Determining the size of shoulder lesions in sows using computer vision

S. Bery[1,*], T. M. Brown-Brandl[1], G. A. Rohrer[2] and S. R. Sharma[1]
[1]*Department Biological Systems Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588, USA*
[2]*Genetics and Breeding Research Unit, USDA-ARS U.S. Meat Animal Research Center, Clay Center, NE 68933, USA*
[*]Corresponding author: Shubham Bery, sbery2@huskers.unl.edu

Abstract

Shoulder lesion in sows is a common welfare concern with prevalence ranging from 4.6 to 50% occurrence. They represent not only a welfare concern but also have an economic impact due to labor needed for treatment and medication, in addition to the premature culling of affected sows. The objective of this study is to evaluate the use of computer vision techniques in detecting and determining the size of shoulder lesions. A time-of-flight (Microsoft Kinect V2) camera was used to capture the top-down depth and RGB images of sows in farrowing crates. This camera has an RGB image resolution of 1920 ×1080. To ensure the best view of the lesions, images were selected with sows lying on their right and left sides with all legs extended. A total of 320 RGB images from 28 sows with lesions at various stages of development were identified and annotated. Two deep learning models (YOLOv5 and FRCNN) pre-trained with the COCO dataset were implemented. The model was tested on 90 images (70 images with lesions, and 20 images without lesions) outside the training set. YOLOv5 was able to detect lesions better than FRCNN with an mAP@0.5 of 0.921. Next, image processing techniques like binarization were used to estimate lesion sizes in measurable units. Lesion size was determined using hand-selected ROI with ImageJ software. The estimated size matched the hand-determined size with an R^2 = 0.91.

Keywords: shoulder lesions, ulcers, sows, deep learning, YOLO

Introduction

Shoulder lesions, amongst lactating sows are commonly seen in the swine industry. Lesions are commonly developed during the first two weeks of farrowing (Herskin et al., 2011). These are formed due to the deficiency of oxygen to the underlying shoulder tissue caused by pressure incited from the flooring. The tissues lose blood supply and die in a similar way as human pressure ulcers (Nola, G.T.; Vistnes 1980). Anatomy of a sow's shoulder consists of a large bump, known as the prominent tuber, when a sow lies on its side, extra pressure is exerted on the tuber tissue making it vulnerable to ulcer formation. Severity of lesions can vary from mild lesions to bone deep ulcers if left untreated. They are associated with poor animal welfare because of the pain and increased infection risk that can lead to euthanizing the animal (Rioja-Lang, Seddon, and Brown 2018). This is not limited only to welfare problems but also adds to the production costs. Bone deep ulcers often negatively affect the sow's carcass value, failing the final quality check. Therefore, it becomes highly important to treat these ulcers at an early stage to reduce the welfare and economic concerns. Early lesion detection will lead to timely veterinary treatments like applying zinc oxide to the affected area. Most research concerning shoulder lesions has been focused on possible causes, side effects and ways to prevent or treat the ulcerations, but not much research has been done on lesions' monitoring. Automatic monitoring technology of shoulder lesions will help reduce the workload of farm workers and lead to timely treatment. In recent years, smart animal husbandry solutions are being developed that can keep track of individual animal's well-being in a continuous and an automated way using various sensors and cameras, while increasing farm productivity and detecting early onset health issues.

Machine vision techniques like deep learning-based models have become increasingly effective in solving a variety of problems. For example, convolutional neural networks (CNNs) like Resnet101, Xception and

MobileNet have already been successfully implemented to classify different postures in swine (Shao, Pu, and Mu 2021). YOLO (You-Only-Look-Once) has been used for the early detection of estrus behavior in cattle by modifying the spatial pyramid pooling (SPP) module in the architecture (Wang et al., 2022). Although there has not been much work done in lesion detection in animals, studies have been done using deep learning models to detect and localize ulcers from the images of diabetic human feet (Yap et al., 2021). However, images this study used for model training had a significantly larger ulcer-to-background area ratio.

With shoulder lesions in sows, it is likely the lesion portion of the image will only be a few pixels as cameras were mounted at a height to avoid inference with day-to-day activities. In addition, for training any deep learning model from scratch, one needs to have large amounts of supervised data, for example, the pictures of sows manually annotated bounded boxes around the lesion region to make the model understand the difference between lesion and all other objects in an image. To make these algorithms better at generalizing, focus should not only be on the dataset size but also on the data quality as well so that the model can extract different patterns and features from the data during the training phase.

