IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 12, 2020, accepted November 23, 2020, date of publication November 26, 2020, date of current version December 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3040805

Deep Learning Based Real-Time Body Condition Score Classification System

KERIM KÜRŞAT ÇEVIK
Department of Management Information Systems, Akdeniz University, 07070 Antalya, Turkey
e-mail: kcevik@akdeniz.edu.tr

ABSTRACT The number of animals worldwide is increasing day by day to meet the increasing animal protein needs. Depending on the increase in animal production, yield amount which can be obtained from per unit area can be increased by increasing the number of animals. In dairy cattle farms, it is necessary to group the animals according to their body condition score (BCS) and to care and feed the animals at certain times. Under normal conditions, these processes should be conducted by animal caregivers or experts coming to the enterprise. BCS ratings conducted by experts on farms based on visual examination may give unreliable results and may include misinterpretations. Therefore, technology-supported systems are required. In this study, the prediction of BCS, which is the most important indicator of proper feeding of dairy cattle, is aimed. In addition, by adapting the designed system to simple, fast and user-friendly mobile software, it will be possible to provide tests in enterprise environments in a shorter time. In the design of the system, deep learning models, which have been used frequently in recent years in computer science, have been used. The CNN model, which was trained with a 94.69% success rate through these data, has been converted into a mobile-friendly format for real-time tests. It is aimed to make real-time tests and provide easy access for dairy producers with the help of the designed mobile software. In order to increase the success of the CNN architecture, pre-trained networks have been utilized. In the study, VGG19 pre-trained network, whose success rate has been proved in the previous studies conducted in the literature, was used in model design. The 78.0% performance results obtained from the study indicate that pre-trained CNN architectures based on deep learning are successful for the real-time BCS classification problem.

INDEX TERMS Animals, computers and information processing, computer aided engineering, neural networks, mobile applications.

I. INTRODUCTION

In dairy cattle, care and nutritional requirements may change through early, mid and late lactation periods and also during dry period. Therefore, animals should be grouped according to their levels in dairy cattle farms. Many enterprises have dairy cattle that are too fat or too thin from birth to dry period (stop of milk production). Failure to identify these animals at the right time increases the expenses of the dairy cattle enterprises due to disease treatments, losses in milk production and decreased fertility rates. In order to eliminate these problems, the animals are inspected regularly by experts and a scoring is made. This scoring system, involves visual inspection of each animal and is called body condition score (BCS).

BCS is a method based on evaluation of fatness or thinness of cattle according to a five-point scale ranging from 1 to 5. Here, Score 1 indicates very thin cattle while score 5 indicates the very fat [1]. At birth, 1 is classified as very thin, 2 as thin and 5 as very fat, while 3 is the expected average score. Figure 1 shows the explanation of the BCS scores [2].

BCS offers many advantages for dairy cattle enterprises. Since the cows are scored between 1 and 5 in different periods, the ones with the desired BCS values exhibit lower prevalence rate for metabolic diseases. In addition, reproductive disorders that may arise among animals that are too fat or too thin can be prevented. Unnecessary feeding consumption can be avoided as well as increase in yield can be achieved with BCS, which is a unique indicator that shows the needs of animals are fully met [3].

BCS also indicates the energy balance estimation in dairy cattle. It is sometimes required to ensure that animals with undesirable BCS levels are reached to the desired levels by caring the animals. This process is carried out according to the method of grouping the animals by observation in line

The associate editor coordinating the review of this manuscript and approving it for publication was Donghyun Kim.

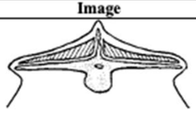
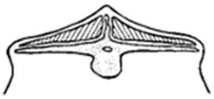
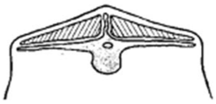
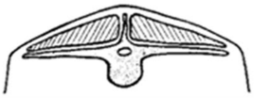
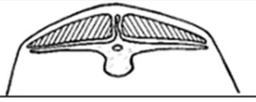
BSC	Image	Description
1		This cow is very emaciated and rarely seen on a farm.
2		This cow is too thin and causes low milk production and reproduction problems. It may be identified as healthy. The vertebrae and short ribs are easily visible, and, the short ribs appear scalloped, but individual vertebrae are not apparent.
3		This cow is likely to be a healthy and high-producing cow. However, if a cow calves in at a score of 3.0 or less, it may not have enough body fat to use for high peak milk production and to carry her through until dry matter intake increases.
4		This is a fat cow. Therefore, cows calving in at this condition should eat less and lose more body weight. Naturally, these cows will have more metabolic problems.
5		This cow is extra obese and will have metabolic and breeding problems. The backbone and short ribs cannot be seen and are not easy to feel.

FIGURE 1. BSC scoring system in dairy cattle [2].

with the individual skills of the experts or staff experienced in this field. In this sense, the execution of these processes on farms may be ignored from time to time based on the cost of retaining expert staff permanently in the enterprises and the lack of experience of the staff [4].

Kellogg separated the body condition score in dairy cattle according to the periods of the animal. According to Table 1, BCS, which is between 3.5-4 in the dry period, is indicated as 2.5-3 during one month postpartum [5].

TABLE 1. Ideal BCS Range [5].

