Deep Learning Techniques for Beef Cattle Body Weight Prediction

Mikel Gjergji[1], Vanessa de Moraes Weber[2,3], Luiz Otávio Campos Silva[4], Rodrigo da Costa Gomes[4],
Thiago Luís Alves Campos de Araújo[5], Hemerson Pistori[2,6], Marco Alvarez[1]

[1]*Departament of Computer Science and Statistics - University of Rhode Island - Kingston - USA*
[2]*Dom Bosco Catholic University (UCDB) - Campo Grande – MS – Brazil*
[3]*State University of Mato Grosso do Sul (UEMS) - Campo Grande– MS – Brazil*
[4]*Brazilian Agricultural Research Corporation - Embrapa Beef Cattle - Campo Grande - MS - Brazil*
[5]*Federal University of Ceará (UFC) - Fortaleza – CE – Brazil*
[6]*College of Computing, Federal University of Mato Grosso do Sul (UFMS) - Campo Grande – MS – Brazil*

***Abstract*—Following the weight of beef cattle is of great importance to the producer. The activities of nutrition, management, genetics, health and environment can benefit from the weight control of these animals. We explore different deep learning models performance in the regression task of predicting cattle weight. This is a hard problem since moving from 3-D space to 2-D images presents a loss of information in object shape, making weight prediction more difficult. A model that produces good results in this problem could potentially be applied more abstractly to similar problem spaces. We analyzed convolutional neural networks, RNN/CNN networks, Recurrent Attention Models, and Recurrent Attention Models with Convolutional Neural Networks, and show that convolutional neural networks achieve the highest performance. Our top model averages a MAE of 23.19 kg. This is nearly half the error as previous top linear regression models which reached an error of 38.46 kg.**

***Index Terms*—deep learning, weight, cattle, attention based models, convolutional neural networks, recurrent neural networks**

I. Introduction

Monitoring and maintaining the weight history of cattle allows for timely intervention of cattle diet, cattle health, and for greater efficiency in genetic selection. Another great advantage of tracking weight gain is to identify the best time to market animals because animals that have already reached the point of slaughter represent burden for feedlot.

Removing animals from paddocks and leading them to scales is a costly and stressful activity for both the animal and the herdsman. This process can cause injuries or even weight loss [1] [2]. With this in mind, some companies have been working on solutions to track the weight of feedlot cattle and have tools such as GrowSafe (Calgary, Canada), Intergado (Betim, Brazil) and the Bosch Precision Livestock Platform (Gerlingen, Germany). These solutions consist of weighing cell equipment that must be installed in passageways or in front of feeders and drinkers. However these devices need constant maintenance that may encumber the cost of production. Still on weight measurement, some researchers propose research where they relate measurements of body parts of animals with their weight [3] [4] [5].

In addition, researchers have developed livestock-based applications based on image analysis through Computer Vision [6]. These applications allow automation of some farm work in key areas such as animal behavior, health and welfare including nutrition management [7], locomotion [8], identification [9], body conditions [10] [9], diseases [11], and cattle weighing [12] [13] [14]. To this end, equipment is described for acquisition of these images and can be divided into two large groups, the first for 2-D images such as RGB cameras and thermal cameras, and the second for 3-D images such as depth and Kinect sensors, stereo vision and stereo photogrammetry according to the review by Nasirahmadi et al. [15].

In addition to classical feature extraction techniques detailed in the related work section, deep learning for computer vision has stood out in the last decade. Deep learning architectures are known to have often outperformed even humans in classification problems [16] [17] [15] [18]. These same architectures adapted to regression tasks can solve problems aimed at predicting continuous values [19], such as estimating head positions and detecting facial expressions [20] [21].

In this paper, we analyze convolutional neural networks, recurrent convolutional neural networks, recurrent attention models [22], and recurrent attention models with convolutional neural networks [23]. We find that convolutional neural networks outperform all of the other tested models. We were able to produce a model that nearly halved the error of a previous regression model based on more traditional computer vision techniques [24]. Our top model averaged a MAE of 23.19 kg while the previous paper produced a top model who's MAE was 38.46 kg. This demonstrates that convolutional neural networks vastly outperform models trained on hand picked features for this task.

In our related work section we review work done on weight calculation of animals, advancements and applications of convolutional neural networks, and applications of Recurrent Attention Models [22] with convolutional neural networks. In our materials and methods section we describe how we collected data and split it into train, validation, and test sets.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

In our models section we give an overview of different models tested, and provide a table of results for those models. In our analysis section we analyze the results of each model. In our future work section we talk about where to move forward with this problem based on problem areas in our models. We summarize our results in our conclusion.

II. Related Work

A. Weight Calculation of Animals

The use of computer vision techniques to predict weight of cattle has been applied for both 2-D and 3-D images [15]. Morphological characteristics such as rump height, rump width, body size, rib height and contours [25] were automatically measured and subsequently subjected to regression algorithms or Fuzzy logic. [26].

