International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 5, No.1 (Jan-2016)

http://dx.doi.org/10.12785/ijcds/050107

Archival and Retrieval of Lost Objects using Multi-feature Image Matching in Mobile Applications

Mohammed Ghazal, Yasmina Alkhalil, Samr Samir Ali, Fasila Haneefa, and Eman Rashed

Department of Electrical and Computer Engineering, Abu Dhabi University, Abu Dhabi, UAE

Received 15 May 2015, Revised 20 July 2015, Accepted 19 Sep. 2015, Published 1 Jan. 2016

Abstract: The rapid development of mobile computing technologies, as well as enhanced wireless communications, has paved the way for the development of the e-Government and m-Government systems. With the aim of increasing the accessibility of government services, such systems are still proven to be under-utilized, especially in the area of retrieving lost personal objects. In this paper, we propose a multiplatform mobile application for reporting and retrieval of lost objects in an efficient manner, rather than going through a manual procedure of filling up forms. The backend sever runs an object retrieval algorithm on the reported lost and found objects in the database, which is comprised of a textual search, a geographic information system filter, an image matching algorithm using Speeded Up Robust Features, followed by color validation using L*a*b* color space. Experiments have been performed on 40 lost objects against the database of 50 found objects with the accuracy of 95%, while providing a precision value of 100% and a 90% of recall. The efficiency and overall good performance of the proposed system can help in reducing the manual labor, as well as in providing fast feedback to the users.

Keywords: Mobile archival; Lost item retrieval; Matching objects; SURF algorithm.

1. INTRODUCTION

The evolution of mobile computations and wireless communications technologies has embedded itself in our lives due to its rapid development. Ubiquitous wireless devices used in daily tasks have led to the revolution of Mobile Government (m-Government) systems aiming to deliver services and information anywhere and anytime to citizens through the utilization of mobile technologies and applications [1], [2]. However, only the performance of the delivered services establishes its usefulness with many undiscovered applications to such innovations, especially since this is a desired area for further development with relation to smart cities.

One such application is the lost and found items retrieval utility. Effectively, concerned authorities and citizens' efforts for completion of a lost and found item procedure is an important affection factor. Manual traditional reporting of lost or found items requires a great deal of both time and effort. Consequently, citizens are hesitant in reporting such items due to the complex and prolonged procedure, with authorities required to expend many work hours in attempts to provide a hasty response to such reports.

To solve this critical problem, a large number of mobile applications has been developed pursuing improvement in the area of lost and found items services. Radio Frequency Identification (RFID) tags and Global Positioning System (GPS) mobile applications used for the tracking and detection of the tagged items locations are an example of such systems. However, RFID-based systems suffer from the disadvantage of its short range (1.5m), limiting its application to only locating personal items within the premises of ones house [3]. Nevertheless, losing and locating items outside the parameters of ones home is a far more serious problem where possible search locations for such an item are countless. Lost and Found Dog is another mobile application example; offering dog specialized database approach where key information about dogs are stored minimizing the time that is spent trying to locate lost dogs when found [4].

Nonetheless, report and location focused mobile applications for lost items are rare; therefore, we propose a new approach to improve the lost and found items reporting procedure with a multi-platform mobile application, Lost and Found. This application offers a user-friendly interactive form for items' reporting with the added feature of easy pinpointing to lost or found items'

E-mail: mohammed.ghazal@adu.ac.ae, 1029391@students.adu.ac.ae, samr.ali@adu.ac.ae, 1014898@students.adu.ac.ae, 1032497@studentsaduac.onmicrosoft.com

location interfacing the Google Maps Application Program Interface (API).

In addition, the mobile application permits users the utilization of item photos to attach with the submitted report. These images are used on the system's server to run image searching and matching algorithms for identification of a lost item within the database of found ones. Image matching algorithms are an active area of research with various approaches as in [5-11]. In this paper, our algorithm locates the point correlations between two images that would represent a similar sight or an object. The usual overall procedure for detection of discrete image correlations is selection of the significant points of interest at specific locations in the image, representing the surrounding area of every interest point using robust feature vectors, i.e. descriptors, and of matching feature vectors between different images [12]. Various descriptors and detectors are recommended in the literature with detailed performance comparisons in [12-16].

There are different visual search techniques and image retrieval systems used in mobile phone applications, such as Scale Invariant Feature Transform (SIFT) [13] or Compressed Histogram of Gradients (CHoG) [17] features. In this paper, we primarily use the Speed Up Robust Features (SURF) matching algorithm, considering that it is one the best descriptors [18]. SURF is a well-established matching algorithm based on SIFT feature detection method that is suitable for diverse applications as in [19-21]. We implement the SURF algorithm to extract the feature vectors from a query image of a lost item and compare it to those of the existing found items images in our database. When the lost item is identified as found by the algorithm, the system notifies the user in order for him to claim the lost item. Furthermore, we developed other versions of the mobile application to successfully integrate with different government sectors, organizations and the elderly. Our system's advantage over recent SURF sophisticated algorithms [22], [23] is the Global Positioning System (GPS) Euclidean distance filtering of candidates after a test query for generation of an initial set of candidates is executed. GPS applications such as [24-27] have a great impact in many fields with [27] directly applicable to the field of image matching research.

