Computers and Electronics in Agriculture, Volume 173, June 2020, Article number 105386
DOI: 10.1016/j.compag.2020.105386

An adaptive pig face recognition approach using Convolutional Neural Networks

Mathieu Marsot[a], Jiangqiang Mei[b,a], Xiaocai Shan[c,a], Liyong Ye[e], Peng Feng[d], Xuejun Yan[e], Chenfan Li[e], Yifan Zhao[a,*]

[a]*School of Aerospace, Transport and Manufacturing, Cranfield University, UK, MK43 0AL*
[b]*School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin, China, 300222*
[c]*Key Laboratory of Petroleum Resources Research, Institute of Geology and Geophysics, Chinese Academy of Sciences, Beijing, China, 100029*
[d]*Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian, China*
[e]*DaHaoHeShan agriculture and animal husbandry technology Co., Ltd, Inner Mongolia, China*

Abstract

The evolution of agriculture towards intensive farming leads to an increasing demand for animal identification associated with high traceability, driven by the need for quality control and welfare management in agricultural animals. Automatic identification of individual animals is an important step to achieve individualised care in terms of disease detection and control, and improvement of the food quality. For example, as feeding patterns can differ amongst pigs in the same pen, even in homogenous groups, automatic registration shows the most potential when applied to an individual pig. In the EU for instance, this capability is required for certification purposes. Although the RFID technology has been gradually developed and widely applied for this task, chip implanting might still be time-consuming and costly for current practical applications. In this paper, a novel framework composed of computer vision algorithms, machine learning and deep learning techniques is proposed to offer a relatively low-cost and scalable solution of pig recognition. Firstly, pig faces and eyes are detected automatically by two Haar feature-based cascade classifiers and one shallow convolutional neural network to extra high-quality images. Secondly, face recognition is performed by employing a deep convolutional neural network. Additionally, class activation maps generated by grad-CAM and saliency maps are utilised to visually understand how the discriminating parameters have been learned by the neural network. By applying the proposed approach on 10 randomly selected pigs filmed in farm condition, the proposed method demonstrates the superior performance against the state-of-art method with an accuracy of 83 % over 320 testing images. The outcome of this study will facilitate the real-application of AI-based animal identification in swine production.

*Corresponding author:yifan.zhao@cranfield.ac.uk

Keywords: Face Recognition; CNN; Deep Learning; Face Detection; Machine Learning; Computer Vision

1. Introduction

The evolution of agriculture towards intensive farming has made farms more productive. The demand for animal identification and traceability is constantly increasing, driven by the need for quality control and welfare management in agricultural animals. For example, some Artificial Intelligence (AI) algorithms based on digital images or other sensing technologies (e.g. infrared cameras) can monitor the disease status of pigs, but they cannot track the sick pigs and link with their historic information (e.g. specie, behaviour, food and vaccine, etc.). Additionally, alterations in feeding patterns are considered to be one of the first warning signs of health, welfare and productivity problems in growing-finishing pigs [1]. Automatic registration of pigs' feeding patterns could support pig farms in their daily management routine [2]. As feeding patterns can differ amongst pigs in the same pen, even in homogenous groups, automatic registration shows the most potential when applied to an individual pig [3]. In the EU for instance, it is required for farmers to be able to identify their animals for certification purposes [4]. Therefore, some procedures have been developed to identify and control animals. The first idea was the use of plastic ear-tags or skin tattoos to give information on the origin of the food. Even though this is enough to comply with the law, it did not address a major issue of intensive farming: diseases outbreaks. As the pigs are now living in small spaces, diseases outbreaks can have disastrous consequences. For instance, the recent swine flu outbreak in China in 2018 caused a considerable loss for the farmers as an estimated number of 200 million pigs have been culled or killed. To perform more advanced monitoring, RFID (radio-frequency identification) chips have been widely used to replace the simple ear tags. It allows more advanced and automated monitoring but it is costly for farmers, particularly for large scale farms with thousands of pigs, because every pig needs its own RFID chip. Another problem of RFID is that metal parts and other electronic materials presented in farms can cause trouble to the antennas of the chips. Jarissa et al. [5] found that even with 2 chips per pig the identification of the animals at a close range has an accuracy of only 88.6%.

To overcome these limitations, as an alternative solution, the computer vision and AI based approaches start to attract interests, which has been used to automatically score pigs posture [6], recognise aggressive episodes of pigs [7] [8], estimate pig body components [9], predict tail-biting, fouling and diarrhoea in pigs [10], predict stress in piglets[11, 12], count pigs [13], track outdoor

animal [14], recognise feeding behavior [15], estimate pig weights from images [16], detect pigs in camera images [17], and measure pig body size [18]. For these approaches, only a few cameras are needed at specific places to identify and monitor the animals and the cost of the system has much less dependency on the number of pigs, which is especially attractive for large farms. In addition, cameras are cheap and non-contact leading to better animal welfare and are not perturbed by other electronic materials. Modern farms are often well illuminated which significantly helps the computer vision system as it facilitates every detection and recognition process.

In terms of improving animal welfare and increasing farming efficiency, the Internet of Things (IoT), edge computing, cloud computing and data-driven technologies have been attracted stock farming. Iwasaki et al. showed that IoT technology can make a breakthrough in livestock management by connecting biological information of livestock and environmental information obtained by IoT sensors to farmers who are in a remote location from the farm via the cloud [19]. Zamora-Izquierdo et al. proposed a smart farming IoT platform based on edge and cloud computing [20]. Treiber et al. discussed the connectivity for IoT and presented a solution that integrates sensor systems, the control of actuators and existing information systems on dairy farms into one central information- and control- system [21]. For animal behaviour analysis and health monitoring in a dairy farming scenario, Taneja et al. presented SmartHerd, a fog computing–assisted end-to-end IoT platform, and a fog computing assisted application system [22, 23]. Jukan et al. made a systematic review of smart computing and sensing technologies for animal welfare [24]. Although there is an increasing push of smart farm management solutions, the leverage of cutting-edge technologies, such as AI and Big Data to improve the productivity of stock farming is relatively slow in comparison with other sectors, such as healthcare, surveillance, and manufacturing. The importance of such AI-related research and development is underestimated and more efforts are demanded. Combining AI with the end-to-end Internet of Things (IoT), fog computing, and cloud computing will definitely further accelerate the development of smart farm management.

