Vision Transformers for Medical Images Classifications

Rebekah Leamons, Hong Cheng, Ahmad Al Shami

Department of Computer Science
Southern Arkansas University - Magnolia, AR, USA
hcheng@saumag.edu
aalshami@saumag.edu

Abstract: Image classification method based on Vision Transformers (VT) are gaining popularity as the standard models in natural language processing (NLP). The VT do not depend on the convolution blocks, rather captures the relative relations between image pixels regardless of their three-dimensional distance. In this paper, three different Deep Learning models were developed to detect the presence of Invasive Ductal Carcinoma, the most common form of breast cancer. These models include convolutional neural network (CNN) was used as a baseline, a residual neural network (RNN) and a Vision Transformer (VT). To test these models, we used a dataset of breast cancer tissue images. These images were used to train and validate our three models, which led to different levels of high accuracy. Experimental results demonstrate that the VT model outperforms the CNN and RNN on different tasks by up to 93% accuracy classification rate, while the other models highest rate was 87%.

Keywords: Deep Learning, Classification, Predictive Analysis, Computer Vision, Convolutional Neural Network, Residual Neural Network, Vision Transformer, Breast Cancer

1. Introduction

Breast cancer is the most common form of cancer, and Invasive Ductal Carcinoma (IDC) is the most common form of breast cancer [4]. More than 180,000 women in the United States are diagnosed with invasive breast cancer each year, and about 8 out of 10 of these women are diagnosed with IDC [5]. Automating the diagnostic process could lead to both faster and more accurate diagnoses.

The primary goal of our research is determining which deep learning model is most effective in detecting IDC. This could help make the diagnostic process much more efficient, both for IDC diagnosis and, eventually, for many other medical conditions. These models were trained separately on the same dataset to determine their success rates. The objectives of this study are:

- To create Residual Neural Network and Vision Transformer models to classify an image as IDC negative or IDC positive,
- To test the Residual Neural Network and Vision Transformer's accuracy against a Convolutional Neural Network, and
- To determine which model is most efficient at IDC detection.

2. Background

2.1 Convolutional Neural Network (CNN)

Traditionally, a Convolutional Neural Network would be used for image classification. From here, a Convolutional Neural Network will be abbreviated to CNN. The primary purpose of Convolution in case of a CNN is to extract features from the input image. Convolution preserves the spatial relationship between

pixels by learning image features using small squares of input data. A CNN contains two phases: feature learning and classification. In feature learning, an input is first given to the program. In this case, input includes an image. Then, convolution layers are used to create a feature map. Convolution is an image processing technique that uses a weighted kernel (square matrix) to revolve over the image, multiply and add the kernel elements with image pixels. Fig. 1, depicts the full CNN process as follows.