Moreover, we implement image retrieval algorithms [28], [29], [30] to enhance the performance of our system. Image retrieval mobile applications such as [28] are mainly designed for efficiency and speed. An added advantage of our system is that we incorporate various levels of consecutive filtering to not only speed up processing but also reduce the number of candidates in each level according to the searching criteria desired and reported by the user. Real-time systems suffer from computational delays as discussed and improved on in [29]. A further advantage of our system to this is the consecutive filtering techniques that reduce the number of candidates run in the SURF image matching, significantly boosting the speed and performance of our proposed algorithm. Our proposed system also implements validation of matches based on color and image segmentation techniques [31], [32] using L*a*b* space color with a detailed comparison of color spaces in [33].

The rest of the paper is organized as follows. Section 2 describes the proposed solution, the mobile application and its implementation, versions, and features, and the various integrated image matching and retrieval algorithms. Section 3 presents the experimental results and a brief overview of the user journey. Section 4 concludes the paper.

2. PROPOSED SYSTEM

A. *Initial System Design*

The workflow of our proposed system is described in Fig. 1. Lost and Found application allows the user to report lost or found items through a smartphone and save them to a database on the server. The items go through a set of validation steps to find the lost items within the list

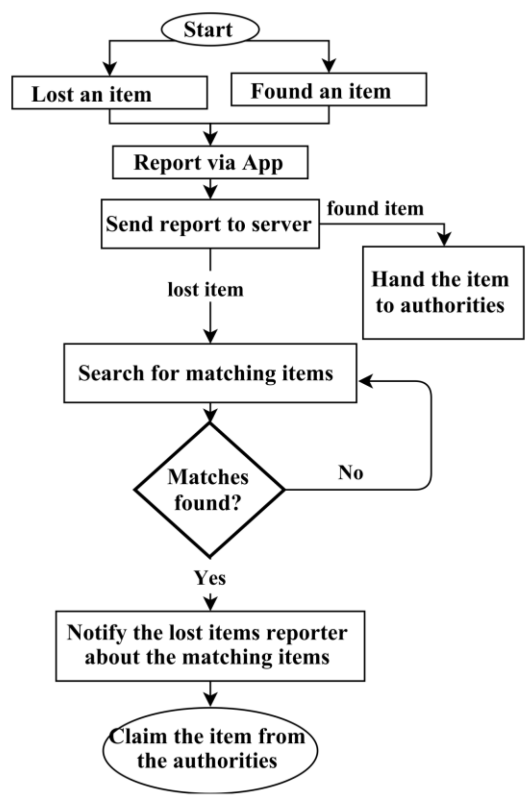

Figure 1. Lost and Found System Workflow

of found items. Once a set of matching objects is identified, the user of the lost item is notified about the possible matches. The actual item can be retrieved from the authorities provided the owners submit proof of their identity. The application is multi-platform, running on all current mobile operating systems. Additionally, the application is available in both Arabic and English.

Fig. 2 refers to the server workflow. Once a report is saved to the database, the server runs a text query search on the type of the item. When possible candidates are found, a location search using a geographic information system (GIS) is used to filter those items found closer to the lost item locations. We use SURF image matching algorithm to find matches between the possible candidates filtered by the GIS results. If the algorithm locates possible matches, we propose a further validation using color matching. We convert the images of possible matches into the L*a*b* color space and use the Euclidean distance to compare the histograms of color images. The lesser the distance, the higher the color matches between the images. If an item passes the color validation, it is reported back to the user as a possible match.

B. Variants of the Proposed System

Our proposed system caters for three different audiences: government sector, private sector, and the special needs. For the government sector, our application maintains the Police department as the managing authority for the delivery and safe keeping of found items for owners' reclaim. On the other hand, in private sector's businesses and facilities, such as companies, entertainment parks, universities, and malls, their respective security staff or entity is the responsible handling authority. Last but not least, we cater to the special needs' necessities and limitations through the utilization of various techniques to further facilitate the system's interface.

C. Localising Lost and Found Objects

To offer an easy to use interface for location reporting of lost and found items in our government-oriented proposed system, we incorporated Geographic Information System technology. This allows the user to simply pinpoint locations by adding markers upon the touch of the area of the map desired. Several location selections are also permitted for consideration of user's uncertainty of an exact location of the lost item. For the private sector system requirements, we utilize the building's floor plans for selection of locations in a similar approach to the government-oriented proposed system, providing the user with reporting task convenience and simplification compared to the traditional applications' style of form filling.

