University of Modena and Reggio Emilia
XXXIV cycle of the International Doctorate School in
Information and Communication Technologies
Doctor of Philosophy dissertation in
Computer Engineering and Science

Machine Learning applications deployment for future farming and animal health

Ercole Del Negro

University of Modena and Reggio Emilia
XXXIV cycle of the International Doctorate School in
Information and Communication Technologies
Doctor of Philosophy dissertation in
Computer Engineering and Science

Supervisor: Prof. Simone Calderara

PhD Course Coordinator: Prof. Sonia Bergamaschi
Modena, 2022

Review committee:

Giuseppe Marrucchella, University of Teramo

Valerio Morfino, DXC Technology

To me and to all

Abstract

Farm4trade is an Italian startup that develops high-tech systems for the breeding of production animals and for animal health in general. Among the applications developed there are computer vision systems that take advantage of machine learning. Among these the most important are Photo Animal Identification (PHAID) and Automatic Detection of Abattoir Lesions (ADAL).
In the case of these computer vision systems, the business challenge is to release into production to maximize application use to the company clients and to obtain predictions and overall results in real time. These systems must also seamlessly integrate with all business solutions and deliver information to third-party systems. The research work led to making important decisions with respect to the deploy technique of the systems that aimed to the business needs for efficiency and scalability. For the deployment, an MLOps approach was introduced for the integration of Machine Lerning models with CI / CD and DevOps technologies and the IT stack of corporate web solutions. The release of Machine Learning systems according to the model used was found to be efficient and responsive to business needs. All the models make inference in real time and are capable of transmitting information through the API to corporate systems and those of third parties.
We expect to improve the MLOps model through the development of web solutions that are increasingly aimed at integration with the ML model that the various applications require.

Abstract (italian)

Farm4trade è una startup italiana che sviluppa sistemi di alta tecnologia per gli allevamenti di animali di produzione e per la salute animale in genere. Tra le applicazioni sviluppate ci sono sistemi di computer vision che sfruttano il machine learning. Tra questi i più importanti sono Photo Animal Identification (PHAID) e Automatic Detection of Abattoir Lesions (ADAL).
Nel caso dei questi sistemi di computer vision, la sfida aziendale è quella di rilasciare in produzione per un utilizzo massimo delle applicazioni per ottenere predizioni e in generale risultati in tempo reale. Questi sistemi devono inoltre perfettamente integrarsi con tutte le soluzioni aziendali e fornire informazioni a sistemi terzi. Il lavoro di ricerca ha portato a prendere decisioni importanti rispetto alla tecnica di rilascio dei sistemi che puntassero al massimo verso i bisogni aziendali di efficienza e scalabilità. Per il deployment è stato introdotto un approccio MLOps che prevede l'integrazione dei modelli di Machine Lerning con i sistemi di CI/CD e DevOps aziendali e lo stack tecnologico delle soluzioni web aziendali. Il rilascio dei sistemi di Machine Learning secondo il modello adoperato si è rilevato efficiente e rispondente ai bisogni aziendali. Ci si aspetta di migliorare il modello di MLOps attraverso lo sviluppo di soluzioni web sempre più mirate all'integrazione con il modello di ML che i vari applicativi richiedono. Tutti i modelli effettuano inferenza in realtime e sono capaci di trasmettere attraverso le API le informazioni ai sistemi aziendali e a quelli di terze parti.

Contents

Chapter 1 Introduction

This thesis is the result of an Industrial PhD funded by Farm4trade (F4T) and made thanks to a fruitful collaboration among public and private sector. In fact, all the contents of this work is a result of the fully integration of AImage lab of the University of Modena and Reggio Emilia (UNIMORE) and Farm4trade team (lead by the PhD candidate and author of this thesis). The project that is the pillar of the entire work, the PhAID application, is fully supported by the "Istituto Zooprofilatico Sperimentale dell'Abruzzo e del Molise, G. Caporale" (IZS).

The challenge of the F4T is to integrate the Machine Learning solutions with the web app solution and the all the suite infrastructure. At the time when this study started, very few examples of ML model deployment were circulating in the industry and high tech companies and F4T was searching the best way to integrate both ML and web. The needs of the new model deployment for the company came out soon when the F4T started the collaboration with IZS for the PhAID system development that we used as case study to develop this work. Thanks to the evolution of the technologies, F4T reached the objective to deploy and integrate the ML solutions in the web. In few years the ML has evolved from purely an area of academic research to an applied field [0]. Still we have a challenge to transform an academic domain such is the ML to the industrial environment, but thanks to the introduction of MLOps (ML+DEVops), a new branch of the deployment phase where the DEVops approach is applied to the ML, we have now a definition of the deployment of the ML in production and in the Industrial sectors.

1.1 Thesis Statement

The ultimate aim of this work is to find a solution to deploy the ML projects of the company in the web and integrate is with the suite products of the company. The research activities started in parallel with the PhAID development that is the core of this thesis describing all steps made for the product development and deployment ready for the IZS and the integration with other applications. In the work the steps to develop and deploy the system are fully described demonstrating the complexity and all objective reached in the IZS-F4T project. In order to understand all the process to develop the deployment of the ML model, all steps for the PhAID development have been integrated on the work as fundamental part to understand the needs of a Model of ML application.

Chapter 2 Farm4trade

In this chapter the Farm4trade company is described.

2.1 F4T Company profile

F4T is a livestock health, traceability and data solutions company founded in 2016 in Italy. F4T applies Artificial Intelligence and other new technologies to precision livestock farming with the aim to improve animal health and welfare, productivity, food quality and traceability. Farm4trade mission is to become the livestock production chain data reference company, by capturing and leveraging full-circle data from farm to abattoir. The idea is to bring the latest mobile technology even in the most remote regions of the world to innovate animal farming and trading systems towards a more sustainable and equitable development of the livestock sector, while preserving the environment. Farm4trade aims at introducing automation along the entire value chain, from raising animals to trading and slaughtering. Farm4trade believes that improving livestock industry production practices will guarantee food security as well as animal and people health and welfare. Fram4trade's flagship product is the Fram4trade Suite, a set of fully integrated synchronized cloud-based applications, whereby multiple stakeholders can manage data across the entire supply chain. In 2017 Farm4trade began working on R&D projects in the fields of Machine Learning (ML) and Computer Vision intending to make significant improvements in the livestock sector for identification and traceability, monitoring health and productivity, as well as developing geo-referenced big-data analysis technologies. The first research project, made in cooperation with the "University of Modena and Reggio Emilia" (UNIMORE) and is AImagelab, entitled "**PhAID**", was aimed at the development of a contactless biometric recognition system for cattle identification (Bergamini *et al.* 2018) and it is currently being piloted for the Italian cattle registry in cooperation with IZS. Another ongoing R&D activity is the **"ADAL" (Automatic Detection of Abattoir Lesions)** developed in cooperation with the University of Teramo (UNITE). This is the first AI system capable of automating the image acquisition process and the scoring of lesions on the organs of slaughtered animals (Trachtman *et al.* 2020).

2.2 F4T Business Strategy

All the Farm4trade products are characterized by a scalable business model, based on the needs of different stakeholders in the agribusiness sector.

Suite of Apps - Farm4trade offers a Free subscription with basic access to most of the apps included, a Pro Plan billed monthly or yearly, and an Enterprise custom plan. By purchasing a Pro Plan, the users get customizable services, add-ons and full integration with third party applications. Both Free and Pro users can access tailored added value services and both can purchase pay per use services outside their plan.

ADAL - The ADAL revenue model is based on fixed system installation costs in each abattoir plus pay-per-unit for each carcass analyzed. The ADAL system price includes a complete reporting system, which fully integrates with our Business Intelligence Application and data analysis platforms. This model undertakes an initial investment for each company and provides large abattoirs the option to offer their clients access to ADAL reporting systems and become resellers of the product they have purchased.

PHAID - The animals' photographic identification system will be distributed as a stand-alone solution or an additional functionality of the Farm4trade's Suite of Apps. The former option is oriented to fill the national governments and institutions' needs on traceability regulation; while the second is aimed at offering a complete management tool for different stakeholders and registry systems in use.

2.3 F4T Market & Positioning

The market size of the Suite of Apps and PhAID can be considered the total number of cattle bred at global level that is close to 1.5 billion. Only 6% of them are bred within the EU-27 with 122 million units and 3,3 million farms. Instead, 324 million is the number of cattle bred in Africa, almost three times the number of animals bred in Europe. Farm4trade for its Suite App and PhAID is targeting the following market: Italy in Europe and Namibia, South Africa and Botswana in Africa. Other markets of interest are the USA, Canada, Australia and New Zealand. Breeders have been divided into 3 clusters: small-scale farmers, commercial farmers and stud and professional farmers. The Commercial Farmers are considered the market segment with the closest features to Farm4trade's ideal client, both because Farm4trade's products best meet their needs and because they have the ability to afford the purchase.

Concerning ADAL, Farm4trade's commercialization strategy will prioritize the key stakeholders, that were identified with the "European Largest Abattoirs", located in the 6 European countries with the highest number of pigs reared (Spain with 29 million pigs, followed by Germany 27M, Netherlands 12.4M, France 12.3M, Poland 11.3M and Italy with 8.5 M), slaughtering on average above 300.000 pigs per year and representing around 60% of the total pig production of each country.

Currently, the App Suite main competitors mostly serve only one phase/aspect of the value chain. Most of them are not focused on providing the outputs that could maximize sustainable production and animal health through the whole supply chain. Many of them offer more expensive and less comprehensive solutions for all the farmers' needs. Companies that develop data keeping applications, without using advanced solutions to improve livestock management, identification and traceability are: CattleMax, Farm Brite, Agritec, Herdmaster, Agriwebb, Bovisync, Tambero, Digital Beef and many others.

Projects similar to PhAID, although in limited numbers, are being developed but they don't have accuracy levels that can serve government registries and they are far from becoming commercially available to the farmers. ADAL, instead, does not have any direct competitors, and it strides ahead of existing solutions and any regulatory standards.

Machine Learning related competitors: Cainthus has a machine vision software that uses stationary application for livestock and crops identification; RSIP Vision has a "deep learning-pattern recognition-computer vision" project that obtains interesting results for pig monitoring but is not a marketable product. Academic studies based on "Pattern Recognition" only in some cases will generate a test open source product.

Chapter 3 PhAID system development and its model

According to [1], the process of developing an ML-based solution in an industrial setting consists of four stages:

- **Data management**, which focuses on preparing data that is needed to build a machine

learning model;

- **Model learning**, where model selection and training happens;
- **Model verification**, the main goal of which is to ensure model adheres to certain functional and performance requirements;
- **Model deployment**, which is about integration of the trained model into the software infrastructure that is necessary to run it. This stage also covers questions around model maintenance and updates.

Following this clear definition, with the development of PhAID for the IZS we followed all the steps listed up to have all the instruments to apply the MLOps techniques to the Model developed.

PhAID is a system that is able to re-identify cattle thanks though the face's picture of the animal. PhAID is a Deep Learning (DL) solution developed by F4T and AImagelab thanks to the support of IZS. After the development of the DL neural network, the extraction of the model was released ready to apply the modern MLOps approach. In this thesis the PhAID development and deploy web solution is described following the listed phases:

Phase 1: PhAID first neural network solution to demonstrate the feasibility of the project;

Phase 2: PhAID scale up and first deployment;

Phase 3: PhAID architecture improvement;

Phase 4: PhAID large dataset test and web prototype solution;

Phase 5: PhAID web application prototype evolution and deployment.

3.1 PhAID first neural network solution and project feasibility demonstration

The PhAID implementation for IZS and its model definition started soon after the neural network was developed by Imagelab. Objective of this was to demonstrate the feasibility of the project.

The research activity in this new sector of photographic recognition of cattle using technologies based on Deep Learning and Machine Learning has made it possible, in less than 6 months, to demonstrate that such activity is possible.

The main challenge of this phase was the photo collection and the low number of animals' picture available (only 550) suggested to use the generative adversarial network (GAN) to "create" artificial images of cattle starting from human faces photos. But this approach failed due to the diversity of the cranial conformation between humans and animals is substantiated by the existence, for the latter, of 3 different profiles which are a front profile, a right side, and a left side. This anatomical diversity, in addition to the technical difficulties

already described, made it necessary to confront the recognition problem in a new way. New approach on image acquisition.

High-resolution videos and/or images of over 550 individuals from 10 different farms were acquired. The animals mainly belong to the Friesian breed, with a limited number of Swiss Browns and Pezzata Rossa, and with about 150 individuals belonging to the Rendena breed. The original images and videos are stored within the Amazon S3 (Simple Storage Service) Buckets using the Amazon Web Service cloud infrastructure that guarantees secure upload or download of data to Amazon S3 via SSL-encrypted endpoints using HTTPS protocols. All Amazon S3 storage classes have been designed to provide 99.999999999% durability of objects for a given year. This durability level corresponds to an estimated average annual item loss of 0.000000001%. This ensures that the data is stored securely and that the risk of loss is almost at zero.

The early stages of the research activity immediately highlighted unsatisfactory results concerning the strategy that was initially adopted. Activity confirmed that it was possible to "transform" images of human faces into images of cattle heads, but not to maintain the cardinality of the embedding [1] in the transfer of the domain, thus preventing the ability to create a unique animal identity for each transformed human face. It was therefore decided to proceed with a different strategy, not excluding the possibility of making new attempts in the future on the use of the domain transfer technique. A work unit was then set up dedicated to the collection of images for the construction of the dataset directly on the field for the new phases of the project.

Experiment

Data Acquisition - Snap Animal

Farm4trade developed an application for Android devices dedicated to the acquisition of the images and videos of cattle necessary for the construction of the datasets. The purpose is to simulate a situation as close as possible to reality and to allow a standard enough acquisition that is both simple and functional for the research activity, rather than using professional cameras or tools natively installed on different mobile devices.

This application is available on Google Play and has the following features and functionality:

- Create and add a new animal to the list, adding the following information: company name, national ID, company ID, any other identifier, sex, date of birth, and animal species;
- Video capture interface. In the options menu it is possible to choose the maximum duration of the videos and the desired resolution;
- Photo capture interface. Depending on the projection (front, right, and left), photos are acquired within a specific folder;
- Log of all animals in the database, with information on the status of synchronizations performed with the cloud storage platform;
- Gallery of captured photos and videos, where images can be deleted or edited;
- An options menu for account settings and some camera settings;
- Synchronization of images and videos with S3 storage;

[1] ***Embedding*** *is a low-dimensional representation of high-dimensional data where information is compressed to obtain only* ***the useful part of the data****. Data can be represented as a rather long text while embedding represents the summary of the text. Embedding uses a small series of numbers to describe the data.* ***Embeddings also allow for better represent distances and similarities between data****.*

- Synchronization with the farm management tool created by Farm4Trade.

Procedure and metrics

The acquisition procedure has undergone numerous changes since the start of the project, moving from an image-based approach to a mixed one based on videos for training and images for testing. The validity of the acquisition method is evaluated in terms of:

- Number of total images extracted (NIT);
- Number of distinct animals (NA);
- The acquisition time for a single image (TPI);
- Annotation time (manual) for a single image, in terms of "clipping" of the animal's face (TPA);
- The variance between photos of a single animal in terms of visual appearance, as the mean-variance between pixels (VA).

The results obtained from the experiments carried out in the initial phase of the project, in terms of the metrics described above, are shown in Table 1.

Method	NIT ↑	NA	TPI (seconds) ↓	TPI (seconds) ↓	VA
Photo	770	32	3	7	10
Video (about 10 seconds)	1356	20	0.33	0.25	3
Video (about 1 minute)	10324	34	0.33	0.25	8

Table 1: results of acquisitions using various methods

The following considerations emerge:

- Thanks to the use of single images, it is possible to obtain a high variance. Every single image is acquired at the discretion of the operator, who directly controls the information content by trying to acquire distinct frames;
- The use of video increases the number of total images by an order of magnitude and decreases the time required for a single acquisition and annotation. These factors are of fundamental importance for an approach based on Deep Learning, which requires a substantial amount of data;
- The use of videos collected by an operator trained in the field and trained with a defined protocol improves the variance for each animal, increasing the information content for neural networks, and bringing the metric closer to the value of an acquisition based on single images.

Training dataset

To create the training database and to assign an embedding to the identities of each of the animals, we have chosen to adopt an acquisition method based on the collection of a video, in HD, with a minimum duration of 30 seconds (preferably 1 minute), annotated with the identification of the animal (ear tag).

The operators who gradually collected the videos were instructed in advance to respect the following instructions as much as possible:

- Move by rotating around the animal's head by 180 degrees at least 2 times;
- Don't dwell in the same position;

- Slightly vary the distance from the animal during shooting;
- Slightly vary the angles on the vertical axis;
- Always keep the animal's head (including horns and ears) in the frame, making sure you have a small percentage of the background around the framed profile.

By respecting these basic rules, with a single acquisition of at least 30 seconds, it is possible to extract hundreds of frames and have complete information on all angles of the animal's head and a wide variety of positions. The images selected from the extracted frames were then used to train the convolutional networks built for the training phase.

Testing dataset

For the creation of the testing database, we decided to adopt an acquisition method based on the collection of single images, associated with the identifier of the animal (ear tag) and with an identifier of the view of the animal (respectively front, right, and left side), as in Figure 1. At least 2 images per day and side were collected of the animals selected for system testing, possibly with a different smartphone than the one with which the videos were acquired. The reason for sampling on different days and with different tools than those used for acquiring training images is to simulate as much as possible the reality in the field. Furthermore, the use of images not extracted from video guarantees the reproducibility of the experiments.

