HEALTH DETECTION ON CATTLE COMPRESSED IMAGES IN PRECISION LIVESTOCK FARMING

Sebastián Tapias Gómez
Universidad Eafit
Colombia
stapiasg1@eafit.edu.co

Valeria Cardona Velásquez
Universidad Eafit
Colombia
vcardonav@eafit.edu.co

Simón Marín
Universidad Eafit
Colombia
smaring1@eafit.edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

ABSTRACT

The constant population growth brings the needing to make up for food also grows at the same rate. The livestock provides one-third of human's protein base as meat and milk.[1] To improve cattle's health and welfare the pastoral farming employs Precision Livestock farming (PLF). This technique implementation brings a challenge to minimize energy consumption due to farmers not having enough energy or devices to transmit large volumes of information at the size are received from their farm's monitors. Therefore, in this project, we will design an algorithm to compress and decompress images reducing energy consumption.

Keywords

Compression algorithms, machine learning, deep learning, precision livestock farming, animal health.

1. INTRODUCTION

In 2015 almost 1.7 million cattle were lost to nonpredator causes. Respiratory problems accounted for the highest percentage of losses (23.9 percent), followed by unknown nonpredator causes (14.0 percent) and old age (11.8 percent). [2]. Some of these deaths can be prevented with accurate and real-time information about the behavior, health, and movements of cattle. This issue is one of the multiple reasons why emerged the Precision Livestock Farming which implements information and communication technology. It optimizes in different ways the cattle rearing, offering continuous and automated information about the animals to detect health issues at an early stage or even improve cattle's welfare, livestock economy, sustainability, and others.[3]

Monitoring cattle's health through an algorithm which uses pictures of the animals to distinguish healthy from sick cattle is one of the best ways to improve efficiency related to a better farming process. Due to the number of animals in these farms, such algorithm would consume lots of energy. A way to reduce this consumption is by compressing the pictures taken to be then processed.

1.1. Problem

The problem we are facing is the number of images and the energy consumption it takes to transmit data in the farm to devices or cloud to get process and at that point works with precision livestock farming. Consequently, we will design an algorithm that compress and decompress images with the minimum amount of energy consumption possible. Hence, the algorithm will be evaluated through a classification animal health algorithm with the goal to reach the maximum compression rate with the classification accuracy affected below 5%.

1.2 Solution

In this work, we used a convolutional neural network to classify animal health, in cattle, in the context of precision livestock farming (PLF). A common problem in PLF is that networking infrastructure is very limited, thus data compression is required.

To solve the problem, we first developed a lossy compression algorithm to reduce the size and quality of the images by a high percentage to reduce the processing cost. This algorithm was created by us based on the nearest-neighbor image scaling for its speed in reducing the size of the data and its effectiveness when compressed images are required for a classification algorithm, this implies saving memory and time. The loss of information in this procedure will be the minimum possible in order not to lose effectiveness in the convolutional processing network. After this compression, we will use a lossless algorithm, which will further compress the image without losing information and saving memory space and time to work with the images in the convolutional classification network.

1.3 Article structure

In what follows, in Section 2, we present related work to the problem. Later, in Section 3, we present the data sets and methods used in this research. In Section 4, we present the algorithm design. After, in Section 5, we present the results. Finally, in Section 6, we discuss the results and we propose some future work directions.

2. RELATED WORK

In what follows, we explain four related works on the domain of animal-health classification and image compression in the context of PLF.

2.1 A systematic literature review on the use of machine learning in precision livestock.

In this article, the authors focus on getting all information related to the recent use of machine learning in PLF and what are the techniques to collect and process data, which is fundamental on taking better decisions to drive farming into a healthier (not only for cattle but for the environment) and more productive industry. With all collected and analyzed data, the article supports some of the best ways to gain advantage from machine learning (ML) to use it in PLF [4]. Due to the nature of this work, such thing as an algorithm wasn't used to solve the problem of gathering and choosing the better uses of ML in PLF.