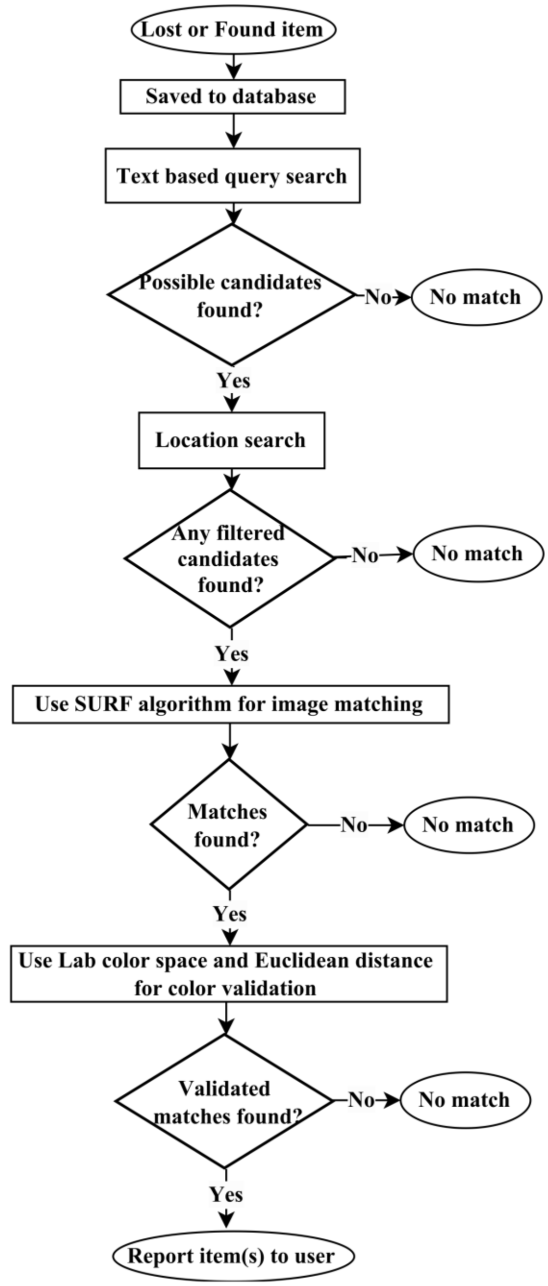

Figure 2. Server Workflow

D. Text and Location Search

As the lost or found report is saved to the database, a text based query search is run on the type of the object such as bag, wallet, electronic device, etc. This process generates a list of possible candidate matches. To eliminate false candidates, we perform a location search. For organization version, a search is run according to the location(s) chosen on the floor plan while reporting. On the other hand, for the General version, we obtain the geolocation coordinates from the global information systems. The geolocation search checks if the found item is reported within a 1 km radius of the location(s) where the lost item was reported. We use the Euclidean distance metric to calculate the distance between two points, where each point represents the coordinates of the found and lost objects, respectively. For instance, if $A=(x_s, y_s)$ is the location of the found item and $B=(x_i, y_i)$ is the location of lost items, the distance is given by

$$d(A, B_i) = \sqrt{(x_s - x_i)^2 + (y_s - y_i)^2} \quad (1)$$

where i is the number of the location where the item was lost. The values of B, or lost item locations, are stored in an array and each is compared to the found item location. The location check generates a filtered set of candidates, which we utilize in the object-matching algorithm explained next.

E. Object matching algorithm

To identify the lost item among the found items in the database, we use an object-matching algorithm. We present the most important steps of the algorithm in Fig. 3. As we can observe in Fig. 3, the algorithm extracts the query and the target images representing the lost and found objects, respectively. Next, the algorithm extracts individual descriptors of each image using the SURF method, as described in [12]. We use the descriptors to find the matching points between the two images. Using the found matching points and by eliminating deviating points, the algorithm attempts to detect the query object in the target image. In case of successful detection, we extract the images of the query and target objects that are further processed using color validation.

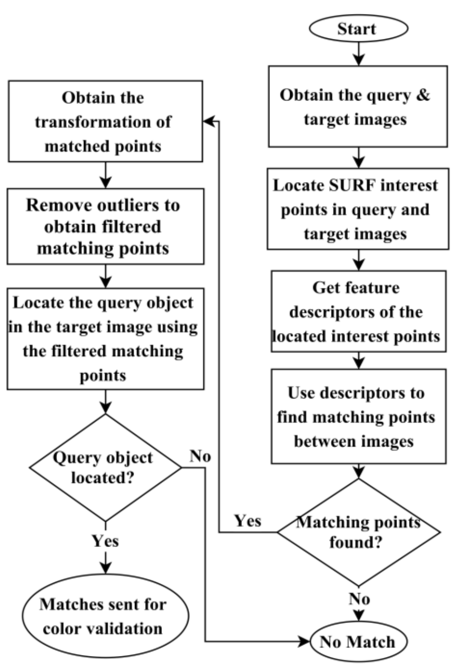

Figure 3. Object Matching Algorithm

The SURF blob detector uses an approximation of a Hessian matrix by employing integral images. Such an approximation decreases the complexity of the object-matching algorithm and enhances its computational performance. The SURF descriptor defines the order of Haar-wavelet responses nearby the area of the interest points. For better performance, the algorithm uses integral images for the approximation of SURF descriptors, as well.