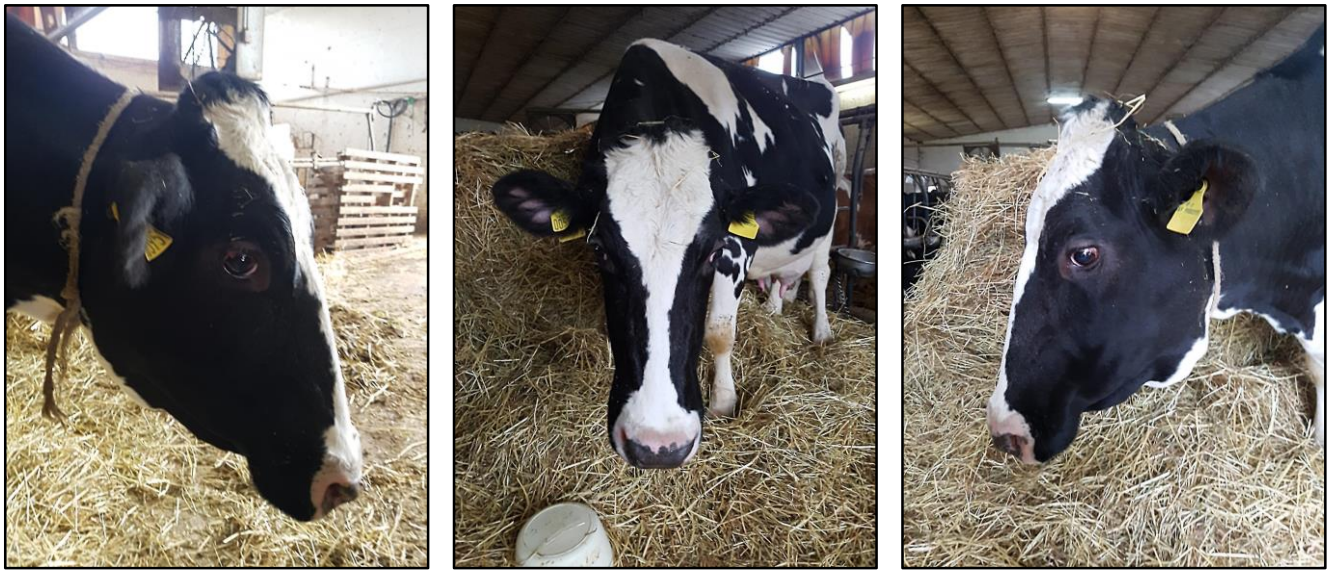


Figure 1: profiles acquired using the procedure established

Test dataset

The two datasets acquired to identify individuals and then test the recognition system involve two acquisition sets from two different companies, the Costantini company (Castel Frentano, CH) and the Martin company (Rocca Santa Maria, TE). Table 2 shows the information for the Costantini dataset, while Table 3 shows the information relating to the Martin dataset.

# Animals Database	# Images Database	# Test Animals	# Test Images

35	3477	24	202

Table 2: Statistics dataset Costantini

# Animals Database	# Images Database	# Test Animals	# Test Images
42	2548	28	359

Table 3: Statistics dataset Martin.

Data annotation

Identification of animals in videos

From each of the captured videos, it was necessary to extract the individual frames and select those with the most variability. This operation, initially performed manually, was automated by training a neural network developed for this type of task. In the first phase, the images were annotated using a tool developed by AImageLab, which allows to draw a rectangle within the image, and to move and/or resize it (manual annotation). Then, shifting to videos and being able to take advantage of the temporal continuity of the frames, we have chosen to use Vatic (Video Annotation Tool from Irvine, California: http://www.cs.columbia.edu/~vondrick/vatic/) which is a free and interactive tool available online for annotations in computer vision (semi-automatic annotation). The manually annotated images and videos were then used to train the SSD (Single Shot Detection) neural network. Figure 2 shows the annotation phase, which converts the initial image into a frame, ready to be submitted during the training phase.

Figure 2: example of annotation and its result

Projections Annotation (Front and Side)

The use of Vatic was expanded to annotate the animal's projections, which were reduced to the front, right, and left sides to avoid ambiguity.

To mitigate the problem of treating each profile as equally representative of the animal, the annotation of the corresponding profile has been introduced for each image, and the nets are forced to recognize not only the animal but also the corresponding profile. This allowed the recognition network to use this information to bring even very different image representations of the same individual closer together.

Approach to the recognition system

Recognition strategy

To tackle the problem of cattle recognition, we decided to take advantage of the biological differences with humans and therefore make the most of the information contained in the diversity of the 3 profiles available for each animal. We first assessed the performance of a recognition system based on a single profile, as it occurs for humans, and then decided to tackle the problem in a completely new way, using 2 profiles and considering in the future the possibility of doing recognition using all 3 profiles.

Testing Protocol

Our algorithm demonstrates superior performance if the two animal profiles (front and side) are provided as input rather than just one of the three. It is therefore necessary to form pairs extracted from the test images. To make the experiments replicable, it was decided to generate all the possible pairs obtained by combining the images of the two animal profiles. For example, in a subject with 4 frontal and 2 lateral images, there are a total of 8 possible pairs. The steps for the test procedure are as follows:

- loading of the trained Neural Network on the data in our possession, consisting of images of part of all the animals collected (433 of over 550 in the database);
- extraction of the database from the network (embeddings) using the training dataset (Costantini and/or Martin) using single images of the training (no increases occur using all the possible pairs in this phase);
- compression (average) of the embeddings of each identity of the training. After this phase, each identity has a single reference embedding in the database;
- generate all pairs of the test dataset;
- extraction of related embedding;
- matching to the nearest or in the first next k

Experimental Results

Tables 4, 5, and 6 show the results in terms of the MR (Match Ratio) metric for k equal to 1, 3 (TOP 3), and 5 (TOP 5). In these last two cases, we check whether the correct animal is present in the first 3 or 5 predictions of our model. The MR Random metric is calculated by considering a random predictor that expresses a random identity by choosing from those present in the training. Both metrics have the same wording; defined as follows:

- I: number of animals;
- Ji: total number of pairs per animal i;
- y_j^i: annotated identity for pair j of animal i;
- p_j^i: predicted identity for couple j of animal i;

so:

$$MR = \sum_{i}^{I} \frac{\sum_{j}^{J_i} y_i^j == p_i^j}{I * J_i}$$

Data Base	Test Set	MR ↑	MR Random ↑
Martin	Martin	0.85	0.028
Costantini	Costantini	0.91	0.023
Martin + Costantini	Martin + Costantini	0.87	0.012

Table 4: benchmark results

Data Base	Test Set	MR TOP 3 ↑	MR Random MR TOP 3 ↑
Martin	Martin	0.97	0.084
Costantini	Costantini	0.97	0.069
Martin + Costantini	Martin + Costantini	0.97	0.036

Table 5: TOP 3 benchmark results

Data Base	Test Set	MR TOP 5 ↑	MR Random MR TOP 5 ↑
Martin	Martin	0.99	0.14
Costantini	Costantini	0.99	0.115
Martin + Costantini	Martin + Costantini	0.99	0.06

Table 6: TOP 5 benchmark results

Test Protocol

This section defines the test protocol used to obtain the results shown in Tables 4, 5, and 6. To obtain the **MR** metric, two data sources are required:

- Source Database (**SB**): This source contains images associated with the identity of the corresponding animal. These images were seen by the network in the Training phase.
- Probe Source (**SP**): This source contains images associated with the identity of the corresponding animal. These images were not seen by the network in the Training phase, **but they all belong to animals present in SB**.

Once the sources are defined, 3 phases follow:

Submission of SB

The images are submitted individually to the network. The network receives the image on the front or side channel, depending on the profile of the animal in the photo, while the other channel receives a black image (meaning the information is absent). Each image is then converted into an **embedding**. The set of embeddings (associated with the corresponding identity) forms the **DE** - Extended Database. This database has cardinality equal to the number of images submitted and therefore is not suitable for scaling on large amounts of data. Therefore, a search in this database is expensive.

Compression of DE

The embeddings contained in **DE** belong, by construction, to a locally Euclidean space, as imposed by the loss function during the training phase. Therefore, the embeddings of a single identity are clustered using a simple arithmetic mean because they are all located close together in the constructed space. To do this:

- embeddings are grouped by identity. Each identity has a variable number of embeddings which depends on how many photos of the animal were present in **SB**;
- the average of the embeddings of each identity is calculated, forming the **DC** Compressed Database. Each **SB** identity now has **only one embedding** to represent it;

This reduces the number of comparisons to perform in the test phase, without impacting the performances (provided that the number of images for each identity contained in SB is sufficiently high).

Submission of SP

The images are submitted in pairs to the network. All possible pairs are generated for each identity (a pair is formed by two images of different profiles) to make the process repeatable and deterministic. For example, an identity with 3 front and 5 side images has a total of 15 possible pairs. Each pair of images is then converted into an **embedding**.

The set of embeddings (associated with the corresponding identity) forms the Extended Probe **PE**.

Test nearest Neighbor of PE

For this test, we compare **PE** versus **DC**. For each element of **PE**:

- The distances L2 between the embedding and all those of **DC** are calculated;
- The distances are ordered in ascending fashion;
- If the closest element in **DC** has the same identity as that of the **PE** embedding, the **MR** score is increased;

- If one of the 3 closest elements in **DC** has the same identity as that of the **PE** embedding, the **MR TOP 3** score is increased;
- If one of the 5 closest elements in **DC** has the same identity as that of the **PE** embedding, the **MR TOP 5** score is increased;

The values of **MR**, **MR TOP 3**, and **MR TOP 5** are then divided by the number of test pairs to obtain the average value, reported in tables 4, 5, and 6.

CONCLUSIONS

The preliminary results (Table 4), obtained by comparing the identities of the test set containing 52 animals (Costantini + Martin; Tables 2 and 3) with the 77 animals belonging to the same farms and present in SB, show that it is possible to recognize single individuals with a high degree of accuracy (MR = 0.87, Table 4).

To date, these are the first available results in the world that demonstrate that this is possible. Since the research activity is only in the initial stages, so far much of the activity to has been dedicated to:

- developing a data acquisition system;
- acquiring the datasets;
- fine-tuning networks to annotate images and cut them out;
- understanding the criticalities of this new domain.

It is therefore clear how much it is still necessary to study the problem and try new approaches to improve what has been done so far. The results obtained are the result of preliminary work that will have to be investigated to explore different and improved paths. We believe without a shadow of a doubt that there is ample room for improvement even if the results achieved so far are extremely satisfactory and promise excellent developments for the future.

Now understanding much better the needs and the domain in which we operate, it is necessary to specify that the identification and re-identification tasks, producing similarity measures between the test identities and those present in SB, do not allow today and they will not allow the achievement of 100% accuracy, i.e. the total absence of error, as the work proceeds. It will therefore be necessary, as the activities proceed, to define threshold criteria within which it is possible to accept a certain percentage of error that does not compromise the recognition ability according to the needs and objectives agreed from time to time with the IZSAM and/or for the specific scope of application. This "tolerance" is justified because no type of test, including traditional identification and recognition systems (brands, boluses, RFID chips, markings, etc.), allows 100% accuracy. An animal registered in a database with an ear tag number is recognizable and correctly identifiable, provided that the operator does not make any errors in reading or transcribing the code, only if the tag is present on the animal. If the ear tag had been lost or removed, it would be impossible to identify the animal, bringing the virtual accuracy from 100% to 0%. The same is true for microchips and ruminal boluses where reading or transcription errors, as well as malfunctions of the reading instrumentation, cannot guarantee the re-identification of the animal always and in all circumstances; for example, the absence of an RFID chip reading tool would make this system unusable for recognition.

We expect to face critical issues with this new photographic recognition system as well. It will be necessary, in the progress of the research activity, to fully understand them and understand how to limit them, but, to date, the results are promising and it is necessary to continue what has been done so far to improve the technology and evaluate the scalability of the results.

3.2 PhAID scale up and first deployment

After the demonstration of the project feasibility new photos were acquired and used to scaleup the PhAID project. A first integration with the model deployment was also made in this specific phase.

High-resolution videos and/or images of over 1000 new cattle were acquired from farms based in Italy, Namibia, and the United Kingdom. The animals belong to different breeds, mainly meat production breeds, with both pure and hybrid subjects. The total number of animals photographed or videotaped exceeds 1700 individuals. Where possible, to ensure a sufficient amount of data to achieve the objectives, multiple images or videos of each individual were acquired, even on different days, thus significantly expanding the dataset. The images and videos are stored within the Amazon S3 (Simple Storage Service) Buckets using the Amazon Web Service cloud infrastructure that guarantees secure upload or download of data to Amazon S3 via SSL-encrypted endpoints using HTTPS protocols.

From the results obtained during the first phase of work and described in the 3.1, it was immediately clear how the increase in performance of the networks derived from the use of two "views" (between right, left and front) per animal, rather than from one view. In particular, one for the front profile and one for the side profile. The greater the number of "views" available for a given animal, the greater the accuracy of the system in the recognition phase. We thus worked to "stress" this aspect as much as possible, replacing the approach based on only two "views" with a new version, capable of working with any number of input profiles (Multi-shot Re-identification or Set to Set Recognition).

This approach, which is quite widespread in human recognition, is also fully applicable to the data available for cattle recognition. It is the most convincing result, assuring the possibility of exploiting a huge amount of data, such as those present in a video acquisition, where generally more images (frames) are generated every second. From a technical point of view, a neural network was implemented capable of extracting, for each image present in an input set, a representation (feature vector) that describes its visual characteristics and, then, aggregating these representations into a single representation, which is more descriptive than the individual representations that compose it. This model does not place any constraints on the number of images present within the input set, thus managing to work correctly with sets of varying sizes. In this new architecture, knowledge of the nature of the view (front or side profile) is no longer required. Each sample set is created by randomly extracting a predetermined number of images from the video of interest (variable between 2 and a maximum of 7 images per set). Random extraction, in the presence of movement of the camera or the animal, ensures that each set is made up of different views of the animal.

This approach, compared to the one adopted previously, is more flexible and less expensive. It emphasizes that the network with two branches requires a precise indication regarding the nature of the image profile.

In this new scenario, the new and adapted neural network were tested

Testing of the architecture

The tests were conducted by comparing two different backbones, namely the Deep ResNet34 and DenseNet121 networks. Furthermore, for each architecture, tests were carried out with a variable number of N_{views} views. All the metrics of interest (Top-1, Top-3, and Top-5) increase as more N_{views} of input views are provided to the model. This demonstrates how the approach adopted is more effective than using a single view. When N_{views} = 2 the proposed architecture collapses on its previous version, therefore we can identify this variant as the baseline for the new work. Comparing the results obtained when N_{views} = 7, it becomes clear how the multi-shot approach provides significantly better results.

The use of DenseNet121 returned in all cases a higher degree of accuracy than ResNet34. In support of this,

it is reported that this result is in line with what has been highlighted by recent articles on human recognition [1]. The basic idea is that the connectivity present in DenseNet preserves the possibility of propagating local details of the starting image on the output. In many cases, these details are crucial for recognition, or to discriminate subjects of similar appearance, who, however, exhibit distinctive features only at a local level (eg: spots on the animal's head, as well as their shape).

The performance of the identification system, as shown in the table below, fully satisfies the requests. By testing the system's ability to recognize an animal among the animals registered in the Costantini and Martin farms (56 animals in total), the accuracy was shown to be 100% both using these two farms as a Gallery Set, then comparing an identity with the 56 present in the same, both by expanding the Gallery Set to three farms, then comparing an identity with the 115 present in the total of the 3 farms.

Farm	Total queries	Correct queries	Correct match %	Total animals	Correct animals	Correct match %
Costantini	47	47	**100%**	24	24	**100%**
Martin	95	95	**100%**	32	32	**100%**
Total	**142**	**142**	**100%**	**56**	**56**	**100%**

Table 7: Test results performed on Costantini and Martin farms (Total Cattle: 56) using DenseNet121 and the infrastructure described in the report as the backbone.

First PhAID deployment

For the test, a dockerized web application was built with the purpose of visually and practically testing the architecture.

Using a Torchserve server, a Flask application interacts with a python server for recognizing cattle on the farm. The structure includes a worker developed in python which contains the model and the handler with the weights of the networks to input to the Torchserve server. A Flask client that communicates with the worker via the python API to send the bovine to be identified and display the results. The APIs are developed in python and involve reading a database that collects the images of the farm dataset. The docker environment is orchestrated by an nginx proxy.