Many applications with livestock species have the limitation of a small, annotated dataset. To tackle the limitations of a small sized dataset, transfer learning has been used to minimize this gap. It is a machine learning technique which reuses a model trained for one task applying it to another related task. It means to use the weights or features of a model that has been trained on a much bigger datasets like MS COCO, ImageNet and feeding them into the target network. Then, modifications are done in the initial and final layers to accommodate predictions on the custom dataset. The middle layers of any CNN learn general features of objects like specific edges, color blobs, etc. and are applicable across many datasets and tasks. The target model freezes its middle layers and uses the mid-layer features from the base model and retrains its initial and last layers to learn specific features related to the custom dataset. The objectives of this study were:

- To analyze performance of object detection models like YoloV5 and FRCNN in localization of shoulder lesions.
- The test the ability of automated image processing techniques to determine each lesion size.

Materials and methods

Data collection

Experiment was set up at the U.S. Meat Animal Research Centre (USMARC) in Clay Center, Nebraska, USA. All animal husbandry protocols were performed in compliance with federal and institutional regulations regarding proper animal care practices and were approved by the USMARC Institutional Animal Care and Use Committee (2015–21). The facility at USMARC is a farrow to finish swine production unit. The study utilized 28 Yorkshire-Landrace cross bred sows. This dataset is from a study designed to test behavioural and production characteristics of sows housed in 3 different crate sizes (Leonard et al., 2021). The images were captured in one of the two farrowing facilities. The farrowing facility was designed with 3 farrowing rooms, each room having twenty farrowing crates. For this test, the data were limited to twenty-eight farrowing crates across all 3 rooms. An aluminium triangle truss 21.6 m in length was placed above each row of crates. Bottom end of the truss was at an approximate height of 2.6 m above the ground. Time-of-flight depth sensor with an integrated digital camera (Kinect V2™, Microsoft, Redmond, WA, USA) was centered above each crate and mounted on the truss's bottom at a height of 2.55 m from the floor. Sensors were enclosed within a waterproof housing to protect the cameras during pressure washing and disinfecting. The setup can be seen in Figure 1. One mini-PC with Windows 10 Home Edition (Windows 10 Home, Microsoft, Redmond, WA, USA) was connected to a single camera. Total of 20 mini-PCs were connected to an external disk station (DS1517+, Synology Inc, Bellevue, WA, USA), having five 10TB each hard disk drives (ST10000VN0004, Seagate Technology LLC, Cupertino, CA, USA).

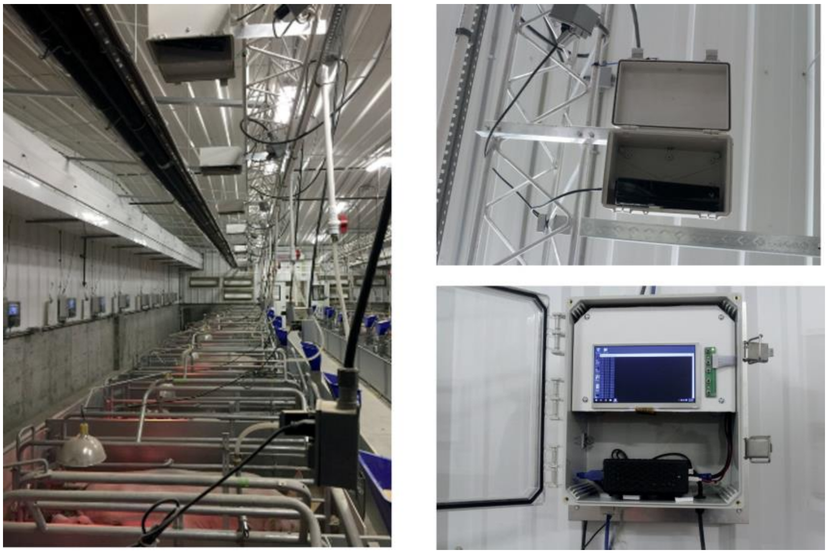

Figure 1: Shows the installed system with Kinect V2™ mounted on aluminum truss

An image capturing program was developed in MATLAB (R2017a, The MathWorks, Inc, Natick, MA, USA). The program captured and stored one digital image every 5 seconds. A total of 320 images having sores/shoulder lesions were identified. Also, fifty images of sows without lesions were added to the dataset to better handle the false positives making the final total 370 images in the dataset, Figure 2 shows a sample image. As the lesions occupied very few pixels in the image, cropping was done to remove the redundant information from the images.