In order to define the scale and the location of the SURF descriptor, the algorithm calculates the determinant of the Hessian matrix. Given a point $X=(x,y)$ in an image I, we can define the Hessian matrix $H(X,s)$ in X at scale s as

$$H(X,s) = \begin{bmatrix} L_{xx}(X,s) & L_{xy}(X,s) \\ L_{xy}(X,s) & L_{yy}(X,s) \end{bmatrix} \quad (2)$$

where $L_{xx}(X,s)$ is defined as the convolution of the Gaussian second order derivative with the image I in point X, as shown in (3). $L_{xy}(X,s)$ and $L_{xy}(X,s)$ are similarly calculated, as shown in (4) and (5) respectively.

$$L_{xx}(X,s) = I(X) * \frac{\partial^2}{\partial_{x^2}} g(s) \quad (3)$$

$$L_{xy}(X,s) = I(X) * \frac{\partial^2}{\partial_{xy}} g(s) \quad (4)$$

$$L_{yy}(X,s) = I(X) * \frac{\partial^2}{\partial_{y^2}} g(s) \quad (5)$$

The algorithm approximates the second order derivatives of the Gaussians using box filters in order to avoid discretion and cropping [12]. Furthermore, to speed up the computation of the convolution, the algorithm utilizes integral images and relative kernels. By employing the approximations of the Gaussian second order derivatives, D_{xx}, D_{xy}, and D_{yy}, we can define the approximate determinant of the Hessian matrix as

$$Det(Happrox) = D_{xx}\, D_{yy} - \left(0.9 D_{xy}\right)^2 \quad (6)$$

with the scale s being 1.2. Using the maximum of the determinant, which is interpolated in scale and image space, the algorithm can locate the interest points in the image [12]. Fig. 4 shows an instance of located interest points that the algorithm obtains using the SURF detector.

Figure 4. SURF Interest Points

The SURF feature descriptor uniquely represents the localized interest points. To ensure the descriptor invariance to the rotation of the images, the algorithm computes the specific orientation for each interest point. This is done with the help of the approximated Haar-wavelet responses. The algorithm determines the approximation of Haar-wavelet responses using integral images, which yields a better performance. Then, the algorithm rotates the surrounding area of each interest point towards its computed orientation. The application of rotated interest points aids in the calculation of the feature descriptor, as shown in [12].

The algorithm uses the feature descriptors for matching similar features of the query and target images. Similar features include outliers, which represent false matches. The utilization of the affine geometric transform eliminates these outliers. Moreover, the algorithm applies the transform in order to convert the query image to the coordinate system of the target image.

We reiterate this algorithm on all of the found objects in the database. The resulting matches are saved as candidates that could represent the possible match between the found and the lost object. To filter the candidates, we process them using color validation, where the objects are matched based on color. Since the SURF algorithm can only detect objects by their shape, the candidate matches can have the same shape, but can differ in color. Therefore, color validation is a necessary step to eliminate false matches. If the algorithm fails to detect any matches for a particular lost object, we run it whenever a new report for the found object enters the database. In that case, the user does not need to regularly file the report for the same lost object.

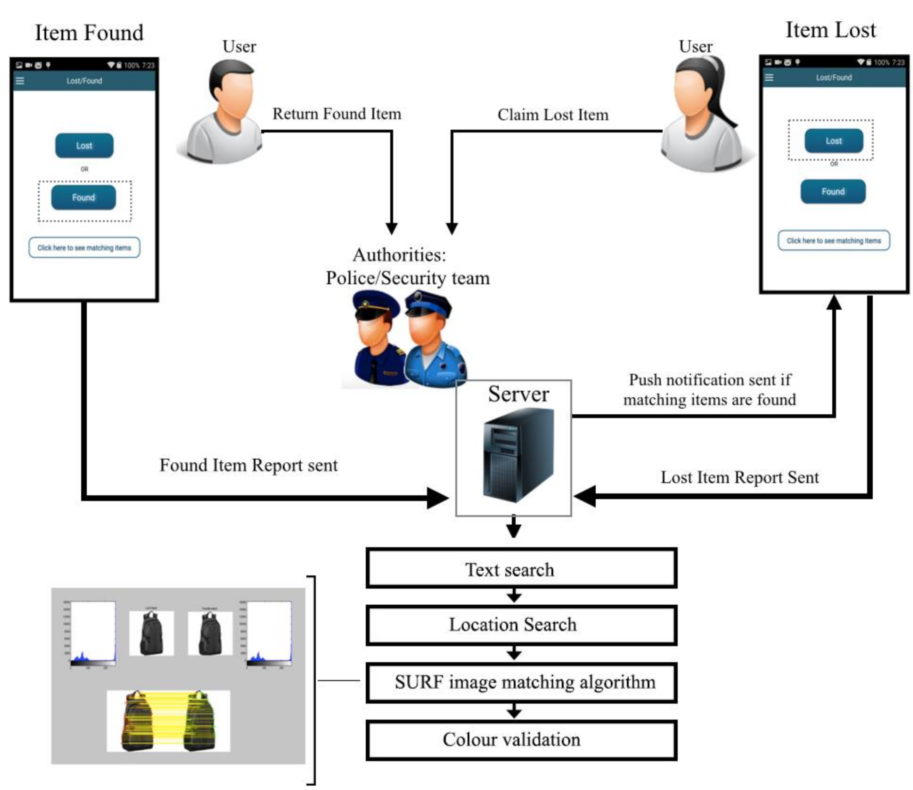

Figure 5. Overall System Diagram

F. Color validation

Color validation is used as a filter for the images matched by the SURF algorithm since the latter uses grey scale images to find matches. Once a match is located between images representing lost and found items, the matching objects within the image are extracted and converted into L*a*b* color space.

