Automatic Defect Detection for X-Ray inspection: Identifying defects with deep convolutional network

Roger BOOTO TOKIME[1,2], Xavier MALDAGUE[1], Luc PERRON[2]
[1]ECE, Université Laval, Québec, QC, Canada
[2]Lynx Inspection Inc., Québec, QC, Canada
E-mails: rogerbooto@gmail.com,
xavier.maldague@gel.ulaval.ca,
lperron@lynxinspection.com

Abstract

In this day and age, parts manufacturers are expected to meet the stringiest quality standards despite the increasing diversity and complexity of the parts they are making. Traditional inspection techniques are often inadequate or too expensive for mass production, so companies rely on a statistical approach to meet their quality objectives. To help reduce the costs associated with parts inspection and make 100% inspection possible, we propose a new method, based on artificial intelligence, that can perform automated defect recognition (ADR) in a high volume part production environment. In this paper, we will discuss the application of a state of the art deep convolutional network method that has been proven effective in the automated detection and identification of defects through semantic segmentation of radiographic imagery. The performance of a deep convolutional network, which is based on an encoder-decoder architecture called SegNet, are presented using real-life porosity samples in welded parts.

Keywords— Deep learning NDT, Automatic weld inspection, Automatic defect detection (ADR), X-Ray inspection, Non-destructive evaluation (NDE), X-Ray convolutional network, X-Ray Artificial Intelligence

1. Introduction

Since their discovery in 1895 by Wilhelm Conrad Röntgen[1, 2, 3], x-rays have been used extensively for evaluating and validating the quality of industrially manufactured products [4, 5, 6] in the context of non-destructive testing (NDT). Nowadays, several techniques and methods make it possible to guarantee a certain level of quality of the parts just manufactured and among these methods, there is the non-destructive evaluation method. Unlike destructive evaluation, non-destructive evaluation is the art of being able to inspect a manufactured part without physically destroying it. Several technologies make it possible to achieve such a feat [7, 8, 9], including technologies based on the exploitation of x-rays, the technology on which this article is based. Traditionally, x-ray (radiographic) assessment was done manually by an NDT technician through the illumination of a radiographic film. This way of doing things has many advantages, but its biggest drawback is that it is not scalable. Indeed, with the increasing number of parts machined each year, it would take a huge number of technicians to inspect the parts properly. As a result, manufacturers have fallen back on the digitization of radiographic films, also known as computed radiography (CR) and have developed processes for creating digital radiographic images (DR). Statistical methods have been developed to analyze a large quantity of parts by inferring an acceptance quality level value [10, 11] through the analysis of a representative sample set taken from multiple batches of manufactured parts. Doing so allows the inspection to be expandable, but does not guarantee a 100% inspection ratio. We have developed a new method based on artificial intelligence that is able to perform automated defect detection using radiographic images in the context of a high volume production environment while being sufficiently economical to allow 100% inspection. In the field of deep learning by computer vision, this is the equivalent of performing what is called semantic segmentation [12], in other words, assigning a class number to one or several pixels that corresponds to the type of content, in this case, defect (1) or no defect (0). This paper discusses the application of a deep convolutional network based on an encoder-decoder architecture called SegNet. The paper is broken down as follows: 1) A review of the literature on encoder-decoder architecture, 2) The decomposition of the database, 3) A description of the network used, 4) The results obtained and 5) A summary of the results obtained.

2. Literature review

In the field of digital imaging, several problems are solved by drawing on a set of techniques from several related disciplines. With such a foundation, it is then possible to design systems to solve complex problems. In this chapter, we will explore in detail how we can design systems based on techniques from related disciplines, such as Convolution and Signal Processing, Neural Network, Convolutional Neural Network and Encoder-Decoder.

2.1. Convolution and signal processing

In mathematics, convolution is an operation composed of two functions, $\boldsymbol{f}$ and $\boldsymbol{g}$ on the same infinite domain, consisting in finding the integral (or the sum if the domain is discrete) around a point in the domain. In the field of signal processing, functions $\boldsymbol{f}$ and $\boldsymbol{g}$ can be one-dimensional time signals or two-dimensional spatial signals but we will focus on two-dimensional signal processing for this article. In image processing, functions $\boldsymbol{f}$ and $\boldsymbol{g}$ correspond to the input image ($\boldsymbol{f}$) and the convolution kernel (ω) is also called a convolution filter. As we can observe in Equation 1, we can extract an interesting property of convolution, which is the fact that the application of a particular filter can extract the same information in all input images. Later in this article we will see how this is used.

