Detection of Tumors in Breast Cancer Patients with Simulated Motion Blurs

Chloe Liu **Genesis Qu** **Kristi Van Meter**
BME548 Spring 2023
Duke University

Abstract

Contemporary literature has focused on using various convolutional neural network (CNN) model architectures for machine learning classification problems in medical imaging. In this study, we created a modified version of Ren et al.'s [1] Faster R-CNN architecture to classify breast cancer tumors that have been simulated to contain motion-blurred artifacts and understand which blur types are most disruptive to detection accuracy. Before running the pre-trained Faster R-CNN model, a custom kernel layer simulating blurring of the images in the eight cardinal directions was implemented, as well as a custom probability layer implementing softmax or Gumbel-Softmax probability distributions. Results from training on this model showed that the blurs simulating north and west movement were the least disruptive to detection accuracy and movement in diagonal directions (northeast, southeast, northwest, southwest) was the most disruptive to detection accuracy in models using both probability distributions. Performance of the model was better using Gumbel-Softmax distributions, with Intersection over Union (IoU) scores of 0.71 and 0.39 at the 0.4 confidence level compared to an IoU of 0.53 at the 0.2 confidence level using softmax distributions.

1 Background

Artificial intelligence mechanisms have gained momentum as aids to overburden physicians, proving to be helpful in triaging patients and as a supportive diagnostic tool. AI tools have been sought particularly in medical specialties that require close scrutiny of images, like anatomical pathology and radiology [2]. A well-established hurdle in the use of machine learning classification for diagnosis is blurring caused by patient movement during imaging. The process of MRI scanning is slow and sequential, and patient movement during the time raw MRI signals are first encoded in k-space or while these signals are encoded as human-readable scans using an inverse Fourier transform disrupts encoding signals and causes a motion artifact. These artifacts reduce the ability to detect clinically relevant abnormalities [3].

Given the abundance of interest in using deep learning applications to resolve motion blur artifacts in MRI imaging, there have been promising advancements made in object detection by estimating the unknown motion kernel. Convolutional neural networks (CNNs) and derivations of CNNs have been the focus of many studies, including the estimation of motion kernels with the use of local patches [4], the introduction of DeblurNet by Noroozi et. al, a novel CNN used to deblur images in challenging circumstances including occlusions, motion parallax, and camera rotations [5], and a fully-convolutional deep neural network (FCN) that uses motion flow to unblur images [6].

With a goal of object detection and localization, we elected to focus on a CNN model that has had success in this area [7], the region-based convolutional neural network (R-CNN). In contrast to other CNN methods, ours uses a series of fixed motion kernels to consider the extent of each motion's effects on object detection performance. The model we focused on was the latest advancement in R-CNNs, Faster R-CNN, which uses deep neural networks to achieve real-time results [1].

The pre-trained Faster R-CNN architecture from Ren et. al was utilized in this study. The Faster R-CNN detection system is comprised of two modules. The first module is a deep, fully-convolutional network that uses attention to propose regions of interest. This region proposal network (RPN) takes in an input image and outputs a set of rectangular object proposals each scored for their "objectness", or rating of relationship to an object class versus the background [1]. The second module of the model is the Fast R-CNN detector by Girshick [8], which uses regions proposed by the RPN from module one to detect objects. The Faster R-CNN model used for this study was pre-trained on the Microsoft COCO dataset [1].

Both softmax probabilities and Gumbel-Softmax probabilities were used in the creation of the custom probability layer in our model. The Gumbel-Softmax distribution was used because of its ability to approximate the sampling process of discrete data when working with a stochastic neural network with discrete variables. Given the success of Jang et. al while using Gumbel-Softmax probabilities in their work on semi-supervised classification using a dataset with categorical latent variables [9], we elected to utilize Gumbel-Softmax in addition to traditional softmax distributions in the creation of our predictive layer.