Human face recognition using computer vision approaches has been well developed and now has been applied in various applications [25, 26, 27]. However, there are limited researches on animal face recognition. Kumar et al. [28] developed a cattle face recognition system where PCA (principal component analysis), linear discriminant analysis and ICA (independent component analysis) are used as features and SVM (support vector machines) is used as the classifier. They also applied the histogram equalisation to enhance the input images but diverse illumination and rotation of the cattle faces were not addressed. Kumar and Singh [29] introduced a Fisherface-

like dog face recognition algorithm using an enhanced fisher linear discriminant analysis called fisher linear preserving projection (FLPP) and SVM for classification. Methods of contrast enhancement and noise removal were used to improve the results. However, the images used are all well captured with ideal illumination condition and proper face alignment, which is usually difficult to be achieved in real-world applications. Tu et al. [30] proposed a face recognition algorithm for huskies and pugs using CNNs (Convolutional Neural Networks). The algorithm firstly identifies the most likely breed of the dog with a pre-trained CNN model and then performs dog identification using an own CNN. This approach divides the population of animals before performing the classification, which could be attractive for animals with different races. Wada et al. [31] attempted to recognise pig face using Eigenfaces with a KNN (k-nearest neighbours) classifier. The method does not deal with the alignment of faces or any kind of perturbation and assumes that the pig face is oriented towards the camera. Hansen et al. [32] proposed a pig face recognition algorithm based on CNN. A visualisation tool was also used to confirm that the CNN benefits from facial features rather than background information. It seems that pigs with black marks can be recognised relatively easily and those without marks could be problematic. One limitation of this study is that they painted the pigs to create artificial features for recognition.

It is concluded that there are very limited researches on pig face recognition. Even for the published works in this topic, there is no consideration of the challenge of data capture in real-world applications, such as diverse background, illumination and alignment of pig faces. Considering the fact that the existing works select the training images manually and the high demand of scalability for large farms, this paper proposes an adaptive approach to automatically select high-quality training and testing data before applying a deep CNN for pig face recognition. This automation will be attractive for any viable applications as manual extraction requires too much labour and sometimes it is not feasible. Indeed, pigs are not behaving like humans in a photo booth and are not necessarily looking at the camera all the time. In addition, a data augmentation approach is proposed to improve the accuracy. It should be noted that the data used in this paper were captured from an industrial environment and there are no artificial marks on pigs.

2. Methods

The used data in this study consist of 30 randomly selected pigs. A normal smartphone was used to capture the pig faces from different directions when they were in the positioning bar for feeding. For each pig, a duration of 60 seconds with a sample rate of 30 frames per second

(FPS) was used with the HD spatial resolution (1980 x 1080 pixels). There are therefore 1,800 images available for each video of each pig. The data were labelled based on the ID of ear tags. It should be noted that the pigs were not always looking at the camera and exhibited various natural behaviours, such as mouth opening, and noise such as dirty on faces. In addition, the background is relatively complex, e.g. the appearance of piglets, and metal bars create shadows on the pigs. Automatic selection of high-quality images for training and testing is crucial for developing an adaptive pig recognition solution.

This paper proposes a novel framework of pig face recognition, as illustrated in Figure 1, which includes 8 steps. The details of each step are presented below.

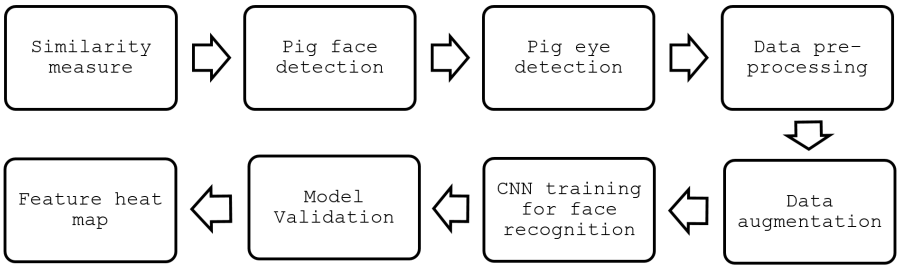

Figure 1: The proposed framework of the adaptive pig face recognition solution

2.1. Similarity measure for image filtering

Since the data were captured at 30 FPS, the structural similarity measure (SSIM) was firstly used to prevent identical frames from being selected for training. The similarity measure between two $N \times M$ images, I_1 and I_2 is given by :

$$SSIM(I_1, I_2) = \frac{(2\mu_1\mu_2 + c_1)(2\sigma_{1,2} + c_2)}{(\mu_1^2 + \mu_2^2 + c_1)(\sigma_1^2\sigma_2^2 + c_2)} \tag{1}$$

where c_1 and c_2 are two small constants to prevent division by zero, μ_1 and μ_2 are the means of the images, σ_1 and σ_2 are the standard deviation of the images, and $\sigma_{1,2}$ are the co-variance between these two images. In this paper, c_1 and c_2 were selected as 0.001 and the threshold of SSIM was chosen as 0.95. It should be noted that the color images were used for this step.

2.2. Pig face detection

Although CNN has been successfully attempted on pig face recognition [32], the pig faces were manually extracted which requires significant work if the number of samples is considerable. It is not an ideal solution for the automatic pig face recognition. A Haar Cascade classifier [27] was

proposed by Viola-Jones and had been trained to perform the pig face detection. This solution is chosen over colour segmentation because it can avoid detecting not only the pig ears/body but also piglets that can appear in the background. The grey-scale images were used for this step. A detailed process is stated below.

2.2.1. Data preparation

To train the classifier, two sample sets are required: negative and positive samples. Negative samples are any images that are not pig faces. To create the negative samples, random areas of the images were selected from the videos (see Figure 2). The size is at least 100 x 100 pixels. After the initial selection, the images containing a partial pig face were manually removed. In the end, a total of 2,110 negative images were extracted from the 30 videos for training.

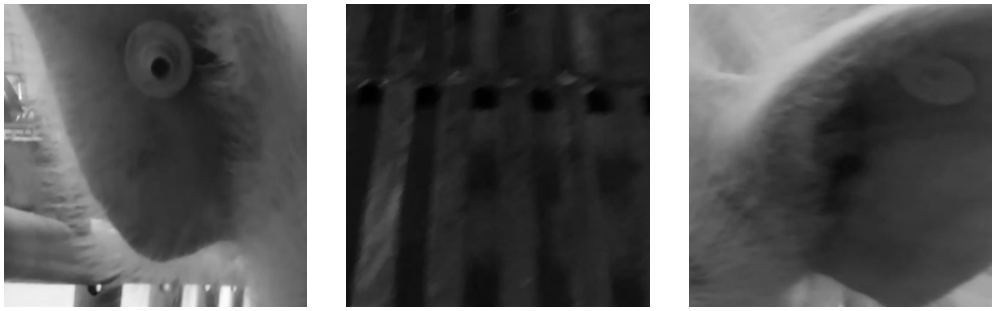

Figure 2: Examples of negative samples used to train the classifier for pig face detection

The positive samples were manually selected from randomly selected 17 videos, and the remaining 13 videos were used for the testing of pig face detection. The manually selected regions exclude the pig ears and focus on forehead, eyes and snouts (see Figure 3). In total, 564 positive samples were selected for training.