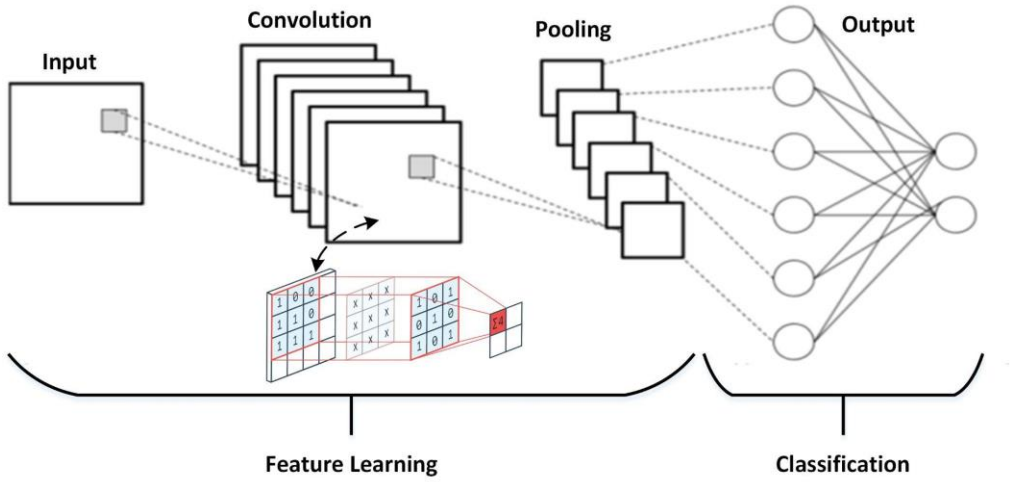

Fig. 1: *This diagram shows the phases of a CNN, from taking input, running it through a convolution layer (2D Convolution with 1 filter (channel), 3x3 h & w and 0 paddings), pooling, and classifying the data for output*. [8]

The convolution of $\boldsymbol{f}$ and $\boldsymbol{g}$ for example written as $\boldsymbol{f * g}$, it is defined as the integral of the product of the two functions after one is reversed and shifted.

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau \qquad (1)$$

The convolution layers apply filters and padding to the input to create a map of any important features on an image. The padding increases the size of the convoluted image. The relationship between padding and the output size of the convolutional layer is given by:

$$O = \left(\frac{n_x + 2p - n_h}{S}\right) + 1 \qquad (2)$$

where O is the output size, S is the stride parameter which indicates the jump sizes the convolution kernel needs to slide over the image, default value $S=1$, meaning that the kernel moves one pixel at a time. However, $S=2$ is also common. P is the padding, how much zero-padding should be placed around the input image. n_x is the length of the input signal and n_h is the length of the filter [9].

Consider a CNN trying to detect human faces; it would likely map out eyes, lips, and noses on its feature map. After the convolution layer, a pooling layer will be used. The pooling layer is a dimensionality reduction technique used to reduce the size of the layer, which reduces the necessary computational power. Finally, the data is classified into a predetermined number of categories.

2.2 Residual Neural Networks (RNN)

Residual Neural Networks are another common model used for image classification. From here, a Residual Neural Network will be referred to as an RNN. In a traditional neural network, the input runs straight through weight layers and activation functions, shown in the below diagram as f(X) [6]. An RNN is very similar to a traditional neural network in terms of its layer structure. However, in an RNN, there are connections between layers known as "skip" connections. These skips connect the layers in a round, which allows the program to return to a previous weight layer or activation function and run it again [7]. This optimizes the program without having to add additional layers, which also improves computational efficiency. The below diagram (see Fig. 2), shows a traditional Neural Network (left) and an RNN (right).

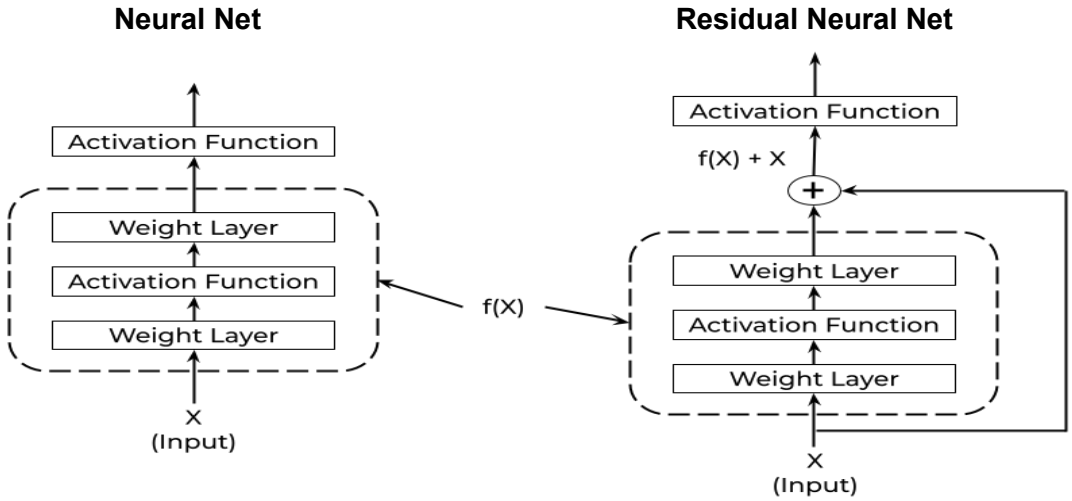

Fig. 2. *This diagram compares a Residual Neural Network to a standard Neural Network. In this diagram, we can see how a neural network only works through the weight layers and activation function, f(x), once, while a residual neural network can "skip" back and run through f(x) multiple times.*