2 Methods

2.1 Data

The data used in this study was gathered for the Saha et. al 2018 study on using machine learning for radiogenomic analysis of breast cancer patients [10]. This dataset contained images from 922 patients with invasive breast cancer who received preoperative MRI imaging. Of these 922 patients, 306 were randomly selected to comprise the dataset for our analysis due to computational limitations. The images are pre-operative dynamic contrast-enhanced (DCE)-MRI images. Axial images in the prone position were taken with 1.5T and 3T scanners. The original dataset contained the DICOM formatted sequence of a non-fat saturated T1-weighted sequence, a fat-saturated gradient echo T1-weighted pre-contrast sequence, and mostly three to four post-contrast sequences for each patient [10]. For our study, we selected the middle image of three post-contrast MRI images for each patient. By selecting the middle slice of the image, we hoped to retain the most prominent view of the lesion which switching to a two-dimensional dataset of pictures using X and Y coordinates from a three-dimensional dataset using X, Y, and Z coordinates for ease of computation.
In addition to the imaging data, the dataset contained the location of lesions in DCE-MRI that are labeled by radiologists. These annotations contained the row, column, and slice information needed to draw the bounding boxes which signified the presence of a lesion in the image. Images were also standardized and augmented before the modeling stage. The images were resized to uniform dimensions, and the dimensions of the bounding boxes were proportionally resized to accommodate these changes. These images were then augmented using the augmentation library.

2.2 Modeling

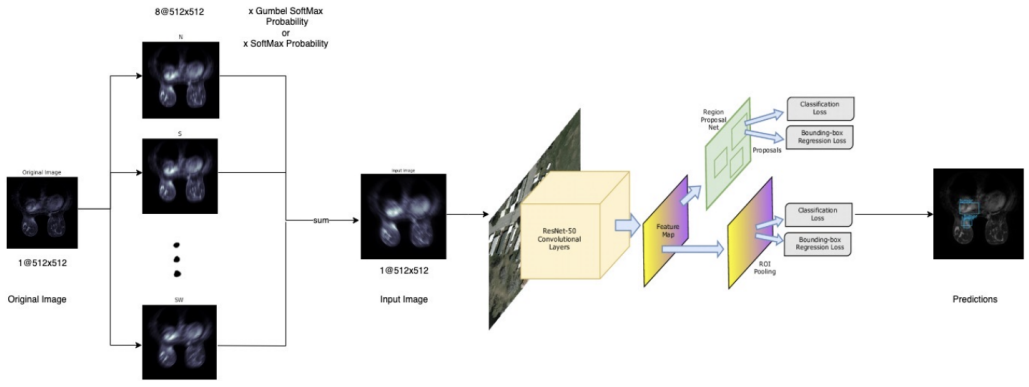

Figure 1: Diagram of custom layer and Faster R-CNN structure. Faster R-CNN structure sourced from Mohan et. al [11].

The first structure of our model was a custom layer to simulate motion blurs using kernels and a probability layer that computed a weighted aggregate of the eight blurred images. These were implemented before the convolution layers of the first module of the Faster R-CNN model. These convolutional kernels had fixed, 30x30 dimensions to simulate the eight cardinal directions (north, east, south, west, northeast, southeast, northwest, and southwest). The kernel arrays were matrices containing probabilities in the range of zero to one extending in the cardinal direction of the filter from the center of the kernel to simulate movement in a single direction (rather than a horizontal or vertical blur, which would consist of an entire row/ column of the matrix).

The probability layer of the model was the next component, with a function that computes the weighted aggregate of the eight blurred images from the convolution kernel. The probability layer would, therefore, have a kernel dimensionality of 8x1x1 comprised of the sums of the weighted images. Both softmax and Gumbel-Softmax probability distributions were used to determine these weights, with Gumbel-Softmax simulating a hard selection. For the Softmax probability model, initialized logits were used to create a discrete probability distribution. The Gumbel-Softmax density distribution used was the same as that used in Jang et. al [9], with the distribution representing an interpolation between discrete one-hot-encoded categorical distributions and continuous categorical densities. To determine the sums of the weighted images, the custom layer picked one of the eight motion blurs depending on the softmax or Gumbel-Softmax distribution.
At this point, the pre-trained model with weights from Ren et. al [1] was loaded to begin training using the training and validation dataset. The structure of the model from this point forward mirrors the Faster R-CNN, with convolutional layers followed by regions of interest proposed by the RPN. Finally, in region of interest (RoI) pooling, the regression loss of the proposal box coordinates was scored and the classification of the proposals was determined.