2.2 An Animal Welfare Platform for Extensive Livestock Production Systems.

Recently, the USA reformed its agricultural policy to support livestock well-being. Keeping up with new technologies and EU requirements, the authors introduce a recent development of wireless sensors, a collar device designed at a low cost. It records different parameters of animal's well-being as movement, geolocalization, and animal behavior in livestock farms. Since the battery life was a challenge, the device works in two radio frequencies, the first one can transmit real-time data in a short-range and fast, but it takes more energy consumption. The second one is slower, but its range to transmit data information is wider and, its energy consumption is lower than the first, also, works offline. The system developed can process pattern recognition through Deep Neural Network algorithms. In addition, it uses cloud computing for both data and Deep Learning model storage and usable and effective visualizations in mobile devices that provide farmers with valuable information.[5]

2.3 Cloud services integration for farm animal's behavior studies based on smartphones as an activity sensor.

The authors studied the iPhone sensors for tracking livestock and the challenges sensors presented. First, the data size must be reduced to minimize energy consumption. Next, the storage and processing of a large amount of data. Finally, matching multiple data from different sources.

They developed a data storage architecture able to share, treat, collect, and process from other platforms. Also, they used a UDP (User Datagram Protocol) protocol on WIFI to transmit data, but this could lead to data loss problems if a large amount of data is sending at the same time. The data compression was performed in two ways; eliminating redundancies, that is replacement of redundant data by a time interval during which the value remains constant was applied to preserve data integrity, and truncating data to 3, 4, and 5 decimal digits. [6]

2.4 Visual Localisation and Individual Identification of Holstein Friesian Cattle via Deep Learning

Since holstein fresian cattle are the highest milk-yielding bovine type [7] and many countries have taken rule against the traditional ear-tagging; identification and traceability look like a new challenge for farms and industry. Under this context, and aiming to a sharp technology solution, a localization using cameras and algorithms to identify these fresian bovine seems to be one of the best ways to keep tracking the cattle.

After taking the photos of the cattle from a UAV, some neural networks such as LSTM, R-CNN, Inception V3 are used to count the correct type of bovines. [8]

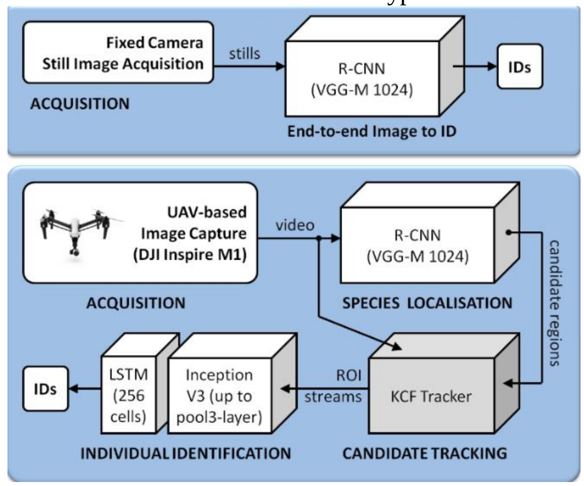

Image 1. Visual localisation and individual identification of holstein friesian cattle via deep learning.

3. MATERIALS AND METHODS

In this section, we explain how the data was collected and processed and, after, different image-compression algorithm alternatives to solve improve animal-health classification.

3.1 Data Collection and Processing

We collected data from Google Images and Bing Images divided into two groups: healthy cattle and sick cattle. For healthy cattle, the search string was "cow". For sick cattle, the search string was "cow + sick".

In the next step, both groups of images were transformed into grayscale using Python OpenCV and they were transformed into Comma Separated Values (CSV) files. It was found out that the datasets were balanced.

The dataset was divided into 70% for training and 30% for testing. Datasets are available at https://github.com/mauriciotoro/ST0245-Eafit/tree/master/proyecto/datasets .

Finally, using the training data set, we trained a convolutional neural network for binary image-classification using Google Teachable Machine available at https://teachablemachine.withgoogle.com/train/image.

3.2 Lossy Image-compression alternatives

In what follows, we present different algorithms used to compress images.