$$
\begin{aligned}
&g(x,y) = \omega \circledast f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} \omega(s,t) f(x-s, y-t) \\
&\circledast : \text{convolution operator} \\
&g(x,y) : \text{Filtred image} \\
&f(x,y) : \text{Original image} \\
&\omega : \text{Filter kernel} \\
&\text{-a} \leq \text{s} \leq \text{a and -b} \leq \text{t} \leq \text{b} : \text{Every element of the filter}
\end{aligned}
\tag{1}
$$

2.2. Neural network

In biology, animals with brains transmit and receive information to different parts of their bodies through a system of neurons. The information that circulates in the system is in the form of an electrical signal and the cell responsible for the transmission is the neuron. A neuron is composed of five main parts: dendrite, the cell body, the nucleus, the axon and the terminal axons. When a signal is emitted in the system, the neuron receiving the signal via the dendrites will be activated if the electrical disturbance is high enough [13] and then transmit a signal via the terminal axons. This property then allows us to model the neuron as a function receiving a signal from one or multiple sources (input signal) and the image of this function corresponds to the activation value of the neuron (output signal). This model is the one used by designers of artificial neural networks as shown in Figure 1. However, it is important to note that this model is a simplification of what is actually happening in biology, therefore, this model cannot be used as a substitute for studying the behaviour of biological neurons.

2.3. Convolutional neural network

In sections 2.1 and 2.2, we introduced two important properties in signal processing: convolution and artificial neurons. Based on these properties, we can then create a particular type of artificial neural network that is being called a convolutional neural network. Convolutional networks are neural networks with filters designed to extract specific type of information from the signal and use it as input signal for the activation function. This process allows a machine learning system designer to narrow down the information necessary to learn from the available general features available in order to solve a particular problem. Depending on the network design being used, the machine learning system can even learn by itself, in which case we are talking about an automatic learning system. As we have just seen, an automatic learning system based on a convolutional neural network architecture is a system capable of generating a filter bank to extract the information necessary to solve a given problem. Such learning can be done supervised, unsupervised[14] or by reinforcement[15]. In this article, we will describe a supervised learning system. Note that the convolutional network has the interesting ability to spatially reduce the signal once filtered. This allows the representation of spatial information in a smaller space, which essentially encodes the information. This principle is the basis of a network architecture called encoder-decoder that we will detail in the next section.

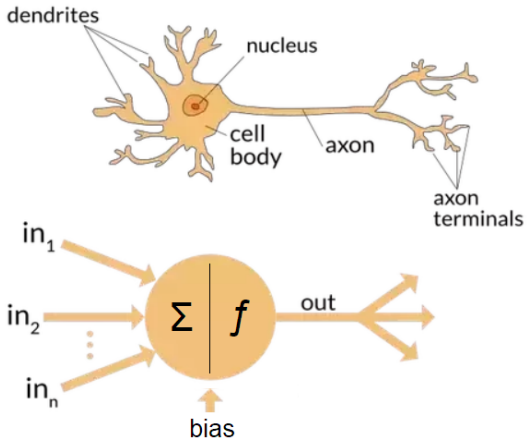


Figure 1: Artificial and biological neuron analogy

2.4. Encoder-decoder

An encoder is a network that takes an input signal and outputs a map of characteristics. These characteristic vectors contain the information, the features, that represent the input. The decoder is a network similar to the encoder, but unlike the encoder, it takes the characteristic vector of the encoder and gives the best possible correspondence with the intended actual input or output. Encoders are trained with decoders. A loss function is based on the calculation of the differences between the actual and reconstructed input. An optimizer will try to train both the encoder and the decoder to reduce this loss of reconstruction. Once trained, the encoder will provide a vector of characteristics for the input that can be used by the decoder to build the input with the characteristics that matter most to make the reconstructed input recognizable as the real input[16]. In our article we wish to perform a semantic segmentation of the porosity defects present in welding samples.