Figure 3: Examples of positive samples used to train the classifier for pig face detection

2.2.2. Classifier training

The classifier was trained using the OpenCV library. The library provides two programs to train a Haar cascade classifier. First, a file was generated containing the information on the positive samples via the *opencv_createsamples* function. This function takes an input file listing the filenames of all positive images. Each line of this file also contains information about one or more face regions in the image. In this paper, as the faces were previously selected and saved as independent images, each file is associated with only one region and the region is the entire image. The function also takes the window size as a parameter to specify the resolution of the algorithm, which defines the maximum size of the Haar features that are used. In this paper, a size of 20 x 20 pixels was used. The same process was applied to create a file for negative samples.

Once the sample files are generated, the classifier was trained using the *opencv_traincascade* function. The inputs of this function include a positive sample file, a negative sample file, a window size which has to be the same as the one used in the previous step (20x20) and the number of samples to be used for the training. Here 500 positive samples and 1000 negative samples were used. The remaining 64 positive samples and 1,110 negative samples were used for the validation purpose.

2.2.3. Classifier inference

The detection function provided by OpenCV, (*detectMultiScale*), adapts to different size of pig faces by resizing the input image multiple times before performing detection. However, it has been found that the detection works better when the features of pig face are not too small or too big compared to the resolution of the algorithm which here is 20x20 pixels. Therefore, before applying the function, images were resized to 100 x 100 pixels. This size was empirically chosen. Once the face is detected on the downsized image, the result is then re-scaled to fit the original image. Figure 4 shows the process of inference.

2.3. Eye detection

It is assumed that the feature of eyes is important for classification. Due to the way we captured data, it is inevitable to detect some pig faces where only one eye or even no eye is visible. As shown in Figure 5, the left image is what we are interested in while the right one is not interested in this study. This paper proposes to use a second Haar Cascade classifier [27] to detect eyes. It should be noted that the second classifier for eye detection generates more

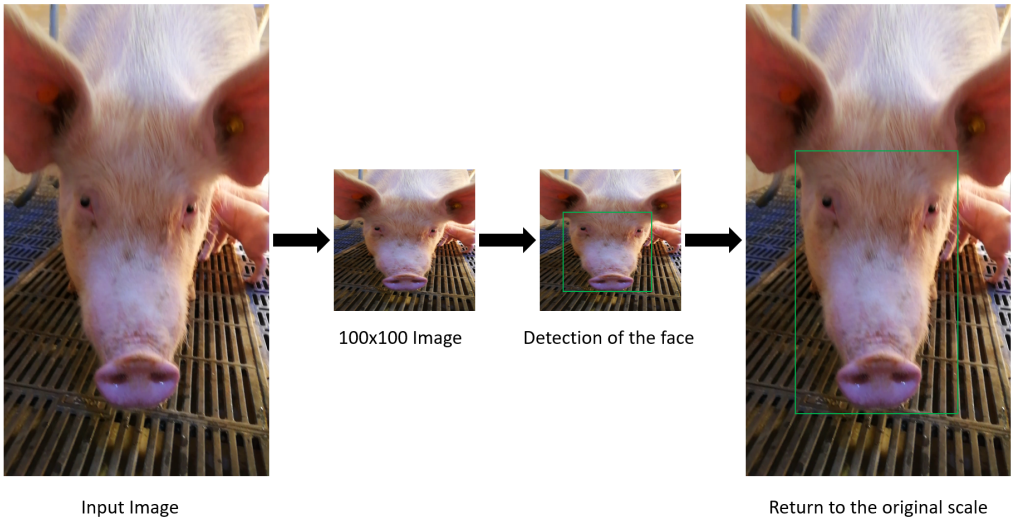

Figure 4: Steps of the pig face detection process: firstly the image is downsized to 100 x 100 pixels, then the classifier detects the face and finally the detected face is re-scaled back to the original image

false-positive than the first classifier for face detection. This is mainly caused by a much smaller area of the target, which can be easily confused by dirty on pig faces. Geometrical constraints and a shallow convolutional network are proposed to reduce the false positive.

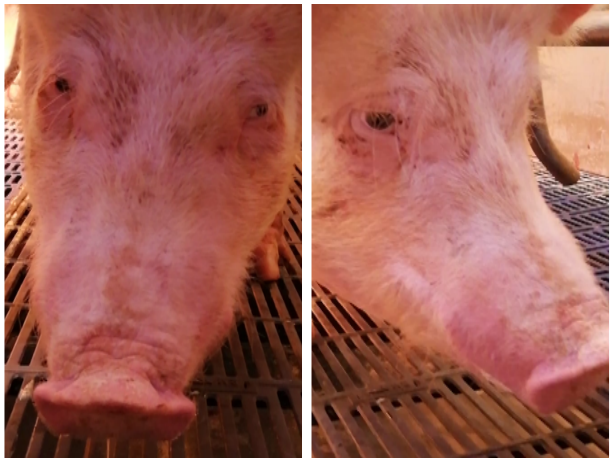

Figure 5: Two faces extracted by the face detection algorithm where the left image has two eyes visible and the right one has only one eye visible

165 The process of training is the same as the one presented for face detection. This time, 3,093 randomly selected negative samples and 546 manually selected positive samples were extracted

for training. A 10 x 10 pixels resolution was chosen because the eyes are smaller than the faces. For the inference, the images were downsized to 500 x 500 pixels rather than 100 x 100 pixels as the eyes are smaller. The second Haar Cascade classifier outputs a list of eye regions. However, unlike the face, the eye detector generates a lot of false positives. The detector classifies a lot of black dots either on the pig faces or on the background grid as eyes, as shown in Figure 6 where the red regions are false positives.

Figure 6: Examples of false positives of eye detection. The green rectangles indicate the true positives and the red ones are false positives. A vast majority of false positives comes from one of these 4 cases : background grid, mouth, snout or black spot on the face

However, even manually, it is almost impossible to differentiate black dots from eyes when only considering a small region around them. Therefore, to reduce the number of false positives, a geometric constraint is imposed on pairs of regions. Each pair of the candidate should be within a certain distance from each other and the line formed by the centre of two candidates should not be inclined by more than a certain angle. Mathematically, assuming there are N eyes detected, C_i is the center of the eye i, for each pair of (C_i, C_j) where $i \in \{1, N\}$, $j \in \{1, N\}$ and $i \neq j$, the pair is accepted if

$$150 < \|C_i\vec{C}_j\| < 500 \quad \text{and} \quad abs(arg(C_i\vec{C}_j)) < \frac{\pi}{4} \tag{2}$$

If $abs(arg(C_i\vec{C}_j)) > \frac{\pi}{2}$, the orientation of the vector is reversed so that $C_i\vec{C}_j$ always points to the right. It should be noted that the selection of 150 and 500 is subject to the image size.