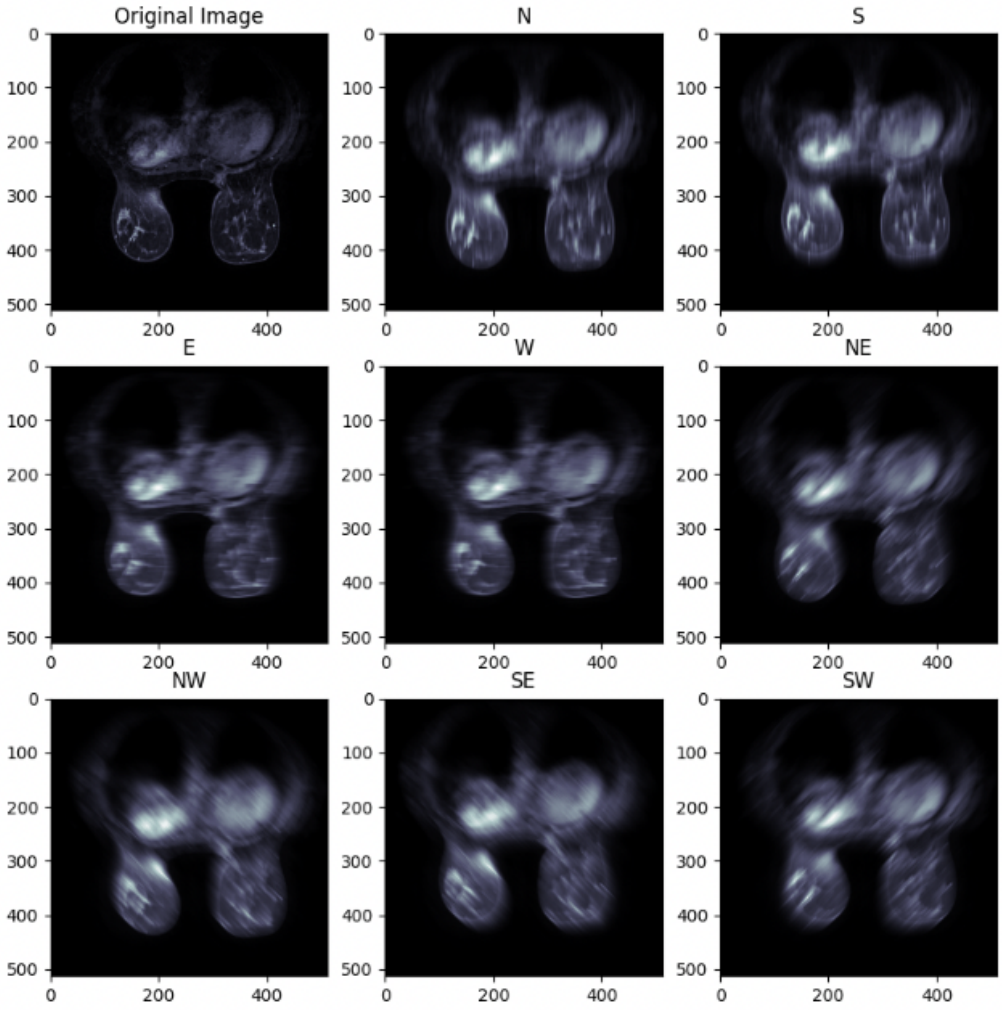

Figure 2: Example of image with blur kernel applied. From left to right beginning in the top row are the original image followed by images blurred in the north, south, east, west, northeast, northwest, southeast, and southwest direction.

Table 1: Performance results using softmax probabilities

IoU Threshold	Precision	Recall	F1
0.2	0.73	0.13	0.21
0.3	0.64	0.11	0.19
0.5	0.36	0.07	0.11
0.7	0	0	0

3 Results

Implementation of the softmax probability distribution in our custom layer indicated that blurs simulated in the north and west distributions had the highest probabilities over 60 epochs, while the diagonal cardinal directions (northeast, southeast, southwest, northwest) had the lowest probabilities, with higher probabilities indicating less disruption of detection accuracy.