Using 3-D images, features were extracted from 234 images of Nellore beef cattle for regression algorithms and Artificial Neural Networks (ANN) to estimate the live weight of cattle [14]. The images were collected at various stages of the animal's life, namely: Weaning, Stocker, Beginning of Feedlot, and End of Feedlot phase.

Although some authors describe that biometric measurements extracted from 3-D images are highly correlated with animal weight [14] [27], ease of access and costs, images from ordinary cameras, such as security equipment for example, can also considered.

The segmentation of animals in their natural environment has been the subject of research and constitutes a challenging task for automated animal body mass prediction systems [27] [28]. Like segmentation, extracting frames with the best positioning of the cattle is also a challenging task. Therefore, deep learning techniques that can predict the weight of cattle through 2-D images without the specific task of segmentation and frame extraction seem promising.

B. Convolutional Neural Networks

Weight calculation in the 2-D space is difficult because there is an implicit loss of information when migrating from 3-D to 2-D. Convolutional neural networks have been applied to tasks facing similar problems with success, for example, 3-D pose estimation using 2-D data [29]. Convolutional neural networks have also been used to create 3-D point clouds from 2-D images [30], an important example for our use case since it provides a concrete example of a convolutional neural network learning a mapping from 2-D to a 3-D space. These examples demonstrate an ability of convolutional neural networks to work effectively on 2-D images when the problem space lies more in 3-D space, and make them a good candidate for weight prediction.

C. Combination RNN/CNN Attention Networks

Attention based RNN/CNN networks have seen success in several application domains. Some examples of this include image captioning [31] and object detection [32]. Attention based RNN/CNN networks have also seen use in a variety of different regression based tasks. A RNN/CNN network with attention was used to predict the price of precious metals [33]. A RNN/CNN network with attention was also used for fine-grained visual emotion regression [34]. These successes, particularly those of regression tasks, make these more advanced models a good candidate for testing in weight prediction. In our implementation, we replace a fully connected layer in the glimpse network for a CNN, and modify the action network for a regression task. The structure of a RNN/CNN network can heavily vary. For example, the model used for prediction of precious metal prices additionally implements a Regularization Self-Attention Mechanism, which improves performance through the use of regularization functions. The model used for fine-grained visual emotion regression implements channel-wise attention maps, and spatial attention maps alongside a novel polarity-consistent regression loss.

III. Materials and methods

A. Data collection

The images were collected from October 8 to October 20, 2018 at the Embrapa Beef Cattle in Campo Grande MS, Brazil. As can be seen in the aerial image in Figure 1, 10 male Nellore and 10 male Angus are distributed in two paddocks.

For the collection of images, the experiment included the installation of a DVR set: MD-1004NS MD-DVR41 of MIDI brand, cameras with AHD 720p image quality and a HD with 1Tb recording capacity. Two cameras shown in Figure 2 (b) of the equipment were installed so as to be fixed in the structure of the water trough shown in Figure 2 (a) of the equipment known as Intergado ® (Intergado Ltd., Betim, Brazil), on an adapted rod, so that each camera collects the image from the dorsal area of the animal when drinking water in one of the possible entries. Two other cameras shown in Figure 2 (c) were installed in the trough cover structure to acquire the profile images of the animals moving to the trough. The collected videos were stored in the DVR and later transferred to computers for the purpose of preprocessing and extracting

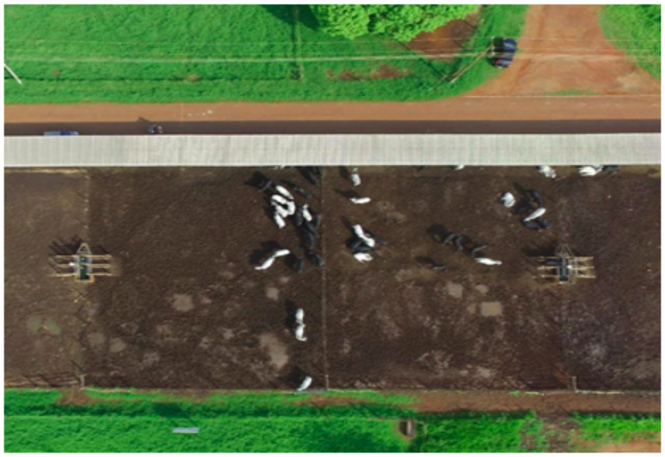

Fig. 1. Aerial view of Embrapa beef cattle feedlot with 20 male Nellore and Angus cattle distributed in two paddocks.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