To further reduce the number of false-positive, a shallow convolutional network that classifies false positives apart from true positives is used. The network takes the result from the classifier

as input and outputs a probability of being a true detection. If this probability is greater than 0.5 the candidate is considered as an eye otherwise it is rejected. This approach has the advantage of using RGB images as input and can therefore easily reject the false positives from the background. However, it is not enough for the ones due to black spot on the faces. To train this network, outputs from the Haar cascade eye detector are classified manually. Then the data was resized to 32x32 pixels RGB images and is fed to the network without any augmentation or pre-processing. The training dataset consists of 1143 negative samples and 1480 positive samples. The network is composed of two convolution layers and two fully connected layers for classification.

2.4. Face recognition using a deep CNN

To perform pig face recognition, the deep learning methods have been considered as they produced the best results in [32]. A total of 10 pigs (see Figure 7) out of the 30 were selected for training and these 10 pigs were filmed again in another day (30 days later) for the testing purpose. This is to better evaluate the provenance of the proposed method by considering the growth of pigs.

2.4.1. Data pre-processing

Before going through the network, the extracted images were first converted to grayscale to force the network to learn the face patterns rather than the colour. Although the colour could be used for classification, it will be affected by illumination and the parameter setting of the camera that records the video, which will limit the generalisation of the network. It should be noted that the colour information is used for face and eye detection.

The next step is applying the contrast limited adaptive histogram equalisation (CLAHE) [33] that performs local histogram equalisation of the image. To prevent the noise from being amplified in uniform areas, if there is a peak in the histogram, it is cut according to a preset threshold. This contrast enhancement aims to make facial patterns (e.g. feather) more visible (see Figure 8) and thus help the network better learn the features.

2.4.2. Data augmentation

To augment the number of training images and make the network more robust to certain changes, five operations were randomly performed on the training images, which include (1) shifted by at most 6 pixels, (2) rotated by at most 30°, (3) scaled up or down by 10%, (4) varied the global brightness within 20% of the mean of the image, and (5) adding random dark polygons

Figure 7: The ten pigs used for the classification process (images taken before face detection)

Figure 8: An example of image enhancement. Left: original face; Right: resulting face after CLAHE

on the pig faces to simulate random shadows. Examples of augmented images are given in Figure 9.

Figure 9: Examples of transformations performed to augment the data, all these images have the same original image and are showing a transformation, in practice the different transformations were mixed

2.4.3. Network hyper parameters

The network was trained using the categorical cross-entropy loss function. The metric used for network selection is the validation accuracy, meaning that the model with the best accuracy over the validation dataset. It is defined as the number of true classifications over the total number of samples in the validation set. The optimizer used for training is the Adadelta optimizer. For all the layers, ReLu activation is used except for the last dense layer where a softmax activation is performed.

2.4.4. Network structure selection

Hansen et al. [32] used a 64 x 64 pixels input image and alternated between 3x3 convolution layers and max-pooling layers for the feature extraction, followed by 3 fully connected layers for the classification. They also used dropout layers to prevent over-fitting after each max pooling and dense layer. In this study, however, there are less strong facial patterns on the pigs so the features are supposedly harder to extract. To take that into account, this paper proposes another structure with an input image dimension of 128 x 128 pixels allowing for 2 additional convolution layers and an additional pooling layer (see Figure 10).

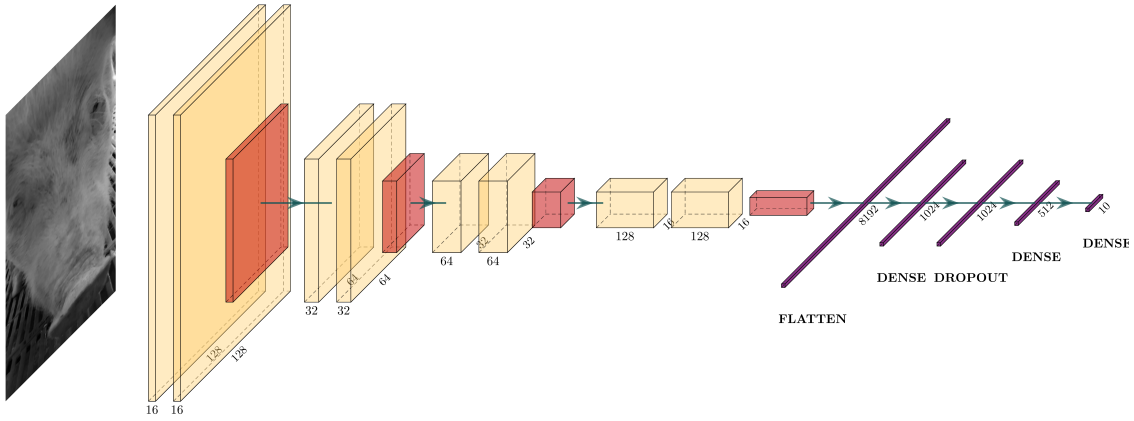

Figure 10: Structure of the network used for the classification where all convolution layers use 3x3 filters except the two first layers using filters of size 3x3, 5x5 or 7x7 depending on the tested structure.

To select the best structure, multiple combinations have been tested for the filter size of the two first layers. Each model structure has been trained for 5 times over 300 epochs and the maximum accuracy of the 5 models is used as a score for the structure selection (see Eq. (3)). The multiple model training is necessary to limit the influence of randomness in the training process allowing comparing structures more precisely. The network outputs a vector of 10 probabilities, each component of which corresponds to a certain pig. The vector with the maximum probability is chosen as the final result.

$$score(s) = max_{e\in[1,300],i\in[1,5]}(Acc_{e,i}(s)) \tag{3}$$

The training process was conducted in a laptop with an Intel i5-5265U CPU and Nvidia 1050M GPU (8GB memory). It took around 15 minutes to complete the training process.

2.4.5. Network evaluation

To evaluate the performance of the proposed network, the accuracy over the testing data is the main criteria. For each pig, 2 independent videos captured at different times were available. One was used for training and the other one was used for testing.

