International Journal of Advanced Smart Convergence Vol.9 No.3 183-191 (2020)
http://dx.doi.org/10.7236/IJASC.2020.9.3.183

IJASC 20-3-23

Photo Management Cloud Service Using Deep Learning

Sung-Dong Kim, Namyun Kim

Professor, School of Computer Engineering, Hansung University, Korea
sdkim@hansung.ac.kr, nykim@hansung.ac.kr

Abstract

Today, taking photos using smartphones has become an essential element of modern people. According to these social changes, modern people need a larger storage capacity, and the number of unnecessary photos has increased. To support the storage, cloud-based photo storage services from various platforms have appeared, and many people are using the services. As the number of photos increases, it is difficult for users to find the photos they want, and it takes a lot of time to organize. In this paper, we propose a cloud-based photo management service that facilitates photo management by classifying photos and recommending unnecessary photos using deep learning. The service provides the function of tagging photos by identifying what the subject is, the function of checking for wrongly taken photos, and the function of recommending similar photos. By using the proposed service, users can easily manage photos and use storage capacity efficiently.

Keywords: *Photo Management, Cloud-based Service, Deep Learning, Photo Classification, Photo Storage Management.*

1. Introduction

Today, people are taking a lot of photos with their smartphones, and taking photos has become the most important function of a smartphone. Therefore, whenever a new smartphone is released, the function is being improved so that people can take better photos, so anyone can take as good a photo as a professional photographer. In the meantime, the resolution of photos is getting bigger, and the size of one photo is getting bigger enough to be 10 MB. In everyday life, people take a lot of photos, and especially when they go on a trip, they take a lot of photos much more. Therefore, the management of photos is becoming more necessary and important. Platform companies such as Naver, Google, and Microsoft are providing cloud services for photo management. People are using these services to back up photos on their smartphones. However, as the number of photos increases, various problems arise. First, the storage capacity becomes insufficient. Second, it becomes difficult to find the photo you want. Third, there are many unnecessary photos such as shaky photos, dark photos, and similar photos. In particular, the third problem is causing insufficient capacity and difficulty in searching photos. Existing photo management services mainly provide storage space with some search and classification functions. However, they don't provide a solution to the third problem.

Manuscript Received: July. 24, 2020 / Revised: July. 28, 2020 / Accepted: August. 5, 2020
Corresponding Author: sdkim@hansung.ac.kr
Tel: +82-2-760-4161, Fax: +82-2-760-4488
Professor, School of Computer Engineering, Hansung University, Korea

In this paper, we mainly propose a cloud-based photo management service PICCL (PICture CLoud) that provides the following functions to solve the third problem. The cloud-based services can be said to be the most effective method due to the development of communication networks and personal terminals such as notebooks, tablets, and smartphones. For example, they propose a cloud-based context-aware system, in which they solve the problem of server overload and waiting time increase by using a cloud-based method [1]. Also, [2] proposes a cloud-based environment for machine translation system improvement, where the cloud-based system supports the improvement of machine translation systems without time and place restrictions.

First, it provides a function to distinguish wrongly taken photos. Wrongly taken photos are dark and shaky photos, and they don't have to be saved. A model to classify dark photos is generated using deep learning, and shaken photos are categorized using OpenCV's photo analysis module. Second, PICCL provides a function to search similar photos. People tend to take photos of the same subject multiple times to get a better photo. As a result, the number of similar photos becomes very large, which leads to difficulty in searching and wastes storage space. In order to identify the subject of the photo, deep-learning based YOLO object recognition model is adopted, and similar photos are found by performing a similarity test on photos having the same subject. Deep learning technology is recently being used to create practical solutions to various problems. For example, a deep learning model was used for license plate recognition in video [3], and a fire and smoke detection solution using a deep learning model was proposed in [4]. In addition, [5] presented a deep learning model for tourism recommendation. PICCL provides the above two functions to check unnecessary photos and recommends deletion of these photos so that users can delete unnecessary photos by themselves or automatically. This saves storage space and supports effective photo search.