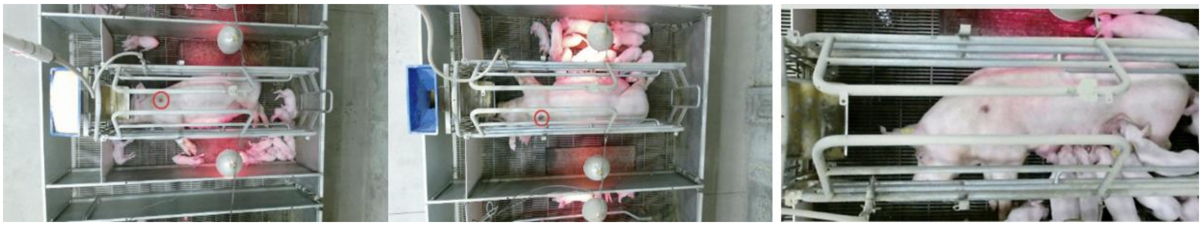

Figure 2: Sample RGB images collected of a sow and litter in a farrowing crate. Images were collected using Kinect V2. Red circle indicates the location of the shoulder lesion and the RGB image after cropping out the extra information

Data processing

Figure 3 illustrates the order of steps that were followed to implement the proposed system. First, data cleaning was completed to remove all the redundant information using a python script. Next, the lesions were annotated manually with bounding boxes using an online annotator tool, MakeSense.AI. The prepared dataset was then fed into a convolutional neural network-based object detection model which outputted coordinates for the lesion class. Detected regions containing lesions were cropped to determine the size of the lesion. Image binarization was done using Otsu's method to isolate the lesion pixels from the background. The size of the lesion was changed from pixels to cm^2 using the reference of the anti-crush bar. The bar is at a similar height as the sow's shoulder. The bar had a diameter of 25.4 mm which equated to 15 pixels. Therefore, the 1 pixel=1.7 mm or an area of 2.89 mm^2.

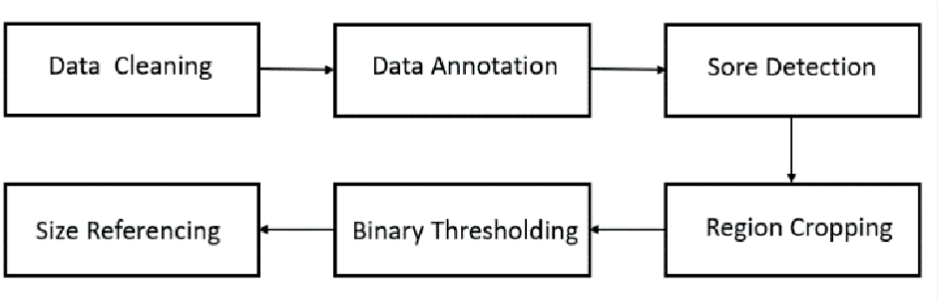

Figure 3: Flowchart

Model development

YOLOv5: The latest version of YOLO family of object detection models was used. These are also called single pass detectors because bounding boxes with their associated class probabilities are predicted in one evaluation. It uses the PyTorch framework which makes it easier to implement as compared to its previous DarkNet versions. The YOLOv5 architecture is formed of three blocks called as Backbone, Neck, and Head. Backbone performs extraction of significant features from the input image using CSPNets (Cross Stage Partial Networks). Neck is the second block that generates feature pyramids using Path Aggregation Network (PANet), it aggregates all the features detected by the first block and pass them onto Head where the final bounding boxes, output vectors with class probabilities and feature maps are generated. These feature pyramids help to detect the small objects as each map has both spatial and semantic information (Liu et al., 2021). As stated above, we used transfer learning where the model fine-tunes a pre-trained detector for a custom dataset. YOLOv5 uses mosaic data augmentation where each training image goes through scaling, color space adjustments, and translation to create a mosaic of nine images which helps the model to receive higher detection accuracy with smaller training sets.

