International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 15 (2018) pp. 12055-12062
© Research India Publications. http://www.ripublication.com

Development of a Mobile Application that Allows the Search of Lost Pets in Bogota using Geographical Location and Image Recognition

María Alejandra Ortiz Mora[1], Octavio José Salcedo Parra[1, 2], Miguel J. Espitia R.[1]

[1]Faculty of Engineering - Universidad Distrital Francisco José de Caldas, Bogotá D.C., Colombia.

[2]Faculty of Engineering - Universidad Nacional de Colombia, Bogotá D.C., Colombia.

Abstract

The next paper exposes the implementation of a mobile application, which allows the localization of lost pets in Bogota, through image recognition and GPS, considering the coincidences between people who lost their pets and people who reported them as found. To do this, it was used the PCA technic, one of the most used in the computational image recognition, and ArcGis, a set of software products that makes easy the compilation and analysis of geographic information. The signed images are transformed to a vector representation, choosing representative parts of the animal face that allows a better identification. In this way, with this mathematic tool, comparisons are made in the app in order to find of coincidences between dogs reported as found or lost. Meanwhile, the given places are compared to each other, finding coincidences between the data stored, allowing to report to the user that its pet could have been seen or found.

Keywords: Pets, Image recognition, Search, GPS.

INTRODUCTION

Throughout history, pets and companion pets have gone from being used to perform specific tasks or services to being a fundamental part of the family, considering them more than just an animal. This has meant that pets do not do housework and that is why they are kept in it, but have become a member of the family. For example, care about them is much broader, and this can be seen in the market, where there is now a wide range of beauty products, places dedicated exclusively to their care, food places where they can enter, . It is for these reasons that families do not want their pets to be in any danger, and therefore it is not among their plans to lose sight of them. This phenomenon of change of perspective has occurred worldwide, but has also been seen in Colombia where, according to Fenalco, six out of six households have a pet, with dogs being preferred [1]. Because of this, and given the growing importance of pets in every household in the country and its different cities, each family makes valuable efforts not to lose sight of them and always preserve them safe in their home.

However, given the friendly nature of pets, there may be situations where they are out of sight and away from home. Here a search is undertaken, which has been well supported by social networks and their wide range of scope. Despite this, and the facilities provided by social networks, there is a decentralization of information, as there are too many groups on dissemination of losses of pets, as well as pages, where it is often impossible to achieve the necessary dissemination, in addition to Not having any other type of search tools.

Given this scenario, and the importance that each family has a pet, a solution is needed to unify information regarding loss of pets, so that their search is much easier, faster and more effective.

The next project proposes an alternative to solve this problem and allow lost pet cases to be solved in the shortest possible time and with the greatest effectiveness available. For this, the proposed alternative is a mobile application that allows to consolidate all available information on pets that have been reported as lost or found, to facilitate their search. In this way, it will not only be possible to concentrate such information but also to carry out operations of image recognition and geographic location to make the search an effective and agile process.

BACKGROUND

At the level of scientific investigations have been developed various projects that perform facial recognition of canines, used for various purposes.

One was developed by Pusit Prasong and Kosin Chamnongthai in Thailand. In their project, the main problem is identified as the difficulty of identifying the breed of a dog, since in many cases the animals, despite belonging to the same breed, have different faces or there are similar dogs but belong to a Same race. This, the authors point out, generates a poor rating. One possible solution to this problem is to take some parts that are characteristic of the breed and use them for identification. However, this is a time-consuming endeavor to locate each of the local parts of the pet's face. In this way, the research aims to propose a method to improve the speed with which the classification of races is made using the size and position of local parts in the face of the dog. These parts are represented by means of own vectors, which are compared with templates designed for each race. Before testing the results of the research and its performance, the system was loaded with 3 images of each race divided into important parts (left and right ear, face without ears) to train the system. Subsequently, 350 dog images were used for 35 different dogs to test their speed. [2]

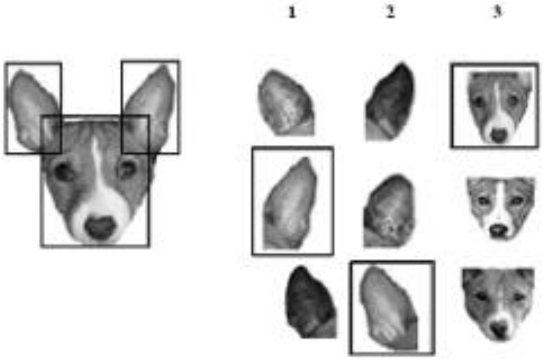


Figure 1. With each of the parts it is searched which one more matches the ones stored in the database. *(Source: [2])*

In an extensive collaboration between several universities and institutes (Nanjing, Dalian, IBM Watson Research Center), research was conducted with a curious purpose. This time, image recognition was used to generate an automatic model that recognizes tender images In order to validate the system, 3 datasets of different animals (rabbits, cats, dogs) were collected with tender images And not tender, obtaining that the model can recognize them automatically.In the proposal has a mathematical model that also uses vectors to recognize the parts of the animal [3]