The paper is organized as follows. Section 2 outlines the structure of the proposed photo management service. Section 3 explains the functions of the PICCL in detail. Section 4 presents the experimental results of the deep learning-based models which identify unnecessary photos. Section 5 concludes the paper by presenting the expected effects of the PICCL.

2. Structure of PICCL

The proposed photo management service (PICCL) has a structure as shown in Figure 1. PICCL is composed of a photo storage part and a photo analysis part.

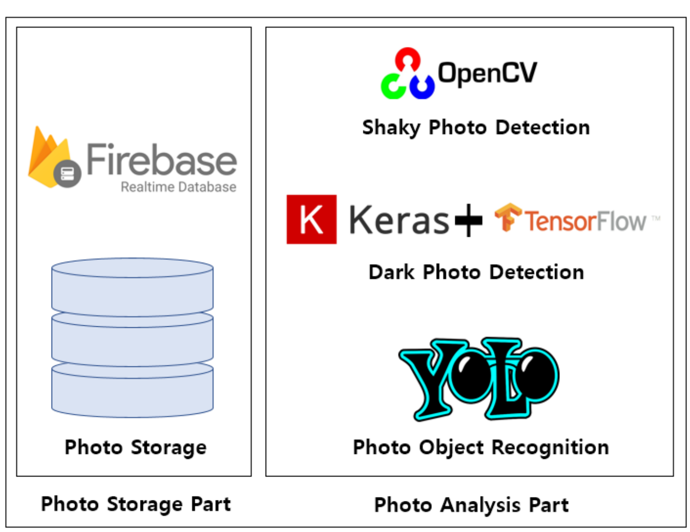


Figure 1. Structure of PICCL

The photo storage section uses the Firebase database to manage photos and provide space to store users' photos. In the database, meta information (latitude/longitude, date, time, file name, etc.) of photos taken with a smartphone and the storage location of the photos on the server are stored. The database structure is shown in Figure 2. The *user table* stores user id and email. The *photo meta information table* stores meta information provided by the smartphone camera and a boolean value indicating whether it can be deleted or not. The meta information is used in the similar photo search process, and the value of *delete* field can be deleted is used in the deletion recommendation function. The *photo tag table* stores a boolean value indicating what the subject is, which is automatically recorded when the user saves the photo to the PICCL storage. The tag information is used in the process of searching for photos and classifying similar photos. The *deletion info table* stores a boolean value indicating the reason for the deletion of a photo. This is a table that stores photo analysis result values in the photo processing part.

User table

ID	email
string	string

Photo Meta Information table

id	title	path	date	latitude	longitude	delete
string	string	string	string	string	string	string

Photo Tag table

id	title	person	animal	vehicle	furniture	book	bag	sports	food	device
string	string	boolean	boolean	boolean	boolean	boolean	boolean	boolean	boolean	boolean

Deletion Info table

id	title	similar	shaken	dark	screenshot
string	string	boolean	boolean	boolean	boolean

Figure 2. Database structure

The photo analysis part performs photo analysis to classify unnecessary photos, the main purpose of the proposed service. Unnecessary photos consist of dark photos, shaky photos, and similar photos. The PICCL photo analysis part provides a function to automatically classify unnecessary photos. It recommends deleting unnecessary photos to efficiently manage storage space and make it easier for users to manage photos. The *Shaky Photo Detection* module is implemented using OpenCV's image analysis modules, and determine whether the photo is shaken or not. The *Dark Photo Detection* module is a deep learning model generated using Tensorflow and Keras and determines whether a given photo is a dark or not. The *Photo Object Recognition* module identifies the object in a photo using YOLO. The photo tag is used for photo search, and the similar photo search function searches for similar photos using the tag and meta information of the photo.

3. Functions of PICCL

This section describes the main functions and methods for providing the functions of the proposed cloud-based photo management service. First, it provides a photo storage function. This is a function that stores photos in the user's smartphone, and is a storage function such as Naver Drive and Google Photos. In order to manage physical photo files, the Firebase database described in Section 2 is used.