Stage of Lactation	Score Value
Drying-off	3.5-4.0
Calving (older cows)	3.5-4.0
One-month postpartum	2.5-3.0
Mid-lactation	3.0
Late lactation	3.25-3.75
Calving (first lactation)	3.5

It has been stated by Kellogg that BCS is used as a useful tool in providing the herd's nutritional management and daily productivity. In their study, it has been stated that with the improvement in nutritional level, animal health, reproductive performance and milk productivity can be improved. However, thin cows in the negative energy balance in the herd are unable to perform at maximum capacity. Cows that are too fat are also prone to metabolic problems. Kellogg stated that using the BSC scoring of the herd could allow dairy producers to achieve adjustments in nutritional status more accurately [5].

In a different study published by Heinrichs and Ishler, it has been stated that BSC value should be 3+, 4- in the calving period, and the scores below 3+ indicate that the cows receive an inadequate energy supply in the late lactation and/or dry period. The researchers indicated that the failure to replenish energy reserves during this period would limit milk production during lactation. They also stated that the scores above 4- indicate that the energy intake in the late lactation and/or dry period is very high. They proposed to separate such cows in the dry period from the milking herd and to feed them with an adequate but not excessive protein, mineral and vitamin additive and a low energy ration [6].

II. RELATED WORKS

In recent years, various initiatives have been proposed to estimate BCS automatically with technological advances. In the early 2000s, BCS was tried to be determined by classical image processing methods. However, deep learning-based approaches have been started to be used since 2015. In addition, the images used in the solution of the BCS problem can be categorized as 2D and 3D. Almost all of these images are professionally taken from the back of the animal. In addition, real-time solutions have been presented in the literature, albeit in a limited number.

The systems designed in studies conducted until 2016 were based on the introduction of classical image processing processes and generally based on determining anatomical and shape-based surfaces and turning them into feature sets. In addition, in some studies feature reduction from these features was conducted by using various methods. The real images taken from the back of the animals were used as a dataset in these studies. As the deep learning methods have been started to be preferred by the researchers, the design methods of the automatic BCS detection systems have also changed. The literature review is shown in Table 2.

When the literature on computer-aided automatic or semi-automatic BCS estimation systems is examined, it is seen that the images are generally obtained from the back region of the animal due to large number of anatomical regions and shape features, and BCS estimations are made based on these images. In addition, many of the images taken in the studies were taken using 3D imaging devices. In the studies conducted using 2D imaging, the devices used were for professional shooting. Therefore, it is understood that the practicality of most of the applications is quite costly, difficult and inflexible. In this study, the developed mobile program is intended for animal husbandry applications, and it is paid attention that the imaging method and imaging devices are accessible so that the study can be applied in daily life. For this reason, images obtained from a mobile device that everyone can access, showing the back of the animal, were used as a data set for BCS determination. In addition, a system using trained CNN pre-trained networks, supported by software based on mobile application, was implemented for testing the study. The study is express to the use of dairy producers both in theory while measuring the effect of CNN architectures on the BCS problem and in practice.

The study consists of four parts and the rest of the study is as follows: In second section, the data set and the pre-trained CNN architectures used in the study are presented. In third section, the designed software and the usage of the software

TABLE 2. BCS Literature Review.

Authors	Methodology	Success Rate
Bewley, Peacock et al. 2008 [7]	They provided accurate identification of 23 different points from the 2D images.	Independent operating level of the system was indicated as low and 92%.
Krukowski 2009 [2]	Canny edge detection, standard deviation, skewness, kurtosis, uniformity and entropy in 3D images	The semi-automatic success of the study was reported to be 46.0%
Azzaro, Caccamo et al. 2011 [8]	23- anatomical points and various shape features labeled using shape based- PCA in 2D images	They reported that the failure rate was 31.0% in results of the study
Anglart 2014 [9]	3D images with image processing techniques	69% success rate was reported to be achieved
Halachmi, Klopčič et al. 2013 [10]	Based on anatomical and shape features using thermal camera 3D images	Thermal cameras may cause problems for the system to become widespread and reported a success rate of 94.0%
Spoliansky, Edan et al. 2016 [11]	3D images obtained by a lower cost imaging device (microsoft kinect), segmentation of various anatomical points	They achieved a 74.0% success rate in their detection system
Alvarez, Arroqui et al. 2018 [12]	2D images obtained via Kinect V2, CNN	They achieved 78.0% accuracy rate with 0.25 sensitivity and 94% accuracy rate with 0.5 sensitivity
Rodríguez Alvarez, Arroqui et al. 2019 [13]	Transfer learning method besides CNN architectures (SqueezeNet and VGG16)	Tests achieved 82.0% accuracy with 0.25 sensitivity and 97% accuracy with 0.5 sensitivity
Huang, Hu et al. 2019 [14]	3D images with CNN architectures such as DenseNet and Inception-v4 in SSD detector.	SSD method could achieve 98.0 % classification accuracy and 86.0% location accuracy
Li, Hu et al. 2019 [15]	3D images with YoloV2 object detection method and ResNet50	Test of the system to achieve an accuracy of 78.0%
Çevik and Boğa 2019 [16]	Pre-trained CNN architectures (AlexNet, VGG-19, GoogleNet, ResNet-101 and Inception v3).	As a result, they reported that AlexNet pre-trained network achieved the highest test performance (64.0%) in the BCS classification process
Çevik and Boğa 2019 [4].	R-CNN architecture with pre-trained networks, AlexNet, VGG-16 and VGG-19	VGG-19 pre-trained network could be used in BCS problem with 94.47% test success rate

are mentioned. In the last section, the obtained results and the contribution of the results to the literature are discussed.