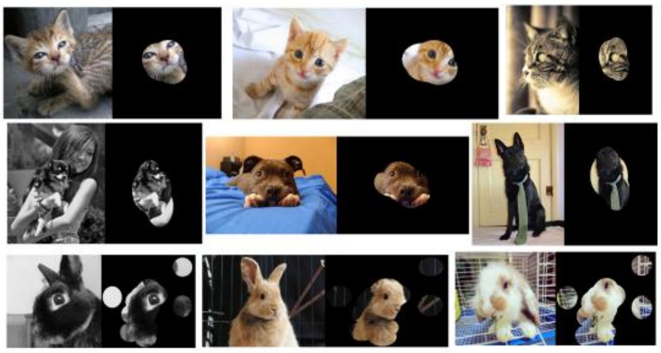
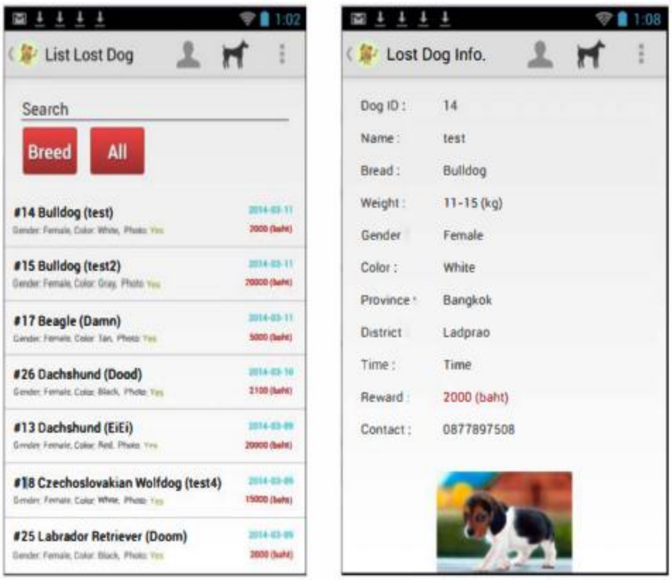


Figure 2. Recognition of tender parts in animals performed by the research model. *(Source: [3])*

LFD: "Lost and Found Dog" is an application implemented in Android which has problems of slowness and inefficiency when looking for pets, so this research proposes an application, LFD, that improves these behaviors. This application is based on the philosophy of collecting an own database that has any type of information about dogs, so that people can easily access it, without wasting time. This application provides the ability for users to easily exchange information about the dog with others. When the pet is lost, the user enters additional information about it, such as the location (specified by GPS, as in the proposed project), contact information, race, gender, among others. Thus, the proposed system will show the four most important matches with the dog that has been reported as lost. [4]



List Lost Dog

Search

Breed All

#14 Bulldog (test) 2014-03-11
Gender: Female, Color: White, Photo: Yes 2000 (baht)

#15 Bulldog (test2) 2014-03-11
Gender: Female, Color: Gray, Photo: Yes 20000 (baht)

#17 Beagle (Damn) 2014-03-11
Gender: Female, Color: Tan, Photo: Yes 5000 (baht)

#26 Dachshund (Dood) 2014-03-10
Gender: Female, Color: Black, Photo: Yes 2100 (baht)

#13 Dachshund (EiEi) 2014-03-09
Gender: Female, Color: Red, Photo: Yes 20000 (baht)

#18 Czechoslovakian Wolfdog (test4) 2014-03-09
Gender: Female, Color: White, Photo: Yes 15000 (baht)

#25 Labrador Retriever (Doom) 2014-03-09
Gender: Female, Color: Black, Photo: Yes 2000 (baht)

Lost Dog Info.

Dog ID :	14
Name :	test
Bread :	Bulldog
Weight :	11-15 (kg)
Gender	Female
Color :	White
Province :	Bangkok
District	Ladprao
Time :	Time
Reward :	2000 (baht)
Contact :	0877897508

Figure 3. Data entry interface in LFD. *(Source: [4])*

In an investigation by the KMUTT (King Mongkut's University of Technology Thonburi), the importance of classifying and identifying dog breeds for training and healing purposes is again highlighted by experts who are difficult to find, Which in this research tries to automate based on images of faces of dogs, using coarse-to-fine processing, which tries to simulate the way in which humans perceive visual scenes and the speed with the That this change of perception occurs. [6] With this method, the images are classified into five groups and, within each of them, the Component Analysis Principle (PCA) is applied to finally classify the race of the In this technique, the characteristics of the face are represented by means of a weight vector. A set of example images is used for each breed of dog, in order to Render the characteristics of it. With this set of images are generated vectors that serve as templates, so that at run time the new images (converted to vectors) are compared with the stored vectors and can be classified into any of the available breeds. Within the investigation we experimented with images of 700 faces of dogs from 35 different breeds and achieved an accuracy of 93% [5]

Many of the developments for monitoring and to prevent pets from losing their homes are more prevention-oriented, so research also aims to create devices that track and jointly locate. For this, Van Kirk Fehr has made a patent consisting of a pet locator system that includes a collar (with a telephone and unique owner code) and a central information core. Thus, the information of the pet and its owner is stored in the central, under the unique code and can be retrieved in response to a phone call from whoever has found the pet [11]