2.3 Vision Transformers (VT)

Transformers have only recently been used in image classification, but they are quickly proving their efficiency. A transformer works by first splitting an input image into patches and giving these patches a "position," which allows the program to keep track of the original image composition. The patches are then run through special encoders to try and relate one patch of an image to another. Transformers were traditionally used in language processing, so it may be easier to think of relating these patches as relating words in a sentence. As humans, we can comprehend which words in a sentence relate to another; such as in the sentence, "Sarah is smart, Ben is creative." We know that smart relate to Sarah and creative relates to Ben. Similarly, the transformer is trying to isolate patches of an image and relate them to one another to find "context," which is in this case IDC positive or IDC negative. After the patches have been through the encoder, the image is run through a neural network and classified. (Fig. 3), below show this process.

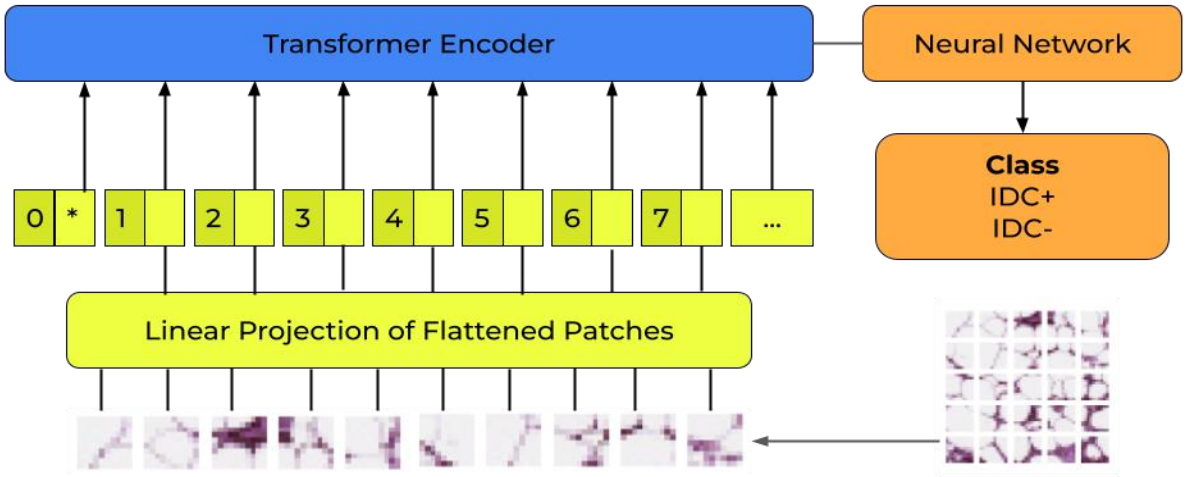

Fig. 3. *This diagram shows how a Vision Transformer works, using an example from the dataset. As seen in the diagram, the transformer takes an image, splits it into patches, creates a linear projection of these patches, assigns a location to the patches, runs them through an encoder, then sends the data through a neural network, before finally classifying the image.*

3. Methodology

This project used a dataset provided by Paul Mooney at Kaggle.com. The code provided was also used as a building block to create our own CNN model. Cornell University also provided the papers *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale* [1] and *Attention Is All You Need* [3], both of which were used as research for our Transformer.

We began by downloading the dataset provided by Paul Mooney. This dataset was then run through a preprocessing program that divided the dataset into positive and negative categories and prepared the data for testing. The data was then trained on the CNN, RNN, and Vision Transformer models. These models run as described in the introduction.