L*a*b* is a representation of color in 3 dimensions where a* and b* are the color channels and L* is used for light intensity [34]. Channel a* goes green to red from negative to positive axis, whereas channel b* goes blue to yellow from negative to positive axis [34]. The values of L range from 0-100, where 100 represents white, while the color channels range from -127 to 127 [34]. The main advantage of using L*a*b* space over RGB, which is the default format for the images, is the fact that L*a*b* is more useful for differentiating color components. L*a*b* has separate channels for chrominance and luminance while RGB does not. Moreover, RGB has 3 color components each for red, green and blue and no separate luminance channel [34]. This results in a reduced color variation. Hence L*a*b* color space is used for color validation.

Once L*a*b* color space components are found for an image, histograms are plotted for both a* and b* channels, which are further concatenated to form one histogram. The difference between two images is calculated using Euclidean distance between two concatenated histograms of the extracted lost and a found object images. If the distance is less than a designated threshold value, the images are considered to be a color match. If multiple images pass the color validation, then multiple items are reported to the user in order to choose the actual item.

G. Overall system

Our system not only archives the data but also utilizes a set of validation tools ranging from text search to SURF image matching to retrieve lost objects. The lost and found reports are stored in a database on the server where frequent searches for lost items are run until they are found. The search is run whenever a lost item is reported or a found item is reported. We go through a set of procedures to find matching items. First, we use text search where a list of candidates is generated based on the type of the item. Next, we filter the set of candidates using geographical positions. SURF object matching algorithm is run to find the possible matches. This list of matches will pass through a color validation algorithm. Once a match is detected between a lost item and a found item, the lost item owner is notified via push notifications. Every phone has a unique ID number, which is attached to the lost or found report sent to the server. By using this ID number, we report probable matches back to the phone. If multiple matches have been found, then the user will see a list of matching items from which the actual object can be claimed. Fig. 5 shows the overall system diagram.

3. RESULTS

Our proposed application compared to existing solutions automates the archival and retrieval of lost objects. We collect lost or found item metadata from the user such as description and the geographical locations where the object has been lost or found to create a user-friendly reporting experience. We also use other measures to increase accessibility to our image-matching powered lost and found service including cross-platform and multilingual development. The use of object matching algorithms reduces the costs associated with the manual search process and helps with locating objects faster. Hence, it reduces the anxiety associated with the prolonged manual processes. Our proposed system caters for large-scale deployment at the government or private sector level, especially during major events attracting a large number of people.

In Fig. 6, we depict the user journey to report a lost object. We deploy an interactive wizard-style form prompting the user to uploading existing or captured

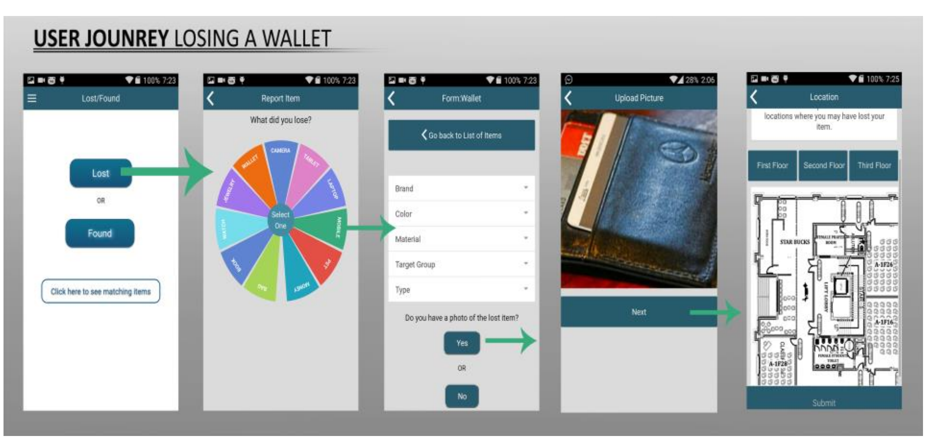

Figure 6. User journey for reporting a lost item.

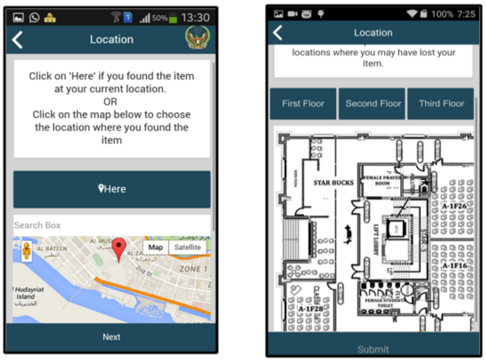

Figure 7. Government outdoor and private indoor versions

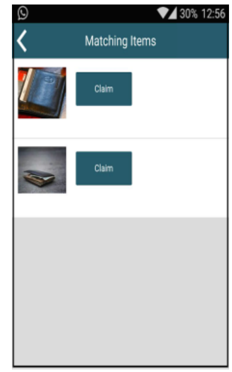

Figure 8. Matching objects list presented to user.

images of the lost object and the indoor or outdoor locations where the objects could have been lost. We also utilize a similar wizard to report found items, which improves the user-friendliness aspects of the proposed system.