3.2.1 Seam Carving

Seam carving algorithm creates seams from images to later eliminate those which have less information (those which have less energy on the energy map created from the picture). Eliminating some of the less important paths on the image, it resizes and becomes lighter. [9]

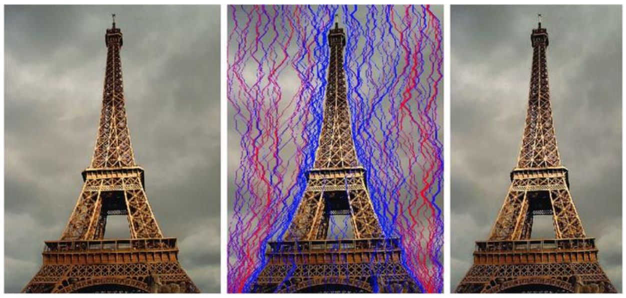

Image 2. Seam Carving

3.2.2 Discrete cosine transform or DCT

The DCT works by separating images into parts of differing frequencies. During a step called quantization, where part of compression actually occurs, the less important frequencies are discarded, hence the use of the term “lossy”. Then, only the most important frequencies that remain are used retrieve the image in the decompression process. [10]

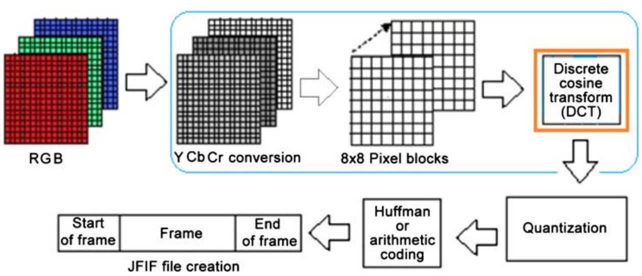

Image 3. Discrete Cosine Transform

3.2.3 Fractal compression

Fractal compression divides an image into sub-blocks. Then, it searches similarities in each sub-block. If found, replace a region with another similar one. This basically storages less information since a single block is used in two of them. [11]

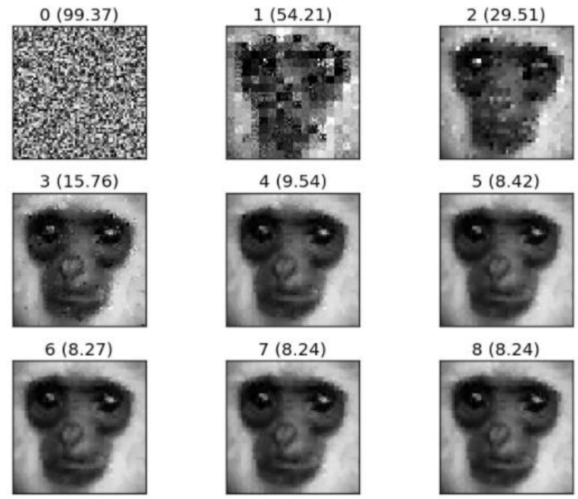

Image 4. Fractal Compression (Decompression)

3.2.4 Wavelet compression

This algorithm is a variant of DCT. First, a wavelet transform is applied to the image. This produces as many coefficients as there are pixels in the image. These coefficients can then be compressed more easily because the information is statistically concentrated in just a few coefficients. After that, the coefficients are quantized, and the quantized values are entropy encoded and/or run length encoded. [12]

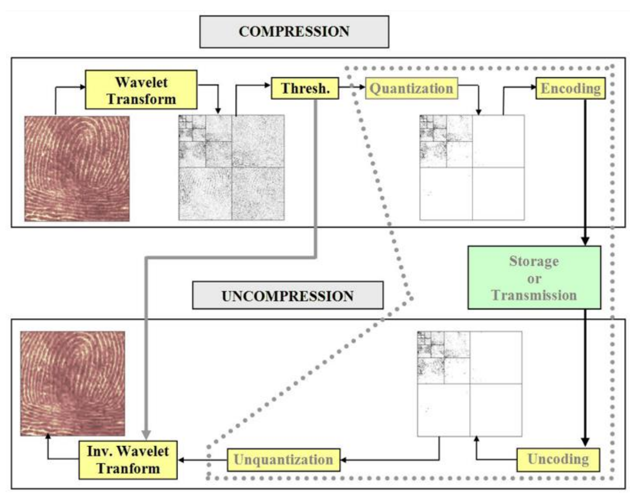

Image 5. Wavelet compression

3.3 Lossless Image-compression alternatives

In what follows, we present different algorithms used to compress images.