In addition, CAM and saliency maps are generated using the Keras-vis library [34] which implements the grad-CAM [35] and the saliency method presented in [36]. The study of CAM aims to investigate the region responsible for a decision, and it is, therefore, valuable to better understand how the network takes its decision. Although it can be hard to understand which

features the network learn, it is possible to observe if the network is learning from unwanted information like the background or other body parts of the pigs. It is also interesting to look at the saliency maps. Saliency maps are representing the derivative of the output relative to the input so it highlights the pixel for which a small change causes a big difference in the classification. It allows a more precise way to inspect which features are responsible for the decision. For these two methods, the last softmax activation was replaced by a linear activation function to limit vanishing gradients during the backpropagation. In addition, for the saliency maps, a modifier was applied to only look at positive values during the backpropagation to reduces the noise in the final image.

3. Results

3.1. Data extraction

To evaluate the performance of the poposed data extraction process, images from 10 videos have been tested. To prevent any biases, the videos used for testing are different from the ones used to train the face/eye classifiers. Geometry constraints and a shallow neural network were used to reduce false positive.

Table 1: Testing results of data extraction process

Extraction Methods	Face and Eyes Detection(FED)			FED with Geometry Constraints			FED with Geometry Constraints and Shallow CNN		
Pig Label	Positive Images	False Positive	False Positive Rate	Positive Images	False Positive	False Positive Rate	Positive Images	False Positive	False Positive Rate
52013	390	28	7.2	326	6	1.8	166	0	0
52986	260	31	12	190	0	0	44	0	0
53194	913	246	27	535	70	13	31	16	50
53322	479	242	50	214	11	5	51	0	0
53466	507	228	45	420	151	36	120	6	5
53468	407	136	33	116	3	2.5	46	0	0
53809	300	73	24	185	47	25	62	3	5
99842	429	149	43	248	90	36	84	1	1.2
99909	549	159	29	186	61	33	57	11	19
99939	263	13	5	124	6	0	5	0	0
Total Number	**4497**	1305	**29**	**2544**	439	**17**	**663**	37	**5.6**

As shown in Table 1, FED represents the process of face and eyes detection without any false positive removal, which extracts a total of 4497 images from the 10 videos which contain around 18,000 original frames. It suggests that almost 75% of raw data are with poor quality, inclusion of which for face recognition training will significantly reduce the performance. It also can be observed that 29% of 4497 selected images (1305 images) are false positives. Adding geometry constraints after FED achieves a significant improvement by reducing the false positive rate from 29% to 17%, while the total number of extracted images declined to 2544 that is almost half less than applying FED only. After applying the shallow neural network, the false positive rate

declined to 5.6% while the total number of images decreased to 663. This step has the advantage of adopting the RGB information and therefore is a very restrictive method which removes almost all the false positives.

As aforementioned, most of existing works select training images manually, our data extraction approach could fundamentally reveal the intrinsic feature of data while extremely reducing the reliance on manual selection. Since the false positive rate dramatically decreases while the number of positive images remains reasonable, the FED with geometry constraints has been utilized for data extraction in this paper. Although a false positive rate of 5.6% is attractive, the number of extracted images is not sufficient for this study. However, if the amount of data set is more than sufficient, the shallow convolutional neural network, which produces much lower false positive rate, is particularly recommended. Consequently, the proposed computer vision pipeline is appropriate for the automated pig face detection and essential for the training of classification neural network.

3.2. Classification accuracy

As mentioned above, 10 pigs were selected for evaluating the performance of the proposed pig face recognition method. For both training and testing images, the condition to be selected is that both pig eyes are visible. After filtering the images through similarity measure, face detection and eye detection with geometry constrain, a total of 2044 images from these 10 pigs were used for training and 320 images from another 10 videos for the same group of pigs were used for testing. Details of selected images are shown in Table 2. It should be noted that each pig has the same amount of testing images, but a slightly different amount of training images.

In this section, the resulting testing accuracy of two different architectures is presented. The first architecture is the one used by Hansen et al. [32]. Another architecture is the one proposed in Figure 10. For the new proposed network, multiple sizes for the first 2 convolution layers have been tested but only the results of the one using 7x7 filters are presented to make the graph more readable (3x3, 5x5, and 9x9 sizes have been tested too but they produced worse results). As shown in Figure 11.(a), the structure producing the best accuracy is the proposed 7x7 filters with 128x128 input images which has a maximum accuracy of 83.75% and an average accuracy of 76% over the epoch 20 to 300. The structure with 64x64 input images yields a maximum accuracy of 81.5% for an average of 74.4%. To present the comparison, Figure 11.(b) plots the average accuracy over batches of 10 epochs, which makes the curve smoother. It is clear that the 128x128 inputs with 7x7 filters outperform the method proposed by Hansen et al. [32].

Table 2: Number of images extracted for the 10 pigs for training and testing after data extracting

Pig Label	Class Number	Training Images	Testing Images
45717	0	204	32
47274	1	146	32
53194	2	221	32
53466	3	243	32
53322	4	215	32
53468	5	248	32
53809	6	172	32
99842	7	214	32
99909	8	208	32
99939	9	173	32
Total		2044	320

As expected, the model performance is not equivalent to each pig. Observed from the confusion matrix shown in Table 3, which is the average of 10 trials, some pigs are recognised almost perfectly with an accuracy of more than 90%, such as Class 0, 2, 5, 8 and 9, while some pigs yield relatively low accuracy of less than 70%, such as 4 and 7. There are multiple potential reasons leading to this observation, such as the number of training data, growth of pigs, dirt or food on pig face, change of illumination etc. The testing accuracy is expected to increase if more data are trained by considering these factors.