III. MATERIAL AND METHOD

In the study, the performance of pre-trained CNN architectures, one of the sub-branches of deep learning, was tested in determining BCS automatically in dairy cattle. In addition, in order to facilitate the implementation of the system, a mobile application that users can easily access was developed and the tests were carried out in this environment. The stages of the designed system are shown in Figure 2.

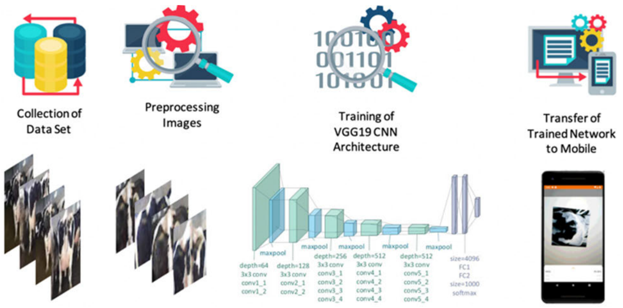


FIGURE 2. Flow chart of the system designed for the mobile application.

Machine learning is used today in many areas such as identifying objects in images, conversion speech into text, matching news items, posts or products with users' interests, and selecting relevant search results. Increasingly, a classification model called as deep learning is utilized in these applications. Since conventional machine learning is limited in processing the raw form of data, before implementing machine learning, various data preprocessing, segmentation, feature extraction, feature selection and reduction procedures are required. In contrary, in deep learning methods, the learning of the data are provided by calculation models composing of multiple processing layers with multiple levels of abstraction. To calculate the value of a machine in each layer, it uses the back propagation algorithm to determine how much the internal parameters used should change compared to the representation in the previous layer. Thus, with deep learning, the complex structure in large data sets can be discovered. Deep convolutional networks have led to breakthroughs in processing images, video, speech and audio while recurrent networks have helped to solve sequential data such as text and speech [17].

Although the concept of deep learning architectures is based on the first general learning algorithm for supervised deep-fed multilayer perceptron's published by Ivakhnenko and Lapa in 1965, the first successful deep learning architecture in literature is the "LeNet" architecture developed by Yann LeCunn in 1989 [18], [19]. LeCunn tried to classify handwritten digits (MNIST) using LeNet architecture in his studies until 1998. In the LeNet network, the sub-layers consist of subsequently followed conv and max pool layers. The next top layers correspond to fully connected traditional MLP [19]. Figure 3 shows a standard LeNet architecture. Although the recurrent neural networks and LSTM models proposed by Hochreiter and Schmidhuber were proposed as well as LeNet studies, Deep Neural Networks were not preferred due to the high cost of computing between 1990 and 2000. Instead, the researchers preferred presenting the input data to models such as SVM, standard ANN etc. and processing the data at a certain level, which provided to solve the problems faster [20].

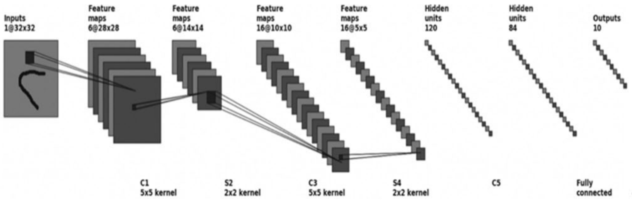


FIGURE 3. LeNet Architecture [21].

With the increase of CPU performance and the emergence of GPUs after 2000, the applicability of deep neural networks have started to be discussed. In 2006, Geoffrey Hinton announced how to train a deep multi-layer feed-forward network, which triggered the idea of deep learning [22]. After the study, interest in deep architecture has increased and studies have focused on this direction. In the following years, many deep architectural designs were realized, and the next step was Hinton and his team's high classification success in the ImageNet competition in 2012. With the networks currently used with the name of AlexNet, the classification success of ImageNet, which was 26.1%, was reduced to 15.3% by Hinton and his team [23]. With the architectures developed in the following years, success rates have been reduced to much lower levels. Figure 4 shows the architectural distribution of ImageNet classification success by years. After 2015, pre-trained deep architectures have surpassed human-level performance on the ImageNet dataset. Depending on this performance rate, researchers and companies have increased the use of deep architectures in their studies, and today deep architectures are used widely in almost all fields.

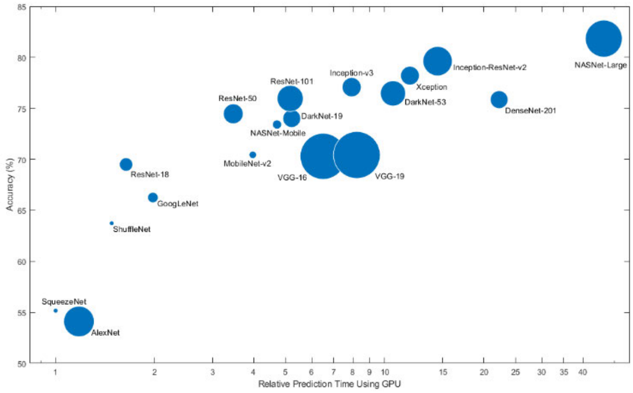


FIGURE 4. Pre-trained network performances of ImageNet accuracy over time [24].