3.3.1 Burrows- Wheeler transformation

The burrows-Wheeler (BWT) permutes the positions of a string of characters to a string of repeated characters. Its data structure is the strings. The algorithm is useful because is easy to compress repeated character strings by techniques as move-to-front transform and run-length encoding. Moreover, BWT is reversible. It means the encoded string can be reconstructed to the initial word which was transformed. [13]

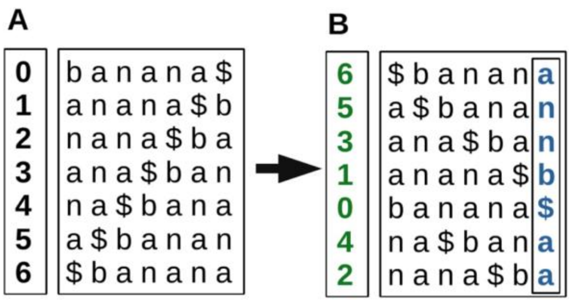

Image 6. Burrows- Wheeler transformation example

3.3.2 Huffman coding

The Huffman coding is a type of prefix code that uses character frequencies to assign a code to a variable. The most frequent character gets the smallest code, and the least frequent character gets the largest code [14]. The algorithm starts with the two minimum frequency variables and sums their frequency. Also, it is assigned an internal leaf code of 0 or 1 determined by its position. Then, the sum done becomes the frequency of a tree node. Next, the two following minimum frequencies are added, it could be the node we built before or different variables and, we continue building a tree until the tree only has one root.

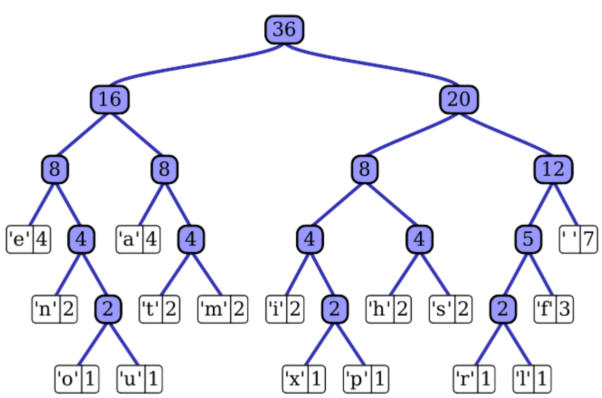

Image 7. Huffman coding

3.3.3 LZ77

Also called search buffer. It is a dictionary-based compression scheme. Codewords for substrings are pointers to the longest match for the substring found in the search buffer. When a matching appears, built a token formed by the steps to go back codeword for the substring, the length of the match, and the character of the next symbol. In software, the PNG file format is based on LZ77. In situations where a pattern repeat over a period larger than the search buffer size, the repetition cannot be taken advantage [15]

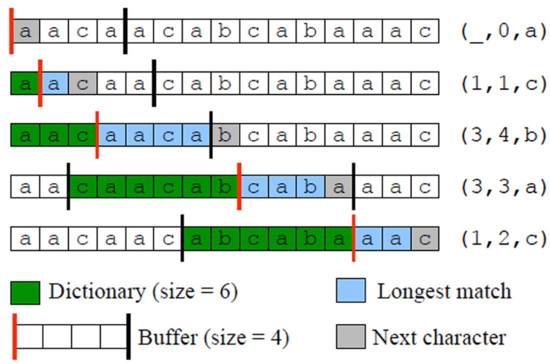

Image 8. Application LZ77 Algorithm

3.3.4 LZW

This algorithm is an improvement of the LZ78 algorithm. It is based on single-string character dictionary coders and is used to compress GIF images. It will maintain a running variable-length dictionary of symbols and trying to optimize for the longest match possible. As the message grows, however, the compression ratio tends asymptotically to the maximum [16]. It works by reading the first character string and check if it is in the dictionary. If not, it's added. Then check the second one and, also join the two first characters and add this joining as an extension of the dictionary to compress. The algorithm decompression works with the initial dictionary that contains a single-string character.