Histograms like the ones presented below (see Fig. 4 and Fig. 5). were also used to plot these images and show their pixel intensity. These histogram models were useful image filters because, to the untrained eye, the primary difference between IDC (-) and IDC (+) images is their coloration.

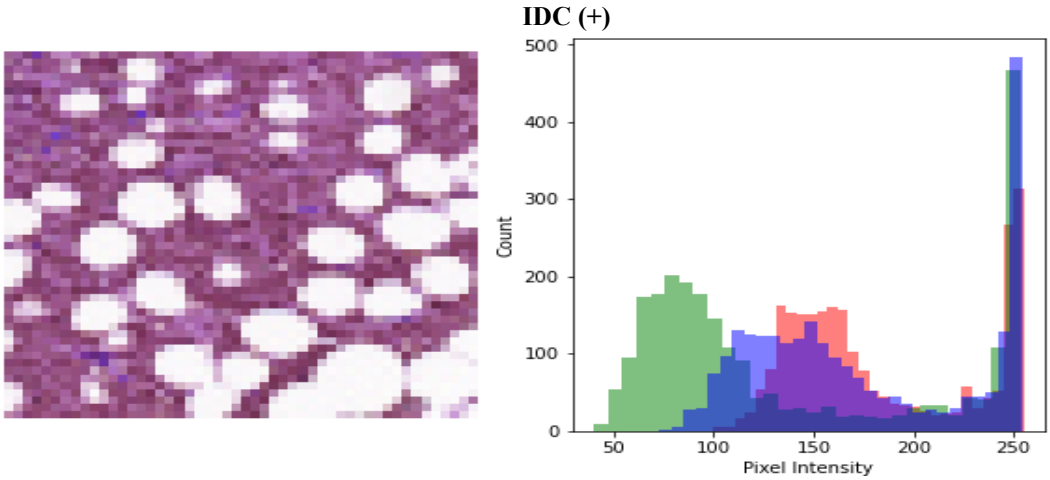

Fig. 4: *The above image shows a breast tissue sample that tested positive for IDC (left) and a diagram displaying the pixel intensity across the image. This intensity can be a factor in classification, depending on the model.*

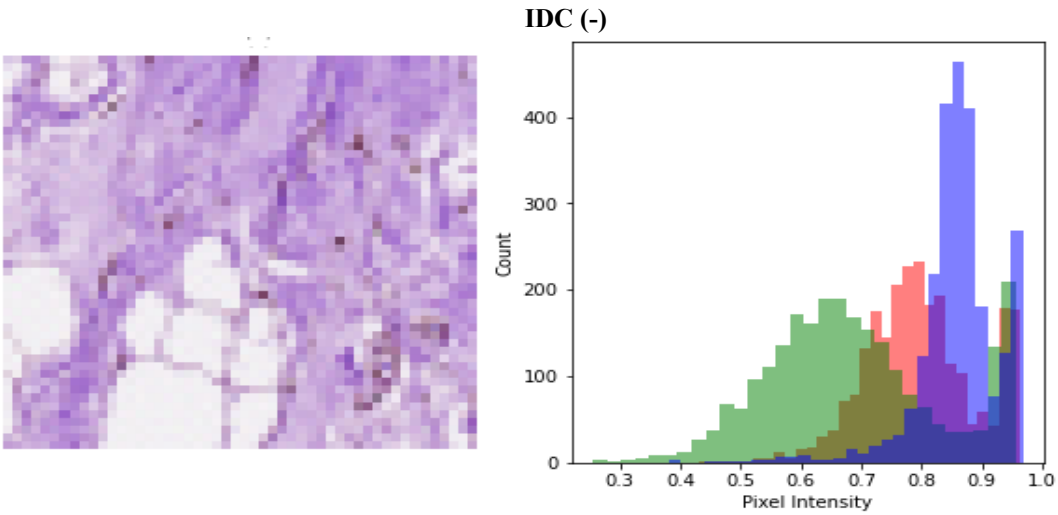

Fig. 5: *The above image shows a breast tissue sample that tested negative for IDC (left) and a diagram displaying the pixel intensity across the image. This intensity can be a factor in classification, depending on the model.*

4. Dataset Description and Preprocessing

This dataset contains approximately five thousand 50x50 pixel RGB images of H&E-stained breast histopathology samples [2]. These images are labeled either IDC or non-IDC (see Fig. 6). The dataset is contained in NumPy arrays, and these arrays are small patches taken from digital images of breast tissue samples. Only some of the cells are cancerous, though breast tissue contains many cells.