Second, PICCL provides a photo object tagging function. This function recognizes the subject of the photo and attaches the appropriate tag. This is an added function to the smartphone's tagging feature and is used for photo search. In PICCL, a photo is tagged as one of nine object types: person, animal, vehicle, furniture, book, bag, sports, food, device. For object recognition, we use YOLO (You Only Look Once), an object recognition model based on deep learning. YOLO system was developed to detect all object in real time by using one neural network [6-8]. The YOLO system looks at the image once and predicts which object is in which position. The object detection process of the YOLO system as follows. The input image is divided into multiple grids. When the center of the object lies in a certain grid, the grid plays a role of detecting the specific object. Each grid predicts a certain number of 'bounding boxes' to identify the position of the object, and each bounding box is assigned a 'confidence score.' One neural network is used to predict the conditional class probability of a bounding box and its objects. The confidence score is the value of intersection over union (IOU) between predicted bounding box and the real object boundary. When the value is equal to or larger than a predetermined value, the object is determined to be included. We can know what the object is by classifying the object and determining the object with highest probability as in the existing image recognition system. Therefore, YOLO system is the combination of the existing image recognition system and the method of identifying object's position. YOLO system defines the object detection problem as the regression problem of estimating the probability of the space-separated bounding box and tries to solve using the neural networks. As a result, YOLO system consists of multiple convolution layers, polling layers, and fully connected layers as in the existing CNN. In this paper, we adjust the object recognition model using the pre-trained weight value of YOLOv3 [8] (yolov3.cfg available from https://pjreddie.com/media/files/yolov3.weights) and 1000 photos. The feature extractor of YOLO v3 is Darknet-53 with 53 convolution layers, and it performs multiclass classification with improved accuracy compared to the previous versions. That is, it provides various binary classification values for the recognized object. For example, when a woman is recognized, both person and woman are judged as 1. Table 1 shows 80 classes of objects provided by YOLOv3 and the corresponding tags. YOLOv3 outperforms other object detection algorithms, so it can be usefully used for the purpose of detecting photographic subjects.

Table 1. Main parameters

Tag	Classes
Person	person
Animal	bird, cat, dog, horse, sheep, cow, elephant, bear, zebra, giraffe, teddy bear
Vehicle	bicycle, car, motorcycle, airplane, bus, train, truck, boat, traffic light, stop sign, parking meter, fire hydrant
Furniture	chair, couch, bed, refrigerator, sink, toilet, dining table, clock
Book	notebook, book
Bag	handbag, backpack, suitcase
Sports	frisbee, skis, snowboard, sports ball, kite, baseball bat, baseball glove, skateboard, surfboard, tennis racket
Food	wine glass, cup, fork, knife, spoon, bowl, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, bottle
Device	hair drier, toaster, oven, microwave, cell phone, keyboard, remote, mouse, laptop, tv, refrigerator

Third, PICCL provides a function to distinguish shaky photos. Shaken photos are recognized using open source computer vision (OpenCV) library. In order to determine whether a photo is shaken or not, we determine the blurriness of the photo. In order to detect blurred images, there are researches such as a method using Gaussian basis filters [9], measuring blur extent of edges and analysis of edge sharpness [10], and etc. In this paper, we adopt the variance of the Laplacian of the photo to determine the blurriness because we can easily calculate the value using OpenCV library. If the variance value is less than the threshold (alpha, here), it is determined as a blurry photo [11]. The Laplacian measures what you could call the "curvature" or stress of the field. It tells you how much the value of the filed differs from its average value taken over the surrounding points. So, it can be used to measure the degree to which the picture is shaken. To calculate the Variance of the Laplacian, we first convert the photo to black and white. Figure 3 shows the algorithm for determining whether a photo is shaken. The algorithm uses the OpenCV library to convert the photo to black and white and calculate the variance of the Laplacian: cvtColor(photo, cv2.COLOR_BGR2GRAY) and Laplacian(photo, cv2.CV_64F).var(). The threshold value (alpha) is set to 400 through experiments using sample photos.

```
function is_shaken_photo(aPhoto, alpha)
{
    aPhoto = cvtColor(aPhoto, cv2.COLOR_BGR2GRAY)
    aLaplacian_Variance = Laplacian(aPhoto, cv2.CV_64F).var()
    if (aLaplacian_Variance < alpha) return true
    else return false
}
```