Measuring loss over the training and validation period showed similar loss numbers at around 0.23. Because these numbers are similar, we are not concerned with overfitting.

Our model's performance using softmax probabilities for our prediction metric yielded an Intersection over Union (IoU) of 0.53. Our model received very few true positives even with an IoU threshold of 0.2. Our maximum IoU was 60.55%, and our mean IoU was 6%.

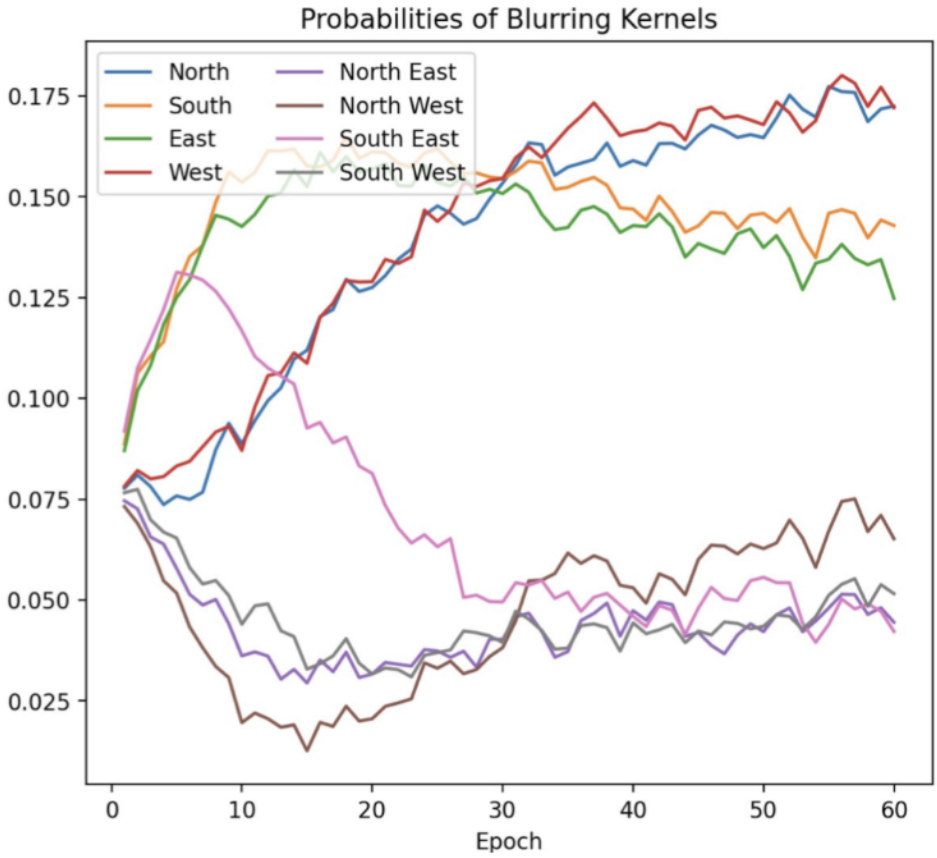

Figure 3: Using softmax probabilities, blurs in the north and west directions have the highest probabilities and the diagonal probabilities (northeast, southeast, northwest, southwest) had the lowest probabilities over 60 epochs.