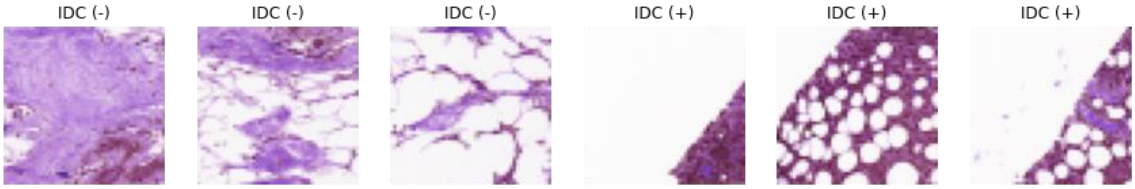

Fig. 6: *The above images are random examples of breast tissue histopathology images taken from the dataset. The three on the left represent images that were classified as IDC (-), while the three on the right represent images that were classified as IDC (+).*

To better process these images, each image was resized to either 50 by 50, in the case of the CNN and RNN, or 256 by 256 in the case of the transformer. Additionally, in the case of the transformer, each image was split into 16 patches as described in the introduction. Otherwise, the data was taken as-is into our suggested model.

7. Results

Each of the models was trained on the full dataset for 100 epochs. The CNN model predicted the presence of breast cancer images with 81.4% accuracy, the RNN model with 87.9% accuracy, and the Vision Transformer with 93% accuracy. The model accuracy of the CNN is shown below (see Fig 7 a). One can see that after each iteration, or epoch, the accuracy increased. However, the model's accuracy begins to plateau around the thirtieth epoch. The model accuracy of the RNN presented below (see Fig 7 b). Much

like CNN, the accuracy increases after each epoch. The RNN also increases in accuracy faster without plateauing as the CNN did, however the model signal no improvement above 87% accuracy rate. The model accuracy of the Vision Transformer (VT) is also shown below (see Fig 7 c). one can see how the accuracy increases after each epoch. The VT also increases in accuracy faster without plateauing as other models did. Overall, this implies that the Visio Transformer is a better model.