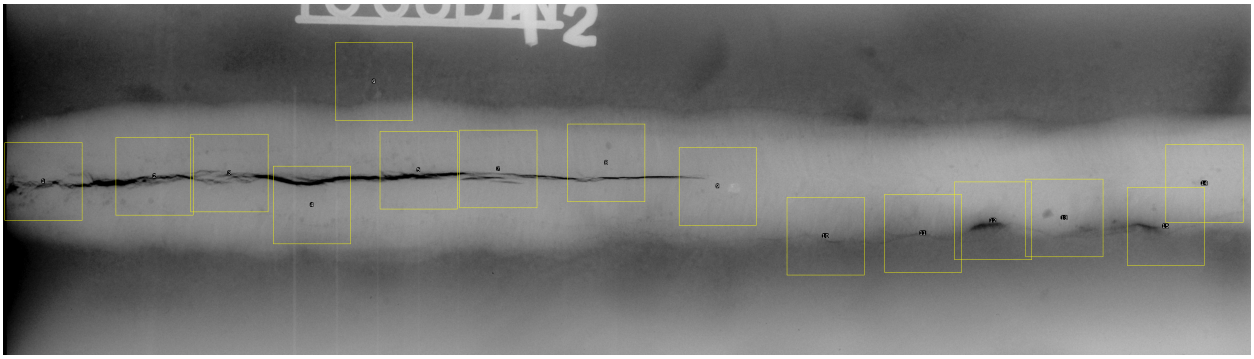
3. Database

For this experiment, we used a public database called GDXray [17]. This database contains several samples of radiographic images including images of welding with porosity defects. The database already contains segmented image samples, which is a good basis for training a small network. On the other hand, the quantity of images available is insufficient for a generalization of the problem. In order to generalize the problem, we decided to cut the large images that were about 1200 px in height and 4000 px in width into several smaller 256x256 images. This had the effect of increasing the amount of data available to the network. It also facilitated training on a graphics card, because a small fixed image size that can easily be stored in memory. In addition to generating small images from the original GDXray images, we have artificially generated several new images by performing basic transformations such as a 342° rotation at 18° intervals and generating noisy images by adding Gaussian noise to each generated image. We also performed x, y and diagonal translations and created negative type images of each image. Finally, we ended up with a database composed of more than 20 thousand 256x256 grayscale and binary images from the original 720 distinct images available in GDXray based on the following calculations:

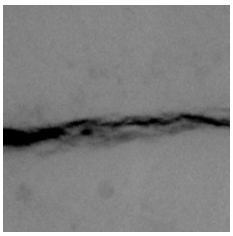
$$720 * (r + n + i + tx + ty + txy) = 23760$$

$$720 * (19 + 1 + 1 + 3 + 3 + 3) = 23760$$

Where **r** is the number of rotations performed, **n** is the number of noisy images generated, **i** is the number of inverse images generated and **tx**, **ty**, **txy** are the number of translations performed on the x, y axes and the diagonal. In our experiments, we separated our data so that we used 90% of our images as training data, and 10% as test data that our network will never have seen. In addition, we performed a cross validation of our training data by separating 75% for training and 25% for validation. However, we did measure the reliability of our predictions using a metric called F1. This metric is used to verify the reliability and generalizability of our classification model, as it takes into consideration precision and recall metrics. Precision is a measure to calculate the ability of our system to predict pixels belonging to both classes in the right regions and recall allows us to calculate the sensitivity of our system when predicting true positives. Thus, the score given by the F1 metric gives a good idea of the prediction and classification capabilities of our model. During our experiments, we managed to obtain an F1 score of **80%** for our SegNet model and this is a very promising result.