Figure 3. Algorithm for shaken photo determination

Fourth, PICCL provides a function to distinguish dark photos. Using Tensorflow and Keras, we generate a deep learning model that distinguishes dark photos. We use Tensorflow version 2.0 and Keras version 2.2.0. Table 2 shows the structure of a model to which the structure of a general CNN is applied. The DarkPhoto model structure is a typical CNN structure and has about 330K parameters. Since it is a relatively small number, it does not require a lot of data and time to learn.

Table 2. DarkPhoto discriminator model structure

Layer	Output Shape	# of Parameters
Convolution	64 x 64 x 32	896
Max Pooling	32 x 32 x 32	0
Convolution	32 x 32 x 32	9248
Max Pooling	16 x 16 x 32	0
Convolution	16 x 16 x 64	18496
Max Pooling	8 x 8 x 64	0
Dropout	8 x 8 x 64	0
Convolution	8 x 8 x 64	36928
Max Pooling	4 x 4 x 64	0
Dropout	4 x 4 x 64	0
Flatten	1024	0
Dense	256	262400
Dropout	256	
Dense	1	257

Fifth, PICCL provides a function to search for similar photos. There are many similar photos due to the tendency of users to take photos of the same subject several times to obtain good photos. Similar photos are searched using the subject, time, and location information of the photo and notified to the user. Figure 4 shows an algorithm to search for similar photos. The algorithm accepts 3 threshold values: alpha, beta, and gamma. The alpha is the threshold for the time range, the beta is for the longitude range, and the gamma is for the latitude range. Photos whose time, latitude and longitude are within the threshold values are selected as similar photos.

```
function search_similar_photos(aGivenPhoto, alpha, beta, gamma)
{
    similar_photos = []
    tag = aGivenPhoto.tag
    time = aGivenPhoto.time
    longitude = aGivenPhoto.longitude
    latitude = aGivenPhoto.latitude
    for (aPhoto of which date is the same as the date of aGivenPhoto)
    {
        if (tag == aPhoto.tag &&
            time - aPhoto.time < alpha &&
            longitude - aPhoto.longitude < beta &&
            latitude - aPhoto.latitude < gamma)
            similar_photos.append(aPhoto)
    }
    return similar_photos
}
```

Figure 4. Algorithm for searching similar photos

4. Experimental Results

In this section, we present the training process and the results of Dark Photo Detection model. For photo object recognition module, we adopt YOLOv3 as it is. The main purpose of PICCL is to find unnecessary photos and recommend them to be deleted. This section presents the deletion recommendation function and the results using the functions provided by PICCL.

4.1 Dark Photo Detection Model

Table 3 shows environment for training the Dark Photo Detection Model. In the training and validation data, we have 354 bright and 366 dark photos. In the test data, there are 46 bright and 34 dark photos. We apply early stopping in the training process, and training completes only in 13 epochs. Since it is a relatively easy classification problem, it is possible to generate a model with good performance with very few epochs.

Table 3. Training environment of dark photo detection model

Item	Value
# of training data	612
# of validation data	108
Batch size	64
Epoch	100
# of training parameters	328,225
Loss function	binary cross entropy
Optimizer	Adam optimizer
Activation functions	Convolution layer, 1st dense layer: RELU Final dense layer: sigmoid
# of test data	80

Figure 5 shows the model training process. The prediction accuracy of the model measured using 80 test data is 95%, and this result supports that the generated model properly distinguishes dark photos.