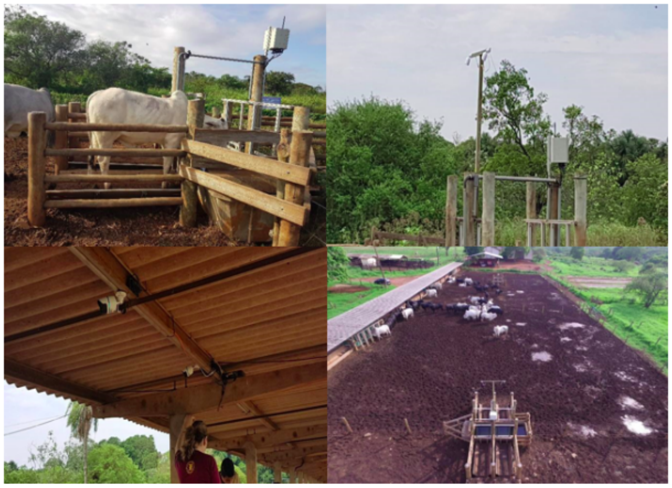

Fig. 2. Embrapa Beef Cattle image acquisition system - (a, top left) Intergado water trough, (b, top right) cameras installed on the water trough, (c, bottom left) cameras installed on the feeding troughs to capture profile images, (d, bottom right) aerial view of Embrapa Beef Cattle feedlot with the collection system.

the frames that contained animal images, as shown in Figure 2.

This drinking system shown in Figure 2 (a) is part of the Intergado equipment and allows individual identification of the animal through an RFID antenna, and every time the cattle moves to the water trough it is positioned on a platform coupled to a weighing scale. Additionally, time and weight data of the animals are transmitted via transmission antenna to the company's software. By relating the weighing time indicated by the software with the video of the corresponding drinker entrance it is possible to identify the animal that is in the drinker and thus extract tables containing the image of the cattle as can be seen in Figure 3 (a) and Figure 3 (b).

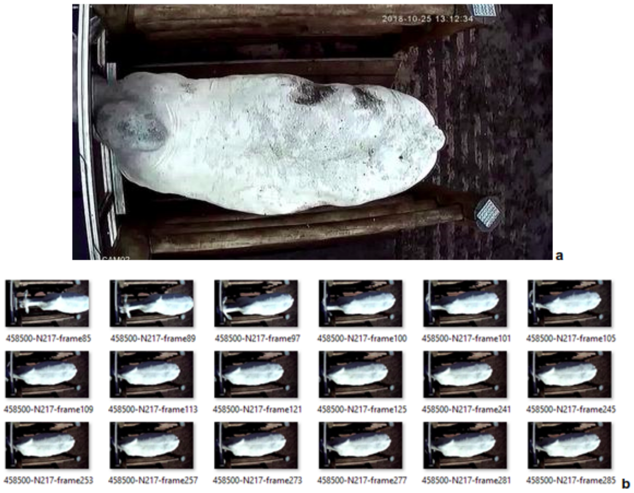

Fig. 3. Image of the dorsal area of bovine collected by the collection structure in the Embrapa Beef Cattle feedlot (a). Sequence of frames extracted from the bovine video in the water trough (b).

After the image collection was completed, they were validated to compose the ESTMASSABOV400 image database.

B. Train, Validation, and Test Set Split

Our data are video frames of individual cattle, moving from one frame of a cow to the next frame in time. We want to avoid training on data points that are nearly identical with data points in our validation or test set in order to get a better gauge of how our model performs on unique and unseen data. As a result of this, we separate our datasets into unique cattle images. Cattle present in the training dataset will not appear in the test/validation dataset and vice versa. We separate 60% of the unique cattle for training, 20% for validation, and 20% for test set. The test set is used by retraining the model on both training and validation and evaluating on test set, to avoid overfitting hyperparameters. In order to provide confidence in our results, we take our final models and train them on 5 seeded random shuffles of training (training and validation set) (80%) and test (20%) set. These results are available in Table 1. All weight labels were scaled by using Equation 1.

$$x' = \frac{x}{x_{max}} \tag{1}$$

where x is the weight of a particular instance of cattle, x_{max} is the largest cattle weight in our dataset, and x' is the new label. We do this in order to squash the range of labels to [0-1] and avoid extremely large gradients in training. For our MAE calculations, we average the difference between predictions and real labels of all batches on a given set of data.

IV. Models

In this section, we detail the different types of networks that were tested and training procedures for each of them. We provide results for our models in Table 1.

A. Convolutional Neural Networks

We trained 3 different convolutional neural networks using the Adam optimizer along with a learning rate of .0005. The Adam optimizer has shown to work well in practice [35] and is easily implemented through PyTorch. We used one of the EfficientNet models [36], EfficientNet-B1 as well as the ResNet18 [37]. Our reasoning for choosing EfficientNet is that it provides a smaller yet high performing model when trained on ImageNet, and better performing models on ImageNet seem to correlate to better accuracy when transferred to other problems [38]. We also tested out EfficientNet-B7 during our hyperparameter search but found it performed worse than EfficientNet-B1 while taking substantially longer to train (Approximately 4.5x). Our reasoning for choosing the ResNet18 model is that it is a highly tested model and easily implementable in PyTorch. All of our convolutional neural networks were trained for 10 epochs. We limited our training to 10 epochs due to excessive training time. For validation and test set results, a batch size of 32 was used for our CNNS. We increase this to 256 to speed up computational time for our 5 shuffles of the test set.