3.3. Where the network learns from

To have a better understanding of how the network works, class activation maps for 10 successfully classified images from the testing dataset have been generated (see the 2nd row of Figure 12) using the grad-CAM technique. Observation from the maps shows that the activation of most pigs comes from the faces indicating that the network does not learn from the background. The only exception is the pig 6 where its leg also contributes to the decision. This error is probably due to the training data. Indeed, a lot of legs appears in a similar position in multiple images, so the network may have learned from them too. These class activation maps should, however, be considered carefully. The fact that the pig 9 has no activation pixels does not mean that the

Table 3: The averaged confusion matrix

Class	0	1	2	3	4	5	6	7	8	9
0	0.92	0.00	0.00	0.00	0.03	0.04	0.00	0.01	0.00	0.01
1	0.03	0.76	0.01	0.00	0.10	0.02	0.00	0.01	0.04	0.02
2	0.03	0.01	0.96	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.04	0.00	0.00	0.89	0.02	0.01	0.02	0.02	0.01	0.00
4	0.07	0.02	0.09	0.00	0.60	0.05	0.02	0.08	0.06	0.00
5	0.03	0.00	0.00	0.00	0.00	0.94	0.00	0.00	0.00	0.02
6	0.03	0.00	0.00	0.02	0.11	0.00	0.80	0.01	0.02	0.00
7	0.20	0.01	0.03	0.00	0.02	0.00	0.01	0.66	0.04	0.02
8	0.03	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.90	0.00
9	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.94

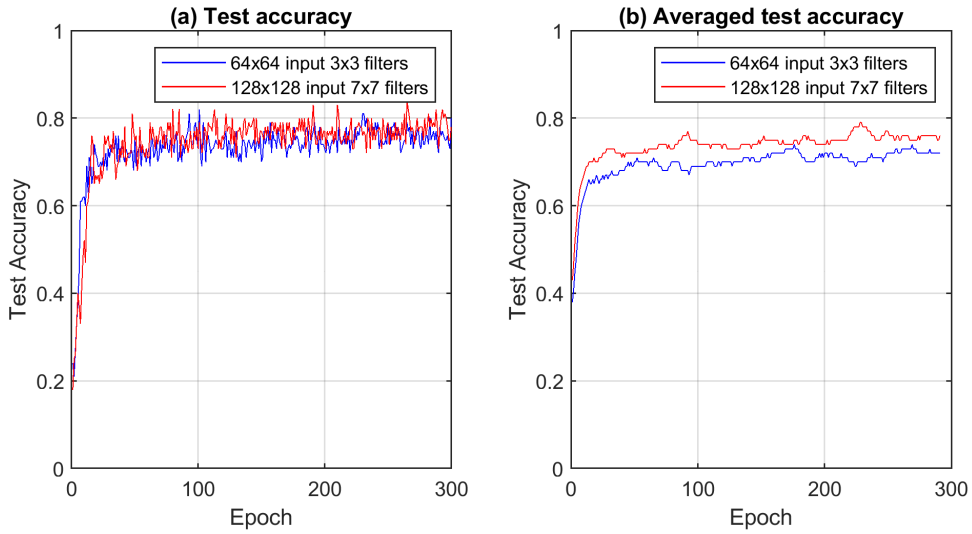

Figure 11: Comparison of the test accuracy between the proposed network structure (red plots) and the one proposed by Hansen et al. [32] (blue plots)

decision has been taken randomly. It is probably a consequence of the gradient becoming too small during the back-propagation.

Another way of visualising important pixels to the decision process is saliency maps. Unlike CAM, saliency does not use the information of the last convolution layer but backpropagates the gradient all the way to the input to find the pixels responsible for the decision with more

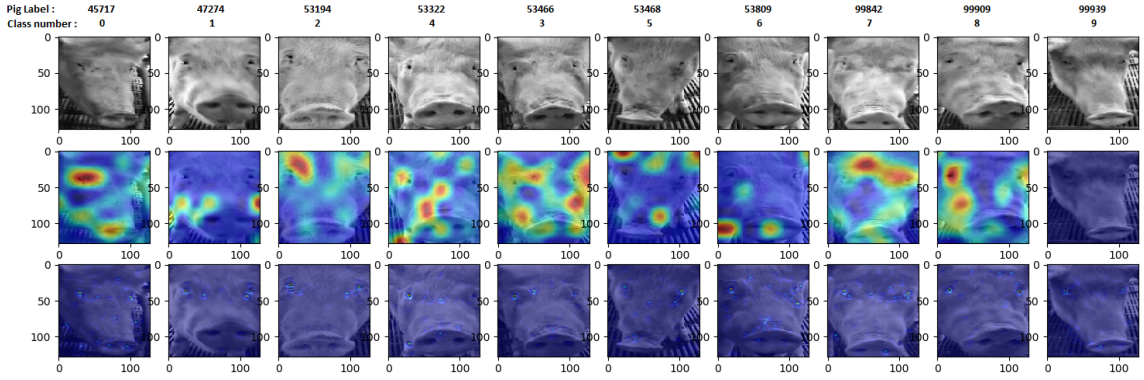

Figure 12: Visualisation of where the network learns from for the 10 tested pigs. 1st row: raw images; 2nd row: class activation maps; 3rd row: saliency maps

precision. As shown in the 3rd row of Figure 12, it seems that the most important area for the classification are eyes. It is not surprising because the data extraction process ensures that both eyes are always visible, giving the network reliable information to learn from. In addition to the eyes, some other patterns also appear important in the decision like black dots on the face. There is no evidence that the network learns from the background.

3.4. Influence of image pre-processing

In this section, the benefits of using gray-scale images over colour images as well as the influence of CLAHE on the accuracy are presented. It has been found that the learning of RGB images leads to unsatisfying results, as shown in Figure 13.(a), where CLAHE was applied. The accuracy oscillates around 40% which is around 35% less than the results using gray-scale images. This is probably because the network has to focus on patterns rather than colours to mitigate the influence of scene illumination and the sensor. In addition, as all the pigs have a similar skin colour, the network will seek small colour variations which means it can easily confuse a pig from another with slightly different testing conditions. In a real-life application, if a lot of videos under various illumination or camera setups are available for training for each pig, the network may not overfit the colour information and RGB images may be a viable solution too but here, it is clearly not an efficient solution.

In Figure 13.(b), the testing accuracy for a structure trained with and without CLAHE is presented, where the gray-scale images were used. It clearly shows that the histogram equalization improves the performances by a few percentages and also makes the convergence of the network faster. It is not surprising as it tends to exacerbate the patterns on pig faces, therefore

making them easier for the network to learn from.