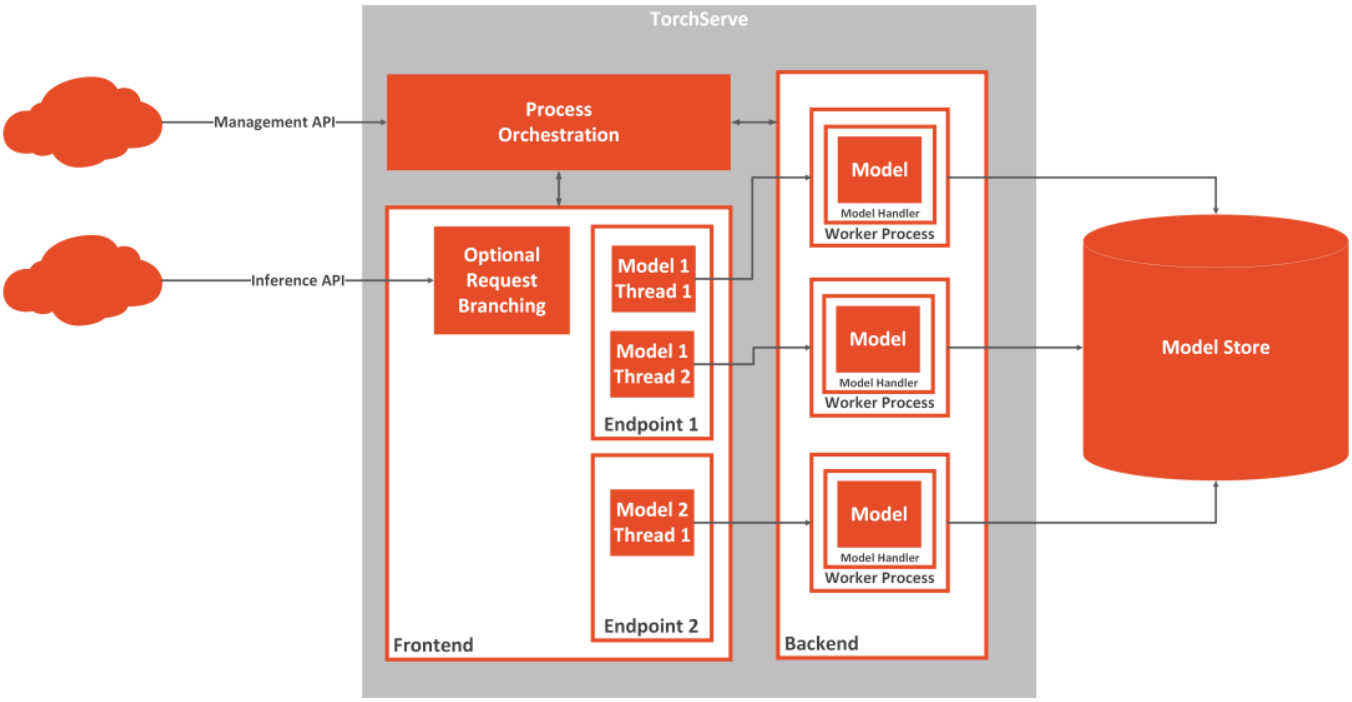


Figure 3: Diagram of the web application created to test the developed architecture

Experiments

Compared to the phase describe in 3.1, the dataset has been expanded in terms of the number of animals (identity). In addition to those acquired on Italy, animals belonging to foreign-based farms have been added, in particular: an experimental farm based in Edinburgh (United Kingdom) and a fattening station based in Windhoek (Namibia). This allowed further standardization of the structure of the training, gallery, and query datasets, described below.

Is important to notice how the increase in acquired and identified individuals has made it possible to employ a total of 115 different identities in the testing phase. This represents an increase of almost 400% compared to the previous report, and more than 100% compared to the article submitted [2] in 2018.

Farm	Location	Country	Number of Animals	Total Images
Costantini	Castel Frentano (CH)	Italy	35	2.198
Martin	Rocca Santa Maria (TE)	Italy	231	10.557
Ritort	Pinzolo (TN)	Italy	52	1.436
Zeledria	Pinzolo (TN)	Italy	46	1.286
Boch	Pinzolo (TN)	Italy	52	1.456

Hombre	Modena (MO)	Italy	126	3.183
Ferrara	Ferrara (FE)	Italy	5	81
Venditti	Castelpagano (BN)	Italy	50	871
Okapuka	Windhoek	Namibia	911	4.797
Howgate	Edinburgh	United Kingdom	218	2.354
TOTAL			**1.726**	**28.219**

Table 8: Training dataset. Location, country, number of animals, and total images are shown for each farm.

Training set

This dataset contains images used during the training phase of the models. These are therefore the only images on which the model has updated its parameters to learn the task. The subjects come from different farms, as described in Table 1. This is the dataset that includes more examples among those collected, as the models require a high number of different subjects during the training phase to generalize to individuals never seen before. In Figure 1, some images extracted from the dataset are proposed: it is possible to appreciate the variety of poses and races emerging from the data collected.

Figure 4: Images extracted from the training dataset.

Farm	Location	Country	Number of Animals	Total Images
Costantini	Castel Frentano (CH)	Italy	24	435

Martin	Rocca Santa Maria (TE)	Italy	32	657
Howgate	Edinburgh	United Kingdom	59	870
TOTAL			**115**	**1.962**

Table 9: Query dataset

Farm	**Location**	**Country**	**Number of Animals**	**Total Images**
Costantini	Castel Frentano (CH)	Italy	24	135
Martin	Rocca Santa Maria (TE)	Italy	32	170
Howgate	Edinburgh	United Kingdom	59	341
TOTAL			**115**	**646**

Table 10: Gallery dataset

Query set
This dataset contains the images whose identity we want to determine in testing. The number of subjects present defines the difficulty of the task since increasing the number of identities increases the possibility of confusion between different identities when searching for a match. Tests were conducted with a maximum number of 115 different subjects, as shown in Table 2. It is important to have a dataset distributed among various farms, as focusing on a single farm would make the results difficult to generalize to other contexts.

Gallery set
This dataset is used as a database against which to compare query dataset images. It, therefore, contains the same subjects as the first, as shown in Table 3.

Testing protocol

Currently, the gallery dataset can be built in 3 different ways:

I. **The gallery dataset is formed of the same images present in the query dataset.** This mode leads to an overestimation of the model's capabilities, as it is extremely easy to solve for any algorithm. In fact, by calculating the distances between an image in the query and all those in the gallery, the minimum distance solution will always be the image itself. This mode is therefore never used, as it does not discriminate between an effective classifier and an ineffective one.

II. **The gallery dataset is composed of images that are different from those in the query dataset but collected on the same day.** This second mode could be used, but from previous experiments, it too

proved too simple to solve. Images acquired on the same day do not present enough differences in terms of external conditions, background, and pose of the animal.

III. **The gallery dataset is composed of images different from those of the query dataset, collected on different days after some time.** This third scenario is ideal and more challenging if you want to evaluate the performance of an algorithm, and is therefore the one currently used during the experiments, discussed in Section 5. In particular, the images for a given individual are collected at least one week apart.

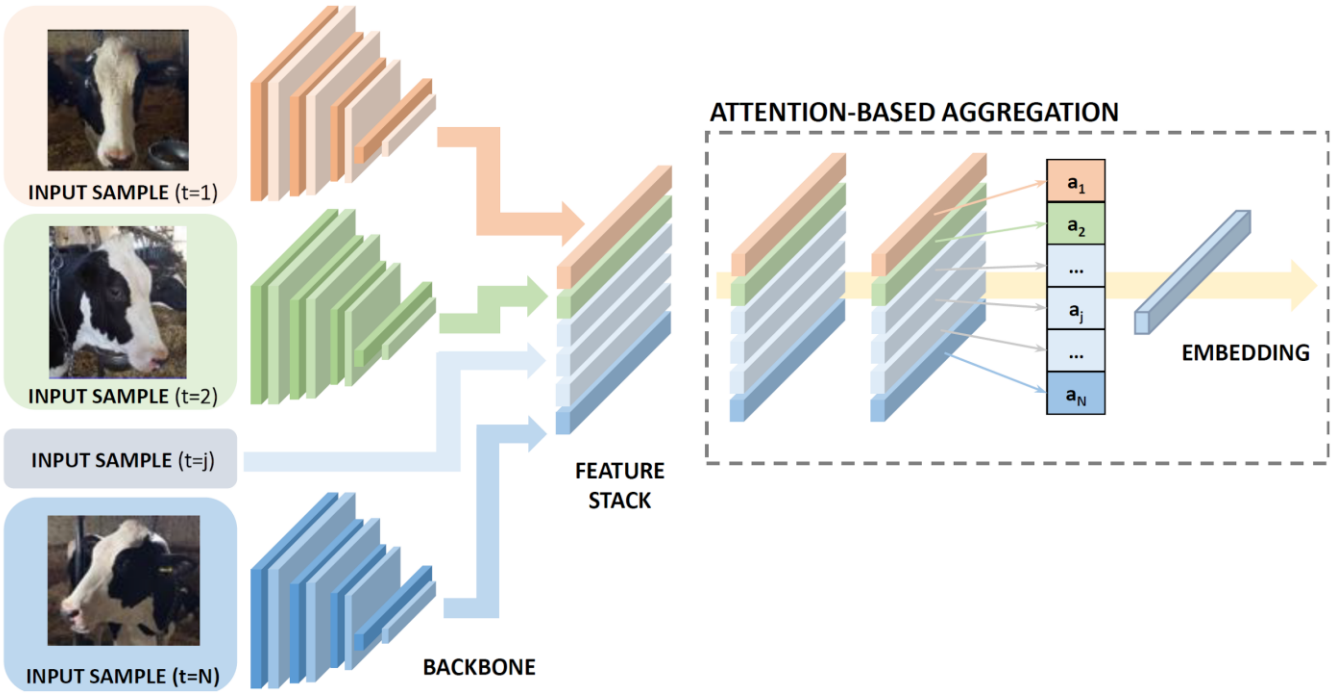


Figure 5: Diagram of the architecture used. Each image that makes up the input set is independently processed by a feature extractor, created using a Convolutional Neural Network (CNN). After that, the intermediate representations are supplied to an Aggregation Neural Network, which has the task of forming a single representation against the Ns that are supplied to it as input. Hopefully, this representation will summarize compactly and efficiently the information content necessary for recognition.

Model

This section lists the architectural choices that have led to a significant improvement in the results. In detail, this improvement was achieved by operating on the following fronts:

- Adopting a **multi-shot** and set-based approach to recognition.
- **Feature extraction**: using a highly performing Convolutional Neural Network (CNN) for the task in question.
- **Set aggregation**: unsupervised learning of a mechanism for merging multiple views into a single final representation.

Multi-Shot Approach

With the results obtained during the first phase [2], it was immediately clear how the increase in performance derived from using two "views" (between right, left, and front) per animal, instead of just one. Specifically, one for the front profile and one for the side profile. The natural motivation behind this improvement is in

the fact that semantically important details can be hidden in the central profile which, however, are hidden from other angles: these factors can still be easily recovered by accessing the side profile (and vice versa). Therefore, the greater the number of "views" available for a given animal, both in the identification phase and then in the recognition phase, the greater the accuracy of the system in the recognition phase.
We, therefore, tried to stress this aspect as much as possible, replacing the approach based on only two "views" with a new version, capable of working with any number of input profiles. In the field of human recognition, this approach is consolidated and is identified with different keywords: 'Multi-shot Re-identification' [3], 'Set to Set Recognition' [4], and 'Video-based Re-identification' [5]. In most cases, these systems exploit the fact that in a video-surveillance context it is possible to extract multiple images of the same target from a single video, using for example a tracker [6].
This aspect is also fully achievable in the data available for the recognition of cattle: the fact that short videos have been acquired with the animal in the foreground makes it possible to extract at a later stage a set of images that describe its features from angles. Ultimately, this approach turned out, in the preliminary phase, to be the most convincing one, guaranteeing the possibility of exploiting an enormous amount of data, such as those present in a video acquisition, where generally more images (frames) are generated every second.
From a technical point of view, a neural network was therefore implemented capable of:

- **extracting**, for each image present in an input set, a representation (feature vector) that describes its visual characteristics.
- **aggregating** these representations into a single one, which will hopefully be more descriptive than the individual representations that compose it.

This model makes no assumptions about the number of images present within the input set, thus managing to work correctly with sets of varying sizes. It is important to underline that in this new architecture knowledge of the nature of the view is not required: in other words, it is not necessary to know in advance whether a given image depicts a front or side profile of the animal. Simply, each sample set is created by randomly extracting a predetermined number of images from the video of interest (during the experiments, this number will vary between 2 and 7 images per set). Random extraction, in the presence of movement of the camera or the animal, ensures that each set is made up of different views of the animal.
This approach, compared to the one adopted previously, is more flexible and less expensive. It emphasizes that the network with two branches requires a precise indication regarding the nature of the image profile. Although finding this information does not represent a particularly expensive or difficult activity, this approach nevertheless requires additional upstream algorithms, whose possible errors will irreparably affect the final performance of the model. Without considering that, within a video clip, only a few images are immediately distinguishable and classified as a front or side profile.

Feature extraction

Within the architecture, depicted in Figure 2, the feature extraction phase represents the first step, during which each image in the input set is analyzed independently from a shared network, in technical jargon backbone. The purpose of this part is to build a compact descriptor (feature vector) for each image presented as input.
During the first phase of the project, a Convolutional Neural Network (CNN) was used, with partial success, for the execution of this task. However, to obtain an increase in performance, it was decided to act on this building block by improving the initialization of the weights: the latter was extracted from a network previously trained on a complex image classification task, such as Image-Net [7]. In this way, it has become possible to exploit an enormous amount of previous knowledge about natural images, which incorporates both low-level notions (e.g. colors, textures, local gradients) and high-level (the shapes and relationships between these in the composition of a complex object). In summary, it was decided to draw on a good knowledge base, which, it should be emphasized, only serves as a starting point for actual training. This choice

turned out to be an important element for the task under consideration, thus obtaining the following advantages:

- The initialization of the first layers of the model is on average excellent, as the features encoded by these layers are general-purpose representations and can be shared even between very different tasks;
- The training times of the architecture are lower, as the initialization provided by the training on ImageNet is an excellent starting point for training the model.

Once defined how to initialize the weights, an analysis was subsequently conducted on the physical architecture of this network. Specifically, on the paradigm that governs how a generic layer must be connected to the others present in the model. Several advances have been made in this regard over the years, effectively overcoming the idea of the first Feed-Forward Neural Networks, in which connections were only allowed between one layer and the next. In this regard, several models have been presented in recent years, most of which are based on residual layers and skip connections, in which the output of a given layer can also be propagated to layers further away than the next.

A study was therefore conducted to verify which paradigm was the most effective for the task in question. In particular, the ResNet [8] and DenseNet [9] models were tested in the following configurations:

- ResNet18, ResNet34 and Resnet50 for the ResNet model. The post-fixed number specifies the number of layers present in the architecture. On a theoretical level, a higher number corresponds to a theoretically more capable model from an expressive point of view. The connectivity of these models is based on the concept of 'residual block', an example of which is provided in Figure 6 (left).
- DenseNet121: a model based on the ideas introduced by the ResNet developers. However, unlike the latter, DenseNet provides that each layer receives the activations of all the layers before this one as input (Figure 6, right).

Despite a lower number of parameters (in this regard, Table 11 offers a summary of the number of parameters of each configuration), the use of DenseNet involves a higher memory load than the ResNet variants. However, after several experiments the use of DenseNet as the backbone of the architecture has led to better results than ResNet and its derivatives.

Configuration	**Number of parameters**
ResNet18	11.689.512
ResNet34	21.797.672
ResNet50	25.557.032
ResNet101	44.549.160
DenseNet121	7.978.856

Table 11: Number of parameters for each possible backbone configuration.

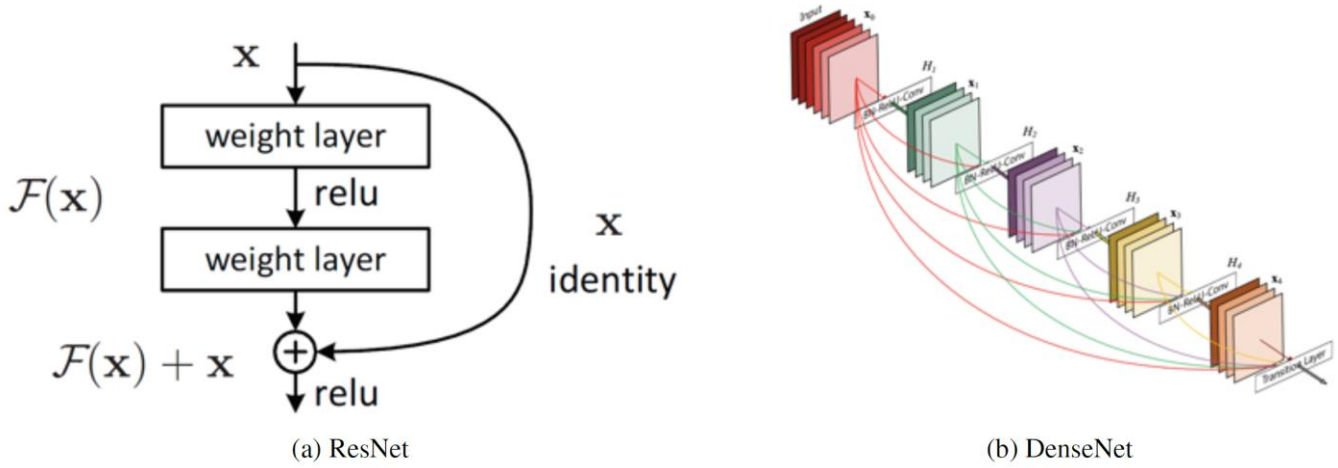


Figure 6: left (a): logic diagram of a residual ResNet block. Each block requires the input to skip two convolutional layers to be added to the output of the block. Right (b): representation of DenseNet. The output of a convolutive block is concatenated to the inputs of all the blocks in the model (thus creating an extremely "connected" model).

Set Aggregation

Once a feature vector has been obtained for each image, the second step involves aggregating this information into a single final representation that describes the animal depicted in the set of images. The use of this operator arises from the complexity inherent in finding matches directly between all the elements of the set and the gallery. In fact, from a computational point of view, this would be very onerous, since the number of comparisons would be proportional to the number of views.
In the literature, there are two categories of approaches:

- Temporal approaches: these techniques consider the set of images as frames within a video, in which there are therefore correlations and patterns at the temporal level. The idea of these methods is to exploit these correlations to extract information that is not accessible by looking at a single frame. These techniques usually employ recurrent neural networks, such as Recurrent Neural Network (RNN) and Long-Short Term Memory (LSTM);
- Set-based approaches: these techniques can be applied to sets of images or objects in which it is assumed that there is no intrinsic order within the sequence.