TABLE I
MEAN AVERAGE ERROR (MAE) IN KILOGRAMS FOR EACH OF OUR TESTED MODELS. THE SECOND AND THIRD COLUMNS SHOW VALIDATION AND TEST ERRORS RESPECTIVELY. THE THIRD COLUMN SHOWS MEAN AND STANDARD DEVIATION CALCULATED FROM 5 DIFFERENT SPLITS OF THE DATA INTO TRAINING AND TEST. THE FOURTH COLUMN SHOWS RUNTIME AVERAGE AND STANDARD DEVIATION, IN SECONDS, FOR EVALUATING THE TEST SET SPLITS. ALL RECURRENT MODELS USED A GLIMPSE SIZE OF 96X96 AND TOOK A TOTAL OF 6 GLIMPSES FOR A GIVEN IMAGE.

Model	MAE Validation Set	MAE Test Set	MAE 5 Shuffles of Test Set	Runtime
Combination RNN/CNN without attention (L1 Loss)	28.10	27.70	28.97 ± 3.94	147.69 ± 4.78
Combination RNN/CNN without attention (L2 Loss)	27.26	27.77	28.80 ± 4.36	146.68 ± 5.00
Recurrent Attention Model without CNN (L1 Loss)	38.31	30.00	30.17 ± 3.66	29.72 ± 0.79
Recurrent Attention Model without CNN (L2 Loss)	33.45	28.31	31.73 ± 3.65	30.52 ± 1.66
Combination RNN/CNN with attention (L1 Loss)	25.03	28.34	26.80 ± 2.52	162.10 ± 22.10
Combination RNN/CNN with attention (L2 Loss)	25.29	26.48	27.53 ± 2.37	154.63 ± 6.64
ResNet18 (L1 Loss)	23.36	24.51	25.15 ± 3.78	5.81 ± 0.38
ResNet18 (L2 Loss)	23.39	24.86	27.09 ± 3.91	**5.66 ± 0.23**
EfficientNetB1 (L1 Loss)	20.67	21.64	**23.19 ± 1.46**	25.62 ± 0.92
EfficientNetB1 (L2 Loss)	**19.83**	**20.49**	24.26 ± 3.01	25.57 ± 0.88
Previous Best Linear Regression	38.46			

B. Recurrent Attention Model (RAM) without Convolutional Neural Network

We trained a RAM model using the same hyperparameters from our top performing RNN/CNN with attention model. This network follows the same architecture as the RNN/CNN with attention model but does not process glimpses through a convolutional neural network, instead concatenating glimpse scales together and processing it through a fully connected layer. Our model replicates the RAM model specified in Recurrent Models of Visual Attention [22] with a few exceptions.

Our action network must reflect that of the regression task at hand, therefore our action network is a fully connected layer with input h_t and output a single continuous value. Another modification that was made to accommodate our change in task was to the baseline network. Rather than using a rectifier activation, we omit this from our baseline network. This is due to how our reward is defined for the reinforcement loss in our hybrid loss. Reward is defined by the following formula:

$$R = -1 * |p - y|$$

where R is our reward, p is our prediction, and y is our label. This is simply the absolute difference between our predicted versus real labels. A smaller difference produces a greater reward, with an exact match giving the highest reward of 0.

For this model, the RNN/CNN without attention model and the RNN/CNN with attention model, we trained for up to 100 epochs and stopped training if validation accuracy did not increase after 10 epochs. We allowed for more epochs in these models because there is a degree of randomness added in training when taking glimpses from the image. Random noise is added to the predicted location of the model which adds more uncertainty on whether the model has seen all of the information present in the training dataset. We limited our epochs to 10 for training on the 5 test set shuffles due to computational constraints. We were able to use a batch size of 256 for this model since it has a low memory foot print comparative to our other models.

C. Combination RNN/CNN without attention model

In this model, we remove the attention portion of the RAM with CNN and instead feed fixed locations to the model. This is accomplished by removing the location and baseline network from the Recurrent Attention Model. Since we do not need to train a location network on this model, our loss function for this network is simply L1 or L2 loss rather than the hybrid loss used with attention networks. For this model and the RNN/CNN with attention model, we used a batch size of 32 during the validation and test set evaluation, but doubled it to 64 to reduce computational time for our 5 shuffles of the test set.

D. Combination RNN/CNN with attention

While the original RAM model concatenates glimpses together and processes it in a fully connected layer, we substitute this fully connected layer with the EfficientNet-B1 CNN. This adds more complexity to the function as well as better extraction of features from the glimpses, as CNNs are known to do.

The rest of the model is identical to the model specified in the Recurrent Attention Model without Convolutional Neural Network specified previously. Figures 4 and 5 show examples of glimpse paths taken for the RNN/CNN with attention networks using L1 and L2 loss.

V. RESULTS AND DISCUSSION

Our runtimes were calculated on a computer with 64 GB of RAM, an AMD 3900x CPU, and a NVIDIA RTX 2080 TI GPU. Our recurrent neural network models take a substantial amount of time comparative to the CNNs.