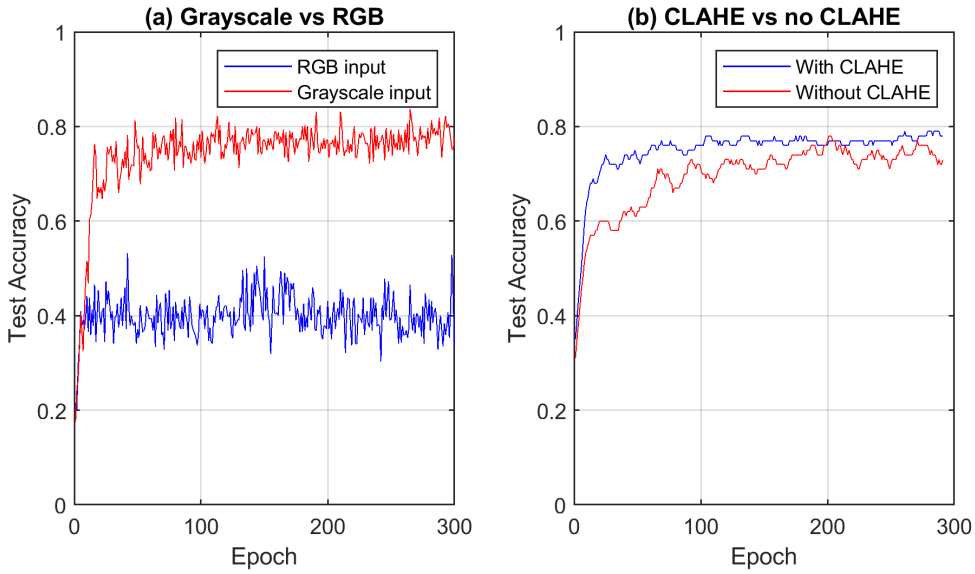

Figure 13: Comparison of the test accuracy between (a) RGB input vs gray-scale input, (b) with CLAHE and without CLAHE

4. Discussion and future work

It should be noted that image/video size is relatively large than data from other sensors, which could be a bottleneck if the internet is required for data transform, particularly for the farms located in regions with low internet connectivity. Our vision is that the proposal pig face recognition will be conducted using edge computing to improve response times and save bandwidth. The processed results (e.g. pig ID) instead of raw videos will be transferred to a centralised server using a local area network using cables or 4G networks (even 5G in future). Such an infrastructure is usually already available for modern farms for the surveillance purpose. All extracted information will be stored and correlated in the server. It should be noted that there is no need to store most of the footage unless abnormal events detected and tracking the raw footage is required. Furthermore, it is our notion that one camera can do multiple things. For example, apart from face recognition, the footage can also be used to analyse the pig behaviour (e.g. movement) which could detect the sick pigs and abnormal events such as sow squash piglets. More and more functions can be added in the edge computing with almost no extra hardware investment. Therefore, including digital cameras in the future swine farm is attractive due to

their advantages on scalability, configurability and extendibility.

The paper proposes a proof-of-concept of using computer vision and AI to recognise pigs based on facial information. To ensure the system to be accepted in the real applications, in the short-term future plan we will focus on a) significantly increasing the number of test pigs to validate the reliability of pig face recognition; b) developing a dedicated data capture system along with the capability of edge computing; c) optimising the number and location of cameras; and d) validating the system through real-time testings in a swine farm. In the long-term future plan, we will develop more functionalities of pig monitoring powered by AI and then integrate them into edge computing. Additionally, fusion with other IoT sensors would be our next consideration to not only improve the performance of pig recognition but also aim to create a fully connected digital network for smart farms.

5. Conclusions

We have proposed an automated pig face recognition framework that achieves a new state of the art result on images captured under farm condition. The presented framework integrates computer vision algorithms and deep convolutional neural network to balance the complicated design rules of traditional feature extraction and recognition strategies. In particular, the pig faces and eyes are extracted by Haar feature-based cascade classifiers whilst face recognition is performed by employing a model trained by categorical cross-entropy loss function and guided by Adadelta optimizer. Through the study of saliency and activation maps, we have highlighted that the neural network benefits from interesting features like eyes and specific marks on pig face but that it can be sensitive to parasite patterns caused by dirt or food.

The differences between this work and other state-of-the-art pig face recognition work are

- There is no artefact mark on pig faces.
- The good quality images for training and testing were selected automatically to increase the scalablity of the proposed solution
- Testing data were captured 1 month later than the training data where the pig growth and uncertainty of farm condition are considered.

It can be concluded from the testing results that

1) The image filtering step is essential to automate pig face recognition. Significant portions of raw data (about 75% in this study) could be poor quality for training.

2) We have demonstrated that our framework for pig face recognition is, by less training iterations, more accuracy than the achievements obtained from a state-of-the-art reference method.

3) The number of training images and the number of false positive must be balanced to achieve the best recognition performance.

4) The image pre-processing step including the conversion to gray-scale and image enhancement is essential to improve the recognition performance including accuracy and convergence speed.

To our best knowledge, this is also the first framework not only delivers comparatively effects but also effectively provides a practical solution to address the challenge of pig face capture and recognition in farm condition.

References

[1] D. M. Weary, J. M. Huzzey, M. A. G. von Keyserlingk, BOARD-INVITED REVIEW: Using behavior to predict and identify ill health in animals1, Journal of Animal Science 87 (2) (2009) 770–777 (02 2009). arXiv:https://academic.oup.com/jas/article-pdf/87/2/770/23640404/770.pdf, doi:10.2527/jas.2008-1297.
URL https://doi.org/10.2527/jas.2008-1297

[2] C. Wathes, H. Kristensen, J.-M. Aerts, D. Berckmans, Is precision livestock farming an engineer's daydream or nightmare, an animal's friend or foe, and a farmer's panacea or pitfall?, Computers and Electronics in Agriculture 64 (1) (2008) 2 – 10, smart Sensors in precision livestock farming (2008). doi:https://doi.org/10.1016/j.compag.2008.05.005.
URL http://www.sciencedirect.com/science/article/pii/S0168169908001476

[3] T. Brown-Brandl, G. Rohrer, R. Eigenberg, Analysis of feeding behavior of group housed growing–finishing pigs, Computers and Electronics in Agriculture 96 (2013) 246 – 252 (2013). doi:https://doi.org/10.1016/j.compag.2013.06.002.
URL http://www.sciencedirect.com/science/article/pii/S0168169913001324

[4] S. Ammendrup, A. E. Fussel, Legislative requirements for the identification and traceability of farm animals within the european union, Revue Scientifique et Technique-Office International des Epizooties 20 (2) (2001) 437–462 (2001).

[5] M. Jarissa, S. Wouter, D. K. Bart, M. Kristof, V. Jürgen, H. E. F, M. Sam, V. N. Annelies, Validation of a high frequency radio frequency identification (hf rfid) system for registering feeding patterns of growing-finishing pigs, Computers and Electronics in Agriculture 102 (2014) 10–18 (2014).

[6] A. Nasirahmadi, B. Sturm, A.-C. Olsson, K.-H. Jeppsson, S. Müller, S. Edwards, O. Hensel, Automatic scoring of lateral and sternal lying posture in grouped pigs using image processing and support vector machine, Computers and electronics in agriculture 156 (2019) 475–481 (2019).

[7] C. Chen, W. Zhu, D. Liu, J. Steibel, J. Siegford, K. Wurtz, J. Han, T. Norton, Detection of aggressive behaviours in pigs using a realsence depth sensor, Computers and Electronics in Agriculture 166 (2019) 105003 (2019).