In the design phase of the architecture, much thought was given to the possibility of exploiting time and temporal coherence within the model. In this regard, it was decided to discard this possibility, thus drawing only from the literature that works at the set level. This choice is justified by two main considerations. In the first place, for the problem under consideration, it is unlikely that the relationships between successive frames will bring out key information for recognition: for example, how the animal moves its head is hardly a truly distinctive trait for the animal. Secondly, the analysis of temporal features requires a considerable effort during the design and training phase of the model, as well as an overhead during the inference. This additional effort is not justified by the observations reported above.
Within the methods based on the concept of set, the fundamental property that an aggregation operator must respect is the following: the result of the computation must not depend on the order in which the elements are presented. Therefore, it is not possible to concatenate feature vectors and perform operations on their concatenation, but only pooling operations and their derivatives.
In this sense, the simplest aggregation method available in the literature (so much so that it is often used as a baseline) involves merging this information using a simple average (in technical jargon, average-pooling):

A B

Figure 7: Images of the same subject of different quality. In A the frames are sharp and with good lighting, while in B unfavorable lighting and focus reduce useful information.

$$f_{AVG} = \sum_{i=1}^{N} w_i * f_i \quad where \quad w_i = \frac{1}{N} \quad i = 1, 2, \ldots, N \tag{1}$$

Where f_i is the feature vector extracted from the backbone for the image *i* of the set. This method proves to be an extremely strong baseline in several papers, given its advantages: it is a light method from a computational point of view, well-motivated from a theoretical point of view, and does not require further parameters to be learned during the model training.

When the weights are not bound to assume the same value for all, the average becomes weighted. In particular, assigning a different weight to each view is equivalent to assigning different importance to each element of the set. As shown in Figure 4, some images contain much more information than others, which appear deteriorated due to a multitude of factors (focus, position of the animal, distance, lighting, etc ...). However, determining on the bench the contribution of each image is very complex, as the score should take into account the measurements of all these factors.

Conversely, the aggregation model developed[2] envisages learning to attribute a quality score [4] to each representation, in an unsupervised way.

This score is what is then used as a value for w_i. This mechanism, defined in the literature as attention and presented for the first time in [10], provides for the presence of a neural network capable of inferring the weights w_i directly from the vector f_i features, through a series of non-linear transformations. The weights are then normalized over the entire set so that their sum is equal to 1. This type of attention is not spatial, as it does not focus on specific areas of a single image, but is performed at the level of the set. As this function is performed by a neural network, which has parameters that can be optimized, the weight values will increasingly reflect a quality metric as the training process progresses.

Experimental Results

In an attempt to make the results obtained comparable with the *in-person re-identification* literature, the metric of *k-Nearest Neighbors Matching Rate* (k-NN-MR) is introduced here, the latter evaluated for multiple values of K. Given the query *embedding* of an animal, this metric returns 1 if the corresponding gallery embedding is among the closest K in terms of distance between embeddings. Then the final score is

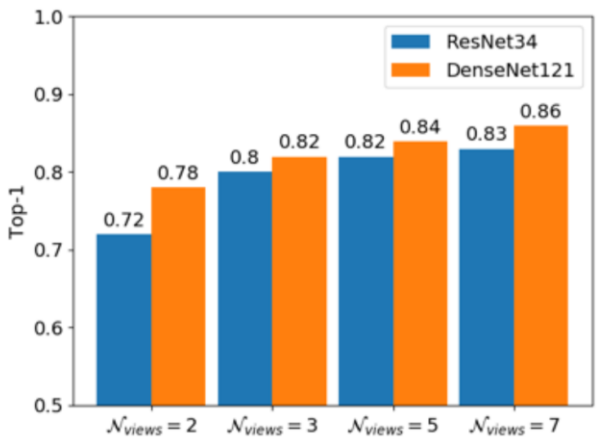
[2] After several experiments aimed at verifying the best architecture, it was decided to use the implementation described in [9] to the detriment of other versions already known in the literature [10, 11, 12]

calculated by averaging these values over all the queries of the test set. Euclidean distance, already discussed in the previous report, was used as a measure of proximity between embeddings.

The tests were conducted by comparing two different backbones, namely ResNet34 and DenseNet121. Furthermore, for each architecture, tests were carried out with a variable number of N_{views} views. The results, shown in Figure 5, highlight some important aspects of the learning process, discussed below.

Impact of the number of views

All the metrics of interest (Top-1, Top-3, and Top-5) increase as more N_{views} of input views are provided to the model. This demonstrates how the approach adopted is more effective than using a single view. When N_{views} = 2 the proposed architecture collapses on its previous version, therefore we can identify this variant as the baseline for the new work. Comparing the results obtained when N_{views} = 7, it becomes clear how the multi-shot approach provides significantly better results.



Backbone	$\mathcal{N}_{views}$	*Top-1*	*Top-3*	*Top-5*
ResNet34	2	0.721	0.831	0.865
DenseNet121	2	0.784	0.877	0.894
ResNet34	3	0.801	0.881	0.898
DenseNet121	3	0.822	0.911	0.936
ResNet34	5	0.822	0.890	0.903
DenseNet121	5	0.848	0.911	0.949
ResNet34	7	0.839	0.898	0.907
DenseNet121	7	0.860	0.924	0.940

Figure 8: results obtained in the Multi-Shot setting: i) Graphic comparison between ResNet34 and DenseNet121 as the number of views varies (left); ii) Overall results as the metric, the number of views, and the backbone used vary.

Impact of the number of neighbors L.

As expected, the accuracy of the architecture increases as K. increases. That is, by relaxing the constraint according to which the predicted identity must be the closest one, and vice versa, contenting itself with finding it among the K closest gallery identities, the system improves its performance.

Farm	Total queries	Correct queries	Correct match %	Total animals	Correct animals	Correct match %
Costantini	47	47	**100%**	24	24	**100%**
Martin	95	95	**100%**	32	32	**100%**
Howgate	95	64	67%	59	40	67%
Total	**237**	**206**	**86%**	**115**	**96**	**83%**

Table 12: Quantitative analysis of the errors for each acquisition present in the test set. The reported results were

achieved using DenseNet121 as the backbone and seven input views. For each farm, the percentage of correct predictions is indicated, as well as the percentage of animals for which no error occurred.

Choice of the backbone

The use of DenseNet121 leads in all cases to a higher degree of accuracy than ResNet34. In support of this, it is reported that this result is in line with what has been highlighted by recent articles on human recognition [14]. The basic idea is that the connectivity present in DenseNet preserves the possibility of propagating local details of the starting image on the output. In many cases, these details are crucial for recognition, or to discriminate subjects of similar appearance, who, however, exhibit distinctive features only at a local level (eg: spots on the animal's head, as well as their shape).

Error analysis

Observing Table 6 it is possible to assess how the errors are distributed on the various farms present in the test set. Currently, all the errors made by the model are attributable to examples from the Howgate herd, while all the predictions relating to Costantini and Martin are correct. To explain this behavior, several reasons are being examined:

- The breeds present at Howgate present a greater degree of difficulty than those present in the other two test farms. The former - belonging to the Red Poll, Aberdeen Angus, and White Park breeds - have on average much less recognizable patterns (spots, etc.) and in general, are characterized by a much more uniform color.
- Howgate's animal background is strongly bimodally distributed. In particular, a blue manger rotates with a view of the stable: therefore, the same animal in the gallery and in the test can have completely different backgrounds.

These two problems mutually influence each other: if two different animals are very similar (item I) and have the same background (item II), the network will tend to confuse them in the test phase, preferring a match like this to one in which the same animal appears with a different background. To support these considerations, reference is made to the first two lines of Figure 6.

Nature of errors

Figure 6 shows some of the errors made by the model. From an analysis of these, it is possible to derive some considerations. First of all, the errors are reasonable: that is, the model never gets confused between completely different individuals, thus testifying to the quality of the features extracted during training. In this regard, the third error reported in Figure 6 is indicative: although wrong, the model projects two different cows, both black with a white spot on the head, in nearby points of the embedding space. Secondly, when the model is confused between animals that share extremely similar visual characteristics, it begins to look at secondary aspects of the image, such as the elements that make up the background. Often the gallery element - mistakenly considered to be the closest - is that precisely because it exhibits the same background as the query image.

Figure 9: View of three mistakes made by the network. For each example: the first line shows the images that appear in the input set; the second depicts the gallery identity erroneously associated with the input one, while the third represents the correct but discarded gallery identity.

CONCLUSIONS

The acquisition of a large number of new images, belonging to over 1,700 different individuals, from 10 different farms in 3 different countries, and the development of a more performing technological

infrastructure with innovative image recognition solutions, have made it possible to advance the results of the project beyond the objectives set for the first year. We can say that a decisive step in the development of the solution for recognizing cattle through images has been taken. The development and implementation of a web application also allows to test the results of the research concretely right now.

This second phase of the research activity also highlights the absolute value of the project. The results achieved to date give hope that in a short time we will arrive at a finished product that satisfies all the set objectives. The results show that today the degree of precision and accuracy is very high and can effectively complement traditional recognition systems. For farms with more than 50 animals, it will be necessary to work not only on the quantity of the dataset but above all on the quality of the images and content of the same to avoid possible errors or difficulties related to the background. After the first phase of the project, the team has grown and expanded, acquiring new figures including researchers, developers, and analysts. The contribution of those in the field who collected the images and videos valuable for this new phase was also important. The greatest effort has been put into the design of the networks that build a robust system capable of adequately satisfying the re-identification task. In the future, the automation of image acquisition in the field will have to be refined to allow the dataset to be available with less difficulty and to rapidly expand the number of images available.

3.3 PhAID architecture improvement

Improvement of the architecture developed and implemented in the first year with the purpose of extending its capabilities to a larger data set
After a first deployment of the PhAID, a new phase started and the evolution of the neural networks was also introduced from the previously Cownet to the new CowNet ++ with introduction combined action of two different depp processing strategies:

- Adoption of features with **different granularity**;
- **Spatial attention** mechanism.

From a purely technological point of view and through the introduction of the innovations, we proceeded to improve the architecture responsible for image analysis. The goal is to calculate the embedding necessary both for the identification process of the individual and for the recognition activity, more precisely when the database is queried in search of the required identity.

Additional dataset was acquired to expand the database and perform in-depth tests on animals never previously analyzed in order to evaluate the recognition performance of races and hybrids other than those used and never used in the training phase (Table 13).
The 288 new identities allow to thoroughly evaluate the ability of the network and to "generalize" the recognition performances also to races and individuals never seen before.

Farm	Location	Country	Breed	Number of Animals
Clementini	Contigliano (RI)	Italy	Chianina	47
Di Marco	Amatrice (RI)	Italy	Pezzata Rossa	116

Krumhuk	Windhoek	Namibia	Nguni and african hybrids	125

Table 13: new dataset. For each farm, the location, country, breed, and number of animals are shown.

The total number of identities available for this phase exceeds 2,000 (two thousand), about 400 of which are used only for testing activities without the same subjects ever being employed in the training phases. This ensures the robustness of the technology and the validity of the networks, which prove to be able to replicate excellent performances even on individuals and races very different from those used for training activities.

Dataset and evaluation protocol

Compared to the previous phase, the number of animals (identities) within the dataset has been significantly expanded. Animals present in some Lazio farms and a Namibian herd were added to the subjects acquired before (Table 14). This allows to further diversify the characteristics of individuals thanks to the acquisition of animals mostly belonging to new breeds (mainly monochromatic) and hybrids with mostly dotted/piebald coats, both very far from the characteristics of the animals already in our possession.

Farm	**Location**	**Country**	**Number of Animals**	**Total Images**
Costantini	Castel Frentano (CH)	Italy	35	2.198
Martin	Rocca Santa Maria (TE)	Italy	231	10.557
Ritort	Pinzolo (TN)	Italy	52	1.436
Zeledria	Pinzolo (TN)	Italy	46	1.286
Boch	Pinzolo (TN)	Italy	52	1.456
Hombre	Modena (MO)	Italy	126	3.183
Ferrara	Ferrara (FE)	Italy	5	81
Venditti	Castelpagano (BN)	Italy	50	871
Okapuka	Windhoek	Namibia	911	4.797
Howgate	Edinburgh	United Kingdom	218	2.354
TOTAL			**1.726**	**28.219**

Table 14: Training dataset. For each farm, the location, country, number of animals, and total images are shown.

In addition to the expansion of the dataset, a substantial cleaning and reorganization of the data both images and videos were carried out. This was necessary to speed up and make the loading and analysis of the same more efficient. The tests of the new network, CowNet++, were carried out both on the new animals, never used in the training phases, and on the same three farms of the previous report as in Tables 15, 16, and 17. The use of the same dataset used previously allows you to make a direct and precise comparison of the

performance of the two networks (CowNet vs CowNe ++) and highlight the actual improvements obtained.

Farm	Location	Country	Number of Animals	Total Images
Costantini	Castel Frentano (CH)	Italy	24	435
Martin	Rocca Santa Maria (TE)	Italy	32	657
Howgate	Edinburgh	United Kingdom	59	870
TOTAL			**115**	**1.962**

Table 15: Query dataset

Farm	Location	Country	Number of Animals	Total Images
Costantini	Castel Frentano (CH)	Italy	24	135
Martin	Rocca Santa Maria (TE)	Italy	32	170
Howgate	Edinburgh	United Kingdom	59	341
TOTAL			**115**	**646**

Table 16: Gallery dataset

Subset	No. of Identities	No. of Examples	No. of Images
Training	1412	2885	24019
Query	118	439	4162
Gallery	118	553	4976
TOTAL	**1530**	**3877**	**33157**

Table 17: Statistics on the dataset

As already described, the test protocol always provides for the subdivision of identities in the dataset into three categories, namely **Training set**, **Query set** and **Gallery set**. However, an important novelty is the introduction of a new metric for the evaluation of recognition performance, known as **Mean Average**

Precision (mAP). This metric is used in a multitude of scientific articles [15,16,17] about human recognition and when combined with the accuracy already used (number of identifications performed correctly on the number of total queries), it allows for a more in-depth evaluation of the generalization skills of a re-identification model.
More in detail, the Mean Average Precision is a value between 0 and 1 and is equal to the average over all queries of the Average Precision (AP) value. Assuming a recognition system based on candidate ranking (given a query, the database of known identities - gallery set - is queried and the records are sorted by decreasing distance), accuracy is limited to assessing whether the identity of the first and first result only matches that of the query; vice versa, Average Precision evaluates how frequently results relating to the query provided (ie same identity) appear in the first places of the ranking. By way of example, Google is a good search engine not only because the first result provided is often what we expect (therefore high accuracy), but above all because the first few pages often consolidate only the results relevant to the key search provided. In summary, the mean Average Precision allows for investigating in a quantitative way the quality of the ordering based on the representations provided by the recognition network.
In this new organization of the evaluation parameter, each animal in the dataset presents a variable number of examples that portray it, each of these examples is presented as a sequence of frames (a set of images previously sampled and cropped from the original video). Finally, these sequences contain between **4** and **10** images (with an average length of **8.6**).

CowNet - Baseline

The new infrastructure is based on the previously developed recognition network: CowNet. During the experimental phase, this network was used as a baseline, to highlight the progress of the changes made. Here we briefly summarize the cornerstones of CowNet, which is based on the following principles:

- **Multi-shot approach**: recognition is performed based on multiple images portraying the target animal. Limiting to a single image the risk is that semantically important details turn out to be hidden and/or occluded: these factors can be easily recovered by accessing a multitude of different views made available for a given animal;
- **Feature extraction**: use of a Convolutional Neural Network (CNN) to carry out the feature extraction part. In detail, the CowNet architecture is built on the concept of residual connectivity, implemented according to the directives and technical specifications of DenseNet [9];
- **Frames Fusion**: learning a mechanism for merging multiple images into a single final representation.

CowNet++ - Architectural improvements

Starting from the original CowNet network, the development and implementation work has led to the birth of an architecture that includes changes that mainly affect feature extraction. Therefore, on the new CowNet ++ network, the multi-shot approach based on multiple images of the target remains valid, as well as the merging scheme of the representations of each frame. Instead, the architecture responsible for analyzing each frame has been revised and improved.
In particular, CowNet++ is based on a combined action of two different deep processing strategies:

- Adoption of features with **different granularity**: breaking up the final representation into different sub-representations, each of which describes the content of a local area of the image.
- **Spatial attention** mechanism: giving the model the ability to choose which areas of the input image to focus one's attention on;

The two strategies introduced are described in detail below.

Multi Granularity Network

Intuitively, CowNet's approach is to extract discriminating features **from the entire body of the image**. However, the images acquired in the farms have a high complexity that risks limiting the learning of the recognition functions: due to the limited scale and the weak diversity among the animals, some no salient or infrequent details can be easily ignored and not contribute to better discrimination of the animal.
To reduce the incidence of this problem, it was decided to resort to identifying significant parts within the body of the images. In this way, the **local information** of the identity can more easily emerge, able to guarantee a better accuracy in the recognition phase.

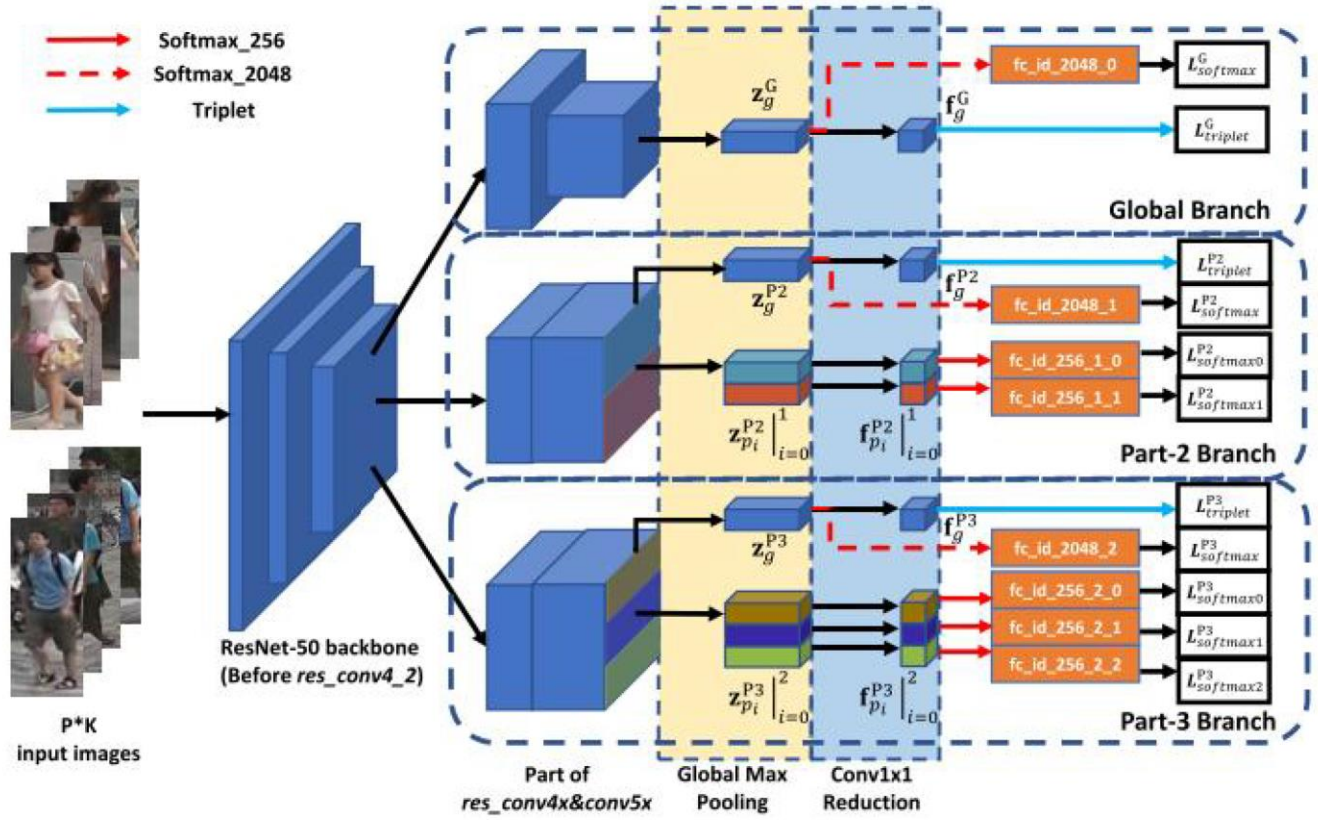


Figure 10: Graphical Representation of Multi-Granularity Network (MGN)

In these approaches [18, 19, 20], also called part-based models, the input image is "broken" into relevant crops (or local regions). Once identified, each of these regions will propose to the model only a small percentage of information from the entire body of the image. In this way, the feature extraction part can focus and adapt to these regions exclusively, avoiding the influence of external regions, whether related or not. Generally, this approach is employed to complement a model that first examines the image in its entirety.
CowNet++ implements these concepts through the adoption of two separate strategies:

- As already happens with CowNet, a part of the model takes care of extracting the features related to the entire body of the image;
- Unlike CowNet, two additional branches examine local regions of the image in different granularities.

From a technical point of view, the followed methodology - based on **Multi-Granularity Network** (MGN) - was introduced in [21] and is shown in Figure 10. Intuitively, the overall network is divided into two logical parts: the first (shared by all subsequent branches) extracts generic features, and the second is divided into three parallel branches, each designed to operate at a different granularity. In particular, the first branch

(responsible for the extraction of global features) examines the entire output tensor, the second branch divides the output tensor into two horizontal bands and examines them independently, and the third branch performs the same operation as the previous, yet working on three bands instead of two.
In this perspective, various numbers of strips (one, two, and three respectively) induce a diversity of granularity of the contents. Therefore, the final representation of the model (embedding) will be given by a concatenation of the embeddings obtained at each granularity.

Spatial Attention

With CowNet it has been hypothesized that some background elements are responsible for some of the erroneous predictions. In particular, some elements of the background (e.g. the light conditions, the manager, and animals in the background related to the target animal) can in some cases become predominant factors, contaminating the final embedding. To mitigate this effect, in the new CowNet++ network it was decided to implement what is referenced in the literature to the concept of spatial attention, or to enable in the model the ability to focus more on certain areas of the image (e.g. the muzzle of the animal) rather than others (e.g. background elements).

This ability tends to be induced in two ways: directly or indirectly. The first of these two options is to provide the model with the segmentation mask as input, the latter estimated from an auxiliary external network. Alternatively, it is possible to proceed "indirectly", leaving it to the recognition model itself to estimate (autonomously and unsupervised) the segmentation masks. These masks are then used internally by the model to concentrate the focus on specific areas of the image.

In this development phase of the project, the second way was chosen. In fact, despite having the segmentation masks available for a subset of the data (and therefore the ability to supply them to the model as input), such information has not yet led to an added value during recognition. This does not necessarily indicate that such information is harmful or useless, but that the optimal way to use it in learning has simply not yet been found.

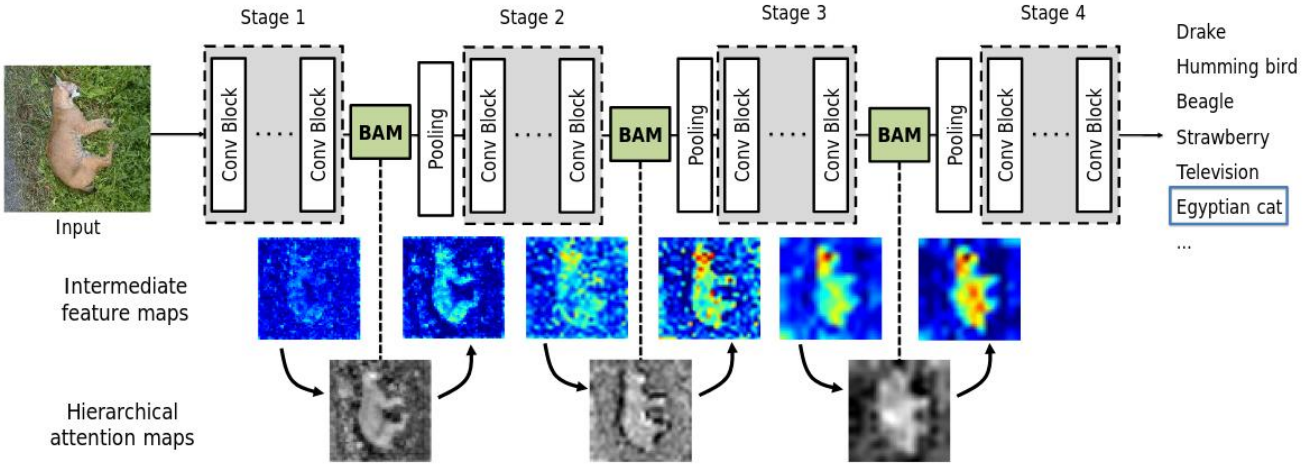


Figure 11: Bottleneck Attention Module (BAM) Graphical Representation

In summary, CowNet++ does not require segmentation masks as input but can estimate them autonomously and subsequently apply them during image analysis. The estimation mechanism of these masks is described

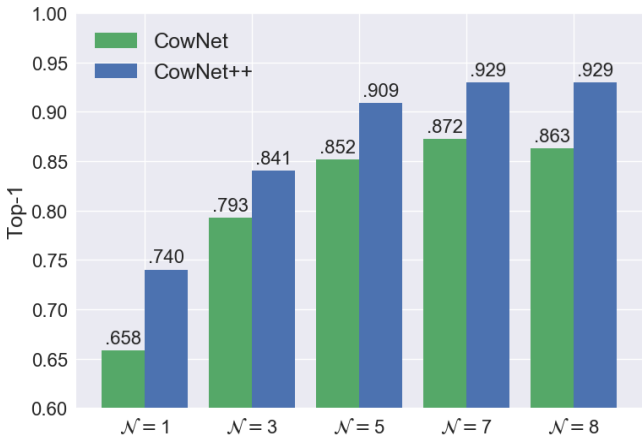
in [21] and is known by the name **Bottleneck Attention Module** (BAM). As highlighted in Figure 11, the idea is to equip the basic architecture (DenseNet in the case of CowNet) with a new module (BAM), whose task is precisely to amplify specific areas of the activation maps (hopefully, those concerning the target of interest). More formally, for a given feature map $F \in R^{C\,x\,H\,x\,W}$, a BAM module first calculates a 3D attention map and only then performs a filtering operation:

$$F' = F + F \otimes M(F).$$

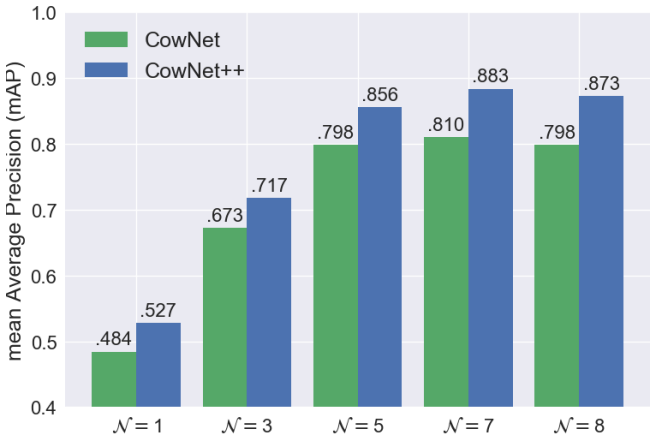
This module is usually inserted in some specific points of the architecture (and not in all); in the case of CowNet ++ a BAM layer was provided before and after each DenseNet Transition block.

Experimental Results

The tests were conducted comparing CowNet ++ with its previous version CowNet. The results, shown in Figure 12, are expressed in terms of top1 (also defined with the term *k-Nearest Neighbors Matching Rate* (k-NN-MR) when **K = 1**) and **mean Average Precision** (mAP). The two different variants were tested in scenarios that present different degrees of difficulty: in this case, tests were conducted in which the number N of images (views) present in each example is progressively increased, thus providing, therefore, gradually more and more visual information to the model.



(a) Accuracy (top-1)



(b) Mean Average Precision (mAP)

	$N = 1$		$N = 2$		$N = 3$		$N = 5$		$N = 7$		$N = 8$	
	top1	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**
CowNet	.658	.484	.743	.598	.793	.673	.852	.798	.872	.810	.863	.798
CowNet++	**.740**	**.527**	**.822**	**.639**	**.841**	**.717**	**.909**	**.856**	**.929**	**.883**	**.929**	**.873**

(c) Results in tabular format.

Figure 12: Comparison between CowNet and its new version CowNet ++

As highlighted by the results shown in Figure 12, the advancement proposed in this new phase leads to significantly higher levels of performance than in the past with CowNet, thus testifying to the quality of the architectural changes introduced. Similar to what has already been observed previously, all the metrics of interest (Top-1, Top-3, and Top-5) improve when a greater number N of input images are supplied to the model.

Ablation study - Through an ablative study, a survey was then conducted to identify which innovation has led to the greatest improvements among all of them. In particular, the intention was to isolate the contribution made by the introduction of a logic based on Multi-Granularity Network. For this purpose, Table 18 compares the results of CowNet (first row), CowNet++ (MGN + BAM, third row) with those of CowNet when it is equipped with MGN (therefore in the absence of BAM modules; second row). As shown, the use of MGN brings substantial benefits for each value of N, an element that fully justifies its use in the recognition phase. However, in the presence of a limited number of images (N = 1 and N = 3), the results introduced by the application of MGN alone are lower than those given by CowNet++ (MGN + BAM), justifying in these cases the introduction of layers Spatial attention BAM.

	$N = 1$		$N = 3$		$N = 5$		$N = 7$	
	top1	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**
CowNet	.658	.484	.793	.673	.852	.798	.872	.810
CowNet (+MGN)	.699	.518	.818	.707	**.911**	**.861**	.920	.880
CowNet++ (MGN+BAM)	**.740**	**.527**	**.841**	**.717**	.909	.856	**.929**	**.883**

Table 18: Ablative study regarding the impact of MGN on the final performance

Explanation of predictions

In the world of deep learning, a recent research trend called **explainability** provides for the possibility of explaining the predictions of a model, that is, trying to understand what are the factors that led the model to make a given decision for a given example. Therefore, an attempt was made to replicate this type of study on the recognition of cattle, trying to understand which parts of the image are judged most relevant by CowNet++ to discriminate the identity of each animal.

To this end, GradCAM [22] was used as the explanation algorithm. In Figure 13 it is possible to see which are the parts that the network is paying attention to during the recognition phase for a subset of images of the training set with the explanations provided by the model.

In all cases, the area of interest (marked in red; the less relevant parts in blue instead) is limited to the face of the animal, with particular emphasis on the areas close to the head and muzzle. Therefore, it can be concluded that the proposed model is sufficiently capable of isolating the predominant and descriptive factors of the animal, eliminating, with a good degree of accuracy, those which may be the external sources of noise and/or distraction.

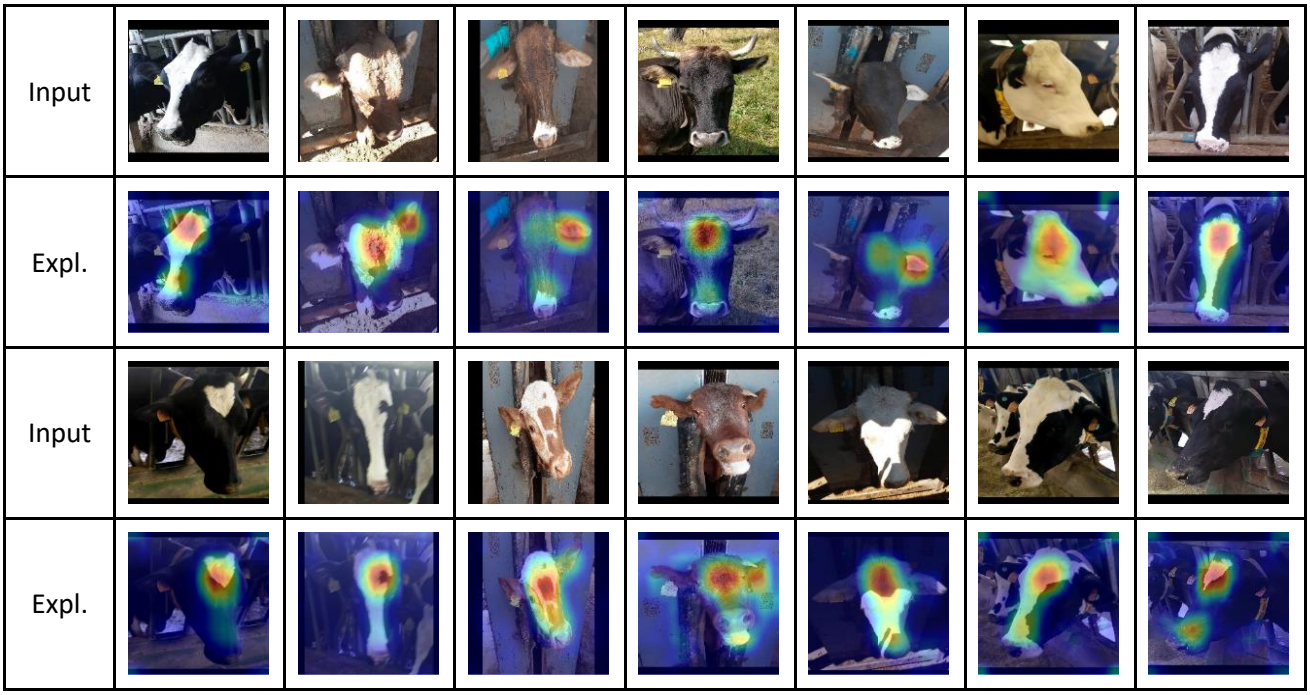


Figure 13: Explanation obtained from CowNet++ on a set of examples from the training set

CONCLUSIONS

Significant technical advances have been made to the recognition network. Thus was born the CowNet ++ network which, thanks to the new solutions introduced, reaches higher performance levels than its previous CowNet version.

The skills acquired over the phases and the close collaboration with AImageLab team, have also made it possible to better understand how the network acts in the recognition phase. Through a qualitative investigation, it was shown that the head of the animal and, secondly, the muzzle are the aspects "judged" most significant by the network and therefore "worthy of attention" during recognition.

The research and development activities have allowed us to achieve excellent results in this area, with accuracy levels (k-NN-MR at the top1) of 93% on a group of 115 animals distributed on 3 different farms (previously it had stopped at 87%). The study phase was also initiated, aimed at defining the best solutions for putting the technology into production on the web and at starting an effective collaboration with the National Animal Identification System working group within IZS.

3.4 PhAID large dataset test and web prototype solution

Parallel, the improvement of CowNet++ network and the development and deployment of the app were carried out.

This phase of the project was mainly concentrated on refining the layers that make up the neural network to improve the accuracy of the calculation of the embedding of each animal. The increase in the number of animals has led to a relative decrease in accuracy, in line with expectations and with what happens in human recognition.

The improvement of the prototype of the web application allows to use and test the system from any computer. The MLOps deployment strategy (Figure 14) is applied and, starting from the embeddings, the weapp can translate the results of the re-identification task on the computer screen with the animal ID identified with the PhAID.

Development of the prototype web application

The objective is to deploy the PhAID model in a web application useful for all stakeholder to use PhAID system. In fact, if the model is not deployed, the action of developers is necessary to use the animal identification system and to run the network on a dedicated computer with all the necessary software and run the task of re-identification. The development and Model deployment of an application allows any operator to be able to test the system and recognize cattle by uploading a photo of the dataset on a web page. Making the animal recognition system usable on any device and to a wide audience of users is the main objective of the study and is always a challenge to find the best solution to deploy ML model.

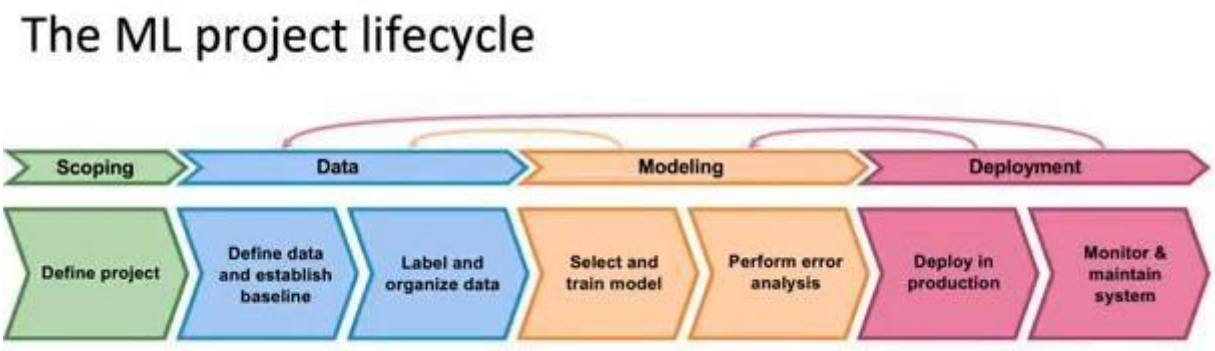


Figure 14: MLOps project lifecycle

One of the main tasks in which the scientific community and code developers all over the world are actively engaged is the deployment of Artificial Intelligence systems. Farm4Trade has created a dedicated research team and adopted a Machine Learning (MLOps) deployment model to transform the networks developed into a web model that can be used on any browser.
The development of the prototype is a solid basis for the final application of the PhAID system. The phase work is focused on the entire cycle which includes four main phases as shown in Figure 14:

1. Scoping and project definition;
2. Data collection;
3. Neural network modeling;
4. Deployment and release of the web application.

The scoping of the project is well defined and it is exactly the task of re-identification the animals.
Data collection takes place in two distinct but related moments: Farm4trade has developed a mobile application called SnapAnimal that can be used to collect photographs or videos of the animals adding a wide range of metadata. This app, usable from a smartphone, sends the data to the company storage, where the images are further refined before being processed by the neural network that calculates the embeddings, building the system dataset.

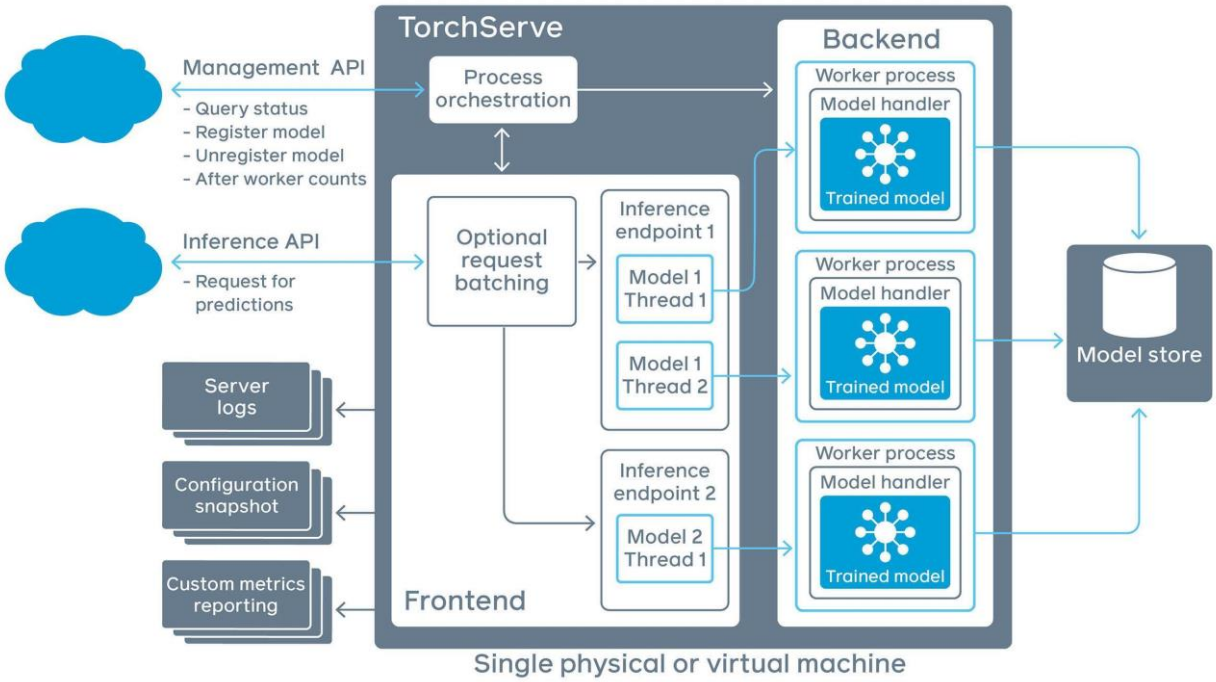


Figure 15: PhAID model deployment

Network modeling occurs immediately after the neural network has been trained with the dataset. This training phase is not always necessary and the network model created is therefore reusable even in different re-identification tasks without the need for substantial modifications. The analysis of the performance of the model and any errors (including the reduction of the accuracy of the re-identification phase of the single garment) is always monitored to make the necessary corrections to the network and consequently to the model.

The system deployment phase involves the translation of everything into a web application that can be used on a browser or as a mobile application. This last phase requires an important effort to translate the network model and the distance reading of the embeddings into web services. The Farm4Trade development team has created new deployment solutions also using dedicated Docker containers. The deployment system can be summarized in Figure 15.

The image highlights the main processes and fundamental parts of the application which is based on the use of Torchserve which is a structured translation system of the python language into web services. Everything is concentrated in a docker-compose consisting of the following containers:

1. The worker with the Torchserve system and models;
2. The APIs and database that read the information in the worker and translate it into APIs for the frontend;
3. Frontend in Flask for display;
4. The proxy for orchestration.

The database that holds the dataset and image embeddings is currently the numpy derived from the Cownet++ network.

Data collection and datasets

The dataset were increased with new picture of animals coming from: Clementini in Contigliano (RI), Di Marco in Amatrice (RI) and Krumhuk in Windhoek (Namibia). Because of the diversity of breeds and hybrids present in these farms, we decided to carry out the two acquisitions, performed after some time, necessary to validate the ability of the network to recognize the same individual even when the acquisition of new images is carried out by a different operator, with a different smartphone and in different environmental and lighting conditions.

Farm	Location	Country	Number of Animals
Clementini	Contigliano (RI)	Italy	24
Costantini	Castel Frentano (CH)	Italy	33
Di Marco	Amatrice (RI)	Italy	57
Martin	Rocca Santa Maria (TE)	Italy	26
Krumhuk	Windhoek	Namibia	1
Howgate	Edinburgh	United Kingdom	59
TOTAL			**200**

Table 19: test dataset (Query and Gallery). For each farm, the location, country, and number of animals are reported

The total number of acquired identities exceeds 2,000, of which 200 are used exclusively for testing activities without the same individuals ever being employed in the training phases. As described in *Table 19*, 200 individuals are currently used to carry out the tests and performance the experiments. This ensures the robustness of the technology and the validity of the networks, which prove to be able to replicate excellent performances even on very different individuals and races.

As previously described, the test protocol always provides for the division of identities in the dataset into three categories, namely the **Training set** (Table 20), **Query set**, and **Gallery set** (Table 19).

Farm	Location	Country	Number of Animals
Boch	Pinzolo (TN)	Italy	52
Ferrara	Ferrara (FE)	Italy	4
Hombre	Modena (MO)	Italy	126

Martin	Rocca Santa Maria (TE)	Italy	197
Ritort	Pinzolo (TN)	Italy	52
Zeledria	Pinzolo (TN)	Italy	41
Venditti	Castelpagano (BN)	Italy	50
Okapuka	Windhoek	Namibia	890
TOTAL			**1412**

Table 20: Training dataset. For each farm, location, country, number of animals, and total images are reported.

Subset	No. of entities	No. of Exemples	No. of Images
Training	1412	2885	24019
Query	200	520	4781
Gallery	200	1108	9301
TOTAL	**1612**	**4513**	**38101**

Table 21: Dataset statistics

The new recognition performance evaluation metric, known as **Mean Average Precision** (mAP) and introduced with the previous Activity Progress Report, was also used for these tests. This metric is used in numerous scientific articles [16,17,18] about human recognition and, if combined with the accuracy already used (number of identifications performed correctly on the number of total queries), it allows a more profound evaluation of the ability to generalize a re-identification model. More specifically, in a recognition system based on candidate ranking (given a query, the database of known identities - gallery set - is queried and the records are sorted by decreasing distance), the Mean Average Precision evaluates how frequently results related to the query provided (ie same identity) appear at the top of the ranking. In summary, the mean Average Precision allows for investigating in a quantitative way the quality of the ordering based on the representations provided by the recognition network.

Network infrastructure

Following what was done in the previous report, we decided to evaluate the performances obtained on both architectural approaches implemented, to further validate the goodness of the CowNet ++ network towards the baseline. The main cornerstones of the models adopted are briefly summarized below.

CowNet - Baseline

CowNet is based on the principles listed below:

- **Multi-shot approach**: recognition is performed based on multiple images portraying the target

animal. If we limit ourselves to a single image, the risk is that semantically important details turn out to be hidden and/or occluded;

- **Feature extraction**: the use of a Convolutional Neural Network (CNN) to carry out the feature extraction part. In detail, the CowNet architecture is based on DenseNet [9];
- **Frames Fusion**: learning a mechanism for merging multiple images into a single final representation.

CowNet++

Here is described again the CowNet++ infrastructure and innovation. As for the CowNet++ network, the multi-shot approach based on multiple images of the target remains valid, as well as the merging scheme of the representations of each frame. Instead, the architecture responsible for analyzing each frame has been revised and improved.

In particular, features with **different granularity** and a **spatial attention** mechanism have been adopted.

Since the images acquired in the farms have high complexity and some non-salient or infrequent details can be easily ignored and do not contribute to better discrimination of the animal, it was decided to identify significant parts within the body of the image.

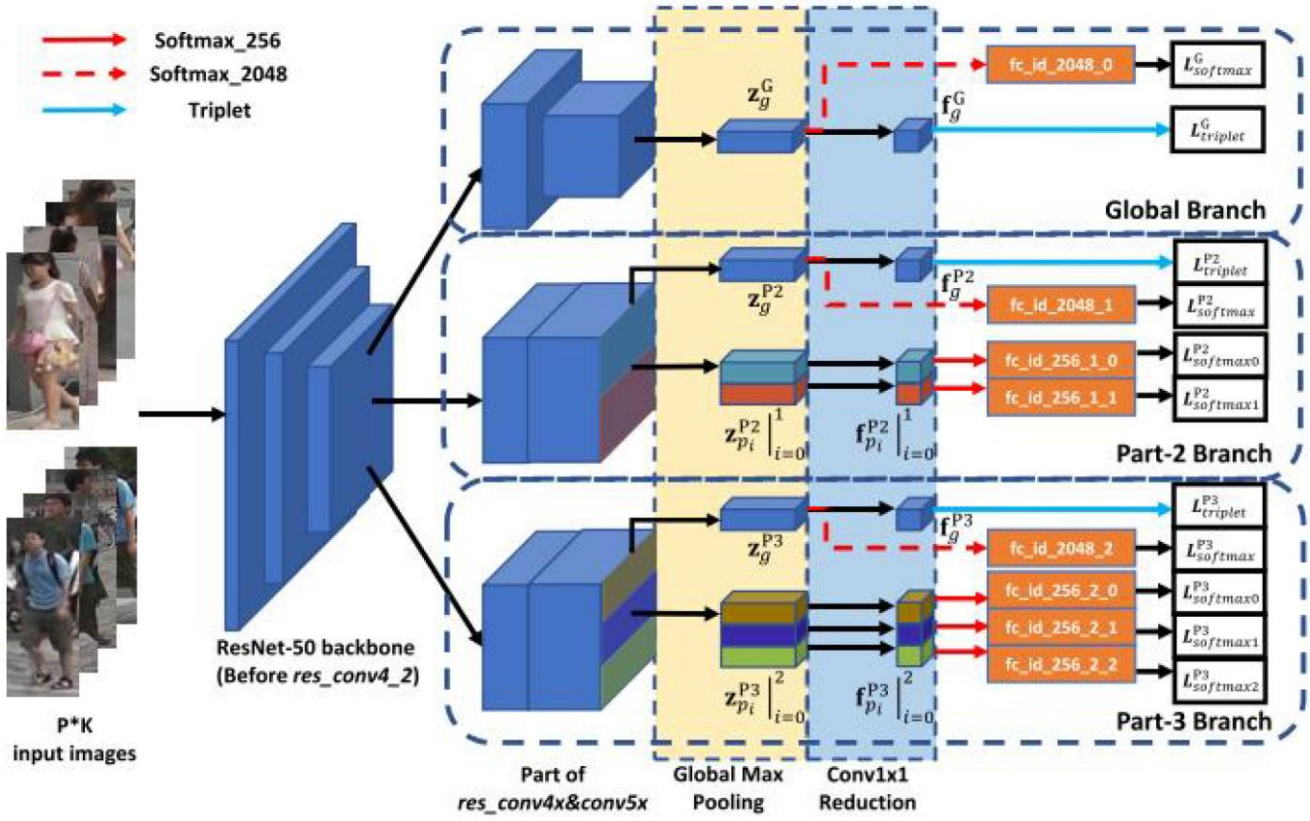


Figure 16: Graphical Representation of Multi-Granularity Network (MGN)

Previous approaches [20,21,22], called **part-based models**, divide the input image into local/relevant regions, which contain a small percentage of information about the entire body of the image. In this way, the feature extraction part can concentrate on these regions exclusively, avoiding the influence of external regions. CowNet++ implements these concepts through the adoption of two separate strategies:

- As already happens in CowNet, a part of the model takes care of extracting the features related to the entire body of the image;
- Unlike CowNet, two additional branches examine local regions of the image in different granularities.

From a technical point of view, the followed methodology - based on **Multi-Granularity Network** (MGN) - was introduced in [21] and is shown in *Figure 16*. Intuitively, the overall network is divided into two logical parts: the first extracts the generic features, and the second is divided into three parallel branches, each designed to operate at a different granularity.
As regards the **attention mechanism**, this was introduced to mitigate any background elements capable of introducing noise into the embedding, effectively affecting the final prediction. Spatial attention provides the model with the ability to focus more on some areas of the image (eg the animal's face) rather than others (eg background elements) and is introduced through attention masks.
In particular, CowNet++ is autonomously able to estimate them and subsequently apply them during the image analysis, through an estimation mechanism described in [24] and known by the name **Bottleneck Attention Module** (BAM). As shown in *Figure 17*, the idea is to equip the basic architecture with a new module (BAM), whose task is precisely to amplify specific areas of the activation maps.
As highlighted in the previous phase, an alternative to the mechanism just described is the use of input segmentation masks (previously estimated by an auxiliary network), however, they still have not brought the same levels of performance so we have continued in the direction currently more promising. Further and specific investigations are delegated to future developments.

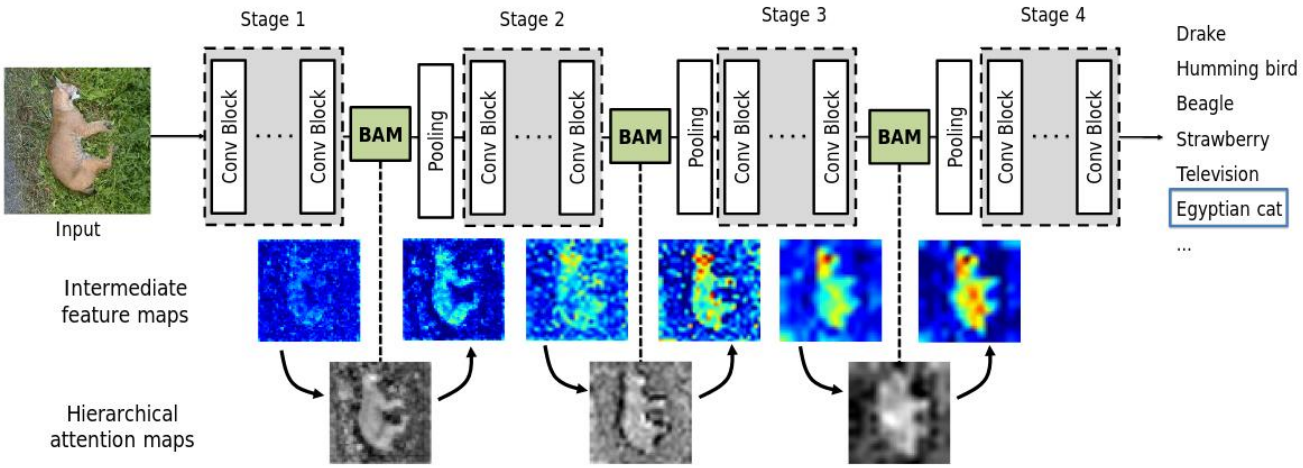
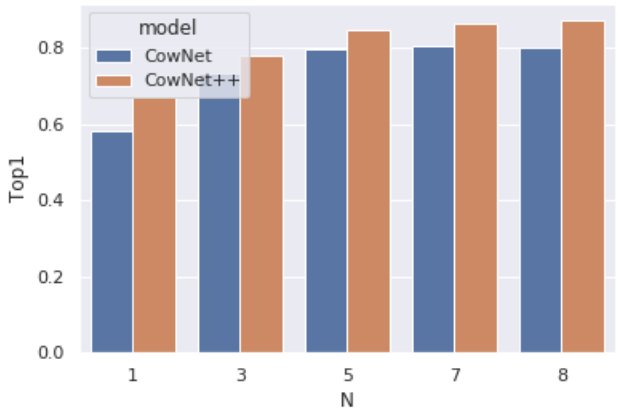


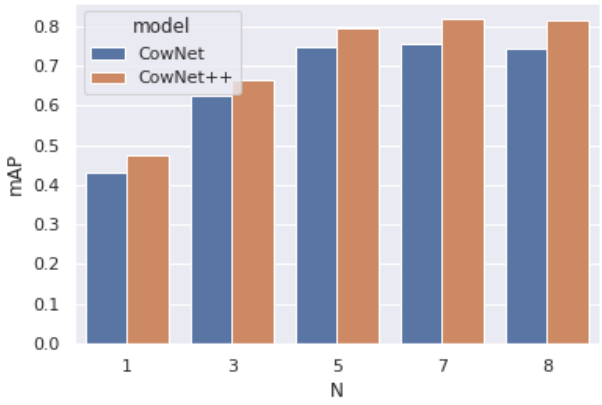
Figure 17: Bottleneck Attention Module (BAM) graphical representation

Experimental Results

The tests were conducted by comparing CowNet++ with its previous version, CowNet, with significantly larger test sets than the previous report (200 identities versus 115 before). The results, shown in *Figure 18*, are expressed in terms of top1 (also defined with the term k-Nearest Neighbors Matching Rate (k-NN-MR) when **K = 1**) and **mean Average Precision** (mAP). The two different variants were tested in scenarios with different degrees of difficulty: in this case, tests were conducted in which the number N of images (views) present in each example is progressively increased, thus providing, therefore, gradually more and more visual information to the model.



(a) Accuracy (top-1)



(b) Mean Average Precision (mAP)

	$N = 1$		$N = 2$		$N = 3$		$N = 5$		$N = 7$		$N = 8$	
	top1	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**	**top1**	**mAP**
CowNet	.581	.430	.683	.548	.735	.626	.798	.747	.806	.755	.800	.745
CowNet++	**.669**	**.474**	**.750**	**.582**	**.779**	**.663**	**.846**	**.793**	**.865**	**.819**	**.873**	**.816**

(c) Results in tabular format.

Figure 18: Comparison between CowNet and its new version CowNet ++

As highlighted by the results shown in *Figure 18*, the advancement proposed with CowNet++ leads to significantly higher levels of performance than that obtained with CowNet, thus attesting the quality of the architectural changes introduced and their robustness to a significant increase in the test set. Similar to what has already been observed previously, all the metrics of interest (Top-1, Top-3, and Top-5) improve when a greater number N of input images are supplied to the model. We also want to underline that the results shown were obtained **without further training of the network** compared to the previous phase. Both models, in particular CowNet++, are therefore able to maintain high metrics without the constant need to train the network again with each increase in test data. Introducing new identities, coming from 6 different farms, and different from the previous ones, the difficulty to which the model is exposed is considerably increased.

CONCLUSIONS

A PhAID web prototype application dedicated to the identification of cattle present in a given dataset has been developed. The application is available on the web and was developed starting from the neural network that performs the re-identification task by extracting the model and making the embeddings available in a Numpy database. The next activities include new phases for a substantial evolution of the prototype through the automation of the modeling and loading phase of the dataset, the use of relational databases for the storage of information.

When the research activity concerning the definition and evaluation of the model reaches a good level, also and in particular concerning the number of animals re-identified correctly in the testing phase, it becomes important to start evaluating the aspect of the hardware impact. In theory, larger models, with the right amount of data, allow for better performance (greater number of trainable parameters), however, they require a higher amount of GPU memory, as well as longer training times. This leads to higher overall costs. To optimize all this and be able to keep the commitment of hardware resources and therefore keep costs low, some network distillation techniques were tested to solve the problems described above. We, therefore, worked to compress the more complex models by testing a small neural network that was trained not on the actual task but to imitate the behavior of the larger network. This particular setting is known as the "teacher-student" paradigm, where a larger model plays the role of teacher and a smaller one plays the role of student. Currently, we have focused on evaluating the current model on a larger test set, trying to stabilize performance as much as possible without the need for further network training. Network distillation is an important stage that will be operated concretely and definitively in the final phase of the research and development activity when the model is completed and validated.

Tests were conducted using the Deep DenseNet121 backbone. Furthermore, tests were carried out with a variable number of N views. All the metrics of interest (Top-1, mAP) increase when a greater number of N views are given in input to the model.

The performance of the identification system is reported in Chapter 5. By testing the ability of the system to recognize an animal among those registered in the Costantini, Martin, Clementini, Di Marco, Krumhuk, and Howgate farms (total of 200 animals), the accuracy was proved to be 87.3% using these six farms as Gallery Set, then comparing an identity with the 200 present in the sameThis Activity Progress Report confirms the functioning of the infrastructure and the required re-identification task on a number of individuals equal to or greater than 200 identities. Compared to the initial work and tests, the number of animals used in both the training and testing phases has increased by almost 5 times and the CowNet++ network has proven to live up to expectations. To achieve these objectives, however, it was not necessary to make substantial changes to Cownet++, compared to what is described in the last report sent.

Some technical refinements, starting from a larger and more varied dataset, have allowed a scale-up of the entire infrastructure. As expected, with the increase of identities in the test dataset, a reduction in the performance of the network was observed in the percentage of accuracy. This is typical of neural networks in re-identification tasks when the units to have to be recognized increase (typical is the case of human recognition networks). The research and development activities have however allowed us to reach very solid results, with accuracy levels (k-NN-MR at the top1) of 87.3% on a group of 200 animals distributed on 6 different farms.

The research will continue to refine the model so that the number of individuals employed for the test impacts less the percentage of accuracy. However, alternative solutions have already been envisaged such as, for example, the adoption of clusters of networks based on the number of animals so that each model making up the cluster considers the number of animals with which the neural network has the best performance.

3.5 PhAID web application prototype evolution and deployment

Starting from what was done previously, in this phase new technical and experimental advancements are made with an important evolution on the deployment stack releasing new RESTful APIs aimed at carrying out the necessary tests and to ensure scalability towards a final web solution of the system. In this phase, no new identities (images) were introduced from new farms, but work continued on the animals already used in the past.

The experiments were carried out using the CowNet++ network to improve it, refine it and make it more and more suitable for the model of the implemented system and to allow future evolutions aimed at improving the inference capabilities, and therefore the recognition of the animals, through web applications. Parallel development and test of a new new architecture with a basic and conceptual structure quite different from the current CowNet++ has begun. This model largely exploits the concepts taken from CowNet++ but employs a network known as Vision Transformer [23]. This network represents an alternative to the classic Convolutional Neural Networks (CNN) and works on patches or predefined image portions. These types of networks are particularly difficult to train, requiring great attention in defining the parameters and usually a large number of images. The first results obtained are not yet equal to those reached up to now, but the conceptual idea behind these architectures and the experimental results obtained in other contexts make further evaluations valid and necessary.

Finally, during this phase, the focus was also focusing on improving the performance of the model for exposing the APIs to third parties and on refining some peculiarities of the networks. New developments are planned to improve the accuracy of the networks themselves, not only through deep learning techniques but also through the introduction of metadata of the animal such as gender, age, particular signs, etc. in prevision of the interoperability and integration with Italian National Animal Registration System. The use of other metadata can contribute on better performances given the further increase in the number of animals that will be registered.

Web application for third parties - connection to existing databases or applications - scalable system - deployment

One of the objective of PhAID development is to integrate it with third-party systems and existing database. For this reason, a substantial evolution has been made to release an integrated and scalable system capable of displaying web services for the identification of cattle also from third-party systems and, after adding new identities, capable of identifying new animals from other databases. In this way, the system can be used by any end-user or third-party applications that can send images of cattle starting from available photo galleries. The next evolution will be real-time network training, or at least at predefined intervals, based on the number of new identities present, to be able to update the system at any time without having to act manually. The new APIs integrated into the current system allow any client to transmit images of cattle to be entered into the system's dataset. The images thus become part of the dataset of the images to be identified. Once the identities have been sent, and possibly carried out a training cycle, the animals can be re-identified via a POST call and have the ID of the garment returned. Among the APIs made available, there are also those of the deployment system used, which allow not only to evaluate the metrics but also to have a broader overview of the percentage of use of the hardware and in general of the system logs. All this allows us to constantly monitor any critical issues at the infrastructure level, and intervene accordingly.

The development of the web application now allows any operator to test the system and recognize cattle by uploading a photo on a web page. Contrary to before, however, the current system can manage multiple requests both from the same user and from different users. Once again it is reiterated that making the animal recognition system usable on any device and to a wide audience of users is the founding goal of the current

project. This is necessary to make the integration with the national registry system (BDN) as little impacting as possible on existing software and to allow anyone who has their animal registered in the BDN to be able to identify them using a photograph.

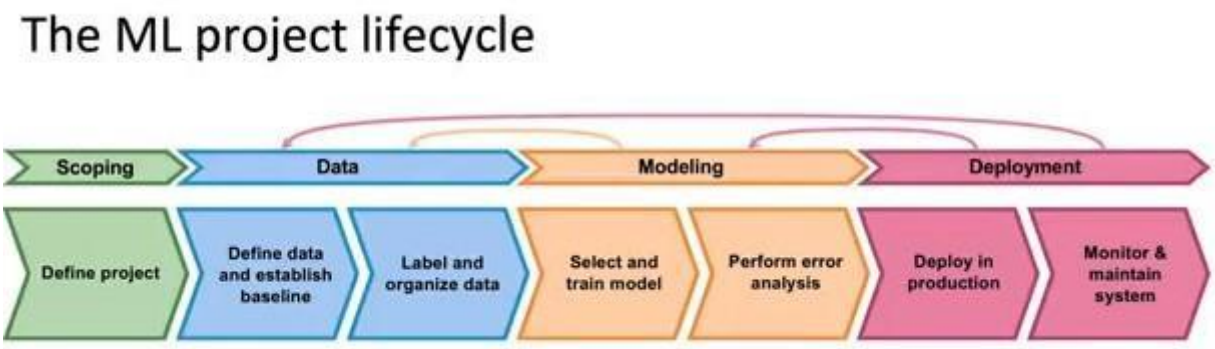


Figure 19: MLOps project lifecycle

The deployment of Artificial Intelligence systems is currently one of the most difficult challenges within the scientific and IT community. There are still several problems to be addressed, including the lack of GPU (graphic processing unit) with inference time (prediction), which makes the network slower in data processing and does not allow real-time analysis, but in "almost real-time"; the management of a large number of users and variable requests, the correct saving of the database and, finally, the monitoring of both the performance and the use of the hardware.
The Farm4trade research and development team focused, after the development of a prototype, on the identification of cutting-edge tools and solutions for scalability and the further advancement of the PhAID project. In this phase, we confirmed the use of MLOps approach for the PhAID deployment. The scoping is obviously always the same: re-identification of the cattle.
Data collection takes place in two distinct but related moments: using SnapAnimal mobile application, the user can collect photographs or videos of their animals and add a wide range of metadata. This app, developed by Farm4trade, sends the data to the company storage where the images are further refined before being processed by the neural network to calculate the embeddings. This constitutes the system dataset.
The modeling of the network takes place immediately after the data collection and labeling phase and the creation of the dataset. This training phase is essential at the beginning to obtain a performing network on the desired task. Once a trained network has been obtained, it is not always necessary to re-train the model for similar tasks, but it can also be reused in different re-identification tasks without the need for substantial modifications. Obviously, for the sake of clarity it is fair to say that with a quick re-training of the network, better performances can be obtained. Although similar, a new task always differs slightly from the previous one. In any case, these evaluations must be carried out by experts on a case-by-case basis by analyzing the performance of the system as a whole.
The analysis of the performance of the model and any errors (including the reduction of the accuracy of the re-identification phase of the single garment) is always monitored to make the necessary corrections to the network and consequently to the model.

Deployment workflow

The system deployment phase involves the translation of all elements into a web application that can be used via a browser or a mobile application. This last phase requires an important effort to translate the network model and the distance reading of the embeddings into web services. The infrastructural skeleton of the web app has been improved with new functionalities and architecture design.
All the components making up the application that will be described below have been collected and standardized each in a Docker container, which offers everything necessary for their correct execution, including libraries, code, and system tools. In short, Docker allows us to deploy the entire application on different hardware resources in a simple way without substantial changes.
In detail, everything was concentrated in a Docker-compose consisting of the following containers:

1. Frontend in Flask for uploading images and subsequent viewing;
2. The reverse proxy for orchestration;
3. Flask and related API endpoints. The database containing the information is managed within this container;
4. Torchserve and related workers for the inference of the trained model.
5.

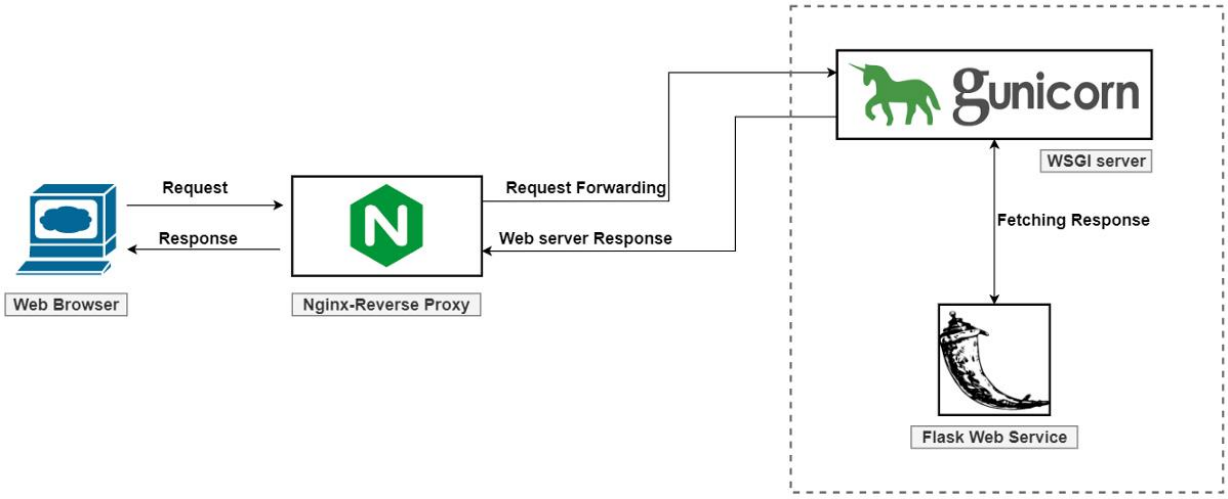


Figure 20: deployment model showing the first three containers

The current frontend is a simple Flask interface that allows the loading of one or more images of different animals, where you get in response, for each animal introduced, a predefined number of cattle with relative identity. The images returned are those images that the network has deemed most similar to those introduced. To make sure that the network has correctly identified the image, two additional pieces of information are returned: the identifier and the distance between the embeddings.

NGINX, a high-performance web server that acts as a reverse proxy (i.e. allows the client to retrieve content from one or more servers) and as a load-balancer for arriving requests, has been inserted as a proxy between the frontend and the API itself. The use of a reverse proxy such as NGINX is essential for the management of multiple and asynchronous requests, also thanks to the load balancing of requests between multiple servers available. The requests made by users, once they arrive at NGINX are sent to Gunicorn, a Python Web Server Interface (WSGI) HTTP Server. This service is compatible with various web frameworks, including Flask, and allows managing multiple parallel workers and therefore moving an application to production. When used stand-alone, Flask cannot manage or create more workers, and if on the development side this is not a limit, it is on the production side.

Once the requests have been managed, we arrive at the actual Flask application, which has been enhanced and improved with three main contributions. The first concerns the security aspect, the prediction endpoint

is now protected from unwanted access; through a login procedure, you get the token to attach to the call to get the answer. The second concerns the introduction of asynchronous programming to manage multiple requests from the same user as quickly as possible, or by parallelizing the computation of embeddings as far as possible from the potential of the current hardware. Finally, the last improvement introduced concerns the use of threads to carry out checks on the inputs entered and prepare the output data as quickly as possible.

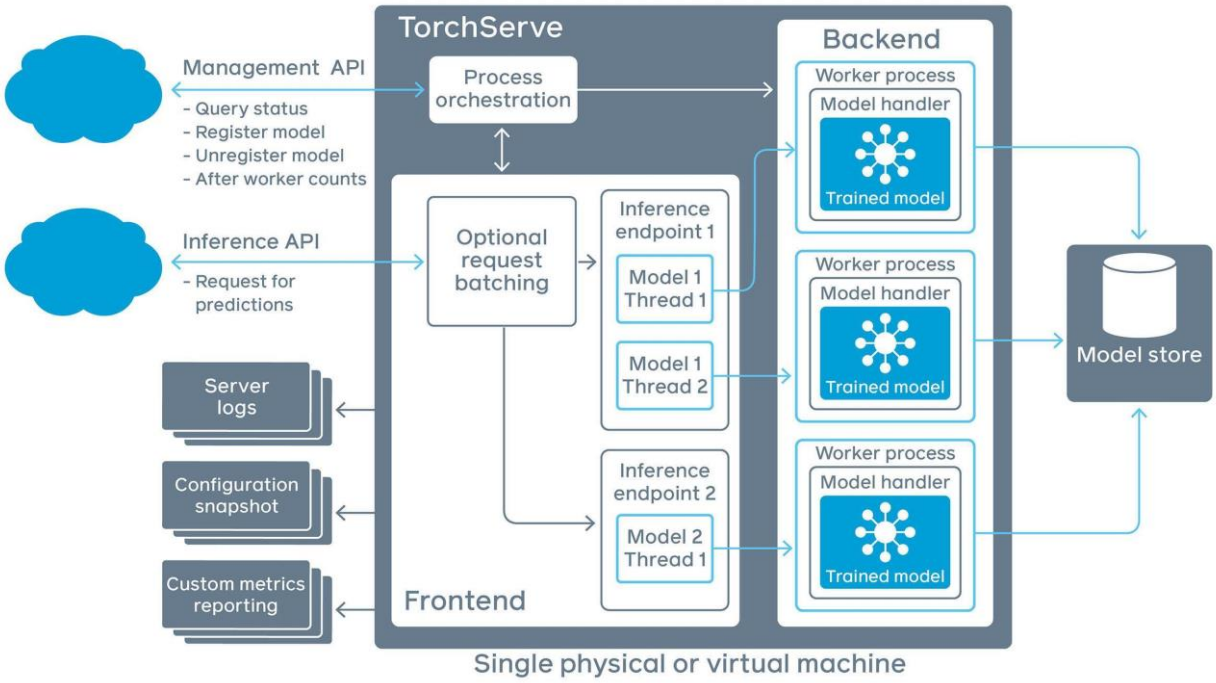


Figure 21: Torchserve general deployment model

The last fundamental component is the flexible and high-performance serving tool open-source available by the developers of Pytorch, called Torchserve (the general infrastructure is explained in Figure 21). Important new integration steps were made on the infrastructure for the full deployment. Torchserve allows in the first instance to save the trained model in a specific format (.mar) that can be used by the different workers, each of which operates in parallel and can handle more than one request. To fully exploit the features made available by this tool, it is necessary to have hardware that has sufficient cores available to parallelize the computation and empirically determine some configuration parameters.

The entire stack permits the deployment of the PhAID system and made available RESTful APIs. The addresses and endpoints of the web services to test the system and possibly exploit it for connections with third-party clients are not shown here for security reasons.

To manage future large amounts of data, work has already begun on the use of a relational database, which will allow easy extraction of only certain portions of necessary data (for example within a geographical buffer), thus not only reducing the memory load due to embeddings but also simplifying the research and prediction work of the algorithm itself. This step will be of fundamental importance to allow the algorithm to work in the most congruous and effective way possible when it will have to scale on a large amount of data.

At that level, automatic fine-tuning will then be necessary based on predefined criteria, such as the geographical area in which the animal resides, as well as the sex or age of the animal. In this case, several versions of the model will be available, suitable, and tailored to the various possible case studies. The balance between generalization of a model and overfitting (i.e. goodness of the model only in specific cases) is subtle and must be carefully managed in the best way possible. The final and most difficult challenge of this infrastructure will be linked first to the creation and then to the monitoring of an automatic retrain and fine-tuning system.

Network infrastructure

As previously mentioned, in this phase we focused purely on the infrastructural and deployment aspect of the model. However, in the field of computer vision and deep learning, there have been recently several innovations that question the hitherto undisputed role of Convolutional Neural Networks. In particular, approaches that exploit in part or entirely the concept of attention are becoming increasingly popular, the main and now best-known approach is called Vision Transformer. Convolutional Neural Networks work very well and learn relatively quickly about vision tasks as they present by construction what we call inductive bias (i.e. it is assumed that certain structures in an image can be repeated spatially). The architectures that derive from the Transformer, on the other hand, were born for natural language processing or time-series processing tasks, so they look for the correlations between every single letter and word within the input. This mechanism can be brought back into computer vision by thinking in terms of pixels and patches. In particular, starting from the input image, small patches are created on which the attention is calculated pixel by pixel. This approach led us to reflect on the fact that a patching network could extract more significant features on certain portions of the head/face of the bovine. As a result, some preliminary tests were carried out to evaluate this idea. Unfortunately, the first results were not particularly encouraging, but given the training difficulty of this type of goal, new attempts will be made. The best model currently remains the use of CowNet++.

To make the web application (prototype) usable and scalable, we have analyzed and tested different storing methods of the database of individuals currently used to test the correct functioning of the algorithm and the prediction capabilities. The current database is stored as a compressed numpy array. This allows, compared to previous versions, a considerable saving of memory and represents a fast way to access and extract the necessary information in real-time. Currently, the database collects three main pieces of information: the embeddings, generated at the time of registration of the individual to re-identify, the past images of these animals, and finally the related identification (ID). In particular, the use of numpy arrays allows us to save this information in just 3 files, one for each type of data. This is advantageous in terms of latency, as loading is done only at the application startup. However, work has begun on the introduction of SQL databases for database management, to make the system more robust, in the perspective of having to create a much larger database, but also to allow the storage of further context information about the animal on the database. This information (which includes metadata and features relating to the animal, but also to the network) will reduce the set within which the algorithm performs searches, making it faster and more accurate in the response, and will provide a more complete description of the animal itself.

CONCLUSIONS

PhAID has now the possibility to be connected to third-party systems. Thanks to the adopted network and the approach used for the development of the APIs necessary to perform the essential tests to ensure scalability towards a web solution of the system. The new APIs integrated into the current system allow any client to transmit images of cattle to enter into the system's dataset. Once the identities have been sent, the animals can be re-identified via a POST call and have the leader's ID returned. Among the APIs made available,

there are also those of the deployment system, which not only allow us to evaluate the metrics and performance on inferences but also to have a broader overview of the use of hardware and system logs. In this phase of work, therefore, no new identities have been introduced from new farms. Work continued on the CowNet++ network, created and presented previously, to improve it, refine it and make it increasingly suitable for the new model of the implemented system. To date, the system can connect to databases containing images of already identified animals and for which the embedding has already been calculated. This is possible thanks to the implementation and subsequent improvement of dedicated web services. The connection with databases containing identities other than those already registered requires further steps, such as re-training or fine-tuning the current model and the subsequent saving of new embeddings within the database itself. The ultimate, more complex, and ambitious goal will be to automate these processes to allow the model to interface with multiple databases. The introduction of this type of automation following the principles of continuous integration - continuous delivery (CI / CD) applied to machine learning.

To create an efficient and scalable system, in addition to the database described above, a Flask web application was implemented that uses concurrent programming, the management of asynchronous requests, and Docker containers. Above Flask, a Web Server Gateway Interface (WSGI) called Gunicorn has been set up, which allows the exploitation of multiple parallel workers and is, therefore, able to manage multiple simultaneous requests, resolving any delays or latencies in system responses. The introduction of concurrent programming and the use of asynchronous requests within even a single synchronous call made it possible to analyze the data in batches. This means that not only multiple users will be able to query the system at the same time but also that the same user will be able to upload multiple images at the same time.

To speed up the model and make it scalable even on higher numbers of parallel requests, we started experimenting with quantization and architectural pruning approaches. These approaches allow and will allow obtaining a less heavy network in terms of memory use and with faster response.

Chapter 4 Conclusions and future work

PhAID is an easy-to-use “contactless” biometric identification system capable of meeting the daily needs of authorities, operators and other actors in the agricultural supply chain. PhAID aims to improve the traceability of all animals and everything related to them, both inside and outside the farm. Farm4trade has now a consistent system able to re-identify the cattle through the web. The identification and traceability of animals are closely linked to the protection of public health and represent a complex challenge.

By exploiting the technologies of Machine Learning, Computer Vision, and the web, PhAID is capable to extract for each cattle image present in an input set, a representation (feature vector) that describes its visual characteristics and to aggregate these representations in a single more descriptive than the individual representations that compose it. The system is able to identify and re-identify cattle instantly in a few simple steps. Through the web the recorded data can be enquired by third-party applications for the management of identification data associated with animal health, livestock production, etc.

Dedicated app has been developed where it is possible to scan or upload one or more photos of the animal and register or associate the animal with an ID number and other metadata. Through the same app the user can made the re-identification task: The app scans the animal's head, the ML model system processes the images and automatically matches them with the animal ID.

PhAID is a system useful also for governments able to guarantee the traceability of data along the entire supply chain and improve the ability to control the spread of infectious diseases and prevent their spread. PhAID guarantees solid traceability necessary both for the authorities and for the daily needs of operators along the entire livestock production and food chain, while acquiring georeferenced mass data.

PhAID is a traceability system that surpasses traditional methods in many innovative ways:

1. Prevents theft, fraud and identity exchanges thanks to the reliability of the AI biometric system;
2. It allows to reduce the production costs of traditional systems;
3. Usable through a simple app running on Smartphone;
4. The animals must no longer be subjected to marking activities with obvious repercussions on well-being.

PhAID could represent a solution to the still unsolved identification and traceability problems in most “World Organization of Animal Health” (WOAH) member countries. Especially for developing countries, but not only, it could give rise to important commercial and development opportunities. Here below is listed the main PhAID use cases:

Establish the identity of one animal in the farm:

The number of animals could vary between 10-20 all the way up to a few thousands (1.000-5.000).

Establish the identity of one animal in a cluster of farms or in a defined geographic area:

The number of animals could vary between 10-20 all the way up to several thousands (10K to 20K).

Establish the identity of one animal in the Region and possibly in the Country:

The number of animals could vary between few hundred thousand (100K to 500K) all the way up to several millions (3M to 6M).

Establish if an animal belongs or not to a herd or a group:

If an animal is lost or found outside its stable / paddock it could be necessary to establish if the animal belongs or not to a certain group in order to be able to put it back from where it originates.

Establish if an animal belongs or not to 2 (maybe up to 5-10) different farmers:

if an animal gets lost and 2 or more farmers claim it is theirs, it would be important to establish if it belongs or not to one of them.

Count how often the same animal does a certain action:

It would be important to be able to establish how often the same animal goes drinking or goes feeding.

Count how much time the same animal spends doing a certain action:

It would be important to be able to establish how much time the same animal spend drinking or feeding.

Establish if the animals that are registered in a cluster (such as a truck or a stable) are all there:

Once the identities of all the animals are recorded, it could be important to be able establish if all the animals are really in that group or if some of the animals doesn't match.

Verify if all the animals in a cluster (i.e. a stable) all performed a specific activity (feeding, drinking, milking) at least once or for the minimum required amount of time:

It would be important to be able to verify or otherwise alert the farmer if an animal doesn't perform a certain activity. For example, if an animal doesn't go drinking for an entire day it could be sick or have a problem or it may be missing

Establish or exclude if an animal is already registered in the database:

The farmers' obligation to register animal in a national database can be supported by PhAID.

The accuracy and performances can be obviously improved in the future through new experiments introducing new solutions in the network development, in the model and in the deployment. The main efforts in the becoming future will be both, for the deployment improvement introducing modern technologies and integrations and for the re-identification performances introducing experiments on the clusterization of PhAID through decentralized deployment following the needs of the final users.

A first study was launched to extend the skills acquired on cattle to the horse species as, in addition to the same individual identification needs, they are characterized by a high similarity of facial features.

The work made for PhAID infrastructure has been also used to develop the ADAL system, an automatic system for the lesions disease scoring in the abattoir. Improvements of the 2 systems are planned both for the model and for the deployment. The objective is to release in production the systems in short period of time available for all the stakeholders and company customers.

Appendix A

List of publications

Here is reported the list of publications made during the PhD course:

- Patrizia Colangeli, Ercole Del Negro, Umberto Molini, Sara Malizia, and Massimo Scacchia, ‘‘SILAB for Africa’’: An Innovative Information System Supporting the Veterinary African Laboratories, TELEMEDICINE and e-HEALTH, 2018
- Bergamini, L., Porrello, A., Dondona, A. C., Del Negro, E., Mattioli, M., D’alterio, N., and Calderara, S. (2018, November). Multi-views embedding for cattle re-identification. In 2018 14th International Conference on Signal-Image Technology and Internet-Based Systems (SITIS) (pp. 184-191). IEEE.
- Trachtman, A. R., Bergamini, L., Palazzi, A., Porrello, A., Capobianco Dondona, A., Del Negro, ., ... and Marruchella, G. (2020). Scoring pleurisy in slaughtered pigs using convolutional neural networks. Veterinary Research, 51, 1-9
- Bergamini, L., Trachtman, A.R., Palazzi, A., Del Negro, E., Dondona, A.C., Marruchella, G. and Calderara, S., 2019, September. Segmentation Guided Scoring of Pathological Lesions in Swine Through CNNs. In International Conference on Image Analysis and Processing (pp. 352-360). Springer, Cham.
- Bonicelli, L.; Trachtman, A.R.; Rosamilia, A.; Liuzzo, G.; Hattab, J.; Mira Alcaraz, E.; Del Negro, E.; Vincenzi, S.; Capobianco Dondona, A.; Calderara, S.; Marrucchella, G.. Training Convolutional Neural Networks to Score Pneumonia in Slaughtered Pigs. Animals 2021, 11, 3290. https://doi.org/10.3390/ani11113290

Appendix B

Activities carried out during PhD

Professional positions:

- CTO at Farm4trade: leading the development team of the company;
- Project Manager at IZS: leading internation projects implementation on animal health sector

Courses:

- Intelligenza artificiale, reti neurali e possibili scenari applicativi
- L'intelligenza artificiale I e II modulo
- Building Scalable Java Microservices with Spring Boot and Spring Cloud
- Apache Kafka and Sprig boot
- Pytorch and Deep Learning
- Google Machine Learning Crash Course with TensorFlow API
- Convolutional Neural Networks for Visual Recognition
- AI for Healthcare
- Developing AI applications on Azure
- Offline Web Apllications
- TorchServe: un framework di model serving PyTorch
- Machine Learning Deep Learning Model Deployment
- Deployment of Machine Learning models
- Spring and AWS
- Docker & Containers
- Deploy Machine Learning in production

Bibliography

[0] Paleyes et al. 2021 Challenges in DeployingMachine Learning: a Survey of Case Studies

[1] Rob Ashmore, Radu Calinescu, and Colin Paterson. Assuring the machine learning lifecycle: Desiderata, methods, and challenges. arXiv preprint arXiv:1905.04223, 2019.

[2] L. Bergamini, A. Porrello, A. C. Dondona, E. Del Negro, M. Mattioli, N. D'Alterio, and S. Calderara, "Multi-views embedding for cattle re-identification," in 2018 14th International Conference on Signal Image Technology & Internet-Based Systems (SITIS), pp. 184–191, IEEE, 2018.

[3] J. Zhang, N. Wang, and L. Zhang, "Multi-shot pedestrian re-identification via sequential decision making," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6781–6789, 2018.

[4] Y. Liu, J. Yan, and W. Ouyang, "Quality aware network for set to set recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5790–5799, 2017.

[5] S. Xu, Y. Cheng, K. Gu, Y. Yang, S. Chang, and P. Zhou, "Jointly attentive spatial-temporal pooling networks for video-based person re-identification," in Proceedings of the IEEE international conference on computer vision, pp. 4733–4742, 2017.

[6] A. Smeulders, D. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," IEEE Transactions on Pattern Analysis and Machine Intelligence, 11 2013.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F. F. Li, "Imagenet: a large-scale hierarchical image database," pp. 248–255, 06 2009.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," CoRR, vol. bs/1512.03385, 2015.

[9] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," CoRR, vol. abs/1608.06993, 2016.

[10] M. Ilse, J. M. Tomczak, and M. Welling, "Attention-based deep multiple instance learning," arXiv preprint arXiv:1802.04712, 2018.

[11] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua, "Neural aggregation network for video face recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4362–4371, 2017.

[12] S. Rao, T. Rahman, M. Rochan, and Y. Wang, "Video-based person re-identification using spatial temporal attention networks," arXiv preprint arXiv:1810.11261, 2018.

[13] M. Zamprogno, M. Passon, N. Martinel, G. Serra, G. Lancioni, C. Micheloni, C. Tasso, and G. L. Foresti, "Video-based convolutional attention for person re-identification," in International Conference on Image Analysis and Processing, pp. 3–14, Springer, 2019.

[14] C. Song, Y. Huang, W. Ouyang, and L. Wang, "Mask-guided contrastive attention model for person re-identification," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1179–1188, 2018.

[15] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in Proceedings of the IEEE international conference on computer vision, pp. 1116–1124, 2015.

[16] L. Zheng, H. Zhang, S. Sun, M. Chandraker, Y. Yang, and Q. Tian, “Person re-identification in the wild,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1367–1376, 2017.

[17] G. Wang, J. Lai, P. Huang, and X. Xie, “Spatial-temporal person re-identification,” in Proceedings of the AAAI conference on artificial intelligence, vol. 33, pp. 8933–8940, 2019.

[18] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, “Person re-identification by multi-channel parts-based cnn with improved triplet loss function,” in Proceedings of the iEEE conference on computer vision and pattern recognition, pp. 1335–1344, 2016.

[19] W. Li, X. Zhu, and S. Gong, “Person re-identification by deep joint learning of multi-loss classification,” arXiv preprint arXiv:1705.04724, 2017.

[20] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, “Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline),” in Proceedings of the European conference on computer vision (ECCV), pp. 480–496, 2018

[21] J. Park, S. Woo, J.-Y. Lee, and I.-S. Kweon, “Bam: Bottleneck attention module,” in British Machine Vision Conference (BMVC), British Machine Vision Association (BMVA), 2018.

[22] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-cam: Visual explanations from deep networks via gradient-based localization,” in Proceedings of the IEEE international conference on computer vision, pp. 618–626, 2017.

[23] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).