The developed architectures have had different success rates in finding solutions for various problems in years. Therefore, it has become possible for the developers to adopt different architectures to their own problems. The process of taking a pre-trained architecture in this manner and training it with developer-specific data is called as transfer learning in the literature. Since transfer learning is built upon the success of the existing architecture, it is more likely to yield more successful results than an empty CNN architecture. Transfer learning process steps are shown in Figure 5 [25].

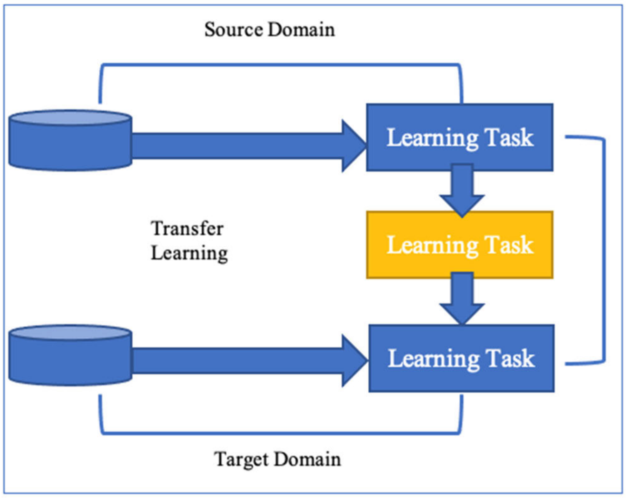


FIGURE 5. Transfer learning process.

Current major CNN models in the literature can be listed as follows: LeNet (1998) [19], AlexNet (2012) [23], GoogleNet (2014) [26], VGGNet (2014) [27], ResNet (2015) [28], Inception (2016) [29], NasNet (2017) [30] vb. When choosing a network to be applied to the problem, it is necessary to consider the different features of the pre-trained networks. The most important parameter in this preference is network accuracy, speed and size. Choosing a network often requires balancing these features.

A. CONVOLUTIONAL NEURAL NETWORK

A Convolutional Neural Network (CNN) is designed to process data that come in the form of multiple arrays such as a color image composed of 2D arrays containing pixel intensities in three color channels [22]. Figure 6 shows the structure of a typical CNN architecture. Layers included in standard CNN architectures are convolutional, non-linearity, pooling, flattening and fully-connected layers.

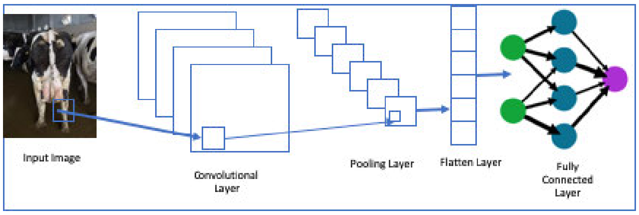


FIGURE 6. Typical CNN architecture [31].

Convolutional Layer: This layer generates feature maps by shifting matrices with different core sizes (feature detectors) in the input image. This core structure is smaller in size compared to the input image and is similar to the input image that requires parameters such as weight and bias sharing between adjacent pixels of the image [31].

Non-Linearity Layer: This layer is applied to prevent the system from linearity and various nonlinear activation functions such as logistic, tanh, Relu are used. The weighted sum of the linear net input value is passed through an activation function for nonlinear transformation [32].

Pooling Layer: This layer is used to combine data from the convolution layer. There are various types such as max, min and average pooling for data reduction. Depending on the type of the problem and the amount of data, the number and type of pooling can be changed [33].

Flattening Layer: It ensures that meaningful data obtained from the system is edited before entering a classifier. When data in 2D form is used as input data of the classifiers such as ANN, SVM, etc., they should be converted to a single output plane. This is the reason it is called as flattening [34].

Fully-Connected Layer: It is a standard machine learning layer. It can include various classification methods according to the structure and type of the network. It has the same output number as the problem classification number. It can include models such as artificial neural networks (ANN), support vector machines (SVM) [35].

B. VGG NET

VGGNet is a homogeneous architecture that takes its name from the developer group (visual geometry group-VGG) at Oxford University. According to the number of convolution layers, there are two types: VGG16 (41 layers, 16 of which are convolution layers) and VGG19 (47 layers, 19 of which are convolution layers). It accepts a 224×224×3 input image size and contains 3 × 3 core matrix. There are two fully connected layers with 4096 outputs. In the final layer, there is a fully connected layer with 1000 channels for 1000 classes in accordance with the ImageNet classification problem and SoftMax elements showing the suitability values of this layer. When performing transfer leaning, it is enough to make changes on this final layer [27]. An average of 73% success is achieved in the ImageNet classification using this architecture. The architecture of the model is shown in Figure 7.