Results for our convolutional neural networks are shown in Table 1. Box plots versus real labels for our EfficientNet-B1 model can be seen in Figures 9 and 10. After performing a random search on our RNN/CNN with attention model for which hyperparameters performed best on validation, we found the highest performing was 6 glimpses, a patch size of $96x96$ pixels, and no additional scales. We took these hyperparameters and retrained a model using both the training and validation data, and tested on our held out test data. We

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

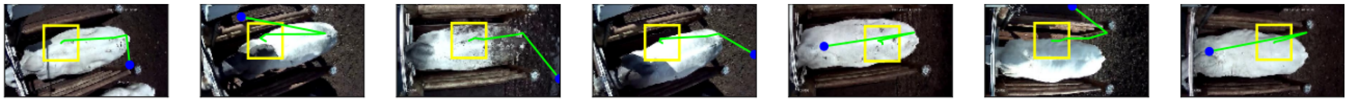

Fig. 4. A random batch of seven cattle from the test set used in our RNN/CNN with attention model and the glimpses taken using using L1 Loss. The blue dot is the starting location that the first glimpse is taken, generated from a random uniform distribution. The green line demonstrates the path taken as more glimpses are taken until ultimately taking its final glimpse at the yellow box.

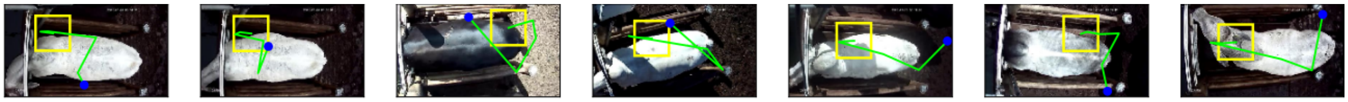

Fig. 5. A random batch of seven cattle from the test set used in our RNN/CNN with attention model and the glimpses taken using using L2 Loss. The blue dot is the starting location that the first glimpse is taken, generated from a random uniform distribution. The green line demonstrates the path taken as more glimpses are taken until ultimately taking its final glimpse at the yellow box.

evaluated the final error using both L1 and L2 loss metrics. A box plot versus true value can be seen in Figures 7 and 8.

Our original intuition behind testing different variations of recurrent models was inspired by a recent paper that shows that CNNs trained on ImageNet prioritize textures versus object shapes [39]. Since mass is much more closely related to the object shape of the cattle rather than the textures of the cattle, we thought that a network that incorporates a location policy will be forced to give greater attention to object shape.

We tried different variations that attempted to exploit this idea, the first being a close implementation of the original recurrent models of visual attention [22]. This model did not converge to a low enough MAE, and actually reached a MAE close to that of the model trained on hand picked features. This is most likely caused by a lack of ability to learn high level features that a convolutional neural network knows to exploit. We then attached the EfficientNet-B1 network to our glimpse network to create a RNN/CNN with attention and found that this reduced error a great deal, but still did not beat the standalone EfficientNet-B1 network. Finally, we tested a model that removed the attention portion of the model, and found slightly degraded results over our attention module.

One possible explanation for these results could be that the convolutional neural network always has full view of the image from input to output, while the recurrent models are selecting subsections of the image. This is a loss of information and can negatively affect performance of the models. Another potential explanation for the poor performance of these models is that shapes of cattle are not very distinct between labels, they follow the same oval shape rather than distinct paths of the MNIST dataset that the original RAM paper was tested on. This could mean that the location policy is ineffective since all optimal location paths will be similar. The negligible performance difference between our attention model and non attention model seems to support this.

An interesting datapoint can be seen in our box plot versus actual weight graphs for our models, shown in Figures 7-10. There is a huge failure of each model of an instance of cattle weighing 392.50 kg, where each model massively overestimates the weight of the cattle. In the frames of the data-points for this label, other cattle are entering and leaving the frame, an example of which is shown in Figure 6. The models could be learning to segment all areas of the picture that have the texture of cattle, and are including the additional cattle area contributed by these straying cattle that do not belong. The only model that makes some predictions near the actual weight of the cattle is the RNN/CNN network using L1 loss. Since it does not observe the entire image, it may have only taken glimpses in the area containing the cattle we are attempting to predict on, leading to better results for some predictions. This can be seen in Figure 7, where there are a number of outlier predictions near the true label for that instance of cattle. We detail why we believe these failures occur and some methods that may combat it in our Future Work section.

VI. Future Work

Our models fail to accommodate for images that have stray cattle. It seems that the models have likely learned to simply segment texture of cattle in an image. Some potential evidence to support this theory is that some of the predictions made from the RNN/CNN with L1 Loss were near correct in predicting the mass, while most predictions were way larger than the

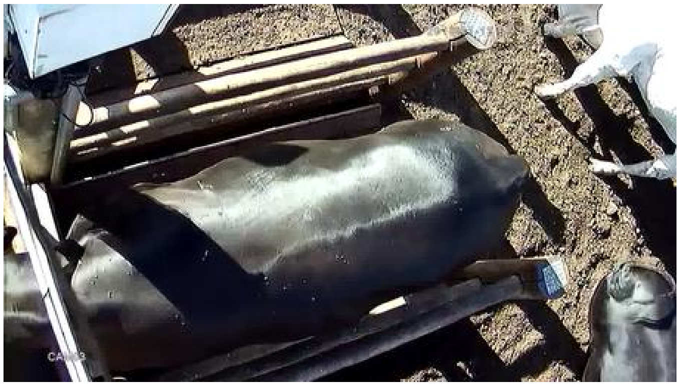

Fig. 6. A frame where other cattle can be seen in the upper right corner, as well as the bottom right corner. This is potentially being picked up by the networks and causing large error rates. We dive deeper into this in our Future Work section.