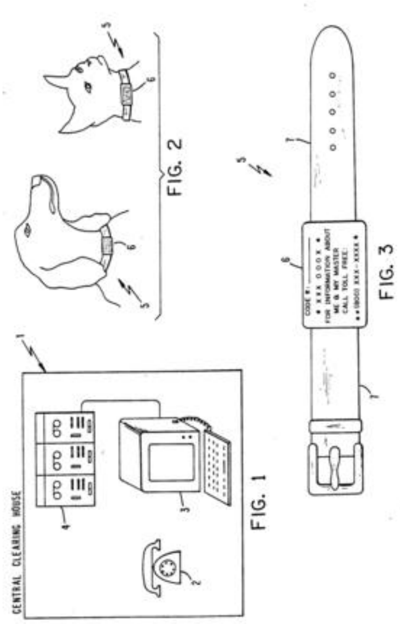


Figure 4. General scheme. *(Source: [11])*

When using geographic location to help the search for pets, you always need a hardware device that has the pet installed to be able to track it. For example, in [12] the system comprises a tag, a handheld device, a remote server and an application, where the tag is carried by the animal; The computer is in wireless connection with the tag; The remote control is used to store information corresponding to the mascot and the application establishes a connection between the server and the user, through a graphical interface. The system consists of identifying, through the handheld device, the code that each tag has the pet. Given this identification, it will be possible to access the information of the mascot (by its unique code) and thus to report to the owners of the loss and encounter of the mascot, by means of a diffusion in the server. It can be evidenced that in this application the owner must be preventive, but not reactive, because if his pet did not have the system, it will be obsolete.

Similarly, there are applications that have been developed not only to report the loss of pets to their owners, but to shelters and animal rescue centers. This is the case of [13], which describes the step-by-step development of an application that offers the possibility of sending warnings regarding the loss or sighting of animals to a remote server for later reception in animal protection centers, Helping them locate lost or abandoned animals much more easily. This application only complies with the requirement to alert the community, but does not expedite the search process for pets

At the Spanish speaking level there is a single application for locating dogs that have been lost. Its name is Back2Gether (return or return) and was developed WundermanBA, an Argentine company. This application was developed only for the iOS platform, so it differs from the current project, being much more inclusive because of the number of people in Colombia who have devices with Android operating system. The application performs the location of the pet due to the coincidence between the person who reports it and the person who finds it (similarly to the project worked). It also allows you to upload a photo and basic pet data such as your breed, size and age. It also reports the date it was lost.

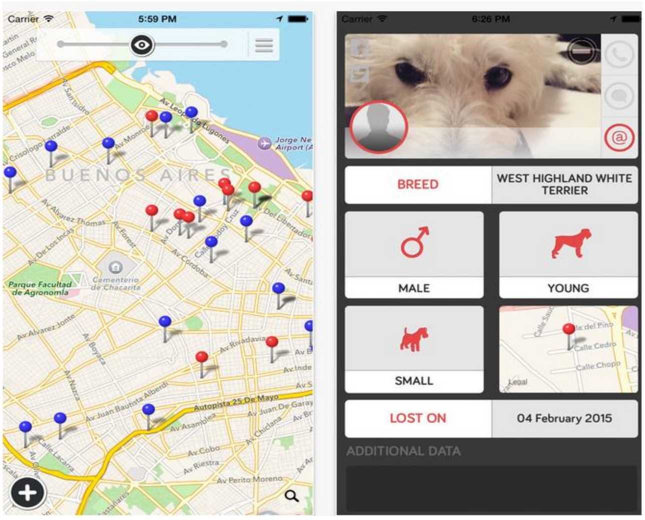


Figure 5. Location in the application. *(Source: [7])*

Likewise, the PiP application, developed for iPhone and Android, uses facial recognition technology to gather lost pets with their families. The application saves the face of the pet and when it is lost, will begin to analyze and match your face with the photos of the pets that have been found. Likewise, the application notifies the veterinarians, animal control entities and rescue agencies of the loss, as well as other users and users of social media in the locality. However, this excellent initiative is not available for Colombia.

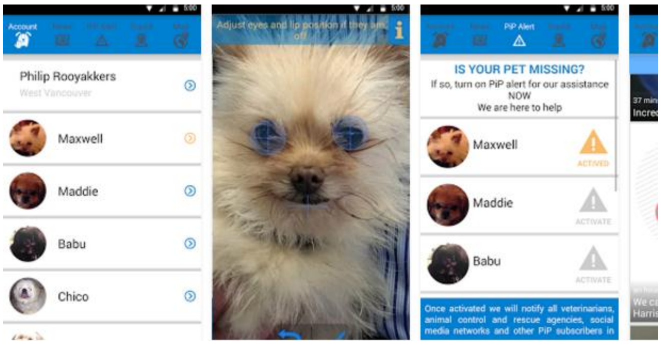

Account
Philip Rooyakkers
West Vancouver
Maxwell
Maddie
Babu
Chico
Adjust eyes and lip position if they are off
PiP Alert
IS YOUR PET MISSING?
If so, turn on PiP alert for our assistance NOW
We are here to help
ACTIVED
ACTIVATE
Once activated we will notify all veterinarians, animal control and rescue agencies, social media networks and other PiP subscribers in