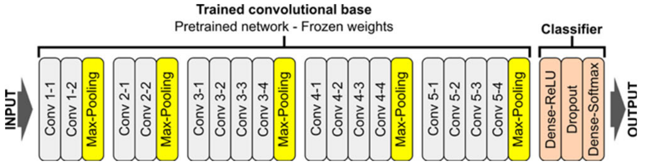


FIGURE 7. Transfer learning with VGG19 network architecture [36].

IV. EXPERIMENTAL STUDIES ON DEVELOPED APPLICATION

In the study, a mobile application was developed to determine and automate BCS values, which is the most important indicator of whether the animal's requirements are met in different periods (early, mid and late lactation periods and during dry period), in hybrid black pied dairy cattle. BCS estimation using real-time images on mobile devices was carried out with the help of CNN architectures, which is one of the deep learning models frequently used nowadays. In addition, pre-trained CNN architectures were used to increase the performance of the system and VGG19 network was preferred among the pre-trained networks [4], [16]. In the design of the system, a user-friendly application that could operate on mobile devices with Android operating system was designed to use the Python programming language and Tensorflow deep learning libraries and to test and implement the system.

In the first stage of the study, the dataset to be used for BCS estimation problem was prepared. The data of the images were evaluated by an expert zootechnician. The images of each of the dairy cattle (with black pied and hybrid cattles) were taken at different periods in 10 different dairy cattle farms in Nigde and Adana region from approximately 1.5-2 m distance. Therefore 505 different images in total were obtained which were obtained with a mobile device with resolutions ranging from 1024 × 768 to 3042 × 4032 pixels (The following image pre-processing techniques were automatically performed by TensorFlow: normalization, quantization, crop image, resize (224 × 224), rotate and basic filters). In this way, resolution limitation in the system was avoided and the system could operate at any resolution value. BCS scores of the animal were determined by the expert using these images. This data set was used in the training of CNN architectures. However, before entering the training, the images were pre-processed and the regions of interest (ROI) to be used in BCS estimation were cut. The data set obtained was divided into classes with a sensitivity of 1.0. Of the 505 images in total, 40 were Score 1, 161 were Score 2, 221 were Score 3, 63 were Score 4, and 20 were Score 5. The obtained images and their pre-processed versions are shown in Figure 8 for each score value.

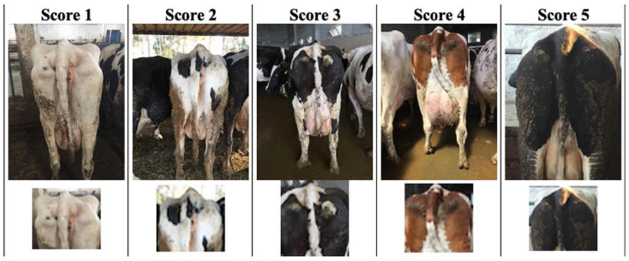


FIGURE 8. Data set and pre-processing.

In the second stage of the study, training of the data set is performed. In the pre-training process of the data set, it is seen that VGG19 network achieved higher success rate. Therefore, in the training of the data set, VGG19 network was used [4], [16]. The Python programming language and TensorFlow and Keras libraries were used to train the network since they are widely used in literature, have a successful infrastructure, and are easy to apply for mobile systems. Table 3 shows the VGG19 based CNN architecture used in the study.

TABLE 3. CNN Architecture used in the study.

Layer (type)	Output Shape	Parameter
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv4 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv4 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv4 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
global_average_pooling2d	(Gl (None, 512)	0
dense (Dense)	(None, 4096)	2101248
dense_1 (Dense)	(None, 5)	20485
Total parameters:	22,146,117	
Trainable parameters:	2,121,733	
Non-trainable parameters:	20,024,384	

The computer on which the data was trained has GTS450 GDDR5 1GB 128Bit Nvidia GeForce DX11 GPU card with Intel i7-2600 3.40 GHz processor, 18 GB RAM, running under 64 Bit Windows 10 operating system. The designed CNN was trained by running 200 steps with pre-processed data (505 original images were used in a data augmentation process to increase the image set to 2020 using such augmentation techniques as rotation, horizontal flip and vertical flip) and this process lasted 2 hours and 45 minutes in total (using these training parameters: optimizer = 'adam', loss = 'categorical_crossentropy', max_epochs = 200, metrics = 'accuracy', BATCH_SIZE = 32). The accuracy and loss values that occurred during the training phase are shown in Figure 9.

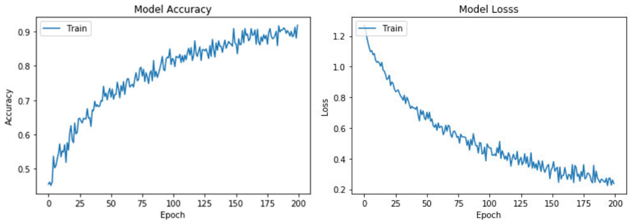


FIGURE 9. Change in loss and accuracy values that occurred during the training process.

Confusion matrix (CM) is the method widely used in the literature to evaluate the results of classification problems. CM includes the display of the number of similarities between the estimated values and the actual values in a table.

TABLE 4. Confusion Matrix [38].