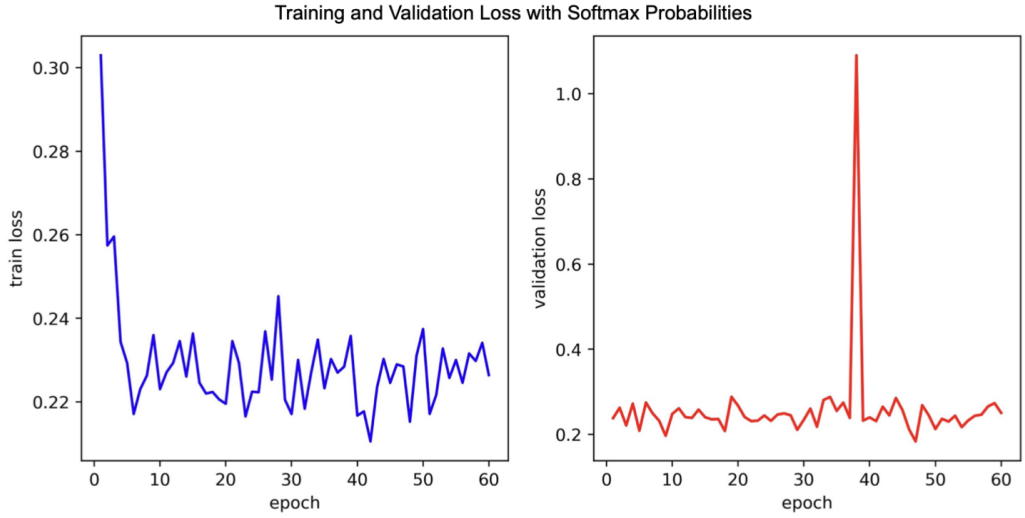

Figure 4: Using softmax probabilities, both training (left) and validation (right) loss showed similar loss numbers around 0.23.

Implementation of the Gumbel-Softmax probability distribution in our custom layer also indicated that blurs simulated in the north and west distributions had the highest probabilities over 60 epochs, while the diagonal cardinal directions (northeast, southeast, southwest, northwest) remained the lowest probabilities.

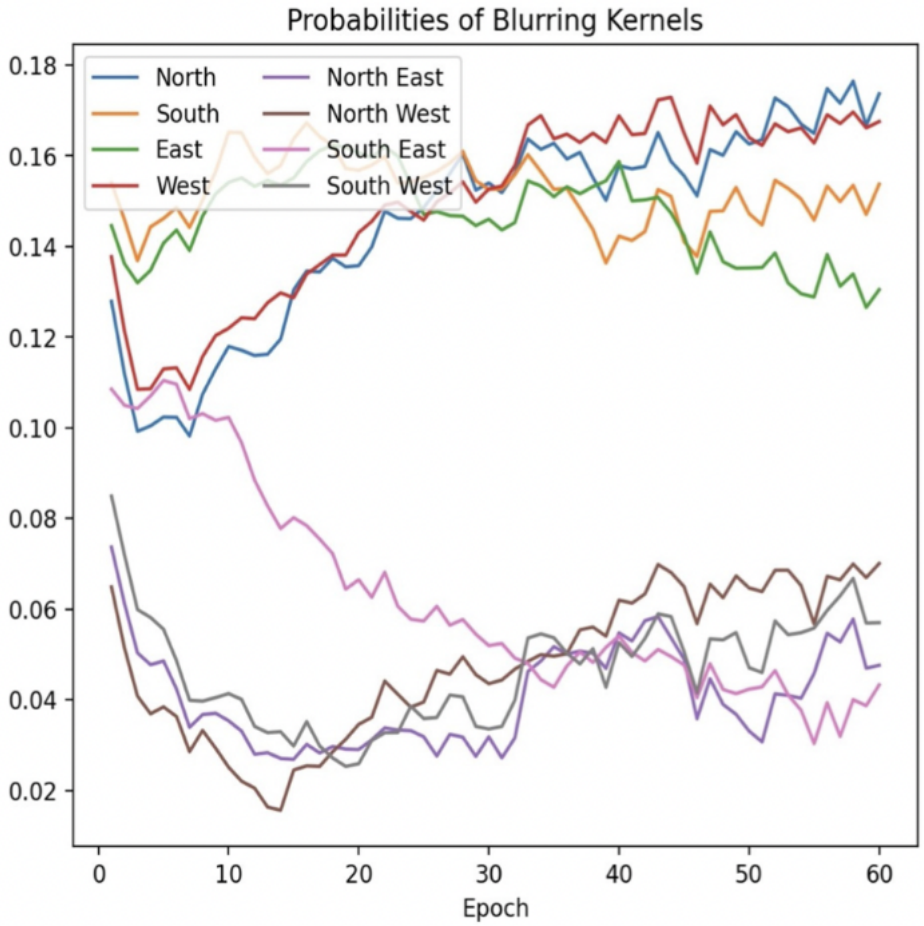

Figure 5: Using Gumbel-Softmax probabilities, blurs in the north and west directions have the highest probabilities and the diagonal probabilities (northeast, southeast, northwest, southwest) had the lowest probabilities over 60 epochs.

Table 2: Performance results using Gumbel-Softmax probabilities

IoU Threshold	Precision	Recall	F1
0.2	0.42	0.94	0.58
0.3	0.33	0.93	0.49
0.5	0.10	0.80	0.18
0.7	0.01	0.35	0.03

Measuring loss over the training and validation period showed training loss as slightly lower than validation loss, averaging around 0.23 and 0.25 respectively. Because validation loss appears to increase with the number of epochs, there is a possibility of the model overfitting based on this dataset.

The performance of our model improved using Gumbel-Softmax probabilities, with an IoU of 0.71 and 0.39 with a threshold of 0.4. The maximum IoU was 79.67%, and the mean IoU was 18.67%, giving Gumbel-Softmax a higher mean IoU than softmax probability by 12.67%.

The Gumbel-Softmax probability model performed better than the softmax model. The softmax tumor prediction was based on a confidence score of 0.2, yet even with a higher confidence score of 0.4, the Gumbel-Softmax version of the model yielded two results. With these results, we understand that the Gumbel-Softmax version is more confident in its results. Additionally, the two Gumbel-Softmax prediction boxes had different area sizes, giving us different IoU scores, with the smaller box having a satisfactory IoU of 0.71. Since the filter confidence score of detection boxes was set to 0.4, the Gumbel-Softmax model had more predictions with a higher confidence score, giving a lower precision but higher recall than the softmax probability model.

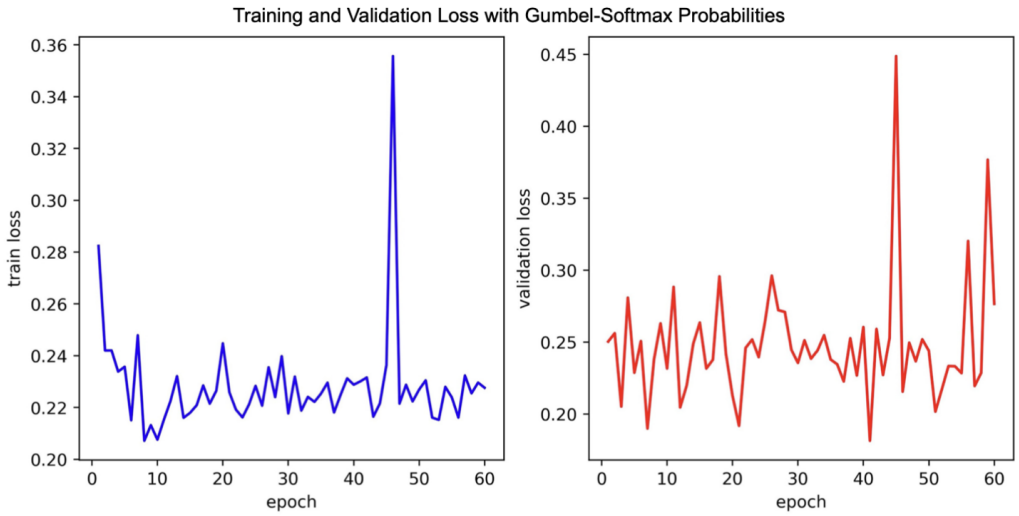

Figure 6: Using Gumbel-Softmax probabilities, training (left) loss averaged about 0.23 and validation (right) loss averaged about 0.25.