[8] C. Chen, W. Zhu, J. Steibel, J. Siegford, K. Wurtz, J. Han, T. Norton, Recognition of aggressive episodes of pigs based on convolutional neural network and long short-term memory, Computers and Electronics in Agriculture 169 (2020) 105166 (2020).

[9] C. Shi, J. Zhang, G. Teng, Mobile measuring system based on labview for pig body components estimation in a large-scale farm, Computers and electronics in agriculture 156 (2019) 399–405 (2019).

[10] Y. Domun, L. J. Pedersen, D. White, O. Adeyemi, T. Norton, Learning patterns from time-series data to discriminate predictions of tail-biting, fouling and diarrhoea in pigs, Computers and Electronics in Agriculture 163 (2019) 104878 (2019).

[11] J. P. da Silva, I. de Alencar Nääs, J. M. Abe, A. F. da Silva Cordeiro, Classification of piglet (sus scrofa) stress conditions using vocalization pattern and applying paraconsistent logic eτ, Computers and Electronics in Agriculture 166 (2019) 105020 (2019).

[12] F. N. da Fonseca, J. M. Abe, I. de Alencar Nääs, A. F. da Silva Cordeiro, F. V. do Amaral, H. C. Ungaro, Automatic prediction of stress in piglets (sus scrofa) using infrared skin temperature, Computers and Electronics in Agriculture 168 (2020) 105148 (2020).

[13] M. Tian, H. Guo, H. Chen, Q. Wang, C. Long, Y. Ma, Automated pig counting using deep learning, Computers and Electronics in Agriculture 163 (2019) 104840 (2019).

[14] M. Bonneau, J.-A. Vayssade, W. Troupe, R. Arquet, Outdoor animal tracking combining neural network and time-lapse cameras, Computers and Electronics in Agriculture 168 (2020) 105150 (2020).

[15] Q. Yang, D. Xiao, S. Lin, Feeding behavior recognition for group-housed pigs with the faster r-cnn, Computers and electronics in agriculture 155 (2018) 453–460 (2018).

[16] K. Jun, S. J. Kim, H. W. Ji, Estimating pig weights from images without constraint on posture and illumination, Computers and electronics in agriculture 153 (2018) 169–176 (2018).

[17] J. Brünger, I. Traulsen, R. Koch, Model-based detection of pigs in images under sub-optimal conditions, Computers and electronics in agriculture 152 (2018) 59–63 (2018).

[18] K. Wang, H. Guo, Q. Ma, W. Su, L. Chen, D. Zhu, A portable and automatic xtion-based measurement system for pig body size, Computers and electronics in agriculture 148 (2018) 291–298 (2018).

[19] W. Iwasaki, N. Morita, M. P. B. Nagata, Iot sensors for smart livestock management, in: K. Mitsubayashi, O. Niwa, Y. Ueno (Eds.), Chemical, Gas, and Biosensors for Internet of Things and Related Applications, Elsevier, 2019, pp. 207 – 221 (2019). doi:https://doi.org/10.1016/B978-0-12-815409-0.00015-2.
URL http://www.sciencedirect.com/science/article/pii/B9780128154090000152

[20] M. A. Zamora-Izquierdo, J. Santa, J. A. Martínez, V. Martínez, A. F. Skarmeta, Smart farming iot platform based on edge and cloud computing, Biosystems Engineering 177 (2019) 4 – 17, intelligent Systems for Environmental Applications (2019). doi:https://doi.org/10.1016/j.biosystemseng.2018.10.014.
URL http://www.sciencedirect.com/science/article/pii/S1537511018301211

[21] M. Treiber, M. Höhendinger, H. Rupp, N. Schlereth, J. Bauerdick, H. Bernhardt, Connectivity for iot solutions in integrated dairy farming in germany, 2019 (07 2019). doi:10.13031/aim.201900561.

[22] M. Taneja, J. Byabazaire, A. Davy, C. Olariu, Fog assisted application support for animal behaviour analysis and health monitoring in dairy farming, 2018, pp. 819–824 (02 2018). doi:10.1109/WF-IoT.2018.8355141.

[23] M. Taneja, N. Jalodia, J. Byabazaire, A. Davy, C. Olariu, Smartherd management: A microservices-based fog computing–assisted iot platform towards data-driven smart dairy farming, Software: Practice and Experience 49 (7) (2019) 1055–1078 (2019). arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/spe.2704, doi:10.1002/spe.2704.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.2704

[24] A. Jukan, X. Masip-Bruin, N. Amla, Smart computing and sensing technologies for animal welfare: A systematic review, ACM Comput. Surv. 50 (1) (Apr. 2017). doi:10.1145/3041960.
URL https://doi.org/10.1145/3041960

[25] H. A. Rowley, Neural network-based face detection, Tech. rep., CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE (1999).

[26] M. Abdel-Mottaleb, A. Elgammal, Face detection in complex environments from color images, in: Proceedings 1999 International Conference on Image Processing (Cat 99CH36348), Vol. 3, IEEE, 1999, pp. 622–626 (1999).

[27] P. Viola, M. Jones, et al., Rapid object detection using a boosted cascade of simple features, CVPR (1) 1 (511-518) (2001) 3 (2001).

[28] K. Santosh, T. Shrikant, S. S. Kumar, Face recognition of cattle: can it be done?, Proceedings of the National Academy of Sciences, India Section A: Physical Sciences 86 (2) (2016) 137–148 (2016).

[29] K. Santosh, S. S. Kumar, Monitoring of pet animal in smart cities using animal biometrics, Future Generation Computer Systems 83 (2018) 553–563 (2018).

[30] T. Xinyuan, L. Kenneth, Y. Svetlana, Transfer learning on convolutional neural networks for dog identification, in: 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), IEEE, 2018, pp. 357–360 (2018).

[31] N. W. M. S. M. Shiraishi, Pig face recognition using eigenspace method, ITE Transactions on Media Technology and Applications (2013).

[32] M. F. Hansen, M. L. Smith, L. N. Smith, M. G. Salter, E. M. Baxter, M. Farish, B. Grieve, Towards on-farm pig face recognition using convolutional neural networks, Computers in Industry 98 (2018) 145–152 (2018).

[33] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, K. Zuiderveld, Adaptive histogram equalization and its variations, Computer Vision, Graphics, and Image Processing 39 (3) (1987) 355 – 368 (1987).

[34] R. Kotikalapudi, contributors, keras-vis, `https://github.com/raghakot/keras-vis` (2017).

[35] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626 (2017).

[36] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps, arXiv preprint arXiv:1312.6034 (2013).