		Real Values 0	 1	Total (%)
Prediction	0	TP	FP	Precision
	1	FN	TN	Negative Predictive Value
Total		Recall, Sensitivity	Specificity	Accuracy

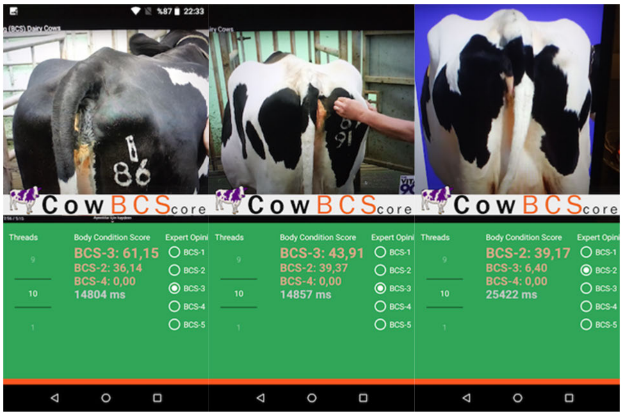
TABLE 5. Confusion matrix resulting from the training of the system.

Expert Values/ App. Values	Score 1	Score 2	Score 3	Score 4	Score 5	Success %
Score 1	**39**	0	1	0	0	97.50
Score 2	0	**150**	9	2	0	93.17
Score 3	0	6	**194**	21	0	87.78
Score 4	0	0	0	**63**	0	100.00
Score 5	0	0	1	0	**19**	95.00
Total	40/39	160/150	221/194	63/63	20/19	**94.69**

The values in the table are True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN). A standard CM is shown in Table 4 [37].

CM resulting from the training of the system with the help of VGG19 pre-trained network is shown in Table 5.

Real-time images were used in the test of the trained CNN. Considering the structure of animal husbandry enterprises, it is difficult to access real-time images with a computer. For this reason, in the study, mobile software was developed for real-time tests in the enterprise environment. The trained CNN was integrated into the software and a user-friendly design was realized. Sample screenshots of the developed mobile software are given in Figure 10.



CowBCScore
Threads: 9 / 10 / 1
Body Condition Score
BCS-3: 61,15
BCS-2: 36,14
BCS-4: 0,00
14804 ms
Expert Opini
○ BCS-1
○ BCS-2
● BCS-3
○ BCS-4
○ BCS-5

CowBCScore
Threads: 9 / 10 / 1
Body Condition Score
BCS-3: 43,91
BCS-2: 39,37
BCS-4: 0,00
14857 ms
Expert Opini
○ BCS-1
○ BCS-2
● BCS-3
○ BCS-4
○ BCS-5

CowBCScore
Threads: 9 / 10 / 1
Body Condition Score
BCS-2: 39,17
BCS-3: 6,40
BCS-4: 0,00
25422 ms
Expert Opini
○ BCS-1
● BCS-2
○ BCS-3
○ BCS-4
○ BCS-5

FIGURE 10. Screenshots of the designed mobile application.

The software designed for the tests was used in an enterprise by the expert zootechnician. The expert used the mobile application to score the animals in the enterprise and then recorded the score information based on his own expertise.

TABLE 6. Confusion matrix resulting from the test of the system.

Expert Values/ App. Values	Score 1	Score 2	Score 3	Score 4	Score 5	Success %
Score 1	**3**	0	2	0	0	60
Score 2	0	**7**	3	0	0	70
Score 3	0	0	**25**	0	0	100
Score 4	0	0	2	**8**	0	80
Score 5	0	0	1	0	**4**	80
Total	5/3	10/7	25/25	10/8	5/4	**78**

Of the 50 images in total, 5 were Score 1, 10 were Score 2, 25 were Score 3, 10 were Score 4, and 5 were Score 5. In Table 6, the confusion matrix obtained as a result of the expert evaluation and software evaluation in the test process is presented.

V. RESULTS AND DISCUSSION

When the literature is examined, it is seen that the researchers have conducted many studies for BCS estimation problem by using the images taken from the back, pins and rump of the cows and using different models. Classical image processing processes and methods were used between 2000 and 2015. However, deep neural networks have been used in the last five years. The researchers tried to increase the solution rate of the problem, especially with the cameras and imaging methods used in the image acquisition phase. However, the chance and rate of implementation of such studies has been very low. In addition, studies generally used the back and profile images of animals during imaging. Although this process requires taking images in a very controlled way, it is not very useful in practice. Therefore, these studies could not go beyond academic studies. In several studies conducted in recent years, BCS has been tried to be determined by using deep architectures with a single image taken from the back of the animal.

In this study, computer-aided software was developed to estimate the BS values using real-time images in order to reduce the error rate that may arise from the expert's interpretation in determining the BCS in dairy cattle. Considering the situation of animal husbandry enterprises, the software developed due to its easy use was designed in a form that could operate on mobile devices. The deep neural network architecture proposed in the study was designed using a pre-trained VGG19 network. Unlike other studies in the literature, the single image of the animal was taken from the back of the animal in real time and the score value was estimated through this image. In addition, thanks to mobile support, it is clear to adapt the study to the field applications.

