JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 24, No. 4, pp. 725-734, Apr. 2023

Check for updates

영상 AI 기반의 자동 선별적 비식별화

김 대 진[1] · 전 윤 걸[2*]
[1]동국대학교 영상문화콘텐츠연구원 조교수
[2*]경북대학교 생태환경대학 강사

Vision AI-Based Automatic Selective De-Identification

Dae-Jin Kim[1] · Youn-Girl Jeon[2*]

[1]Assistant Professor, Research Institute for Image & Cultural Contents, Dongguk University, Seoul 04626, Korea
[2*]Instructor, College of Ecology & Environmental Science, Kyungpook National University, Sangju-si 37224, Korea

[요 약]

최근 개인정보는 경제성이 높아지고 활용 범위가 넓어지고 있어, 개인정보 유출에 따른 위험성과 피해가 커질 가능성이 점점 높아지고 있다. 영상데이터의 경우 이러한 문제점 때문에 개인 데이터 수집 시 사전 승인을 받은 사람들만 노출하고 그 외에는 비식별화 하는 과정이 반드시 필요하다. 그러나, 자동으로 비식별화하는 경우에 모든 대상을 비식별화 처리하기 때문에 허가받은 대상만 식별화하기 위해서는 동영상편집기를 통해서 수작업으로 진행되어 왔다. 따라서, 본 논문에서는 인공지능 기술을 통해서 얼굴/객체인식 기반의 자동 선별적 비식별화를 진행하여, 승인된 대상은 식별화하고 나머지 대상은 비식별화 할 수 있는 방법을 연구하였다. 이를 통하여, 개인 정보의 활용성 및 안정성을 보장하면서 자동으로 선별적 보호를 진행할 수 있다.

[Abstract]

Recently, with personal information being increasingly used in several applications, its leakage and resulting damage have become a relevant concern. In video data, only people consenting to the collection of personal data are to be identified, and other people should be de-identified. In current automatic de-identification, not all objects are identified, and de-identification of only recognized objects is performed manually using video editor tools. Therefore, in this study, face/object-detection-based automatic selective de-identification was conducted through artificial intelligence technology to identify approved objects and de-identify remaining objects. Thus, selective protection can be automatically performed while ensuring the usability and stability of personal information.

색인어 : 선별적 비식별화, 개인정보, 얼굴인식, 객체인식, 자동화

Keyword : Selective de-identification, Personal information, Face recognition, Object recognition, Automatic

http://dx.doi.org/10.9728/dcs.2023.24.4.725

Received 16 February 2023; **Revised** 04 March 2023
Accepted 28 March 2023

***Corresponding Author;** Youn-Girl Jeon
Tel: 054-530-1430
E-mail: youngirl@knu.ac.kr

Ⅰ. Introduction

Recently, CCTVs have become ubiquitous not only in places where access control is required, but also in public places such as streets and subway stations, and opportunities to access various media such as the Internet, OTT, and TV are increasing. At this time, as video data is easily shared on the Internet, there is a problem of portrait rights or privacy infringement in which an individual's unwanted appearance and information are disclosed without realizing it. Although this situation is essential for personal safety and protection, unintended personal information may be collected by images stored by video surveillance, which may pose a problem for personal information protection. In the case of distribution, de-identification must be accompanied [1],[2].

Among them, there are four methods for the de-identification of video content: “image filtering”, “image encryption”, “face synthesis”, and “inpainting” as follows.

"image filtering" is currently most commonly used in such a way that multiple filters are applied to areas where individuals can be identified in images, preventing them from identifying specific individuals [3]-[8]. This is simply processable so that the personal identification area is unrecognizable. However, with the development of deep learning based technology, it is also possible to restore the filtered image to some extent. Blur and Pixelation filters are used for “image filtering”. Gaussian functions for Blur are commonly used to weight each pixel and its surrounding pixels according to a narrow distribution and obtain an average value to blur the image [6]. The Pixelation filter obtains an average value of pixels included in each grid area and filters by replacing all pixel values with an average value [8].