Figure 6. Recognition of the pet. *(Source: [8])*

Tagg: It is an application that uses GPS, and consists of establishing a connection between the collar of the pet (which also includes GPS) and its location. The application allows to visualize in a map the routes realized by the mascot and sends alarms if this one has gone beyond the limits allowed by the owner. Also, it allows to count if the mascot performs sufficient exercise, since it measures its movement. This application is not available in Colombia.

Finding Rover: This app, available for iOS and Android, also handles face recognition to find pets. Whether or not the dog has a collar to contact its owners, or a microchip, with a photo will suffice to do so. As in an application explained above, the owner registers his or her pet in the database, and at the time it is found, it will match accordingly to alert the owner that their pet has been found. Likewise, it allows to share photos and updates of their own pets with their friends and publications to allow adoptions. [9] Not available for Colombia.

Tractive GPS Pet Finder: If the pet has a Tractive type device on its collar, this application will apply perfectly to it. The application allows you to track the animal in real time, follow its trail, set the safe zone for an alert to be sent if it is broken. This application does have coverage in the national territory, but if a Tractive device is not available it will not work. The price of the device is approximately 130 dollars. [10]

METHODOLOGY

For the implementation of the mobile application was used the IDE Eclipse (Luna Service Release 1 (4.4.1), along with the development tools for Android incorporated in it, such as the SDK (Software Development Kit) for the operating system, And the Android device emulator (ADT), where XML files were worked (Version 1.0), but the development of the different modules and the work of the data were done in the JAVA programming language.

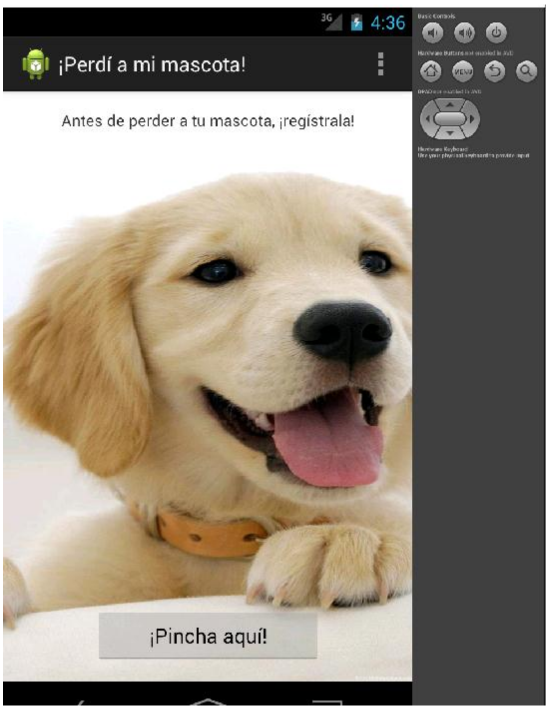
For the development of the geographical localization module, the ArcGis developer tool (ArcGis for Developers, v10.2.7) was used that provides the programmer with an effective way to create and manage geospatial applications. This tool consists of a downloadable kit that complements the IDE and contains libraries and documentation available for handling maps and locations within the application.

The image recognition module was developed within the application and with the Java programming language, using it as a tool to perform the mathematical operations required by image recognition based on PCA. Within this module it is necessary the decomposition of images in vectors and the training of the system with example images that allows to measure the degree of success when comparing images.

The application, for its part, is able to receive information from the user, which will be stored in a mySQL (5.1) database hosted in a hosting known as 000webhost.com. On this database both modules are fed to fulfill their functions (when accessing the images uploaded by the user and to the geographical positions provided by him). Also, the application is developed for devices with Android operating system, version higher than 4.0, having access to camera and GPS.

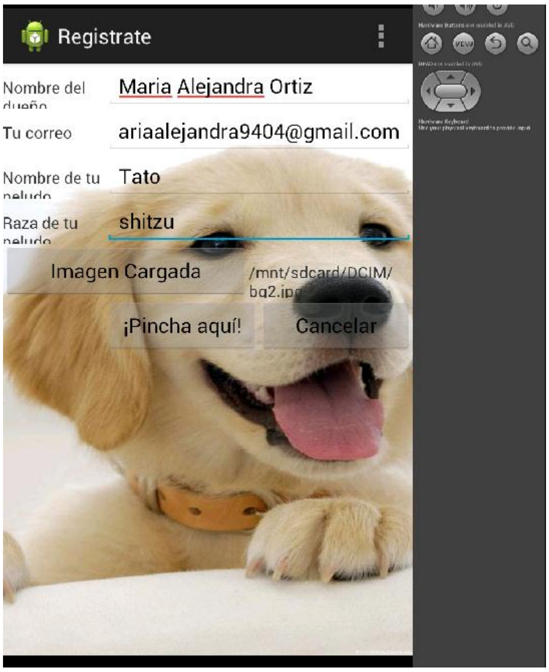
IMPLEMENTATION AND DEVELOPMENT

The user when installing the application the first thing that will have to do is a record of your basic data along with those of your pet. This is done so the pet is not reported as lost or found, as a preventive measure and to perform the respective management of the image that the owner has disposed of his pet.