In the study, firstly, animal images taken from different enterprises were classified into classes by expert zootechnician. 505 images in total were distinguished according to the class they belonged to in the range of 1-5. The regions of these images used by the expert to determine BCS were cropped. These regions were labeled according to the BCS value. Afterwards, transfer learning process was carried out in deep architectures and the data was trained in VGG19 network. At the end of the training, it was observed that the system provided classification with a success of 94.69%. The trained network was transferred to the mobile software designed for real-time testing of the system. For the tests of the study, the expert zootechnician conducted tests in the enterprises where there were animals that were not introduced to the system before using mobile software. As a result of the tests, 78% success rate was achieved. When the test results are examined, it is seen that Score 3 value is successfully distinguished in the tests. In addition, the results of misclassifications obtained in other scores also indicate Score 3. This result is one of the limitations of the system. The reason for this may be the number of data density according to the classes in the training data set. If the amount of data is increased for each class, it is possible that the classification success in the testing phase will also increase.

In addition, it has been determined that the user-friendly mobile-based software is easily used by the expert. With the widespread use of this system in the enterprises, it will be possible to provide good care and feeding conditions, take precautions against possible problems related to care and feeding; so that the health condition of the animal, milk performance, fertility rate, etc. will be increased, which will increase the profitability of the enterprise.

The results of combining deep architectures with mobile-based designs in the realization of BCS determination problem using real-time images are quite promising. In addition, the success of the system can be increased by changing the pre-trained CNN architectures used in the system. However, considering that CNN architecture will operate on a mobile device, the success of networks with more layers may be inversely proportional to the real-time operating performance of the system. In addition, the size of the training data that directly affects the classification performance of the system is important. In future studies, tests can be performed using different CNN architectures and different mobile platforms and data with more training clusters.

ACKNOWLEDGMENT

The author would like to thank to Dr. Mustafa Boğa for his contribution in classifying the images required for training the system and performing the tests of the system.

REFERENCES

[1] E. E. Wildman, G. M. Jones, P. E. Wagner, R. L. Boman, H. F. Troutt, and T. N. Lesch, "A dairy cow body condition scoring system and its relationship to selected production characteristics," *J. Dairy Sci.*, vol. 65, no. 3, pp. 495–501, Mar. 1982.

[2] M. Krukowski, "Automatic determination of body condition score of dairy cows from 3D images-processing and pattern recognition in images from a time-of-flight camera," M.S. thesis, Comput. Sci. Commun., KTH Royal Inst. Technol., Stockholm, Sweden, 2009.

[3] A. Galiç, "Determination of the body condition score and its relationship with milk yield in Turkish holstein cows," *J. Agricult. Sci.*, vol. 23, no. 4, pp. 464–471, 2016.

[4] K. K. Çevik and M. Boğa, ''Body condition score (BCS) segmentation and classification in dairy cows using R-CNN deep learning architecture,'' *Eur. J. Sci. Technol.*, vol. 17, pp. 1248–1255, Dec. 2019.

[5] W. Kellogg, *Body Condition Scoring With Dairy Cattle (Department of Agriculture, and County Governments Cooperating)*. Fayetteville, Arkansas: Univ. Arkansas, 2010, pp. 1–6.

[6] A. J. Heinrichs and V. A. Ishler. (2020). *Body Condition Scoring as a Tool for Dairy Herd Management*. Diary Cattle. Accessed: May 7, 2020. [Online]. Available: https://dairy-cattle.extension.org/body-condition-scoring-as-a-tool-for-dairy-herd-management/#Cows_at_Calving

[7] J. M. Bewley, A. M. Peacock, O. Lewis, R. E. Boyce, D. J. Roberts, M. P. Coffey, S. J. Kenyon, and M. M. Schutz, ''Potential for estimation of body condition scores in dairy cattle from digital images,'' *J. Dairy Sci.*, vol. 91, no. 9, pp. 3439–3453, Sep. 2008.

[8] G. Azzaro, M. Caccamo, J. D. Ferguson, S. Battiato, G. M. Farinella, G. C. Guarnera, G. Puglisi, R. Petriglieri, and G. Licitra, ''Objective estimation of body condition score by modeling cow body shape from digital images,'' *J. Dairy Sci.*, vol. 94, no. 4, pp. 2126–2137, Apr. 2011.

[9] D. Anglart, ''Automatic estimation of body weight and body condition score in dairy cows using 3D imaging technique,'' M.S. thesis, Dept. Animal Nutrition Manage., Agricult. Sci., Swedish Univ., 2014.

[10] I. Halachmi, M. Klopčič, P. Polak, D. J. Roberts, and J. M. Bewley, ''Automatic assessment of dairy cattle body condition score using thermal imaging,'' *Comput. Electron. Agricult.*, vol. 99, pp. 35–40, Nov. 2013.

[11] R. Spoliansky, Y. Edan, Y. Parmet, and I. Halachmi, ''Development of automatic body condition scoring using a low-cost 3-dimensional Kinect camera,'' *J. Dairy Sci.*, vol. 99, no. 9, pp. 7714–7725, Sep. 2016.

[12] J. R. Alvarez, M. Arroqui, P. Mangudo, J. Toloza, D. Jatip, J. M. Rodríguez, A. Teyseyre, C. Sanz, A. Zunino, C. Machado, and C. Mateos, ''Body condition estimation on cows from depth images using convolutional neural networks,'' *Comput. Electron. Agricult.*, vol. 155, pp. 12–22, Dec. 2018.