Faster-RCNN: Faster-RCNN is an end-to-end CNN network, that passes an image input to the ConvNet which generates feature maps. Unlike earlier versions of RCNN family, it uses Region Proposal Network (RPN) instead of using selective search algorithm to output object proposals. Region-of-Interest (ROI) pooling is applied to make all proposals of same size. Then, processed proposals are passed to a fully connected layer that classifies the objects in the bounding boxes. We implemented the architecture using a framework called Detectron2, (it is an open-source PyTorch based object detection framework by Facebook AI research). This provides many state-of-the-art detection and segmentation algorithms like Faster R-CNN for easier implementation.

Evaluation metrics: Object detection performance can be measured by using mean average precision (mAP) which takes account of classification and localization while evaluating. The predicted bounding boxes are compared with the ground truth coordinate boxes if the overlap between them is more than the threshold value of Intersection-over-Union (IoU), then, it is considered as True Positive (TP) otherwise it gets classified as False Positive (FP). If the model fails to detect anything when the object is there, it is a False Negative (FN). After this, precision and recall of a model are calculated. As the model was pre-trained on COCO dataset for both models, COCO's evaluation method is used where mean of the precision values ranging across different IoU thresholds [.50:.95:.05] (IoU's from 0.50 to 0.95 in increments of 0.05) as shown in equation 1.

$$mAP = \frac{\&}{?} \sum_{!@'}^{?} \text{APi} \tag{1}$$

One other metric that is commonly used to evaluate the fit is box loss which measures how well predicted bounding boxes capture objects. It is calculated by taking the mean-squared-error (MSE) between the coordinates of predicted and the ground truth bounding boxes.

Results and discussion

The main objective of this research was to determine the size of developed lesions for which we trained CNN based models for lesion localization and then used image processing techniques to compute the size. For YOLOv5, the original image size of 1920x1080 was decreased to 1280x1280 for improved detection and faster training compared to using the standard 640x640 size. For FRCNN, the default image size range of (800,1333) pixels were used. By setting a range for these sizes, the model can learn to detect objects at different scales, which is important for achieving good performance in object detection tasks. Out of both architectures implemented for object localization, YOLOv5 showed a significantly better performance than FRCNN. Models were trained for 400 iterations or epochs. Both models were pretrained on MS COCO dataset meaning COCO weights were used for transfer learning. As a neural network trains on more parameters and epochs, there is a higher risk of overfitting. Early stopping is a technique which halts the training when there is no significant improvement in model's performance for a certain number of epochs. After being trained for 165 epochs out of a total of 400 epochs, with a learning rate of 0.01 and a batch size of 4, the YOLOv5 model's training was halted due to the lack of any significant improvement in its performance. At the time of stopping, it had achieved an mAP@.5 score of 0.921 and a bounding box loss of 0.029. On the other hand, FRCNN was trained for 400 epochs, with a learning rate of 0.02 and a batch size of 4. It achieved an mAP@.5 score and a bounding box loss of 0.823 and 0.41, respectively. When evaluated for recall at 0.5 IoU, FRCNN and YOLOv5 obtained scores of 0.782 and 0.923, respectively.

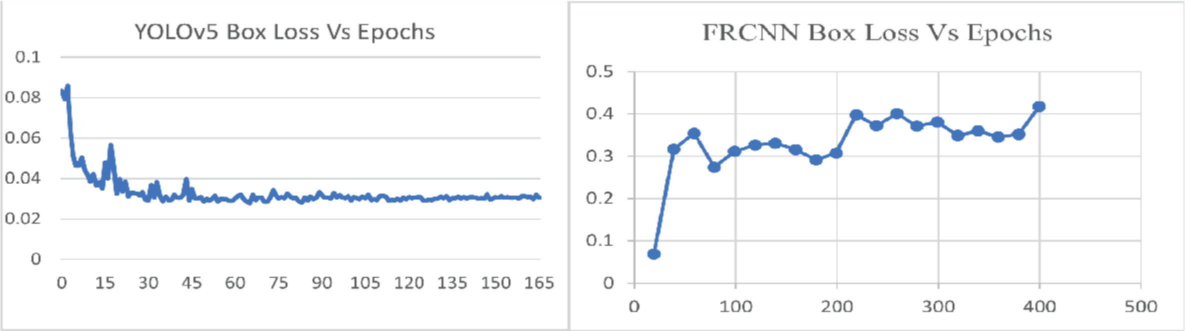

Figure 4: Showing loss curves for both models

YOLOv5 did a better job for localization of lesion pixels from the background. YOLOv5 also had fewer false positives and could even detect lesions which were in very early stages of development when compared to FRCNN.