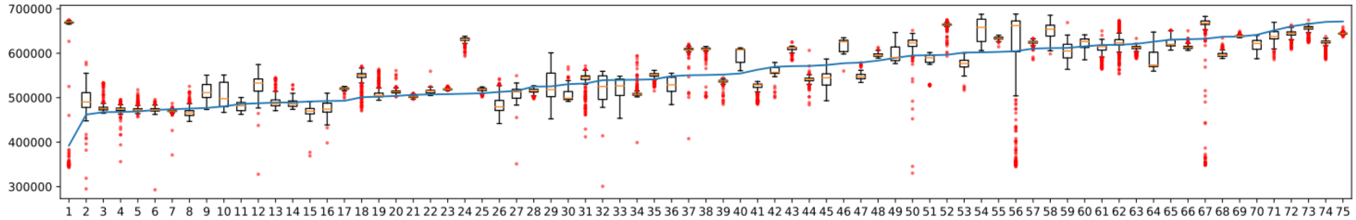

Fig. 7. Predictions for RNN/CNN network with attention using L1 loss. Actual weight values are plotted in sorted order in blue. Box plots of predictions for that given weight are shown for that label, with outliers plotted in red.

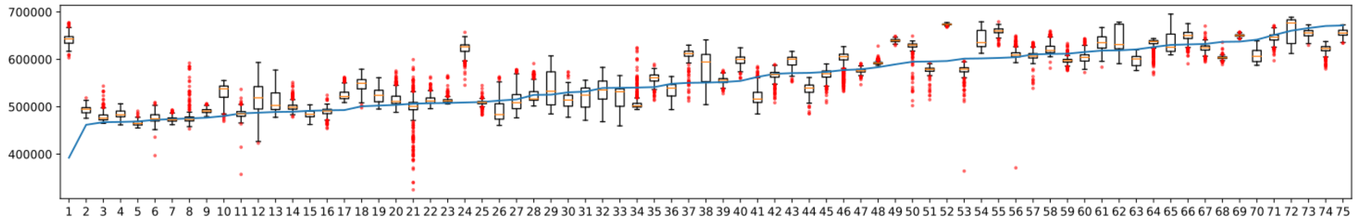

Fig. 8. Predictions for RNN/CNN network with attention using L2 loss. Actual weight values are plotted in sorted order in blue. Box plots of predictions for that given weight are shown for that label, with outliers plotted in red.

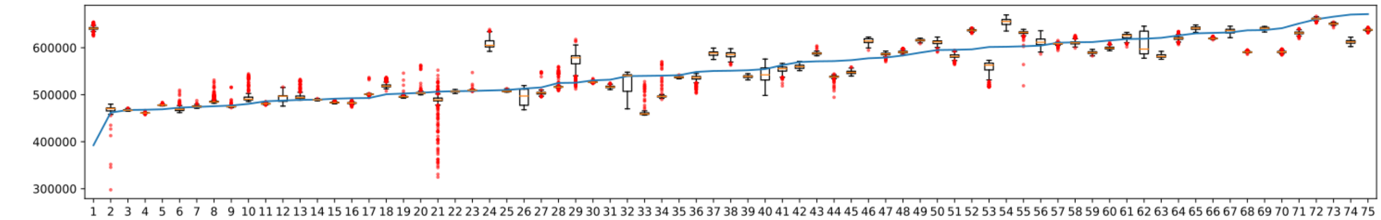

Fig. 9. Predictions for EfficientNet-B1 using L1 loss. Actual weight values are plotted in sorted order in blue. Box plots of predictions for that given weight are shown for that label, with outliers plotted in red.

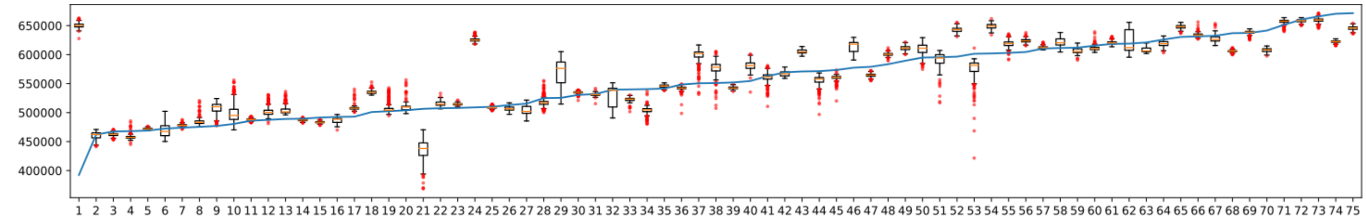

Fig. 10. Predictions for EfficientNet-B1 using L2 loss. Actual weight values are plotted in sorted order in blue. Box plots of predictions for that given weight are shown for that label, with outliers plotted in red.

actual. Likely with some low probability the glimpses that the network took never saw the stray cattle and did not include this in its prediction. The models tested in this paper simply lack the complexity to differentiate between the cattle being predicted and those stray cattle entering the image. We propose some methods below that may counteract this problem.