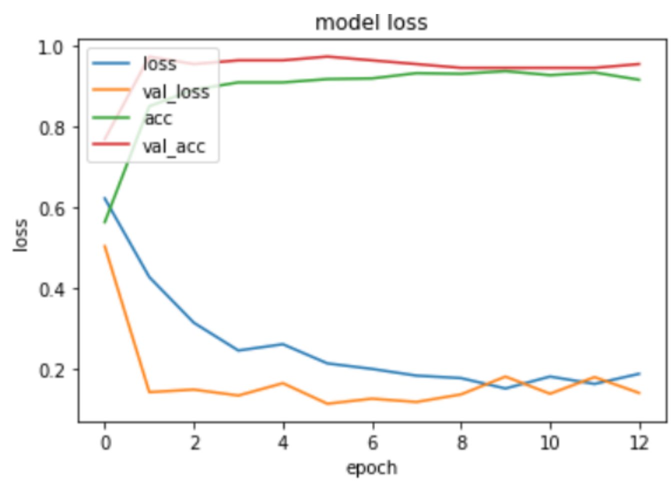
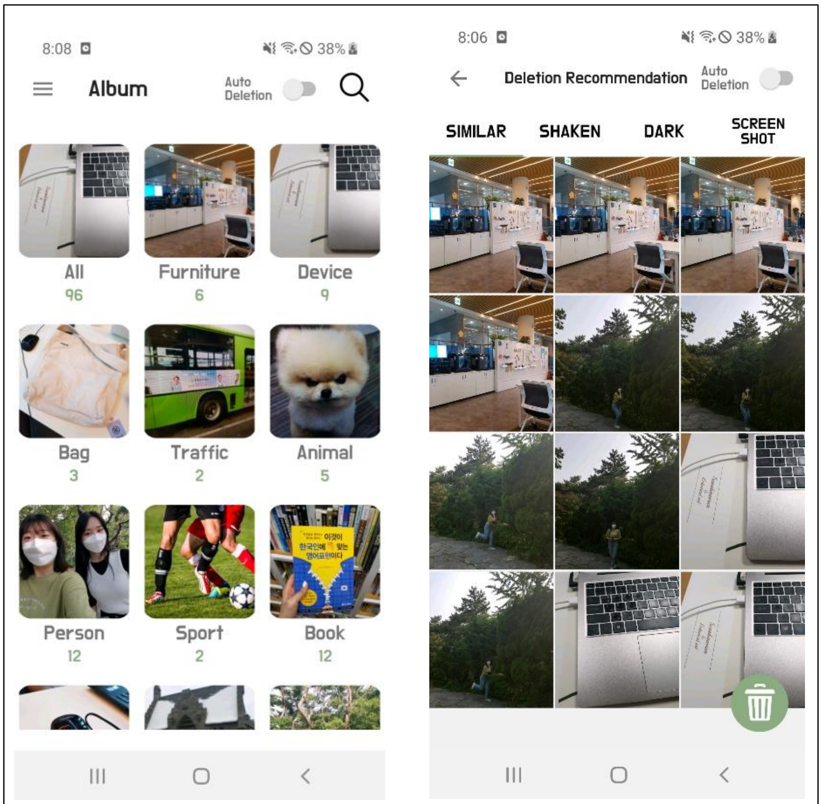


Figure 5. Training process of dark photo detection model

4.2 Photo Deletion Recommendation

PICCL is a cloud-based photo management service to effectively manage photos and improve the efficiency of storage space. Therefore, the most essential function is the deletion recommendation function. In other words, it is recommended to check and delete unnecessary photos such as shaky photos, similar photos, and dark photos. When backing up photos from a smartphone to a cloud storage, photo analysis processes such as the determination of a degree of shake/darkness and the similar photo searching using photo object recognition, are performed to determine whether the photo may be deleted or not.

Figure 6 shows some images of the PICCL smartphone application. In (a), the photos are classified according to the subject recognition result. Figure 6-(b) is the image for photo deletion recommendation. It is divided into similar photos, shaken photos, and dark photos, so you can see why the deletion recommendation was made. The figure shows the appearance of a similar photo tab, and we find that similar photos have been properly recommended.