Current Char	Next Char	CurrentChar + NextChar is in the dictionary ?	Output Index	[New Index] New String
"B"	"A"	No	66	[256]"BA"
"A"	"B"	No	65	[257]"AB"
"B"	"A"	Yes	-	-
"BA"	"A"	No	256	[258]"BAA"
"A"	"B"	Yes	-	-
"AB"	"A"	No	257	[259]"ABA"
"A"	"A"	No	65	[260]"AA"
"A"	"A"	Yes	-	-
"AA"	"A"	No	260	[261]"AAA"
"A"	EOF	-	65	-

Image 9. LZW Example

4. ALGORITHM DESIGN AND IMPLEMENTATION

In what follows, we explain the data structures and the algorithms used in this work. The implementations of the data structures and algorithms are available at Github[1].

4.1 Data Structures

To read and store the csv data from the files, they are stored in Data frames and then in arrays to work with them in a simple way without losing information with this structure.

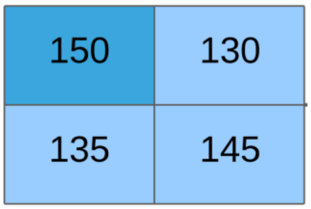

Figure 1: Subdivision of the pixel matrix after being resized.

4.2 Algorithms

In this work, we propose a compression algorithm which is a combination of a lossy image-compression algorithm and a lossless image-compression algorithm. We also explain how decompression for the proposed algorithm works.

For the compression of the images, we will first use a lossy compression algorithm to reduce the size of the images even if some information is lost; for this step we use nearest neighbor image scaling. Then we will proceed to use a lossy compression algorithm to further reduce the compression efficiency of the images and the time efficiency for the classification network through which the images will pass. The second algorithm to use has not yet been defined exactly which one to use.

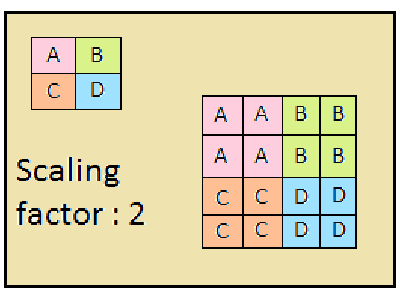

Image 10. Scaling Nearest- Neighbor Example

4.2.1 Lossy image-compression algorithm

For the Lossy image- compression we first redefine the size of the data frame, making the number of rows and columns even to make subdivisions of 2x2 matrixes.

From each subdivision we select the first element, i.e. the element in the upper left corner and move on to the next subdivision. After going through the whole matrix of subdivisions and selecting the element indicated above, the new matrix is formed with each selected pixel, thus creating the new compressed image. Its weight and size reduction is considerable, remaining half of both values compared to the original image.

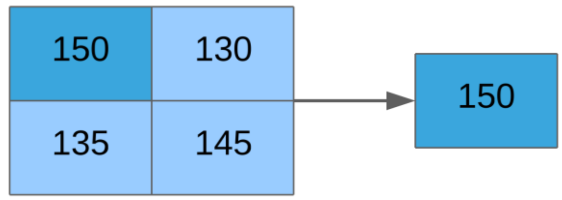

Figure 2: Image scaling using Nearest- neighbor method.