(a) Select a zone from the original image



(b) Cropped 256x256 input image

(c) Associated image ground truth

Figure 2: Example of how we created our database

4. Network description

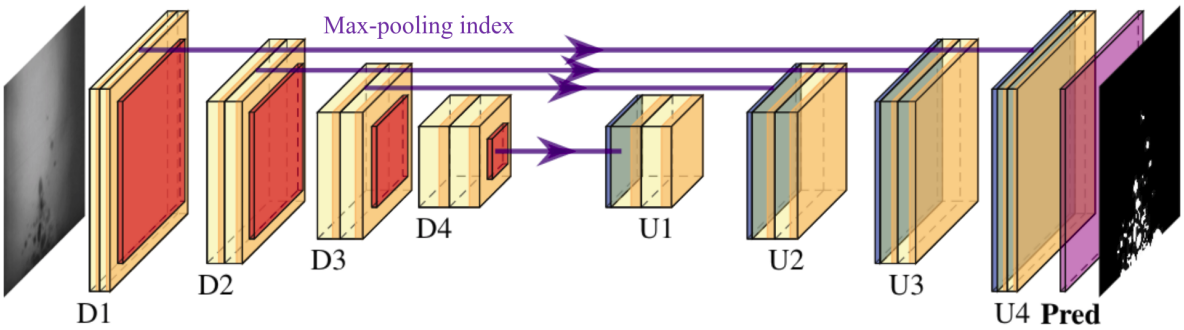


Figure 3: Our SegNet architecture diagram

4.1. Encoder

In Figure 3, we can see the two sections of our encoder-decoder architecture, the encoder being on the left while the decoder is on the right. The encoder consists of 4 convolution blocks (D1 - D4) and 4 pooling layers. Convolution blocks perform the following operations: convolution, batch normalization and application of an activation function. We will now detail the usefulness of each of the operations and, if applicable, explain our choices. Let's start with convolution. As mentioned before, convolution is a mathematical operation consisting of summing the products as shown in equation 1. To convolve an image to a filter, we drag the filter onto the image and depending on the type of dragging, also called stride, the filtered image may or may not be the same size as the input image. Let us now move on to the normalization of batches. Batch normalization speeds up learning when we have features with different value scales, for example, from 0 - 1 and 0 - 1000. This not only reduces the scale of values, but also preserves the importance of each characteristic[18]. As can be seen in Figure 4, the convolution blocks are composed of 6 layers with layers 3 and 6 being activation layers whose activation functions are exponential linear unit (ELU) and scaled exponential linear unit (SeLU)[19, 20] respectively.

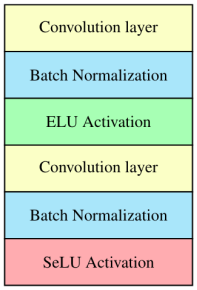


Figure 4: Encoder blocks: Convolution block operations for D1 to D4

The choice of these activation functions is based on the following properties of each function; 1) They keep the simplicity and speed of calculation of the rectified linear unit (ReLU) activation function, which is the reference activation function in most state of the art deep learning models, when the values are greater than zero. 2) They treat values near or below zero in two different ways; as indicated in their name, exponentially and exponentially scaled. As a result, the network is in continuous learning mode, because unlike ReLU, the ELU and SeLU functions are unlikely to disable entire layers of the network by propagating zero values in the network. This phenomenon is known as the dying ReLU[21, 22]. The last layer of each encoder block is a pooling layer that consists of generating a feature map at each resolution level. As shown in Figure 5, this operation, in our case, reduces the size of the image by a factor of two each time it is applied. This operation allows to keep the pixels representing the elements that best represent the image. To do this, we keep the largest pixel value in a kernel of any size and the position of this pixel which allows us to have a spatial representation of the pixels of interest. As a result, the network learns to encode not only the essential information of the image, but also its position in space. This approach makes it easier to reconstruct the information performed by the decoder, the architecture of which we will detail in the next paragraph.

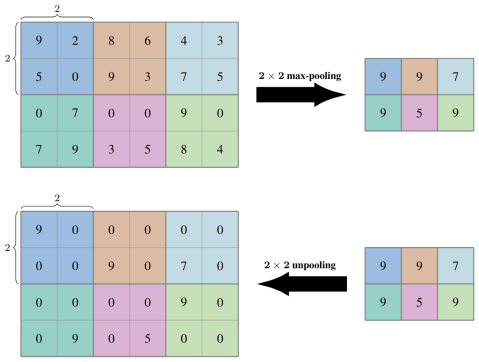


Figure 5: Pooling operation example

4.2. Decoder

The decoder consists of 4 convolution blocks (U1 - U4) and 4 unpooling layers. Convolution blocks perform the same operations as D1 - D4, but are organised a bit differently as described in Figure 6. The blocks U1 and U2 have one more block of convolution, batch normalization and activation, because we unintendedly pasted an extra block to the decoder section during one of our experiments and obtained great results. The blocks U3 and U4 follows the same convention in terms of operations as Dx except that the last layer of U4 is the prediction layer, which means that the activation function will not be ELU or SeLU, but the sigmoid function, in our case. At the end, we compare our prediction with the ground truth image by using the Dice loss function [23].