"image encryption" is a method of encrypting images and disclosing them only to authorized users [9]-[15]. It is an alias processing technology that can be restored as an original image, and can be used to safely transmit an image over a network. However, using the existing encryption technique, it is difficult to process real-time images due to a large amount of computation, and encrypted image data cannot be used without a decryption key. In addition, the encryption method may vary depending on the storage format of the image. The image encryption type includes a method using “frequency encryption” and a method using “pixel encryption”. “frequency encryption” converts an image into a frequency domain through Discrete Cosine Transform (DCT) conversion, and converts AC coefficients to encrypt them [9]. “pixel encryption” is a method of changing the position of a pixel according to a certain rule so that the original image cannot be recognized [15].

"face synthesis" is an extension of the k-same model to fit image data, which can synthesize similar k faces only within the collected set of face images, and should ensure that the image recognition program has no more than 1/k probability of identifying a particular individual's face [16]-[18]. This provides a mathematically guaranteed level of personal identification prevention and replaces it with a synthesized face, so the use of anonymized image data is high. However, since the stored face image is used, real-time image processing is difficult, and usefulness cannot be guaranteed because usefulness is not considered in the anonymization stage. In addition, it is impossible to restore the original face from the synthesized face.

"inpainting" is a technique that removes certain parts of the face and fills in gaps or damaged parts [19]-[25]. This does not leave any visual information on the object removed from the image. In addition, real-time processing is difficult due to a large amount of computation, and recovery may not be possible when the removed area is large. In addition, the usefulness of image data may decrease if it is restored unnaturally. In the case of inpainting, there are "patch based inpainting" methods that find and fill the areas most similar to spaces within a given image frame, [20],[24] and "object based inpainting" methods that separate images into backgrounds and objects to remove specific objects and replace the remaining areas with backgrounds [25].

When video content is de-identified, there are cases where it is necessary to clarify meaning, protect personal information, and selectively de-identify depending on the business model. For example, when watching drama contents, there are cases in which PPL products must be identified and competitive products must be de-identified. It may also be necessary to de-identify unauthorized persons, except for persons in the spotlight. In addition, when collecting raw data to build artificial intelligence learning data, in case of abnormal behavior (fight, fall, dumping, theft, child abuse, etc.), actual occurrences are not collected through CCTV. However, the data can reproduce through the actor's situation. At this time, actors can be identified because they have already obtained permission. However, depending on the situation, non-actors, specific unlicensed trademarks, unlicensed locations, etc., de-identification must be done by video tools. In addition, as I have read articles about cases where CCTV footage of child abuse in a daycare center cannot be disclosed to parents unless de-identified, AI must identify only the child and the abuse daycare center teacher, etc., except for rests in CCTV videos. So, the need for vision AI based selective de-identification automation service is emerging. Therefore, this paper studies a vision AI based automatic selective de-identification technique that can automatically de-identify the remaining objects except for the object requiring identification.

The contributions of this study can be summarized as follows:

1. We present the problems of de-identification which are

existing studies, and present an automated selective de-identification technique to solve them.

2. We performed automatic selective de-identification based on face/object detection through artificial intelligence technology to identify approved objects and de-identify the remaining objects.

3. Centerface and Arcface algorithms were used to perform face recognition based selective de-identification, and YOLOv5 and EfficientNet algorithms were used to perform object recognition based selective de-identification. Effective selective de-identification was possible by linking the above algorithms.

Looking at the structure of this paper, Chapter 2 presents the existing cases for de-identification and the direction of selective de-identification, and Chapter 3 examines face/object recognition based algorithms for selective de-identification. After researching how to interlock based on the algorithm proposed in Chapter 4 and comparing performance in a given environment, the conclusion was drawn in Chapter 5.

II. Related Works

2-1 De-Identification Based on Face Detection

Fig. 1. De-identification based on face detection

Figure 1 shows a general face detection based de-identification process. The image de-identification is progressed using a face detection algorithm depending on various AI networks, which proceeds as follows.

Step 1. Data Upload

-. Upload large-capacity photos and videos that require de-identification to the server by transfer protocols(HTTP, HTTPS, FTP, etc.).

Step 2. Automatic detection of personal information area

-. An AI algorithm automatically detects the facial area in personal information areas. AI algorithms such as MTCNN [26], ArcFace [27], RetinaFace [28], and CenterFace [29] can be used for face detection, and the optimal face detection algorithm is selected according to the characteristics of the data set.