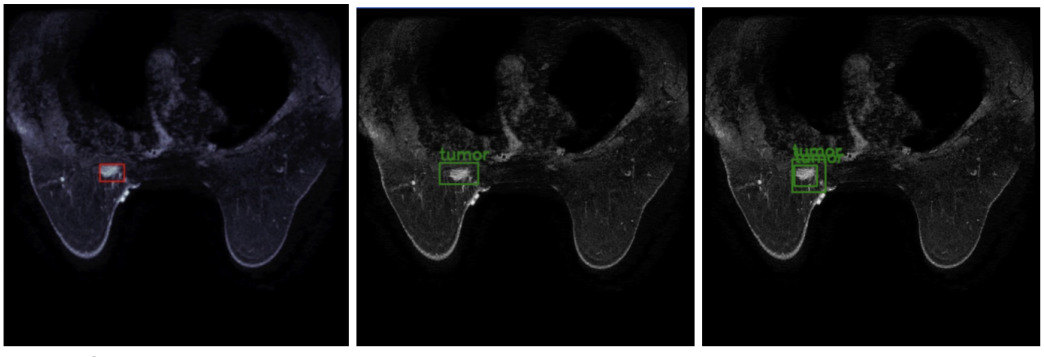

Figure 7: From left to right, the images depict the ground truth, softmax prediction (confidence score = 0.2, and Gumbel-Softmax prediction (confidence score = 0.4).

4 Conclusion

Results from this study indicate that blurring in the north and west directions were the least detrimental to the detection scheme of our model. In contrast, motion blurs in the diagonal directions were the most disruptive to the model. These findings were confirmed by both our models trained using softmax distributions and Gumbel-Softmax prediction distributions. The overall performance of our models was low, with the better performing of the two models, the Gumbel-Softmax-based model, yielding a mean IoU of 18.67%. Based on the relative success of the Gumbel-Softmax probability distribution compared to the softmax distribution, we recommend further research using similar distributions that approximate the sampling process of discrete data when working with a stochastic neural network with discrete variables.

Another potential area of research would be using actual motion-blurred images, with the addition of a deconvolution layer to optimize a physical layer that can deblur the motion artifact and improve model performance, particularly with diagonal blurring. This would also go toward helping resolve the problem of motion artifacts in MRI images in practice.

It's important to note that we trained our model on a subset of available images, around a third specifically, because of computational constraints. With more images and resources, the neural network may improve its performance and reduce overfitting. Additionally, the reality of motion artifacts will not correspond strictly to the motion kernels used in this simulation. We used a kernel with a gradient from one to zero in the direction of the blur, while the motion artifacts captured by MRI images in real life can be more complex and dynamic. A greater variety of simulated motion kernels could be used in future studies.

References

[1] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

[2] Char, D. S., Shah, N. H., & Magnus, D. (2018). Implementing Machine Learning in Health Care - Addressing Ethical Challenges. *The New England journal of medicine*, 378(11), 981–983.

[3] Li, S., & Zhao, Y. (2022). Addressing Motion Blurs in Brain MRI Scans Using Conditional Adversarial Networks and Simulated Curvilinear Motions. *Journal of imaging*, 8(4), 84.

[4] Sun, J., Cao, W., Xu, Z., & Ponce, J. (2015). Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 769-777).

[5] Noroozi, M., Chandramouli, P., & Favaro, P. (2017). Motion deblurring in the wild. In *Pattern Recognition: 39th German Conference, GCPR 2017, Basel, Switzerland, September 12–15, 2017, Proceedings 39* (pp. 65-77). Springer International Publishing.

[6] Gong, D., Yang, J., Liu, L., Zhang, Y., Reid, I., Shen, C., ... & Shi, Q. (2017). From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2319-2328).

[7] Uijlings, J. R., Van De Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, 104, 154-171.

[8] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448).

[9] Jang, E., Gu, S., & Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

[10] Saha, A., Harowicz, M. R., Grimm, L. J., Kim, C. E., Ghate, S. V., Walsh, R., & Mazurowski, M. A. (2018). A machine learning approach to radiogenomics of breast cancer: a study of 922 subjects and 529 DCE-MRI features. *British journal of cancer, 119*(4), 508–516.

[11] Mohan, V. S., Vinayakumar, R., Sowmya, V., & Soman, K. P. (2019). Deep rectified system for high-speed tracking in images. *Journal of Intelligent & Fuzzy Systems, 36*(3), 1957-1965.