REFERENCES

1. Suryawanshi, K., Redpath, S., Bhatnagar, Y., Ramakrishnan, U., Chaturvedi, V., Smout, S., Mishra, C., 2017. Impact of wild prey availability on livestock predation by snow leopards. Roy. Soc. Open Sci. 4 (6).
2. National Agricultural Statistics Service (NASS) and the Animal and Plant Health Inspection Service (APHIS). 2017. Death Loss in U.S. Cattle and Calves Due to Predator and Nonpredator Causes, 2015.
3. Schillings, J. et al. 2021. Exploring the Potential of Precision Livestock Farming Technologies to Help Address Farm Animal Welfare. Frontiers in Animal Science. 2, (2021).
4. García, R. et al. 2020. A systematic literature review on the use of machine learning in precision livestock farming. Computers and Electronics in Agriculture. 179, (2020), 105826.
5. Doulgerakis, V., Kalyvas, D., Bocaj, E., Giannousis, C., Feidakis, M., Laliotis, G.P., Patrikakis, C., Bizelis, I., 2019. An animal welfare platform for extensive livestock production systems. In: CEUR Workshop Proceedings, vol. 2492.
6. Debauche, O., Mahmoudi, S., Andriamandroso, A.L.H., Manneback, P., Bindelle, J., Lebeau, F., 2019. Cloud services integration for farm animals' behavior studies based on smartphones as activity sensors. J. Ambient Intell. Humanized Comput. 10 (12), 4651–4662.
7. M. Tadesse and T. Dessie. Milk production performance of zebu, holstein friesian and their crosses in ethiopia. Livestock Research for Rural Development, 15(26), 2003.

https://github.com/Sebastian-Tapias/ST0245-001/tree/master/proyecto

8. Andrew, W., Greatwood, C., Burghardt, T., 2017. Visual localisation and individual identification of holstein friesian cattle via deep learning. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 2850–2859.
9. Avidan, Shai; Shamir, Ariel., 2007. Seam carving for content-aware image resizing: ACM SIGGRAPH 2007 papers.
10. Ken, C. Peter, G. Image compression and the discrete cosine transform.
11. Northwestern. Digital Image Processing - Open Project: Fractal image compression. Retrieved from https://users.cs.northwestern.edu/~agupta/_projects/image_processing/web/FractalImageCompression/
12. Chui, Charles K., 1992. An Introduction to Wavelets, San Diego, CA: Academic Press.
13. Burrows M., Wheeler D. 1994. A block- sorting lossless data compression algorithm. Algorithm, Data Compression (1994), 2-12
14. Huffman, D. 1952. A Method for the Construction of Minimum-Redundancy Codes. Proceedings of the IRE. 40, 9 (1952), 1098-1101.
15. J. Ziv, A. Lempel, A universal algorithm for sequential data compression, IEEE Transactions on Information Theory 23 (3)(1977) 337–343.y
16. Ziv, J. and Lempel, A. 1978. Compression of individual sequences via variable-rate coding. IEEE Transactions on Information Theory. 24, 5 (1978), 530-536

IMAGES REFERENCES

1. Andrew, W., Greatwood, C., Burghardt, T., 2017. Visual localisation and individual identification of holstein friesian cattle via deep learning. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 2850–2859.
2. ResearchGate https://www.researchgate.net/figure/Removing-inner-parts-of-an-image-based-on-the-seam-carving-technique-original-image_fig6_237007558
3. Nasser, A., Shin-Dug, K., 2014. "Adaptive discrete cosine transform-based image compression method on a heterogeneous system platform using Open Computing Language," *Journal of Electronic Imaging* 23(6), 061110
4. Pierre. 2018. Retrieved from https://pvigier.github.io/2018/05/14/fractal-image-compression.html
5. MathWorks. Wavelet Compression for Images. Retrieved from https://www.mathworks.com/help/wavelet/ug/wavelet-compression-for-images.html
6. Anderson, T. and Wheeler, T. 2021. An optimized FM-index library for nucleotide and amino acid search.). (2021)
7. By Meteficha - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=2875155
8. Shi, P. et al. 2018. A knowledge-embedded lossless image compressing method for high-throughput corrosion experiment. International Journal of Distributed Sensor Networks. 14, 1 (2018), 155014771775037.
9. A little Piece of Me https://philipstel.wordpress.com/2010/08/04/dictionary-based-algorithm-lempel-ziv-welch/
10. RAJAKARUNA, *Nearest- Neighbor Example*