Step 3. View and edit de-identification results

-. Check the de-identified results and, if necessary, modify the relevant area through the editing function by tools.

Step 4. de-identification completed

-. After all, tasks are completed, the de-identified data is stored. If the stored data is an image, it can be saved as a frame-by-frame file, and in the case of a video, the de-identified frames are encoded again to make it a video.

2-2 De-Identification Based on Object Detection

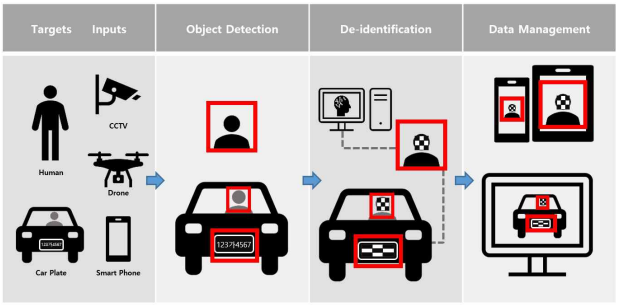

Fig. 2. De-identification based on object detection

Figure 2 shows a typical de-identification process based on object detection. In particular, the license plate number can be used as personal information about the vehicle because information about the vehicle owner can be known through the license plate number. In order to identify the license plate number, the object must first be identified. At this time, AI algorithms such as YOLOv5 [30], SSD [31], and Faster R-CNN [32] are used for object detection. One-Stage models are used such as YOLO and SSD when real-time is required. Additionally, for greater accuracy, object detection is performed using a Two-Stage model.

2-3 Suggestion of Problems with the Existing System

All existing de-identification solutions are de-identified for all specified objects. Although it has the advantage that batch processing is possible, manual de-identification is required because it is impossible to selectively identify and process specific objects and targets. In this case, it can be more inefficient because it takes longer and costs more. Therefore, this issue can be resolved by resolving it with automatic selective de-identification.

III. Selective De-Identification

3-1 De-Identification Based on Selective Face Recognition

1) Face Detection

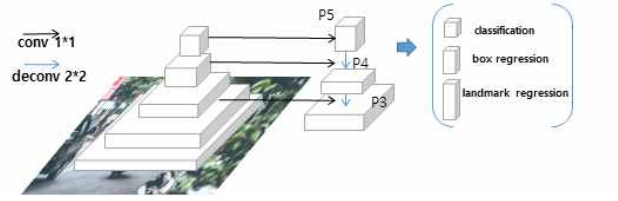

Fig. 3. FPN architecture of the Centerface

This paper uses the face detection algorithm(Centerface) to detect a person's face from a camera or video. The SOTA(State-Of-The-Art) algorithm can accurately detect a face at high speed when detecting a human face and shows a high accuracy of 0.931% on the WIDER FACE dataset [29],[33]. To extract features at high speed, MobilenetV2 is used as the backbone, and FPN(Feature Pyramid Network) is used to represent the face based on the center point of the Bbox(Bounding Box) at the Neck and predict the face size and keypoints. The extracted feature information can reduce the amount of computation by reducing the channel with 1x1 conv, increasing again with 2x2 conv, and restoring the number of channels with 1x1 conv. These keypoints heatmap of critical points can be obtained using FPN. Peaks are extracted for each category from the hitmap, and the peak is considered the center of the face. The horizontal and vertical lengths can be obtained from the Bbox, and the five keypoints can be obtained. Afterwards, the normalized keypoints consist of eyes x 2, and nose, and mouth tail x 2. Figure 3 shows the configuration of the FPN used in Centerface.

2) Face Verification

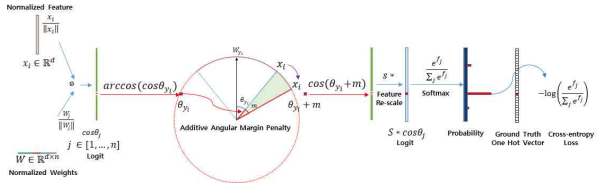

Fig. 4. Arcface architecture

There are many studies on classification using Softmax loss for face identification. In the linear transformation of the fully-connected layer, it has a size of $W \in R^{(d \times n)}$, and the face can be recognized by classifying it into n classes. It works well in a Close-set, but the identification accuracy is reduced in an Open-set. Therefore, as a research direction to overcome this, a study was conducted to discover the discriminating features by introducing the concept of Angular Margin. A representative face identification algorithm is the Arcface face identification algorithm. Figure 4 shows the overall configuration of the Arcface algorithm.