[13] J. R. Alvarez, M. Arroqui, P. Mangudo, J. Toloza, D. Jatip, J. Rodriguez, A. Teyseyre, C. Sanz, A. Zunino, C. Machado, and C. Mateos, ''Estimating body condition score in dairy cows from depth images using convolutional neural networks, transfer learning and model ensembling techniques,'' *Agronomy*, vol. 9, no. 2, p. 90, Feb. 2019.

[14] X. Huang, Z. Hu, X. Wang, X. Yang, J. Zhang, and D. Shi, ''An improved single shot multibox detector method applied in body condition score for dairy cows,'' *Animals*, vol. 9, no. 7, p. 470, Jul. 2019.

[15] X. Li, Z. Hu, X. Huang, T. Feng, X. Yang, and M. Li, ''Cow body condition score estimation with convolutional neural networks,'' in *Proc. IEEE 4th Int. Conf. Image, Vis. Comput. (ICIVC)*, Jul. 2019, pp. 433–437.

[16] K. K. Cevik and M. Boga, ''Body condition score (BCS) classification with deep learning,'' in *Proc. Innov. Intell. Syst. Appl. Conf. (ASYU)*, Oct. 2019, pp. 1–5.

[17] Y. LeCun, Y. Bengio, and G. Hinton, ''Deep learning,'' *Nature*, vol. 521, pp. 436–444, May 2015.

[18] A. G. Ivakhnenko and V. G. Lapa, ''Cybernetic predicting devices,'' School Elect. Eng., Purdue Univ., West Lafayette, IN, USA, 1966.

[19] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, L, ''Handwritten digit recognition with a back-propagation network,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 1990, pp. 396–404.

[20] A. Şeker, B. Diri, and H. H. Balık, ''A Review about deep learning methods and applications,'' *Gazi J. Eng. Sci.*, vol. 3, no. 3, pp. 47–64, 2017.

[21] A. El-Sawy, E.-B. Hazem, and M. Loey, ''CNN for handwritten Arabic digits recognition based on LeNet-5,'' in *Proc. Int. Conf. Adv. Intell. Syst. Inform.* Cham, Switzerland: Springer, 2016, pp. 566–575.

[22] G. E. Hinton, ''Learning multiple layers of representation,'' *Trends Cognit. Sci.*, vol. 11, no. 10, pp. 428–434, Oct. 2007.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[24] *Pretrained Deep Neural Networks*, Matworks, New Delhi, India, 2017.

[25] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, ''A survey on deep transfer learning,'' in *Artificial Neural Networks and Machine Learning—ICANN*. Cham, Switzerland: Springer, 2018, pp. 270–279.

[26] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, ''Going deeper with convolutions,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[27] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[28] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, ''Rethinking the inception architecture for computer vision,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[30] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, ''Learning transferable architectures for scalable image recognition,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8697–8710.

[31] P. Murugan, ''Feed forward and backward run in deep convolution neural network,'' 2017, *arXiv:1711.03278*. [Online]. Available: http://arxiv.org/abs/1711.03278

[32] F. Agostinelli, M. Hoffman, P. Sadowski, and P. Baldi, ''Learning activation functions to improve deep neural networks,'' 2014, *arXiv:1412.6830*. [Online]. Available: http://arxiv.org/abs/1412.6830

[33] L. Deng, O. Abdel-Hamid, and D. Yu, ''A deep convolutional neural network using heterogeneous pooling for trading acoustic invariance with phonetic confusion,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6669–6673.

[34] J. Jin, A. Dundar, and E. Culurciello, ''Flattened convolutional neural networks for feedforward acceleration,'' 2014, *arXiv:1412.5474*. [Online]. Available: http://arxiv.org/abs/1412.5474

[35] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, ''Convolutional, long short-term memory, fully connected deep neural networks,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584.

[36] J. Jaworek-Korjakowska, P. Kleczek, and M. Gorgon, ''Melanoma thickness prediction based on convolutional neural network with VGG-19 model transfer learning,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 2748–2756.

[37] W. Chen, Y. Li, W. Xue, H. Shahabi, S. Li, H. Hong, X. Wang, H. Bian, S. Zhang, B. Pradhan, and B. B. Ahmad, ''Modeling flood susceptibility using data-driven approaches of naïve bayes tree, alternating decision tree, and random forest methods,'' *Sci. Total Environ.*, vol. 701, Jan. 2020, Art. no. 134979.

[38] Ö. Çelik and A. F. Aslan, ''Gender prediction from social media comments with artificial intelligence,'' *Sakarya Univ. J. Sci.*, vol. 23, no. 6, pp. 1256–1264, 2019.

KERIM KÜRŞAT ÇEVIK received the bachelor's degree in electronic and computer sciences, in 2007, and the M.Sc. degree and the Ph.D. degree in computer engineering from Selcuk University, in 2010 and 2016, respectively. He started to work as a Lecturer at the Computer Programming Program, Nigde University, in 2009. At the end of his Ph.D. studies, he was appointed as an Assistant Professor in the same program. He currently works as an Assistant Professor with the Department of Management Information Systems, Akdeniz University. His research and teaching focus on artificial intelligence, image processing, biomedical, machine learning, computer-aided detection, medical informatics, data mining, mobile programming, and computer vision.