5. Results

In Figure 7, we can observe the prediction results of our model on a radiographic image. Since our training images have a size of 256x256, in order to cover the entire surface of the test image, we generated a mask in which we manually delimited the area where the weld is located. Then we used a technique called ”sliding window” that allows us to make pixel-by-pixel predictions in

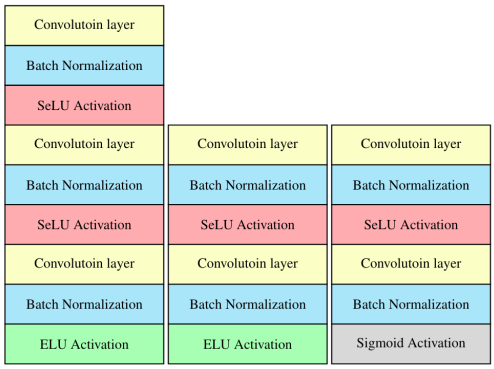


Figure 6: Decoder blocks: Convolution block operations for U1 and U2 (left), U3 (middle) and U4 (right)

the selected area. The results of this manipulation can be seen in Figure 7.n. As explained in sections 2.4 and 4 of this article, an encoder-decoder network architecture can reconstruct an input signal into an output signal by minimizing the differences between predictions and the ground truth. In our case, the output values will be between 0 (no defect) and 1 (defect) so we can interpret these values as a probability value that a given pixel represents an area containing a defect or not. In Figure 7, image **a** represents the manually selected area in which we will predict the location of defects. The images **b** to **e** represent a close-up view of the yellow outlined areas in the original image **a**. The images **f** to **i** represent the predictions made by our network. The representation chosen to show our results is a heat map in which the dark blue represents the pixels where the network does not predict any defect and in red the pixels where the network predicts a strong representation of a defect. The images **j** to **m** represent the ground truth images associated with the framed areas in the original image **a**.

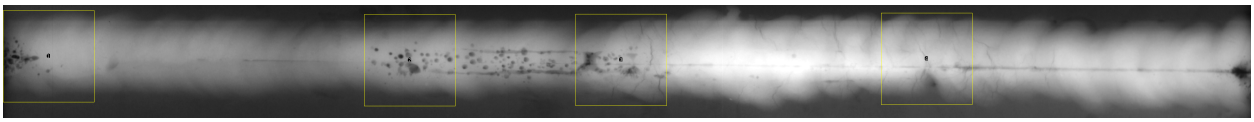
Our experimental results show that our deep machine learning approach to performing semantic segmentation can be applied to detect porosity defects in radiographic images representing a welded area. However, we can observe that a section to the right of our test images was not predicted. This is due to the fact that our images were not a multiple of 256 in width. Therefore, this section is not considered when applying the sliding window. The problem can easily be solved by ensuring that the acquired images are multiples of the size of the cropped images, but this is not the most practical solution in a production environment.

We measured the reliability of our predictions using a metric called F1. This metric is used to verify the reliability and generalizability of our classification model, as it takes precision into consideration and recall metrics as expressed in equation 2. Precision is a measure to calculate the ability of our system to predict pixels belonging to both classes in the right regions and recall allows us to calculate the sensitivity of our system when predicting true positives. Thus, the score given by the F1 metric gives a good idea of the prediction and classification capabilities of our model. During our experiments, we managed to obtain an F1 score of **80%** for our SegNet model which is a very promising result.

$$
\begin{aligned}
Precision &= \frac{truePositive}{truePositive + falsePositive} \\
Recall &= \frac{truePositive}{truePositive + falseNegative} \\
F1 &= 2 * \frac{precision * recall}{precision + recall}
\end{aligned}
\tag{2}
$$

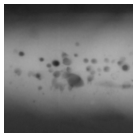
6. Conclusion

In summary, we have demonstrated that deep learning through semantic segmentation can detect porosity defects in a weld. To do this, we implemented a deep convolutional network based on an encoder-decoder architecture called SegNet. The network has been adapted to our needs as described in Figure 3 and the results can be seen in Figure 7. In Figure 7, we can clearly see in a visual way that our approach is able to detect porosity defects in welds and in section 5, we have explained how we have been able to evaluate the quality of our results using statistical measurements that illustrate that our model is able to predict and classify pixels of a radiographic image representing a weld with an **80%** efficiency. Based on these promising results, our future work will be based

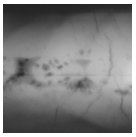


(a)

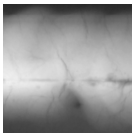
(b)



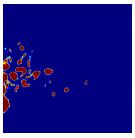
(c)



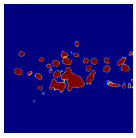
(d)



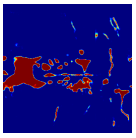
(e)



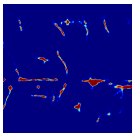
(f)



(g)



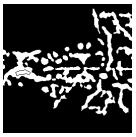
(h)



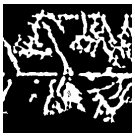
(i)

(j)

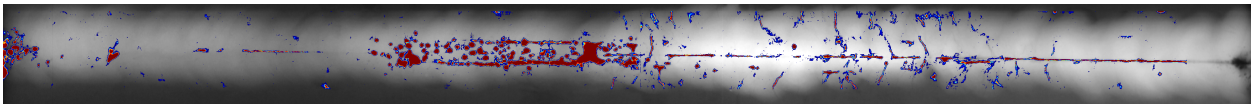
(k)



(l)



(m)



(n) Predictions Results

(o) Ground truth

Figure 7: Prediction results after applying sliding window

on the application of a network model that classifies the different types of porosities, in other words, we want to be able to label the different types of defects that we can predict. In addition, we will explore the application of smaller network models, which could accelerate training time.