One potential point of interest would be a more advanced attention module to accommodate for this, since ours was able to predict fairly accurately on some occasions. This was demonstrated by some near correct results for our CNN/RNN model with L1 loss.

Another route that can be taken to tackle this would be to exploit the format of the data. Since each unique cattle instance are frames pulled from a video, a recurrent model can be trained to integrate information over time (from frame $1...N$) and better segment the cattle we are actually trying to predict on. Essentially the goal would be to "forget" frames in which stray cattle were entering the image and "remember" the good frames. An LSTM based model would be a good candidate for this.

Finally, we could simply use more complex labels than simply mass. We could train a semantic segmentation model to learn to ignore these stray cattle [40]. This requires a great deal of manual labeling, since our data would have to be manually labeled with these segmentation maps.

VII. Conclusion

Producing an accurate model that is able to predict the weight of cattle from raw 2-D images would be of great use to producers. Tracking weight is beneficial in cattle health, genetic selection, and selecting correct point of slaughter for cattle. Producing a model that is able to acheive this from camera image rather than weighing cell equipment can help to avoid the constant maintenance that these machines require, lowering cost and saving time. Our experiments appear to show that convolutional neural networks are high performing on the task of weight calculation in 2-D images. However, they are highly prone to bad data as shown in figure 6. While we reached an error rate much lower than that of the models trained on hand-picked features, there is still work to do to eliminate the large errors that can occur from these bad datapoints as they are likely to occur when a model is implemented in practical use.

References

[1] E. D. G. Santos, M. F. Paulino, R. d. P. Lana, S. d. C. Valadares Filho, and D. S. Queiroz, "Influência da suplementação com concentrados nas características de carcaça de bovinos f1 limousin-nelore, não-castrados, durante a seca, em pastagens de brachiaria decumbens," *Revista Brasileira de Zootecnia*, vol. 31, no. 4, pp. 1823–1832, 2002.

[2] S. Ozkaya and Y. Bozkurt, "The relationship of parameters of body measures and body weight by using digital image analysis in pre-slaughter cattle," *Archiv fur Tierzucht*, vol. 51, no. 2, p. 120, 2008.

[3] M. d. O. Franco, M. I. Marcondes, J. M. d. S. Campos, D. R. d. Freitas, E. Detmann, and S. d. C. Valadares Filho, "Evaluation of body weight prediction equations in growing heifers," *Acta Scientiarum. Animal Sciences*, vol. 39, no. 2, pp. 201–206, 2017.

[4] G. L. Reis, F. H. M. A. R. Albuquerque, B. D. Valente, G. A. Martins, R. L. Teodoro, M. B. D. Ferreira, J. B. N. Monteiro, M. d. A. Silva, and F. E. Madalena, "Predição do peso vivo a partir de medidas corporais em animais mestiços holandês/gir," *Ciência Rural*, vol. 38, no. 3, pp. 778–783, 2008.

[5] G. Bretschneider, A. Cuatrin, D. Arias, and D. Vottero, "Estimation of body weight by an indirect measurement method in developing replacement holstein heifers raised on pasture," *Archivos de Medicina Veterinaria*, vol. 46, no. 3, pp. 439–443, 2014.

[6] J. G. A. Barbedo and L. V. Koenigkan, "Perspectives on the use of unmanned aerial systems to monitor cattle," *Outlook on Agriculture*, vol. 47, no. 3, pp. 214–222, 2018.

[7] S. Nyamuryekung'e, A. F. Cibils, R. E. Estell, and A. L. Gonzalez, "Use of an unmanned aerial vehicle- mounted video camera to assess feeding behavior of raramuri criollo cows," *Rangeland ecology & management*, vol. 69, no. 5, pp. 386–389, 2016.

[8] F. Lao, T. Brown-Brandl, J. Stinn, K. Liu, G. Teng, and H. Xin, "Automatic recognition of lactating sow behaviors through depth image processing," *Computers and Electronics in Agriculture*, vol. 125, pp. 56–62, 2016.

[9] A. Bercovich, Y. Edan, V. Alchanatis, U. Moallem, Y. Parmet, H. Honig, E. Maltz, A. Antler, and I. Halachmi, "Development of an automatic cow body condition scoring using body shape signature and fourier descriptors," *Journal of dairy science*, vol. 96, no. 12, pp. 8047–8059, 2013.

[10] S. Jung and K. B. Ariyur, "Strategic cattle roundup using multiple quadrotor uavs," *International Journal of Aeronautical and Space Sciences*, vol. 18, no. 2, pp. 315–326, 2017.