Table 1: Metrics for YOLOv5 and FRCNN

Architecture	mAP@0.5	mAP@0.5:0.95:0.05	Recall@0.5
YOLOv5	0.923	0.451	0.941
FRCNN	0.823	0.371	0.782

After using a python script to crop the detected lesion regions, image processing techniques were used to calculate the number of pixels that make up the lesion area. Image binarization techniques were used to separate out the lesion pixels from sow's body as lesions were darker in color when compared to rest of the body. We used Python's OpenCV module to implement two automatic image thresholding techniques, namely Otsu's method and Gaussian adaptive thresholding. Otsu's method returns a single intensity threshold that divides pixels into two classes, foreground, and background. The base value is calculated by maximizing or minimizing the intensity variance between both classes. In adaptive thresholding, different thresholds are calculated for different parts of the image for segregating all the pixels. Both methods were used and produced the binary images. Then, the number of black pixels from binarized image was calculated. To evaluate performance of the methods used, lesion pixels were manually annotated to create binary masks using ImageJ's web application, then, plotted intensity histograms to get the number of black pixels. Otsu's method gave better results than adaptive thresholding, with a coefficient of determination of 0.9108 compared to 0.5921 of the latter. One of the possible reasons behind its poor performance can be the non-uniform lighting around lesion regions. In fact, some crates had heat lamps hanging from the truss producing shadows of the iron bar (top part of the crate) to get captured in the detected area resulting in varying intensity values causing problems.

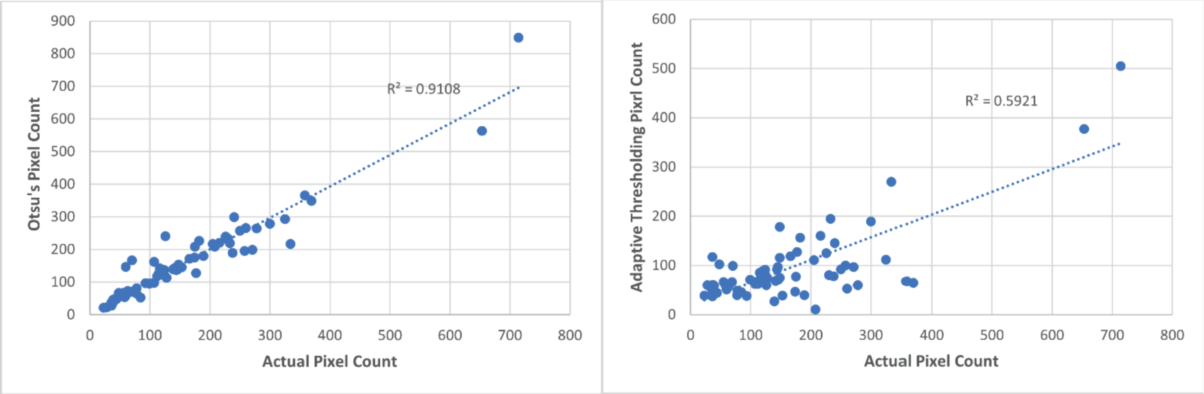

Figure 5: Graphs showing variation in estimated number of pixels against actual pixel count using Otsu's method and adaptive thresholding

For the estimation of area covered by lesions in measurable units, a calibration was performed using a crate's pipe which was close to lying sow and at a similar height of the shoulder as a reference object. The pipe had a diameter of 25.4 mm and 15 pixels wide. We used these values to calculate the one pixel's area in mm which came out to be 2.89 mm^2. Using equation 2 the area of the lesion was converted from pixels to mm^2.

$$\boldsymbol{Lesion\ Area} > \boldsymbol{mm^2}? = \boldsymbol{Lesion\ Area}\ (\boldsymbol{pixels}) \times \boldsymbol{2.89}\ \frac{\boldsymbol{mm}^{2}}{\boldsymbol{pixel}} \qquad (2)$$