¡Perdí a mi mascota!

Antes de perder a tu mascota, ¡regístrala!

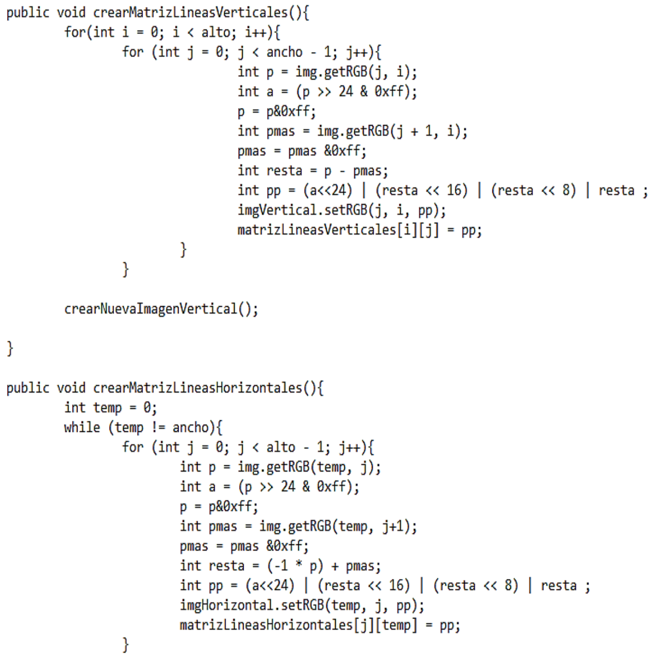
Figure 7. Graphical interface that marks the beginning of the application. *(Source: Authors)*


Registrate

Nombre del dueño	Maria Alejandra Ortiz
Tu correo	ariaalejandra9404@gmail.com
Nombre de tu peludo	Tato
Raza de tu peludo	shitzu
Imagen Cargada	/mnt/sdcard/DCIM/bg2.jpg

Figure 9. Graphical interface that collects the user's data along with the image of the mascot. *(Source: Authors)*

All this data is hosted on the host (000webhost.com). Once the captured image is processed, it is processed through the recognition of edges. To do this, the image is read, converted to grayscale, operations are performed on the matrix that represents it (horizontally and vertically), obtaining a new matrix and therefore a new image.



```
public void crearMatrizLineasVerticales(){
        for(int i = 0; i < alto; i++){
                for (int j = 0; j < ancho - 1; j++){
                                int p = img.getRGB(j, i);
                                int a = (p >> 24 & 0xff);
                                p = p&0xff;
                                int pmas = img.getRGB(j + 1, i);
                                pmas = pmas &0xff;
                                int resta = p - pmas;
                                int pp = (a<<24) | (resta << 16) | (resta << 8) | resta ;
                                imgVertical.setRGB(j, i, pp);
                                matrizLineasVerticales[i][j] = pp;
                        }
                }

        crearNuevaImagenVertical();

}

public void crearMatrizLineasHorizontales(){
        int temp = 0;
        while (temp != ancho){
                for (int j = 0; j < alto - 1; j++){
                        int p = img.getRGB(temp, j);
                        int a = (p >> 24 & 0xff);
                        p = p&0xff;
                        int pmas = img.getRGB(temp, j+1);
                        pmas = pmas &0xff;
                        int resta = (-1 * p) + pmas;
                        int pp = (a<<24) | (resta << 16) | (resta << 8) | resta ;
                        imgHorizontal.setRGB(temp, j, pp);
                        matrizLineasHorizontales[j][temp] = pp;
                }
```

Figure 10. Creating horizontal and vertical images. *(Source: Authors)*

In the previous code you can see the methods that create the images of the following figures. It performs an image manipulation using the RGB methods of JAVA. These allow you to work at the bit level with the RGB numbers of each image. Likewise, a subtraction is performed between the columns of the matrix representing the image within the code to perform the formation of the new vertical image. In the same way, for the horizontal image, but this time doing a subtraction between the rows that make it.

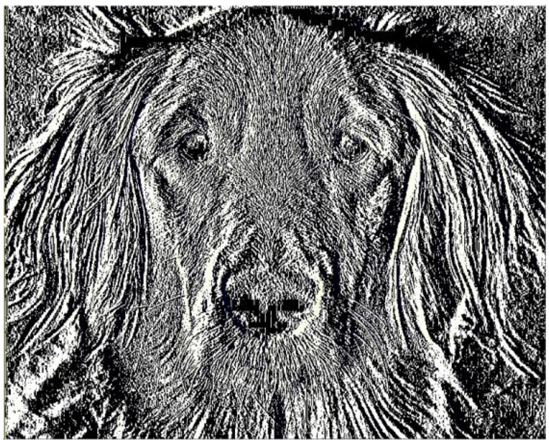


Figure 11. Treatment of the image loaded inside the application. (*Source: Authors)*

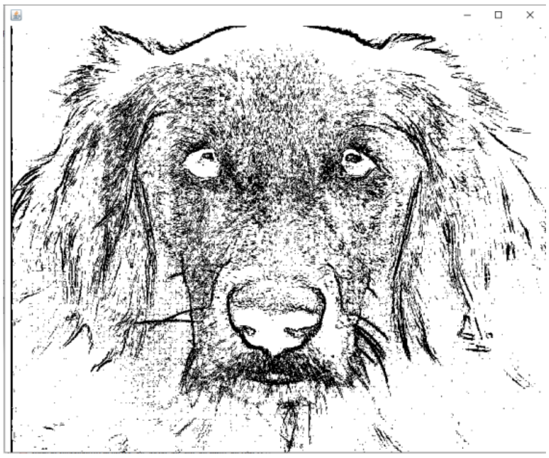
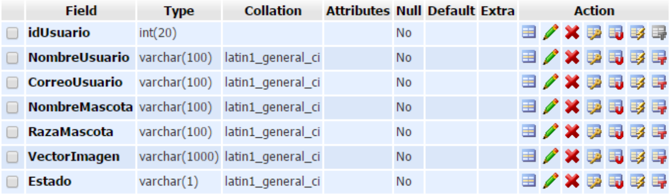


Figure 12. Treatment of the image loaded inside the application. *(Source: Authors)*

This new image (Figure 9) is stored in 000webhost.com, where the mySQL database is hosted. Within this database are stored the vectors that make up the image that the user has uploaded and which has just been processed at the time of registration.

This process is performed for each of the users who download and use the application, so a database is generated (Figure 10) which corresponds to the personal information of the users and the information of the processed image In image recognition.



	Field	Type	Collation	Attributes	Null	Default	Extra	Action
□	**idUsuario**	int(20)			No			
□	**NombreUsuario**	varchar(100)	latin1_general_ci		No			
□	**CorreoUsuario**	varchar(100)	latin1_general_ci		No			
□	**NombreMascota**	varchar(100)	latin1_general_ci		No			
□	**RazaMascota**	varchar(100)	latin1_general_ci		No			
□	**VectorImagen**	varchar(1000)	latin1_general_ci		No			
□	**Estado**	varchar(1)	latin1_general_ci		No			

Figure 13. MySQL database table structure for persistent user information. *(Source: Authors)*

Within the options that users have in their profile within the application, there is one that will report to the pet, or pets, registered as lost. Once this status is reported, it will be changed in that user's database to be taken into account when reporting pet found.

On the other hand, the user can also enter those pets that he has found, in order to do so, he must register a clear photograph of that animal and the location in which it has been found. Thus, when an animal has been reported as found, the following process is followed:

1. The image processing of the animal is performed, notifying the user that has made the report that it has entered the system.
2. A query is made on the images stored in the database, specifically taking into account those reported as lost, making a comparison between the digitized vectors and the processed image of the animal found (process explained later).

3. When matching the image of the lost pet with the reported one, the application will alert the user that the pet has been reported as lost, asking if the pet found might be the one you are looking for.
4. If this is the case, contact between both parties will be allowed. Conversely, digitized images of lost and found pets will continue to be compared until all have been compared (returning to step 2).

For carrying out the tests within the application, and more specifically the image recognition module, 10 processed images of 10 different pets were stored in the database. Then, a different image of some of the pets residing in the database was entered by the application, processing it in the manner explained above. When processing the image, vectors are obtained thanks to the pixel representation of the image. In an abstraction, the image is represented in Table 1.

Table 1: Pixels in the vector

255	212	7	1	3
211	237	3	9	0
218	240	8	12	2
241	241	5	4	0

Performing operations between rows and columns identifies changes in the matrix, which allow the identification of the edges that are highlighted within the figure. For example, the following table sets forth the result of operating the rows between them. Here it can be emphasized that one of the columns shows a significant change with respect to the others. For this, a criterion is chosen where, when the value of the matrix is greater than said criterion, a border within the figure is identified. For the identification of edges was taken into account that the criterion was 10. See Table 2.

Table 1: Identification criteria

43	205	6	-2
-26	234	-6	9
-22	232	-4	8
0	236	1	4

Based on this, when comparing the images housed in the database with the image entered, the image was divided into sections, ie the matrix into 3 sections (based on the image dimension), identifying the edges Of the most representative features (ears, eyes and nose). To do this, within each section, the number of positions in the matrix that have been identified as edges (that is, meet the assigned criteria) were counted. By having both images (saved and uploaded by the user) and performing this division and accounting, it was established that if, for each section, there is a difference of maximum 8, it can be considered as a candidate for the user to be notified that Your pet has been found, ie there is a possible match.

Regarding the geographic location, the tools of ArcGis facilitated the implementation of this module. When importing the SDK provided by the company, a search map could be inserted into the application, which allowed users to register locations that translated into coordinates that were stored in the host-hosted database.

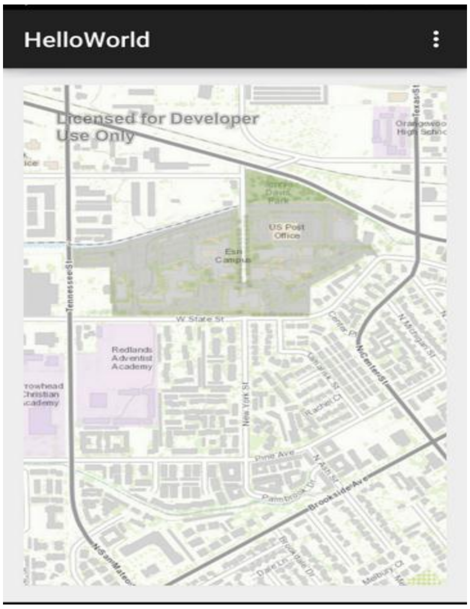


Figure 14. Map that allows inserting ArcGis in the application. *(Source: Authors)*

When comparing with locations inserted by the user that found the pet with the ones housed in the database, 10 examples hosted on the host were also taken into account. ArcGIS has a complete system for measuring distances and closeness between them. The calculation of the distance depends on the type of geometry of the entities but there are three basic rules to calculate this value.

1. The distance between two points is the straight line connecting them.
2. The distance from a point to a line is the perpendicular or the nearest vertex.
3. The distance between polylines is determined by the vertices of the segments.

For implementation, the first rule was taken into account, which can be graphically:

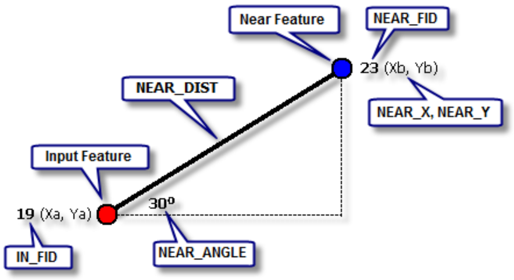


Figure 15. Distance and proximity calculation. *(Source: [14])*

In the previous image, Input Feature corresponds to the point entered by the user (which in this case is fed from the host database and corresponds to the coordinates of the pet that has been reported as lost). The other attributes are derived from an ArcGIS tool called 'Close', which calculates the proximity between one point and another. Near_FID corresponds to the ID of the next closest entity, if none is found, takes the value -1. Near_Dist is the distance between the Input entity and the next to it. This value is in meters. If multiple entities are generated nearby, Near_FC is used to indicate the route to them. Near_x and Near_Y are the X and Y coordinates that allow you to specify the location of the next entity. These data are specifically those that return to the application and are located in the map of Figure 11, so that the user can observe where someone has registered a pet as lost and inquire if it corresponds to the one that is looking for.

Within the specifications of the proximity tool you can define a search radius. According to [15], the longer the radius, the geographical location system will consider more characteristics, so that it will lose its accuracy. In the following image, you can see how the image clarity is lost by defining a search radius of 300 meters, while between 60 and 150 allows to observe better patterns and to perform better searches. For this reason, the radius chosen for the search in the project was 100 meters.

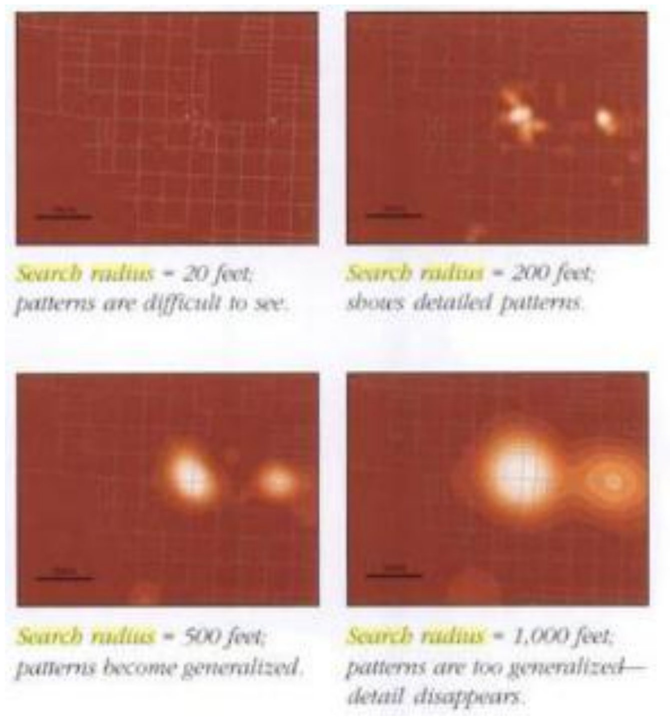

Search radius = 20 feet; patterns are difficult to see.
Search radius = 200 feet; shows detailed patterns.
Search radius = 500 feet; patterns become generalized.
Search radius = 1,000 feet; patterns are too generalized—detail disappears.