[11] S. G. Matthews, A. L. Miller, T. PlÖtz, and I. Kyriazakis, "Automated tracking to measure behavioural changes in pigs for health and welfare monitoring," *Scientific reports*, vol. 7, no. 1, pp. 1–12, 2017.

[12] M. Tscharke and T. Banhazi, "Review of methods to determine weight and size of livestock from images," *Australian Journal of Multi-disciplinary Engineering*, vol. 10, no. 1, pp. 1–17, 2013.

[13] S. Tasdemir, A. Urkmez, and S. Inal, "Determination of body measurements on the holstein cows using digital image analysis and estimation of live weight with regression analysis," *Computers and electronics in agriculture*, vol. 76, no. 2, pp. 189–197, 2011.

[14] A. Cominotte, A. Fernandes, J. Dorea, G. Rosa, M. Ladeira, E. van Cleef, G. Pereira, W. Baldassini, and O. M. Neto, "Automated computer vision system to predict body weight and average daily gain in beef cattle during growing and finishing phases," *Livestock Science*, vol. 232, p. 103904, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1871141319310856

[15] A. Nasirahmadi, S. A. Edwards, and B. Sturm, "Implementation of machine vision for detecting behaviour of cattle and pigs," *Livestock Science*, vol. 202, pp. 25–38, 2017.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[17] A. dos Santos Ferreira, D. M. Freitas, G. G. da Silva, H. Pistori, and M. T. Folhes, "Weed detection in soybean crops using convnets," *Computers and Electronics in Agriculture*, vol. 143, pp. 314 – 324, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168169917301977

[18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[19] S. Lathuilière, P. Mesejo, X. Alameda-Pineda, and R. Horaud, "A comprehensive analysis of deep regression," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[20] G. Fanelli, J. Gall, and L. Van Gool, "Real time head pose estimation with random regression forests," in *CVPR 2011*. IEEE, 2011, pp. 617–624.

[21] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 2879–2886.

[22] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in neural information processing systems*, 2014, pp. 2204–2212.

[23] J. Ba, V. Mnih, and K. Kavukcuoglu, "Multiple object recognition with visual attention," *arXiv preprint arXiv:1412.7755*, 2014.

[24] V. A. M. Weber, F. L. Weber, R. C. Gomes, A. S. Oliveira Junior, G. V. Menezes, U. G. P. Abreu, N. A. S. Belete, and H. Pistori, "Prediction of girolando cattle weight by means of body measurements extracted from image," *Revista Brasileira de Zootecnia*, 2020.

[25] D. Anglart, "Automatic estimation of body weight and body condition score in dairy cows using 3d imaging technique," 2014.

[26] D. Stajnko, M. Brus, and M. Hočevar, "Estimation of bull live weight through thermographically measured body dimensions," *Computers and Electronics in Agriculture*, vol. 61, no. 2, pp. 233–240, 2008.

[27] A. F. Fernandes, J. R. Dórea, R. Fitzgerald, W. Herring, and G. J. Rosa, "A novel automated system to acquire biometric and morphological measurements and predict body weight of pigs via 3d computer vision," *Journal of animal science*, vol. 97, no. 1, pp. 496–508, 2019.

[28] Y. Qiao, M. Truman, and S. Sukkarieh, "Cattle segmentation and contour extraction based on mask r-cnn for precision livestock farming," *Computers and Electronics in Agriculture*, vol. 165, p. 104958, 2019.

[29] S. Li and A. B. Chan, "3d human pose estimation from monocular images with deep convolutional neural network," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 332–347.

[30] D. Crispell and M. Bazik, "Pix2face: Direct 3d face model estimation," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 2512–2518.

[31] K. Chen, J. Wang, L. Chen, H. Gao, W. Xu, and R. Nevatia, "ABC-CNN: an attention based convolutional neural network for visual question answering," *CoRR*, vol. abs/1511.05960, 2015. [Online]. Available: http://arxiv.org/abs/1511.05960

[32] Y. Ji, H. Zhang, and Q. J. Wu, "Salient object detection via multi-scale attention cnn," *Neurocomputing*, vol. 322, pp. 130 – 140, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231218311342

[33] J. Zhou, Z. He, Y. N. Song, H. Wang, X. Yang, W. Lian, and H.-N. Dai, "Precious metal price prediction based on deep regularization self-attention regression," *IEEE Access*, 2019.

[34] S. Zhao, Z. Jia, H. Chen, L. Li, G. Ding, and K. Keutzer, "Pdanet: Polarity-consistent deep attention network for fine-grained visual emotion regression," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 192–201.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[36] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 2019.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

[37] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[38] S. Kornblith, J. Shlens, and Q. V. Le, “Do better imagenet models transfer better?” *CoRR*, vol. abs/1805.08974, 2018. [Online]. Available: http://arxiv.org/abs/1805.08974

[39] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, “Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness,” 2018.

[40] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, “A review on deep learning techniques applied to semantic segmentation,” *arXiv preprint arXiv:1704.06857*, 2017.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE