Section II shows the experiment parameters and results. Section III concludes the paper and outlines directions for future work.

I. METHODS

Continuous progress in computing technologies and increasing availability of data are driving the application of AI to biomedical data. Jointly, these technologies are now an established framework to solve numerous problems in pattern recognition, classification, and segmentation. In medical imaging, these technologies are increasingly ubiquitous and are being successfully applied in radiology, pathology, and dermatology. For example, Viz.LVO software, which uses AI to automatically identify suspected large vessel occlusion strokes in CT angiogram imaging, received the Food and Drug Administration (FDA) clearance in February 2018. In August 2018, Aidoc received FDA clearance for the first product in its AI-based workflow optimization solution helping radiologists to mark acute intracranial hemorrhage cases in head CTs.

In this work, we make a step forward towards developing an AI model for MI detection and quantification with the capability to learn on-the-job.

The proposed AI model is composed of two compartments: a Convolution Neural Network (CNN) and an error correction part based on the Stochastic Separation Theorems (SST). Considering the widely evidenced good performance of CNN in various fields, such as object detection and patter recognition, we used CNN for the initial detection. At the same time, no matter how well a data-driven AI system is trained, mistakes will always exist. The mistakes can be caused by data misrepresentation, imprecise training or uncertainty in empirical data. Therefore, we equipped the backbone AI system (a CNN) with error correction capabilities so that its performance can be fine-tuned on-the-job. As AI aims to bridge the gap between the capabilities of humans and machines, AI error correction aims to overcome AI adoption barriers in clinical applications. The flowchart of the whole proposed system is shown in Fig. 1.

A. Convolutional Neural Network

A CNN is a deep learning model that has originally been introduced in 1980 [19] but has become increasingly influential in the field of computer vision since 2012, initially due to its remarkable performance in ImageNet competitions [20] followed by successful applications to a broad range of computer vision tasks (see e.g. [21] for a comprehensive overview). When processing inputs, CNN models rely on their trainable layers to extract relevant features from data as opposed to being supplied with structured data by subject experts. When the inputs propagate through layers in a CNN starting from the initial layer to deeper parts of the network, more low-level and high-level features can be extracted by CNN. A typical CNN is composed of different types of layers, including an input layer that holds the pixel values of the input images, an output layer for the classification result, convolution layers, ReLU layers, pooling layers and fully connected layers [21]

Convolution layer (CL): CL is the core component of a CNN. CL applies a set of learnable filters to an input: for each filter, it computes a dot product operation (convolution) on the filter and a filter-sized patch of the input across images. Suppose the input's size is $H_i \times W_i \times D_i$, where H stands for the input height, W for input width, D for input depth and i is the layer's index, the filters' hight and width is F, the stride length is S and the padding size is P. Then the i-th CL produces a feature map of size $H_{i+1} \times W_{i+1} \times D_{i+1}$, representing the input, where $H_{i+1}=(H_i - F + 2P)/S + 1$, $W_{i+1}=(W_i - F + 2P)/S + 1$ and D_{i+1} is the number of filters.

ReLU (Rectified Linear Units) layer: To implement and account for non-linear dependencies between different features, feature maps (or their linear combinations) generated by CLs are often passed through a ReLU layer. Each element of the ReLU layer performs a simple nonlinear operation on its input u: $ReLU(u) = \max\{0, u\}$. In addition to acting as a continuous threshold gate, ReLU functions are known to be universal approximators which makes them suitable for computationally efficient implementations of complex decision boundaries.
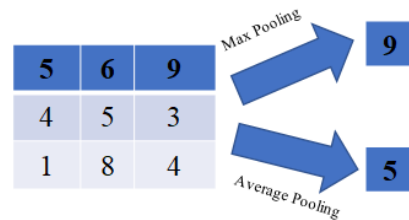

Fig. 2. Illustration of Max pooling and average pooling operations

Pooling Layer: After ReLU is applied to the feature maps generated by a convolutional layer, the pooling layer aims to reduce the feature dimensions by combining the outputs of

several neurons into a single neuron. There are two main types of pooling techniques: max pooling and average pooling, which compute the max or average value of over a group of neurons, respectively. Fig. 2 illustrates the max pooling and average pooling with over a group of 9 neurons organized into a *3x3* cluster.

Fully connected layer: Neurons in a fully connected layer have full connections to all activations in the previous layer, which is the same as the conventional artificial neural network [22]. The structure of CNN was shown in Fig. 3
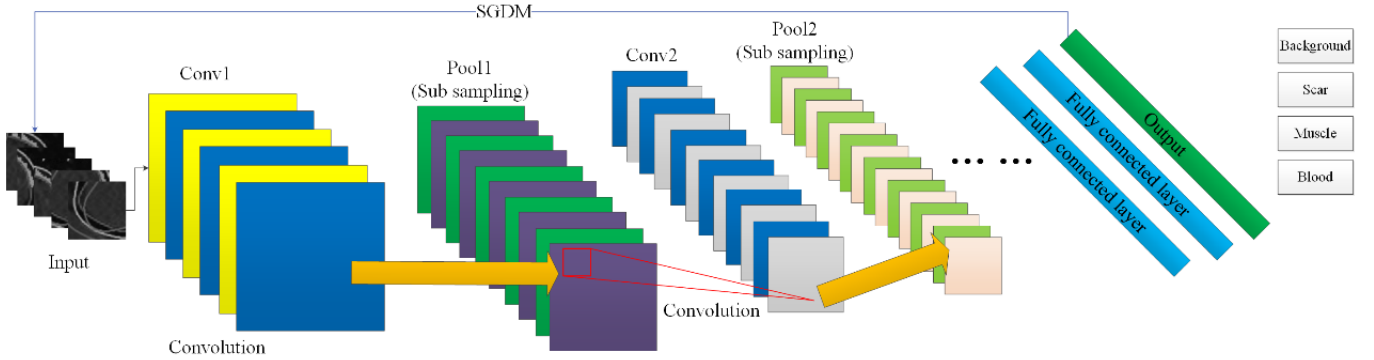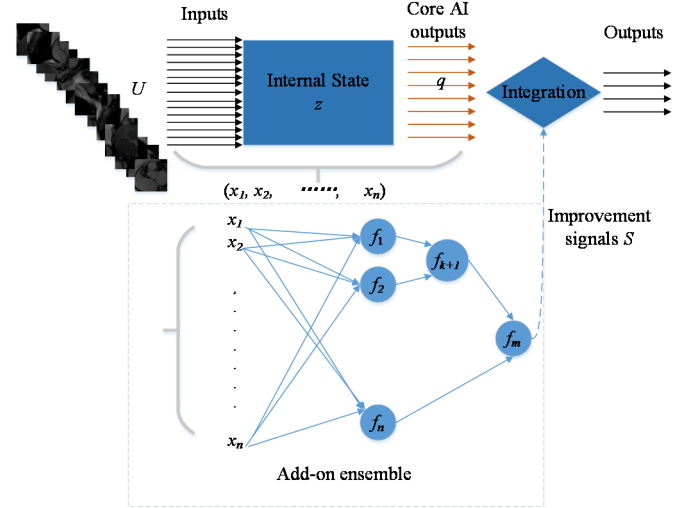

Fig. 3.   The structure of CNN

B. Error correction

Following [15, 17, 18], in order to equip our model with capabilities to learn on-the-job and to further improve the accuracy, we introduced an error correction module into the processing architecture of the system. The module consists of small neuronal ensembles and their cascades and is designed to remove the errors with high probability without negatively impacting on the correct performance of the backbone system. Mathematical justification of this property for a broad class of distributions as well as several cases studies are presented in [15, 17, 18]. Compared to other related techniques, such as the classical cascade correlation [23], neurogenesis deep learning [24], randomized methods for training neural networks [25], greedy approximation [26] and so on, training of the correction cascades can be achieved in a one-shot or few-shot manner using simple Fisher Linear Discriminants. This is particularly advantageous when large-size training datasets are available and preferable. Below we describe a version of the error correction cascade which was used in our model.

A generic AI system can be viewed as an operator that maps elements of its inputs to the outputs. Here, the inputs are MRI images and outputs are labels of pixels in relevant areas of these image. The inputs of the AI system are represented as $u \in U$, the outputs as $q \in Q$, and the internal state as $z \in Z$. The complete state of the AI system therefore can be expressed as a triple *(u, z, q)* (see Fig. 4). With each triple *(u, z, q)* we associate an element $x \in R^n$ constructed from *(u, z, q)*. For example, x may be a vector of the output of the j-th layer. We suppose that elements x are automatically produced from *(u, z, q)*. In addition to the backbone AI system, an add-on module is introduced to map x to the improvement signal $s \in S$. The set S could be a subset of R or simply a two-element set {0,1}. The add-on improves performance of the system by correcting erroneous responses of the backbone AI.

For the add-on part, the operation performed at the individual node can formally be expressed as

$$f_i(x) = f(< w_i, x > -c_i)$$

Fig. 4.   the structure of AI system with error correction as the add-on module

in which, $f: \mathbb{R} \to \mathbb{R}$ is some function, $c_i \in \mathbb{R}$ and $w_i \in \mathbb{R}^n$. For the sake of simplicity, we assume that the function f satisfies: $f(s) > 0$ for all $s \geq 0$ and $f(s) \leq 0$ for $s < 0$.

Let X be a finite set of elements x that have been collected. We partition the set X into two sets:

$$C = \{x_1, x_2, \dots \dots, x_C\}$$
$$E = \{x_{C+1}, x_{C+2}, \dots \dots, x_{C+k}\}$$

The set C contains elements x that correspond AI's expected behaviour. The set E contains vectors x corresponding to AI's errors. Informally, training of the addon ensemble aims to construct a map $f_m(\cdot)$ separating the set C from E.

Different approaches can be used to construct such add-on ensembles, including stochastic configuration networks [27], error back propagation, or algorithms based on greedy approximation [26]. Here we used the algorithm proposed in our earlier work [16], mainly due to its computational efficiency. Basic steps of the algorithm are described in Steps $1 - 8$ below. Further discussion and justification of these can be found in [16].

Step 1. Initialization: $C^1 = C$, $E^1 = E$, $S^1 = S = C \cup E$.

For the i_{th} iteration, the input is C^i, E^i and $S^i = C^i \cup E^i$. We let N be the number of clusters at the i-th iteration, and α be the value of filter threshold.

Step 2. Centering: subtract the empirical mean of the set S^i from all elements of E^i and S^i

$$E_c^i = \left\{ x \in \mathbb{R}^n | x = \delta - \overline{x(S^i)}, \delta \in E^i \right\}$$

$$S_c^i = \left\{ x \in \mathbb{R}^n | x = \delta - \overline{x(S^i)}, \delta \in S^i \right\}$$

Step 3. Regularization (control overfitting by preventing learning too complicated models and ensure sufficient numerical stability of calculations):

$$S_{reg}^i = \{ x \in \mathbb{R}^n | x = H^T \delta, \delta \in S_c^i \}$$

$$E_{reg}^i = \{ x \in \mathbb{R}^n | x = H^T \delta, \delta \in E_c^i \}$$

where, $H = (h_1 | h_2 | \ldots | h_r)$ is the matrix composed of some eigenvectors (principal components) of the empirical covariance matrix of S^i. Exact choice of the principal components may depend on regularisation criteria, including conditioning numbers, the broken stick test, or the Kaiser–Guttman test.

Step 4. Whitening (optional):

$$S_w^i = \left\{ x \in \mathbb{R}^n | x = W\delta, \delta \in S_r^i, W = Cov(S_r^i)^{-\frac{1}{2}} \right\}$$

$$E_w^i = \left\{ x \in \mathbb{R}^n | x = W\delta, \delta \in E_r^i, W = Cov(E_r^i)^{-\frac{1}{2}} \right\}$$

Step 5. Projection (optional): to project the whitened elements S_w^i and E_w^i to the unit sphere via scaling all the elements to unit lengths: $x \rightarrow \vartheta(x), \vartheta(x) = \frac{x}{\|x\|}$.

Step 6. Training: Divide E_w^i into m sets as $e_{w,1}^i, e_{w,2}^i, \ldots, e_{w,m}^i$, of which the elements are pairwise positively correlated. Construct the hyperplanes h_j :

$$w_j = [Cov(S_w^i \backslash e_{w,j}^i) + Cov(e_{w,j}^i)]^{-1} [\overline{x(e_{w,j}^i)} - \overline{x(S_{w,j}^i / e_{w,j}^i)}]$$

$$h_j(x) = \left\langle \frac{w_j}{\|w_j\|}, x \right\rangle - min_{\delta \in e_w^j} \left\langle \frac{w_j}{\|w_j\|}, \delta \right\rangle$$

The hyperplane is retained only when $c_j = min_{\delta \in e_w^j} \left\langle \frac{w_j}{\|w_j\|}, \delta \right\rangle > \alpha$ in which a new corresponding element is created in the ensemble for the retained hyperplane:

$$f_j(x) = f\left(\left\langle \vartheta(WH^T(x - \overline{x(S^i)})), \frac{w_j}{\|w_j\|} \right\rangle - c_j \right)$$

Step 7. Integration: All samples generated by the backbone AI should pass through the ensemble. For any x, if $f_j(x) > 0$, an error correction action is performed. In this application, the error correction action is to alter the incorrect label.

Step 8. Testing and cascading. Assess performance of the combined system on a test set. If needed, generate the new sets $C^{i+1}, E^{i+1}, S^{i+1}$ and repeat Step 2-Step 8.

II. Experiments

A. Dataset

The MRI images were collected from patients with MI and from a healthy control group. MI, muscle, and blood pool areas were contoured by expert clinicians. The contoured images were then cropped to 256 by 256 size. Throughout the entire development phase of the project, we did not have access to non-contoured copies of MRI images. To recover/emulate the original non-contoured MRI scans, contoured greyscale areas in the images were manually processed with the aim to restore the original appearance of each image.

For the reference ground truth, we employed experts to manually label the raw images at the pixel level into the following four categories: background, blood, muscle and scar using Image Labeller integrated into MATLAB.

Based on the reference ground truth, sliding window sampling with a window size of 61 by 61 pixels was employed to slide across the target MRI images from left to right and from top to bottom with stride size as 1 as shown in Fig. 5 to generate training and test samples. 70% of the data (rounded to the nearest integer) was used for training, and the remaining 30% formed the test set. Considering all the left ventricular (LV) located in the central part of the collected MRI images, we used pixels in the interval [80 171] in both dimensions. For each 61x61 grayscale input, the class of the input was determined by the true class of the central pixel in the image. Each 256 by 256 image produced 8, 281 images of size 61 x 61. Figure 6 shows some samples of input.

Overall, 566 MRI images were processed which resulted in 4, 687, 046 input samples, of which 78, 146 corresponded to scar tissue, 1,844,311 to background, 924, 529 to muscle and 1, 840, 060 to blood. The distribution of label frequencies is shown in Fig. 6. As these figures suggest, the data sample is severely unbalanced. It is well-known that unbalanced datasets may bias the model's prediction towards the mostly frequent classes. To compensate for the risk of this bias, we forcefully balanced the data by randomly down-sampling the other the three classes: background, blood pool and muscle. This process resulted in 308, 708 input samples, of which 78, 146 were scar samples.

TABLE I.   STRUCTURE OF THE CNN

Layer name	Output size	Stride	Padding	Filter size and No
Convolution	21x21	2	1	3*3*50
ReLU+maxpooling[3*3]	21x21	1	1	
Convolution	9x9	1	0	5*5*100
ReLU+maxpooling[3*3]	9x9			
Convolution	9x9	1	1	3*3*150
ReLU+maxpooling[3*3]	9x9	1	1	
Convolution	9x9	1	1	3*3*200
ReLU+maxpooling[3*3]	9x9	1	1	
Convolution	9x9	1	1	3*3*250
ReLU+maxpooling[3*3]	9x9	1	1	
Fully Connected Layer				100
Fully Connected Layer				4

TABLE II.   TRAINING PARAMETERS

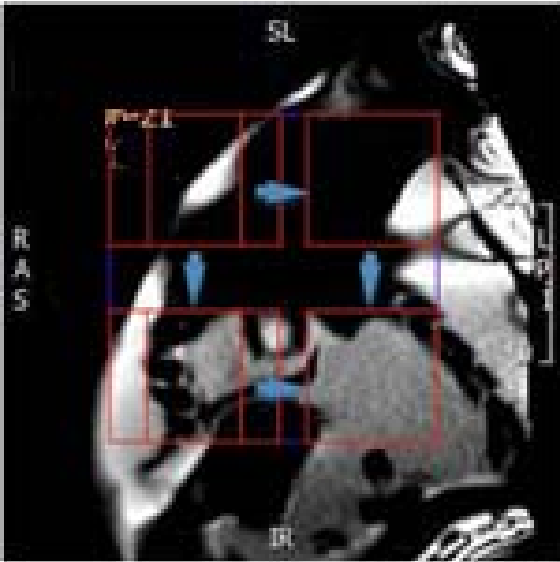|	
Max Epoch	30
Initial Learning rate	0.001
Mini Batch Size	128
Training algorithm	SGDM
Execution environment	GPU

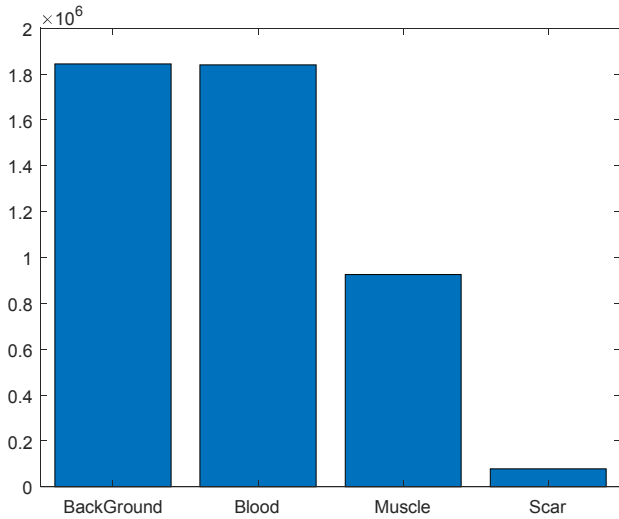Fig. 5. Illustration of the sliding window

Fig. 6. Data distribution of all the elements

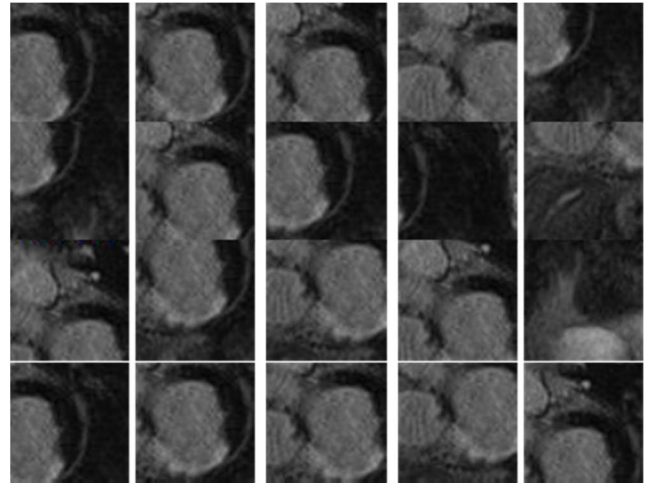
Fig. 7. Examples of input samples

The architecture of the backbone CNN model we developed is shown TABLE I. Dimension of the input layer was set to 61*61*3. After fixing the architecture and training the backbone AI model, an error correction addon [20a], [20b], as described in Steps 1 – 8, was implemented. Training parameters for the backbone AI are shown in TABLE II

B. Results

On the data which has been subjected to manual removal of contouring for the purposes of training, the backbone CNN achieved the accuracy of 94.5% in detecting background, 95.3% for scar tissue, 92.9% for muscle, and 98.0% for blood. In addition, in the task of detecting MI, the backbone CNN model achieved 95.33% sensitivity, 98.70% specificity. Fig. 8 shows the convergence curve line, and Fig. 9 shows the confusion matrix for each category, where '0' stands for the background, '1' stands for the scar tissue, '2' stands for the muscle tissue, and '3' represents for the blood. The Confusion matrix shows that, on our dataset, the algorithm were able to segment MI areas from blood and background. Out of 20316 scar pixels from the test set, 855 pixels were misclassified as muscle, 34 were misclassified as blood and 60 pixels were misclassified as background; 798 pixels were misinterpreted as scar, of which 684 pixels were from muscle.

After training the CNN, we extracted the features from the first fully connected layer, which were then used as an input to the error correction module. Correlations between errors in the feature space is shown in Fig. 10. The error correction cascade was built to improve detection of pixels corresponding to scar tissue only. After the error correction, the accuracy of detection for scar tissue improved to 98.2%. These results are summarized in TABLE III.
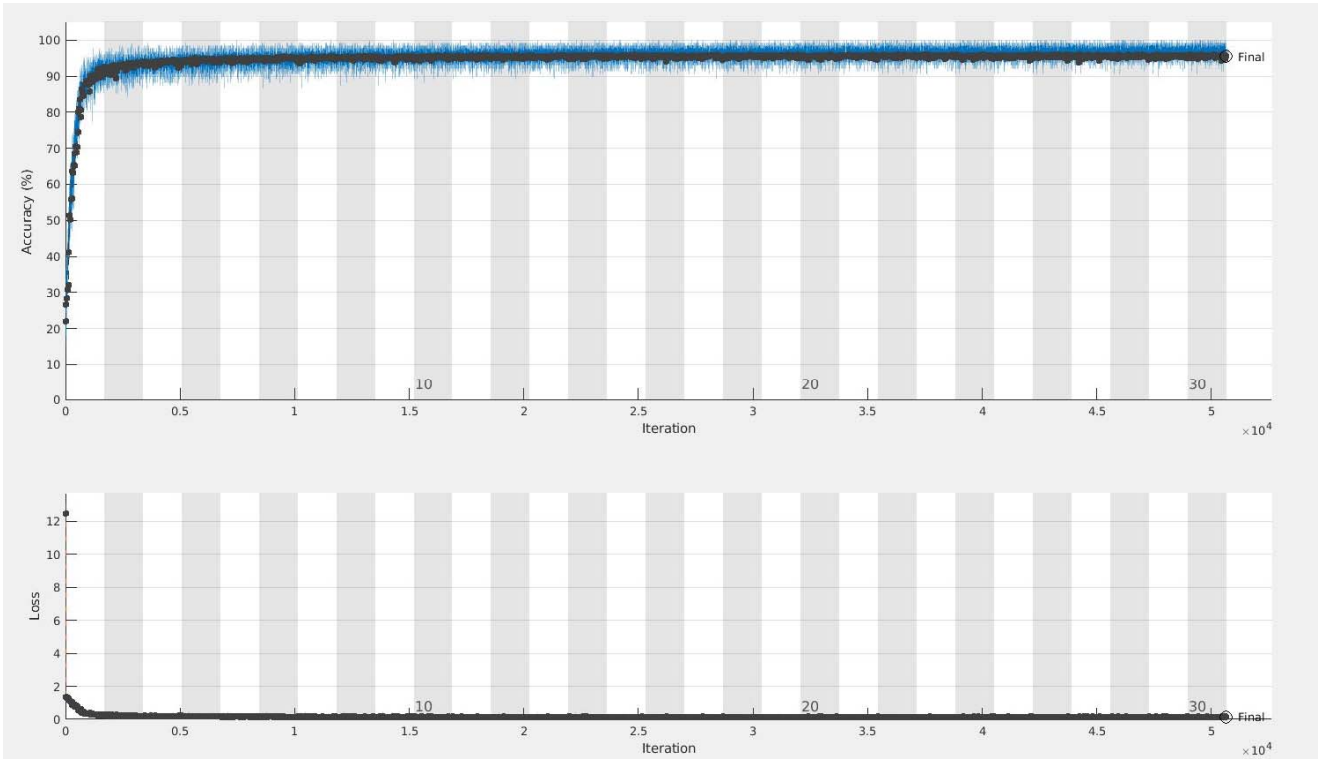
Fig. 8. Convergence curve

Fig. 9. Confusion Matrix '0' stand for background, '1' stands for the scar tissue, '2' means the muscle, and '3' represents for blood)

Fig. 10. Color coded values of $\left|\left\langle \frac{x_i}{\|x_i\|}, \frac{x_j}{\|x_j\|} \right\rangle\right|$ for the error data art

In order to test performance of the model in MI *quantification* task, we compared the CNNEC with the reference ground truth at the level of individual cases (that is, at the level of a patients, as opposed to mere pixel-level performance). The result is shown in Fig. 11. The scatter plot in Fig. 11 is based on 19 independent cases. In Fig. 11, "predicted true scar" label stands for true positive data. The "predicted scar" label includes both true positive and false positives. The largest gap between the reference ground truth for scar tissue and predicted scar tissue was 213 pixels (out of 1006), which means that we missed 213 pixels of scar tissue in the MI quantification task based on the proposed model.
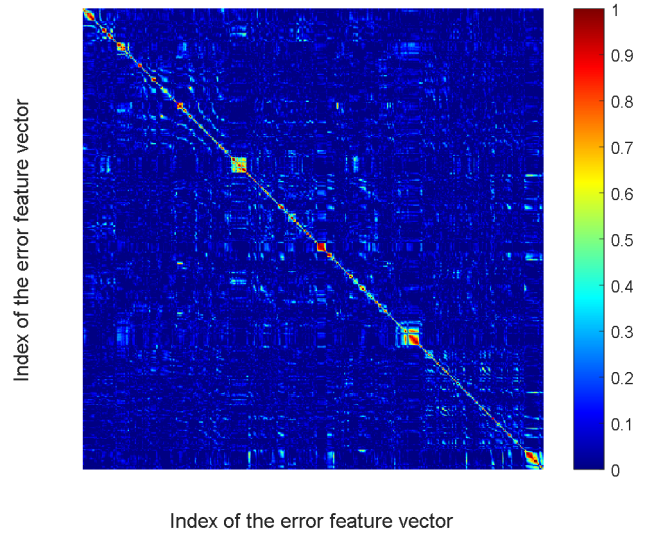
III. CONCLUSION

The current work presents a proof-of-principle study of constructing data-driven automated systems for MI detection and quantification with capabilities to learn on-the-job. With respect to methods listed in TABLE III. , the proposed method compares favourably with state-of-art models of similar complexity. Compared to other significantly more complicated deep learning models, like DSTGAN, the developed model has a much reduced computation cost.

However, the limitation of this work is that the backbone CNN model was built on the data containing traces of prior contouring information despite our best efforts to manually remove these. Hence the research should be viewed as a proof of principle for the AI learning on-the-job functionality rather

than a report on developing a precise AI-based MI detection and quantification algorithm. After the paper was accepted, we have acquired a clean data set, evaluated the network's performance on that clean set and observed a performance drop in the task of MI detection to 73%. This confirms that the technology can easily pick contouring artefacts and that any future research/work must pay close attention to data cleaning and preparation. At the same time, even in this extreme case, the network was able to generalize and learn, even from corrupted data. Our future work will be focused on extensive analysis of how different network architectures perform on both clean and corrupted data sets.
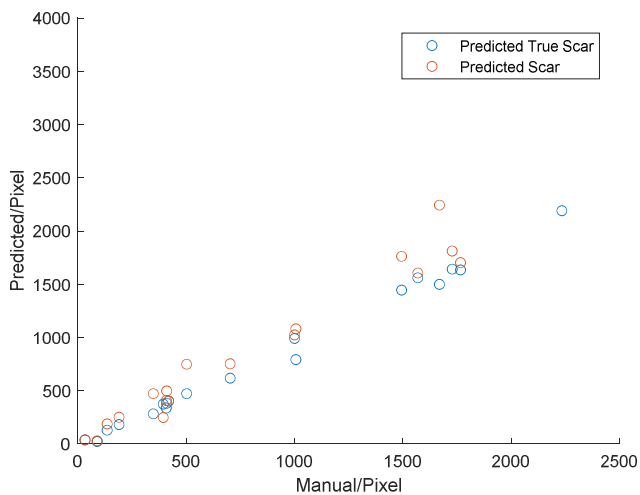

Fig. 11. Comparison to the reference ground truth (blue circles show pixel counts for only correctly detected scar tissue; red circles show pixel counts for all the scar tissue detected including false positives)

TABLE III. PERFORMANCE ANALYSIS IN TERMS OF ACCURACY, SPECIFICITY AND SENSITIVITY

	Accuracy	Specificity	Sensitivity
NAF[13]	84.39%		
Stack autoencoder (SAE) [13]	83.5%		
LSTM-RNN [12]	94.35%	98.42%	91.23%
DSTGAN [11]	96.98%	98.70%	92.15%
CNN with Data balance and *manual removal of contours*	95.3%	98.70%	95.33%
CNN with Data balance, *manual removal of contours*, and with error correction	98.2%	99.8%	98.2%

REFERENCES

[1] I. Biasato, E. Biasibetti, D. Biagini, G. Bruatto, G. Cenacchi, F. Guarda, *et al.*, "Spontaneously occurring intramural coronary arteriosclerosis in regularly slaughtered veal calves and beef cattle: a screening study about prevalence and histopathological features," *Journal of Veterinary Cardiology,* vol. 20, pp. 55-63, 2018/02/01/ 2018.

[2] X. Liu, J. Chen, B. Zhang, G. Liu, H. Zhao, and Q. Hu, "KDM3A inhibition modulates macrophage polarization to aggravate post-MI injuries and accelerates adverse ventricular remodeling via an IRF4 signaling pathway," *Cell Signal,* vol. 64, p. 109415, Dec 2019.

[3] Available: https://www.bhf.org.uk/what-we-do/our-research/heart-statistics

[4] S. Xie, B. Desjardins, M. Kubala, J. Liang, J. Yang, R. J. van der Geest, *et al.*, "Association of regional epicardial right ventricular electrogram voltage amplitude and late gadolinium enhancement distribution on cardiac magnetic resonance in patients with arrhythmogenic right

[5] I. Eitel, S. Desch, S. de Waha, G. Fuernau, M. Gutberlet, G. Schuler, *et al.*, "Long-term prognostic value of myocardial salvage assessed by cardiovascular magnetic resonance in acute reperfused myocardial infarction," *Heart,* vol. 97, pp. 2038-2045, Dec 2011.

[6] R. J. Kim, D. S. Fieno, T. B. Parrish, K. Harris, E. L. Chen, O. Simonetti, *et al.*, "Relationship of MRI delayed contrast enhancement to irreversible injury, infarct age, and contractile function," *Circulation,* vol. 100, pp. 1992-2002, Nov 9 1999.

[7] L. C. Amado, B. L. Gerber, S. N. Gupta, D. W. Rettmann, G. Szarf, R. Schock, *et al.*, "Accurate and objective infarct sizing by contrast-enhanced magnetic resonance imaging in a canine myocardial infarction model," *J Am Coll Cardiol,* vol. 44, pp. 2383-9, Dec 21 2004.

[8] A. S. Flett, J. Hasleton, C. Cook, D. Hausenloy, G. Quarta, C. Ariti, *et al.*, "Evaluation of techniques for the quantification of myocardial scar of differing etiology using cardiac magnetic resonance," *JACC Cardiovasc Imaging,* vol. 4, pp. 150-6, Feb 2011.