Figure 16. Patterns according to distance radius. *(Source: [15])*

DISCUSSION OF RESULTS

Within the image recognition module it was observed that the criterion of 10 allowed to recognize the edges of the images in a correct way. However, by establishing that the difference between the number of edges found in a stored image and a loaded image was maximum of eight, it was possible to show that 6 of the 10 test images entered coincided with the images stored in the data base, Which translates to an accuracy of 60%. Thus, this involved more than half of the successes in the test performed. However, the 4 cases that were not successful have common characteristics and correspond to a greater brightness in their characteristics.

Regarding the geographical location module, 8 of the 10 locations provided were successful in comparing them with nearby locations, showing a wide effectiveness (of 80%) when making the location through the geographic location.

Regarding other related works, it should be noted that they have used a more significant amount of images to perform their tests, as in [2] and [3]. However, their systems need this, since they must be trained so that each time they learn and perform the image recognition better; This, unlike the present project, does not have a feedback module that allows learning of the evaluated images but, for its part, must optimize its programmed recognition algorithms to work more efficiently.

Also, in [4], [5] and [6] use as PCA image recognition technique, as in this article, which demonstrates that it is one of the techniques most used to perform this procedure, yielding favorable results In the process of recognizing animal images, which is not so simple to perform.

Similarly, as shown in [7], [8], [9] and [10], these applications are strictly designed to cover US territory, so it is not appreciated that Colombia has not deployed this type of Application yet.

CONCLUSIONS

Regarding image processing, in spite of achieving more than half of the cases studied were successful (with a percentage of 60% success), it should be noted that, when performing vector analysis of the image based on pixels, If it has some special features (such as a lot of light or a lot of brightness), the analysis may be wrong. This should be corrected, as the user should not be limited when choosing the images that the application will load, since the application must be prepared to receive any type of image. However, it should be noted that for the six successful cases, images other than those housed in the database were used, so that it was possible to verify that the image recognition system is able to find matches between images.

On the other hand, the search by geographic location provided much more accurate and reliable results (with an 80% accuracy with the samples worked), due to the low volatile and always immutable nature of the coordinates provided by the ArcGis tool. This criterion proved to be much more efficient than image recognition and can provide much more confidence to the user using the application. Likewise, since a pet is reported as lost and its image enters the database, as soon as it is reported one close to it or similar in appearance, the system will take about 5 seconds to display all possible options related to The possible encounter of the lost pet. For this reason, the system and its speed will depend on the reports made by the people within the application, but not the technology used, because this responds almost immediately.

REFERENCES

[1] Mascotas, crecen las oportunidades de negocio. Artículo Avalieble in: http://www.fenalco.com.co/contenido/934. 2013

[2] Prasong, Pusit. Chamnongthai, Kosin. Face-Recognition-Based Dog-Breed Classification Using size and position of each local part, and PCA. Department of Electronics and Telecommunication Engineering. Bangkok, Tailandia. 2012.

[3] Bao, Yu. Yang, Jing. Cao, Liangliang. Cuteness Recognition and Localization in the Photos of Animals. Colaboración entre Universidad de Dalian, Chongging, Ninjing y el IBM Watson Research Center. 2013

[4] Chutichudet, Supabuth. Kanthathasiri, Tossapon. Ritsakunchai, Israporn. Wongsawang, Damras LFD: Lost and Found Dog Application on Mobile. 2014 Third ICT Internation Student Project Conference. Universidad de Mahidol, Tailandia. 2012

[5] Chanvichitkul, Massinee. Kumhom, Pinit. Chamnongthai, Kosin. Face Recognition Based Dog Bredd Classification Using Coarse-to-Fine Concept and PCA. Department of Electronics and Telecommunication Engineering. Universidad de Tecnología Thonburi, Tailandia. 2014

[6] J, Hegde. Time course of visual perception: coarse-to-fine processing and beyond. National Center of Biotechnology Information.

[7] Back2gether, desarrollada por WundermanBA. Avalieble in: https://itunes.apple.com/co/app/back2gether/id666053130?mt=8

[8] PiP app. Avalieble in: http://www.petrecognition.com/

[9] Finding Rover. Avalieble in: https://play.google.com/store/apps/details?id=com.findingrover.findingrover

[10] Tractive GPS Pet Finder. Avalieble in: https://play.google.com/store/apps/details?id=com.tractive.android.gps

[11] Van Kirk Fehr. Pet locator system and method. Patente número US 5036610A. 2014

[12] Chi Ho Ng, Wing Haw Ng, Kin Wang Chau. System and method for positioning lost pet. CN 104422944. 2013

[13] Sanjuan Quiles, Alejandro. Desarrolllo de una aplicación móvil en Android para un sistema de aviso y localización de animales perdidos. Universidad Politécnica de Valencia. 2014

[14] Conjunto de herramientas de proximidad. Avalieble in: https://pro.arcgis.com/es/pro-app/tool-reference/analysis/near.htm#C_GUID-8716F481-24E7-4581-95A3-350A2A6EAD68

[15] ESRI Guide to GIS Analysis, Volume 1: Geographic Patterns and Relationships.