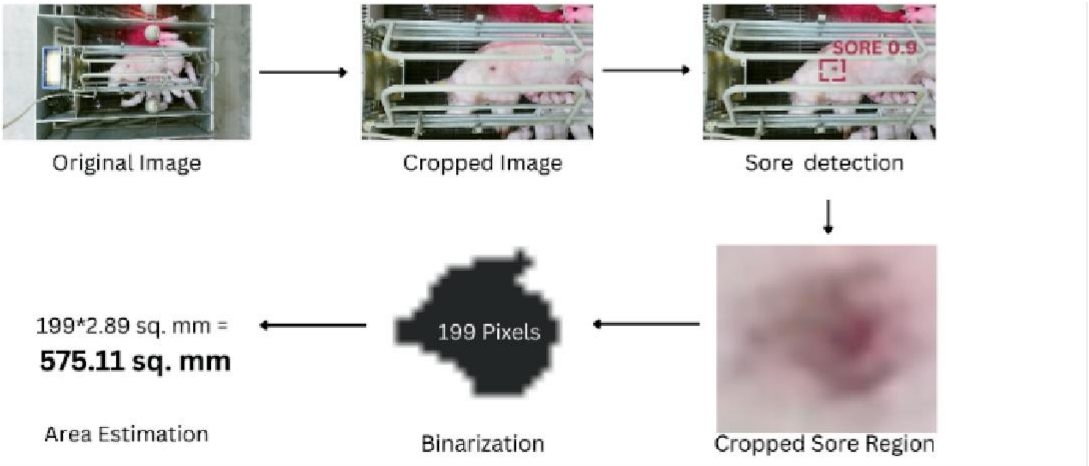

Figure 6: The entire pipeline showing different steps followed from image capturing to area estimation.

The binarization techniques performed poorly when the captured image had a sow being treated for lesions using a topical solution like zinc oxide. The zinc oxide ointment was the same color intensity (deep yellow) as that of lesion. Algorithms mistake the applied solution for lesion pixels outputting a larger lesion area. Figure 6 shows the entire process of the developed program, from data acquisition to lesion area estimation.

Conclusions

This paper proposed a non-invasive computer vision-based technique to measure the size of shoulder lesions in sows. Considering the fast inference speed of YOLOv5, this can be translated into a real time monitoring system that can detect lesions at an early stage, enabling earlier care resulting in better welfare for the animals and reduction in losses for producers.

Acknowledgments

This research was funded in part by USDA, Agricultural Research Service and USDA NIFA Award No. 2016- 67015-24. USDA is an equal opportunity employer.

References

Herskin, M.S., Bonde, M.K., Jørgensen, E., and Jensen, K.H. (2011) Decubital shoulder ulcers in sows: A review of classification, pain and welfare consequences. *Animal* 5(5), 757-766.

Leonard, S.M., Xin, H., Brown-Brandl, T.M., Ramirez, B.C., Johnson, A.K., Dutta, S., and Rohrer, G.A. (2021) Effects of farrowing stall layout and number of heat lamps on sow and piglet behavior. *Applied Animal Behaviour Science* 239, 105334.

Liu, Y., Sun, P., Wergeles, N., and Shang, Y. (2021) A survey and performance evaluation of deep learning methods for small object detection. *Expert Systems with Applications* 172, 114602.

Nola, G.T., and Vistnes, L.M. (1980) Differential response of skin and muscle in the experimental production of pressure sores. *Plastic and Reconstructive Surgery* 66(5), 728-733.

Rioja-Lang, F.C., Seddon, Y.M., and Brown, J.A. (2018) Shoulder lesions in sows: A review of their causes, prevention, and treatment. *Journal of Swine Health and Production* 26(2), 101-107.

Shao, H., Pu, J., and Mu, J. (2021) Pig-posture recognition based on computer vision: Dataset and exploration. *Animals* 11(5), 1295.

Wang, R., Gao, Z., Li, Q., Zhao, C., Gao, R., Zhang, H., Li, S., and Feng, L. (2022) Detection Method of Cow Estrus Behavior in Natural Scenes Based on Improved YOLOv5. *Agriculture* 12(9), 1339.

Yap, M.H., Hachiuma, R., Alavi, A., Brüngel, R., Cassidy, B., Goyal, M., Zhu, H., Rückert, J., Olshansky, M., Huang, X., and Saito, H. (2021) Deep Learning in Diabetic Foot Ulcers Detection: A Comprehensive Evaluation.
Computers in Biology and Medicine 135, 104596.