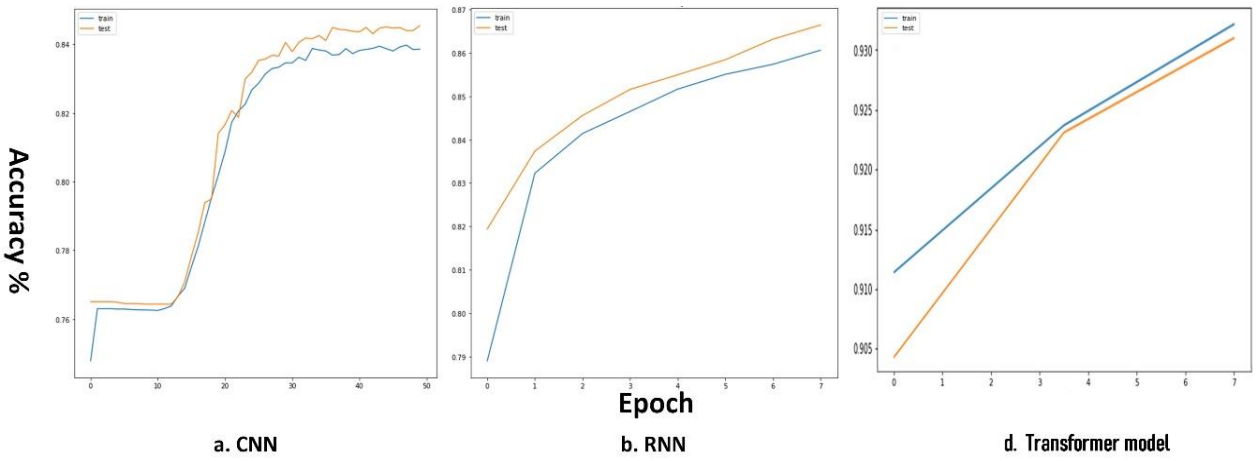

Fig. 7 a. *This image shows the model accuracy overtime for the CNN model. Along the X-axis, we see the epoch, or iteration, while the Y-axis shows accuracy. From this graph, we can see how the accuracy peaks to a max of 84% after 30 epochs, with slight increase after this.*

Fig. 7 b. *This image shows the model accuracy over time for the RNN model. Along the X-axis, we see the epoch, or iteration, while the Y-axis shows accuracy. From this graph, we can see how the accuracy slightly improved without plateauing.*

Fig. 7 c. *This image shows the model accuracy over time for the Transformer model. Along the X-axis, we see the epoch, or iteration, while the Y-axis shows accuracy. From this graph, one can see how the accuracy is gradually improve without plateauing. The accuracy peaked to a max of 93 % after few epochs.*

8. Conclusion

This study confirmed two things. The first is that computer vision can be used to diagnose breast cancer with accuracies up to or over 90%. This means that, with some refinement, these tools could soon be used as a diagnostic tool by doctors. The second is that, of the models tested, Vision Transformers are the most accurate at diagnosing cancer when given photos of histopathology samples with an accuracy of 93%.

In the future, we hope to research Topographic Data Analysis, also known as TDA. TDA can be used in computer vision by isolating the topology of an image and using the filtered data as a sort of feature map, similar to the feature maps created by a CNN. These feature maps can then be used by the computer to make assumptions about the data. In the future, we hope to study TDA as another resource in image classification.

Acknowledgment and Disclaimer

This material is based upon work supported by the National Science Foundation (NSF) under **Award No. OIA-1946391**. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. Dosovitskiy, Alexey, et al. *An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale*, Cornell University, 3 June 2021, https://arxiv.org/abs/2010.11929.
2. Mooney, Paul. "Predict IDC in Breast Cancer Histology Images." *Kaggle*, Kaggle, 6 Mar. 2018, https://www.kaggle.com/paultimothymooney/predict-idc-in-breast-cancer-histology-images.
3. Vaswani, Ashish, et al. *Attention Is All You Need*, Cornell University, 6 Dec. 2017, https://arxiv.org/abs/1706.03762.
4. "Invasive Breast Cancer (IDC/ILC)." *Cancer.org*, American Cancer Society, 19 Nov. 2021, https://www.cancer.org/cancer/breast-cancer/about/types-of-breast-cancer/invasive-breast-cancer.html.
5. "Invasive Ductal Carcinoma: Diagnosis, Treatment, and More." *Breastcancer.org*, Breatcancer.org, 13 Oct. 2021, https://www.breastcancer.org/symptoms/types/idc.
6. Bhattacharyya, Saptashwa. "Understand and Implement Resnet-50 with Tensorflow 2.0." *Medium*, Towards Data Science, 9 Sept. 2021, https://towardsdatascience.com/understand-and-implement-resnet-50-with-tensorflow-2-0-1190b9b52691.
7. "Module: Tf.keras.applications.RESNET: Tensorflow Core v2.7.0." *TensorFlow*, 12 Aug. 2021, https://www.tensorflow.org/api_docs/python/tf/keras/applications/resnet
8. "2D Convolution block." *Peltarion*, 2022, https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/2d-convolution. Last Accessed 10 February 2022
9. The Mathematical Engineering of Deep Learning Home page, https://deeplearningmath.org/convolutional-neural-networks.html#:~:text=A%20convolution%20is%20an%20operation%20on%20two%20vectors%2C,led%20to%20the%20development%20of%20convolutional%20neural%20networks. Last Accessed 10 February 2022