Our proposed system uses alternative methods of data entry when dealing with users with special needs such as audio reporting and text to speech engine. The user is also presented with a gallery of images to select instead of textual menus while inputting the metadata associated with a lost of found object. The interactivity of the app is maintained in audio form in this case. After a question is played, the user is given a specific amount of time to respond with an answer. The answers are recorded and included in the report and can be analyzed in the future using speech recognition technologies to convert the audio report to a textual one. The deployment of geographical information systems to identify potential locations of lost or found item reduces the occurrence of false positives or negatives. In Fig. 7 we show the interface used in our proposed system in its both government outdoor and private indoor versions.

We use client-server architecture for our proposed system. The generated lost or found item reports are passed on to a processing server using a RESTful HTTP API. The report data is stored in a database of lost objects. We also maintain a database of found objects. Our image matching algorithm is then used on the addition of new reports to attempt to match the lost item to found ones using textual meta data supplied by the user, shape information extracted using SURF features, and color information based on the L*a*b* color space histograms. Potential matches are reported back to the user in the form of a push notification that leads into a description of the items. Then, the user can make a final decision and claim the found item as his or hers, thus initiating the process of validation and handing over by the concerned authorities. In Fig. 8, we present the list of matched objects displayed to the user.

In order to study the performance of the proposed algorithm for the retrieval of lost objects, we perform an experiment containing 50 reports of the found objects and 40 reports of the lost objects. Out of the 40 lost objects, 20 have a match among the found objects, 10 objects are of the same type and shape as their potential matches among found objects, but are of different color. The rest of the lost objects do not have a match. We test all 40 lost objects against the database of 50 found objects and observe the results. We successfully retain all of the candidates that satisfy the database query by type, and eliminate those that are of different types. Furthermore, by calculating the distance between the geolocation positions of the lost and found objects, we eliminate those objects that are found outside of the 1 km radius of the reported locations of the lost object. The actual results of the text and geolocation search can be observed in Table a. We consider a report of a lost earring, which is evaluated against 50 existing reports of found objects in the database.

Condition	**Number of objects**	
	Query (lost object)	***Target (found) object/s***
Initial	1	50
Search by type	1	10
Search by geolocation	1	4
Object-matching	1	2
Colour validation	1	1

a. Numerical results of the proposed algorithm

From Table a, we observe that the query search by type retrieves 10 objects as potential candidates. According to the geolocation information about the coordinates of places where the object was lost and found, the algorithm eliminates additional 6 objects. We choose only the objects found within the radius of 1 km of the reported lost object. We select the value of the radius considering the probable human errors while reporting the exact location of the lost/found object. In the next step, the algorithm extracts the images of the lost

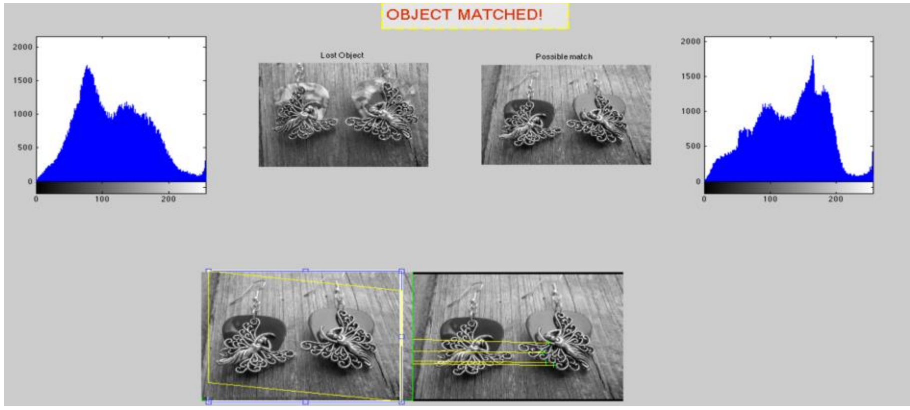

Figure 9. Object-matching results

and found objects, if any exist, and performs the object-matching algorithm proposed in Section 2. In this case, the algorithm reports 2 candidates among the found objects that have similar shape features compared to the lost (query) object. Fig. 9 shows the successful match between the query and target images.

All successful matches in this stage of the algorithm are processed for color validation. This stage eliminates those objects that have a similar shape, but differ in colors. Fig. 10 shows a case of matched objects that differ in color. By observing the concatenated histograms of a* and b* dimensions of the objects, we can visually notice the difference between them. This implies the difference in colors. By calculating the Euclidean distance between the two histograms, we obtain the value of 0.3322, on a scale from 0 to 1. Through testing with different smartphone cameras, we select the threshold value of the Euclidian distance between the histograms of 0.05. Any value of Euclidian distance below the threshold implies a high similarity between the colors of the extracted objects. We can observe that the object-matching algorithm is robust to rotation, as matching occurs even for instances when the query and target objects are under a different orientation. Moreover, object-matching algorithm is also invariant to scaling and size. The robustness of the algorithm can be seen in Fig. 11.