(a) (b)

Figure 6. Example images of PICCL smartphone application

5. Conclusion

Since taking pictures using smartphones has become commonplace, the need for photo management services is increasing. In this paper, we propose a cloud-based photo management service PICCL using deep learning. The PICCL proposed in this paper provides the following functions. First, it provides a function to back up photos stored in a smartphone. Second, it provides a function to check unnecessary photos in order to increase the efficiency of storage space. What differentiates PICCL from other photo management services most is its ability to check unnecessary photos. PICCL can check unnecessary pictures such as dark pictures, shaky pictures, and similar pictures. In the process of developing PICCL, a deep learning-based model was created using Tensorflow and Keras to create a model that classifies dark photos. And a deep learning-based object recognition model, YOLOv3, was applied to recognize the subject of a photo, and a similar photo search function was implemented using this. Also, we implemented a module that recognizes shaken photos using OpenCV. Through the experiments, we confirm that dark photos, shaky photos, and similar photos were properly searched.

Currently, an application that can use the PICCL service on a smartphone is provided, but it is planned to be able to use PICCL through the web and using various terminals. This is expected to contribute to providing convenience of photo management to many people.

Acknowledgement

This research was financially supported by Hansung University.

References

[1] T.W. Kwon, J.-Y. Lee and K.-D. Jung, "Design of Cloud-based Context-aware System Based on Falling Type," *International Journal of Internet, Broadcasting and Communication*, Vol. 9, No. 4, pp. 44-50, 2017. DOI: http://dx.doi.org/10.7326/IJIBC.2017.9.4.44

[2] S.-D. Kim and N. Kim, "Environment for Translation Domain Adaptation and Continuous Improvement of English-Korean Machine Translation System," *International Journal of Internet, Broadcasting and Communication*, Vol. 12, No. 2, pp. 127-136, 2020. DOI: http://dx.doi.org/10.7326/IJIBC.2020.12.2.127

[3] B. Kim and J. Heo, "Semi-Supervised Learning Based Anomaly Detection for License Plate OCR in Real Time Video," *International Journal of Advanced Smart Convergence*, Vol. 9, No. 1, pp. 113-120, March 2020. DOI: http://dx.doi.org/10.7236/IJASC2020.9.1.113

[4] Y. Lee and J. Shim, "Deep Learning and Color Histogram based Fire and Smoke Detection Research," *International Journal of Advanced Smart Convergence*, Vol. 8, No. 2, pp. 116-125, June 2019. DOI: http://dx.doi.org/10.7236/IJASC2019.8.2.116

[5] C.-S. Jeong, K.-H. Ryu, J.-Y. Lee, and K.-D. Jung, "Deep Learning-based Tourism Recommendation System using Social Network Analysis," *International Journal of Internet, Broadcasting and Communication*, Vol. 12, No. 2, pp. 113-119, 2020. DOI: http://dx.doi.org/10.7326/IJIBC.2020.12.2.113

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, pp. 779-7882016, 2016. DOI: https://doi.org/10.1109/CVPR.2016/91

[7] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, pp. 6517-6525, 2017. DOI: https://doi.org/10.1109/CVPR.2017/690

[8] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv*, 2018. DOI: https://arxiv.org/abs/1804.02767

[9] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, Vol. 13, No. 9, pp. 891-906, 1991. DOI: https://dx.doi.org/10.1109/34.93808

[10] Y. Chung, J. Wang, R. Bailey, S. Chen, and S. Chang, "A non-parametric blur measure based on edge analysis for image processing applications," *In Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems*, Vol. 1, pp. 356-360, 2004. DOI: https://dx.doi.org/10.1109/ICCIS.2004.1460440

[11] R. Bansal, G. Raj, and T. Choudhury, "Blur detection using Laplacian operator and Open-CV," *In Proceedings of the 2016 International Conference System Modeling & Advancement in Research Trends (SMART)*, pp. 63-67, Nov. 2016. DOI: https://dx.doi.org/10.1109/SYSMART.2016.7894491