References

[1] A. B. Reed, "The history of radiation use in medicine," *Journal of Vascular Surgery*, vol. 53, no. 1, pp. 3S–5S, 2011.

[2] R. Halmshaw, "The discovery of x-rays and the early history of industrial radiography," *INSIGHT*, vol. 37, no. 9, pp. 669–671, 1995.

[3] R. I. Frankel, "Centennial of röntgen's discovery of x-rays," *West J Med*, vol. 164, pp. 497–501, Jun 1996. PMC1303625[pmcid].

[4] N. Brierley, C. Bellon, and B. Lazaro Toralles, "Optimized multi-shot imaging inspection design," *Proc Math Phys Eng Sci*, vol. 474, pp. 20170319–20170319, Aug 2018. 30220862[pmid].

[5] A. Fischer, T. Lasser, M. Schrapp, J. Stephan, and P. B. Noël, "Object specific trajectory optimization for industrial x-ray computed tomography," *Scientific Reports*, vol. 6, pp. 19135 EP –, Jan 2016. Article.

[6] X. Xiao, A. Ferro, T. Ma, C. Y. Han, X. Zhou, and W. Wee, "Adaptive reference image set selection in automated x-ray inspection," *Journal of Electrical and Computer Engineering*, vol. 2014, p. 7, 2014.

[7] P. Zhu, Y. Cheng, P. Banerjee, A. Tamburrino, and Y. Deng, "A novel machine learning model for eddy current testing with uncertainty," *NDT & E International*, vol. 101, pp. 104 – 112, 2019.

[8] C. I.-C. X. P. V. M. N. P. Avdelidis, T.-H. Gan, "Infrared thermography as a nondestructive tool for materials characterisation and assessment," 2011.

[9] J. García-Martín, J. Gomez-Gil, and E. Vázquez-Sánchez, "Non-destructive techniques based on eddy current testing," *Sensors (Basel, Switzerland)*, vol. 11, pp. 2525–65, 12 2011.

[10] E. Corporation, "Quality control procedures qcp 15.1."

[11] J. A. Melchore, "Sound practices for consistent human visual inspection," *AAPS PharmSciTech*, vol. 12, pp. 215–221, Jan 2011. 21203872[pmid].

[12] R. B. Tokime, H. Elassady, and M. A. Akhloufi, "Identifying the cells' nuclei using deep learning," *2018 IEEE Life Sciences Conference (LSC)*, pp. 61–64, 2018.

[13] Z. S. e. a. Lodish H, Berk A, *Molecular Cell Biology. 4th edition. New York: W. H. Freeman*. 2000. https://www.ncbi.nlm.nih.gov/books/NBK21535/.

[14] A. Jung, "A gentle introduction to supervised machine learning," *CoRR*, vol. abs/1805.05052, 2018.

[15] Y. Li, "Deep reinforcement learning," *CoRR*, vol. abs/1810.06339, 2018.

[16] K. A. Siddiqui, 2019. https://www.quora.com/What-is-an-Encoder-Decoder-in-Deep-Learning.

[17] V. Z. U. M. G.-L. I. Z. I. L. H. C. M. "Mery, D.; Riffo, "Gdxray: The database of x-ray images for nondestructive testing.," 2015. 34.4:1-12.

[18] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.

[19] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," *CoRR*, vol. abs/1706.02515, 2017.

[20] D. Pedamonti, "Comparison of non-linear activation functions for deep neural networks on MNIST classification task," *CoRR*, vol. abs/1804.02763, 2018.

[21] L. Lu, Y. Shin, Y. Su, and G. Karniadakis, "Dying relu and initialization: Theory and numerical examples," 03 2019.

[22] A. F. Agarap, "Deep learning using rectified linear units (relu)," *CoRR*, vol. abs/1803.08375, 2018.

[23] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations," *CoRR*, vol. abs/1707.03237, 2017.