Finally, we evaluate the performance of the algorithm using 40 different test cases or lost object reports. We describe the performance using the measures of precision, recall, and accuracy. We denote the total number of testing cases with N and the number of correctly matched cases as P. Then, the accuracy ACC is given by

$$ACC = \frac{P}{N} \quad (7)$$

We denote the number of true positive cases by TP. TP represents the correctly matched objects. FP represents the number of false positives, when the 2 matching objects are wrongly evaluated as non-matching. Then the precision of the algorithm can be calculated by

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

The recall of the algorithm is the capability of the algorithm to locate the matches correctly. We define FN

as the number of cases when the objects are incorrectly classified as matching, or false negatives. Then, the recall

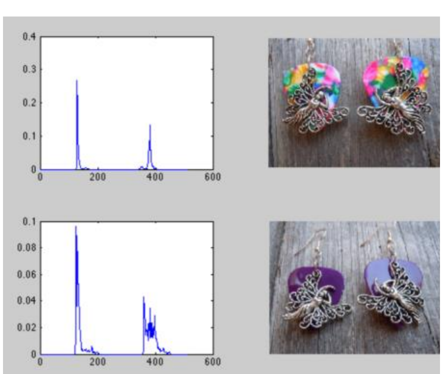

Figure 10. Colour validation

		Condition		
	Total Polulation N: 40	Match	No Match	Accuracy(ACC) (TP+TN)/N=0.95
Test Outcome	Match	TP=18	FP=0	Precision TP/(TP+FP)=1
	No Match	FN=2	TN=20	Recall TP/(TP+FN)=0.9

Figure 12. Confusion matrix for the proposed algorithm

Figure 11. Robustness of object-matching algorithm

is given by

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

The confusion matrix for the proposed algorithm is shown in Fig. 12. We denote *TN* as the number of true negatives, indicating the correct classification of non-matching objects. The accuracy of the algorithm is 95%, as there are 38 correctly classified cases out of the 40 tested. Moreover, the precision of the proposed algorithm is 100 %. We can observe that there are no occurrences of false positive cases, when the objects are incorrectly classified as non-matching. Finally, according to the obtained value of recall, the algorithm has identified the matching objects correctly in 90% of the trials.

4. Conclusion

In this paper, we proposed an innovative system for the archival and retrieval of lost objects using multiple features such as textual description, geographical information, shape matching using SURF features, and color matching in the L*a*b* space. Our proposed algorithm reduces the resources associated with the current, mostly manual process. Our performance on a 40 query lost items utilizing an image data set of 50 found objects shows an accuracy of 95% and a precision of 100%. The algorithm has a high recall percentage of 90%, which proves its efficiency and usefulness.

Acknowledgment

We thank the Office of Research and Sponsored Programs in Abu Dhabi University for its support.

References

[1] N. Ghazali and R. Razali, 'A preliminary review of interface design elements for mobile electronic government systems', 2014, pp. 217-223.

[2] T. Alhussain and O. Alfarraj, 'Biometric authentication for mobile government security', 2010, pp. 114-118.

[3] S. Chan and R. Kamoua, 'RFID for personal asset tracking', 2009, pp. 1-7.

[4] S. Chutichudet and D. Wongsawang, 'LFD: Lost and Found Dog application on mobile', 2014.

[5] P. Sidiropoulos and J. Muller, 'Matching of Large Images Through Coupled Decomposition', *Image Processing, IEEE Transactions on*, vol. 24, no. 7, pp. 2124-2139, 2015.

[6] L. Liu, Y. Lu and C. Suen, 'Variable-Length Signature for Near-Duplicate Image Matching', *Image Processing, IEEE Transactions on*, vol. 24, no. 4, pp. 1282-1296, 2015.

[7] H. Ng, C. Lin and T. Muindisi, 'Fast image matching using multi-level texture descriptor', 2014, pp. 1-5.

[8] K. Yata, G. Koutaki and K. Uchimura, 'Image matching by eigen template method for multi-class classification', 2015, pp. 1-4.

[9] W. Wen-Huan and Z. Qian, 'A novel image matching algorithm using local description', 2014, pp. 257-260.

[10] L. Liu, Y. Lu and C. Suen, 'Novel Global and Local Features for Near-Duplicate Document Image Matching', 2014, pp. 4624-4629.

[11] F. Yan and K. Mikolajczyk, 'Deep Correlation for Matching Images and Text', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[12] H. Bay and L. Gool, 'SURF: Speeded Up Robust Features', *Computer Vision and Image Understanding (CVIU)*, 2008.

[13] D. Lowe, 'Distinctive Image Features from Scale-Invariant Keypoints', *International Journal of Computer Vision*, 2004.

[14] Y. Ke and R. Sukthankar, 'PCA-SIFT: A more distinctive representation for local image descriptors', 2004, pp. 506-513.

[15] R. Hussin and A. Kamarudin, 'Digital Image Processing Techniques for Object Detection From Complex Background Image', *Procedia Engineering*, 2012.

[16] L. K.Velmurugan, 'Content-Based Image Retrieval using SURF and Colour Moments', *Double Blind Peer Reviewed International Research Journal*, 2011.

[17] V. Chandrasekhar and B. Girod, 'CHoG: Compressed Histogram of Gradients', 2009, pp. 2504-2511.

[18] E. Oyallon and J. Rabin, 'An Analysis and Implementation of the SURF Method, and its Comparison to SIFT', *Image Processing On Line*, 2014.

[19] Z. Ding, S. Qian, Y. Li and Z. Li, 'An image matching method based on the analysis of grey correlation degree and feature points', 2014, pp. 157-162.

[20] M. Zheng, C. Wu, D. Chen and Z. Meng, 'Rotation and affine-invariant SIFT descriptor for matching UAV images with satellite images', 2014, pp. 2624-2628.