[9] L. Y. Hsu, A. Natanzon, P. Kellman, G. A. Hirsch, A. H. Aletras, and A. E. Arai, "Quantitative myocardial infarction on delayed enhancement MRI. Part I: Animal validation of an automated feature analysis and combined thresholding infarct sizing algorithm," *J Magn Reson Imaging,* vol. 23, pp. 298-308, Mar 2006.

[10] Q. Tong, C. Li, W. Si, X. Liao, Y. Tong, Z. Yuan, *et al.*, "RIANet: Recurrent interleaved attention network for cardiac MRI segmentation," *Computers in Biology and Medicine,* vol. 109, pp. 290-302, 2019/06/01/ 2019.

[11] C. Xu, J. Howey, P. Ohorodnyk, M. Roth, H. Zhang, and S. Li, "Segmentation and quantification of infarction without contrast agents via spatiotemporal generative adversarial learning," *Medical Image Analysis,* vol. 59, p. 101568, 2020/01/01/ 2020.

[12] L. X. Chen, chu Xu, Zhifan Gao, Shen Zhao, Heye Zhang, Yanping Zhang., Xiuquan Du, Shu Zhao, Dhanjoo Ghista, Shuo Li, "Direct Detection of Pixel-Level Myocardial Infarction Areas via a Deep-Learning Algorithm," presented at the Medical Image Computing and Computer Assisted Intervention (MICCAI), 2017.

[13] J. M. Héloïse Bleton, Hervé Lombaert, Hervé Delingette, Nicholas Ayache, "Myocardial Infarct Localization Using Neighbourhood Approximation Forests," presented at the Statistical Atlases and Computational Models of the Heart (STACOM 2015).

[14] A. S. Fahmy, J. Rausch, U. Neisius, R. H. Chan, M. S. Maron, E. Appelbaum, *et al.*, "Automated Cardiac MR Scar Quantification in Hypertrophic Cardiomyopathy Using Deep Convolutional Neural Networks," *JACC: Cardiovascular Imaging,* vol. 11, pp. 1917-1918, 2018.

[15] A. N. Gorban and I. Y. Tyukin, "Blessing of dimensionality: mathematical foundations of the statistical physics of data," *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences,* vol. 376, p. 20170237, 2018.

[16] I. Y. Tyukin, A. N. Gorban, S. Green, and D. Prokhorov, "Fast construction of correcting ensembles for legacy artificial intelligence systems: Algorithms and a case study," *Information Sciences,* vol. 485, pp. 230-247, 2019.

[17] I. Y. Tyukin, A. N. Gorban, K. I. Sofeykov, and I. Romanenko, "Knowledge Transfer Between Artificial Intelligence Systems," *Frontiers in Neurorobotics,* vol. 12, 2018-August-13 2018.

[18] A. N. Gorban, R. Burton, I. Romanenko, and I. Y. Tyukin, "One-trial correction of legacy AI systems and stochastic separation theorems," *Information Sciences,* vol. 484, pp. 237-254, 2019/05/01/ 2019.

[19] K. Fukushima, "Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol Cybern,* vol. 36, pp. 193-202, 1980.

[20] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Neural Information Processing Systems,* vol. 25, 01/01 2012.

[21] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, *et al.*, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv preprint arXiv:1803.01164,* 2018.

[22] C. Liu, L. Xie, W. Kong, X. Lu, D. Zhang, M. Wu, *et al.*, "Prediction of suspicious thyroid nodule using artificial neural network based on radiofrequency ultrasound and conventional ultrasound: A preliminary study," *Ultrasonics,* vol. 99, p. 105951, 2019/11/01/ 2019.

[23] S. E. Fahlman, C. Lebiere. "The cascade-correlation learning architecture." In Advances in neural information processing systems, pp. 524-532. 1990.

[24] T. J. Draelos, N. E. Miner, C. C. Lamb, J. A. Cox, C. M. Vineyard, K.

D. Carlson, W. M. Severa, C. D. James, J. B. Aimone. "Neurogenesis deep learning: Extending deep networks to accommodate new classes." In 2017 International Joint Conference on Neural Networks (IJCNN), pp. 526-533. IEEE, 2017.

[25] S. Scardapane, D. Wang. "Randomness in neural networks: an overview." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 7, p.e1200, 2017.

[26] A. R. Barron. "Universal approximation bounds for superpositions of a sigmoidal function." *IEEE Transactions on Information theory*, vol. 39, pp.930-945.

[27] D. Wang, M. Li. "Stochastic configuration networks: Fundamentals and algorithms." *IEEE transactions on cybernetics*, vol. 47, pp.3466-3479, 2017.