[21] L. Li and Y. Hongyang, 'Improved SIFT performance evaluation against various image deformations', 2014, pp. 172-175.

[22] B. Zhang, Y. Jiao, Z. Ma, Y. Li and J. Zhu, 'An efficient image matching method using Speed Up Robust Features', 2014, pp. 553-558.

[23] Y. Liu and S. Ma, 'Research on Image Based on Improved SURF Feature Matching', 2014, pp. 581-584.

[24] P. Verma and J. Bhatia, 'Design and Development of GPS-GSM Based Tracking System with Google Map Based Monitoring', *IJCSEA*, vol. 3, no. 3, pp. 33-40, 2013.

[25] A. Biswas, G. Pilla and B. Reddy Tamma, 'Microsegmenting: An approach for precise distance calculation for GPS based ITS applications', 2015.

[26] M. McDonald, J. Wu, Y. Li and J. Guo, 'The potential application of GPS for distance controlled car following', 2002, pp. 25-29.

[27] N. Enami, N. Ukita and M. Kidode, 'Image matching robust to changes in imaging conditions with a car-mounted Camera', 2008, pp. 1-10.

[28] D. Edmundson and G. Schaefer, 'Fast mobile image retrieval', 2013, pp. 1-6.

[29] Y. Luo, Y. Li, J. Tu and C. Xu, 'FIM: A Real-Time Content Based Sample Image Matching System', 2013, pp. 551-555.

[30] B. Li and Z. Miao, 'Image matching using adapted image models and its application to content-based image retrieval', 2014, pp. 3117-3121.

[31] G. Liu, Y. Lin and Q. Wang, 'Image Retrieval Based on the Color Textons Descriptor', 2009, pp. 1-5.

[32] C. Liu, A. Zhou, C. Wu and G. Zhang, 'Image segmentation framework based on multiple feature spaces', *Image Processing, IET*, vol. 9, no. 4, pp. 271-279, 2015.

[33] Y. Du, H. Ai and S. Lao, 'Evaluation of color spaces for person re-identification', 2012, pp. 1371-1374.

[34] S. Westland, C. Ripamonti, and V.Cheung, "CIELAB and Colour Difference," Computational Colour Science using MATLAB, Second edition, 2012, pp, 49-74

Dr. Mohammad Asaad Ghazal obtained his Doctorate and Master's degrees in Electrical and Computer Engineering (ECE) from Concordia University, Canada, in 2010 and 2006, respectively. He was awarded Alexander Graham Bell Fellowship by the Natural Science and Engineering Research Council of Canada (NSERC) and the Doctoral Scholarship of Quebec's Fund for Research on Nature and Technologies. Dr. Ghazal is currently an Assistant Professor of ECE and the Chair of the ECE Department. He has authored/coauthored more than 30 publications in technical journals and international conferences including IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology, and IEEE Transactions on Consumer Electronics. His current research interests are image and video signal processing, robotics, embedded and networked systems, and mobile computing. He is an IEEE member since 2005.

Yasmina AlKhalil received her B.Sc. degree in Computer Engineering from Abu Dhabi University, Abu Dhabi, United Arab Emirates. She has worked on several awarded mobile and web applications, and received the Best mGovernment Service Award for Innovation for the Al Najdah Mobile Application for accident detection in moving vehicles, as well as the award for the Lost and Found Mobile Application for the retrieval of lost personal objects. Among other awards, she has received the first place in the mGov Developer Jam hosted by the Telecommunications Regulatory Authority, and the first place in the Digital Arabic Content Competition organized by United Nations Economic and Social Commission for Western Asia for an Iqraa mobile application. Her research areas include digital signal processing, mobile and electronic applications, health applications, image and video processing, machine learning, and embedded systems design.

Fasila Haneefa was born in India, 1994. She is an undergraduate student at Abu Dhabi University where she is currently pursuing her Bachelors degree in Computer Engineering. She has participated in several competitions such as Best m-Government Service Award, Digital Arabic Content, MAC 2015, and Undergraduate Research Competition. Her research interests include mobile applications, mobile computing, and computer vision.

Samr Samir Ali obtained her Bachelor of Science (Summa Cum Laude) in Computer Engineering from Abu Dhabi University, United Arab Emirates, in 2015; where she was awarded the overall excellence award twice for the most outstanding student in the college of Engineering and the university college respectively as well as several other honours. She is an IEEE member and served as the president of the IEEE student branch in Abu Dhabi University for academic year 2014-2015. She has led several teams towards winning top places in competitions such as a Special Innovation Award of 1,000,000 Dhs from HH Sheikh Muhammad Bin Rashid Al Maktoum as a part of the Best mGovernment Service Award Competition during the 2015 Government Summit for the Al Najdah Mobile Application, the United Nations Economic and Social Commission for Western Asia best Digital Arabic Content Application Award, the Ministry of Interior Award, the Mobile Application Contest, and many others. Her current research interests include artificial intelligence and machine learning, robotics and embedded systems, digital signal processing, and mobile computing.

Eman Farag Rashed is currently a third-year Computer Engineering student from Abu Dhabi University, Abu Dhabi, UAE. She has participated on research competitions and many mobile application contests where among them she has won the Best M-Government Service Award in University Students sector - 2nd Cycle 2015.