Looking at the algorithm, we first normalize feature x_i and the weight W, then the dot product of these two to get the $\cos\theta$ logit. It becomes Raw Prediction Value before Softmax. We calculate the $\arccos(\cos\theta_{y_i})$ and get the angle the angle between the feature x_i and the ground truth weight W_{y_i} that provides a kind of centre for each class. At this time, the possibility of belonging to the class is lower by adding m, which is the angular margin penalty, on the target (ground truth) angle θ_{y_i} and calculate $\cos(\theta_{y_i}+m)$. After that the Feature Rescale is proceeded by multiplying all logits by feature scale S. Based on this value, the Softmax takes, and by calculating the cross-entropy, the Softmax value of the value class corresponding to the correct response brings closer to 1. In contrast, the other values converge to 0.

When the algorithm is tested on a small data set such as CASIA [34] and an extensive data set such as MS1MV2(self-refined data of MS-Celeb-1M) [35], both show high performance. In addition, the Arcface algorithm shows 99.830% accuracy in LFW, which is face data set in the natural environment [27].

3-2 De-Identification Based on Selective Object Recognition

1) Object Detection

Object detection often includes One/Two-Stage Detection, and Two-Stage Detection is often used based on accuracy. However, One-Stage Detection is used in business models that require real-time assurance. This paper uses the representative model YOLO(You Only Look Once) v5. This algorithm can detect multiple objects with only a single forward neural network, has a fast detection speed, and is relatively resistant to false positives because it detects the entire environment of the image. Figure 5 shows the architecture of YOLOv5. The algorithm is mainly composed of Backbone, Neck, and Output parts. Backbone extracts the feature information from the input image, and Neck combines the extracted feature information and produces three dimensions of feature maps. Furthermore, the Output is configured to detect objects in the generated feature map.

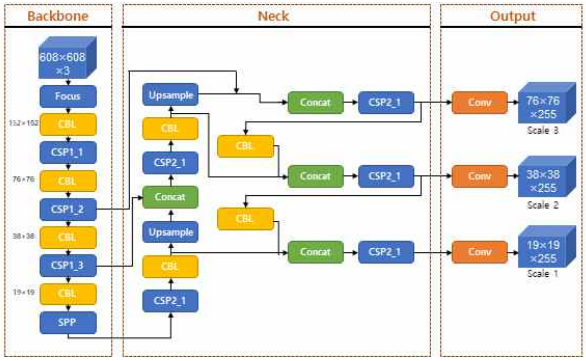

Fig. 5. YOLOv5 network architecture

Backbone is a convolution network that extracts feature maps and constructs four map layers through multiple convolutions and pooling. At this time, 152×152, 76×76, 38×38, and 19×19 pixel layers are generated, as shown in figure 5. The map layers generated in Backbone play a role in obtaining more context

information from the Neck and reducing loss information. FPN(Feature Pyramid Network) and PAN(Path Aggregation Network) can be used in this process. FPN provides meaningful feature information from the upper map to the lower feature map. In contrast, PAN provides robust localization of feature information from the lower feature maps to the upper feature map to connect feature information. The data size consists of fusion layers of dimensions 76×76×255, 38×38×255, and 19×19×255. At this time, 255 represent the number of channels. As the detection process proceeds in a layer with different dimensions, both large and small objects produce object detection results that satisfy SOTA.

The YOLOv5 is available in four models, s, m, l and x, each offering different detection accuracy and performance. This can use a model that best suits our business model. In this paper, the experiment was conducted based on the x model emphasizing the importance of accuracy.

2) Fine-Grained Classification

After object detection using YOLOv5, EfficientNet [36] was used for more accurate classification. A large class of objects can be detected with object detection, but to proceed with selective de-identification, a classification technology capable of distinguishing detailed items is required.

The EfficientNet measurement method was top-k(k=1) accuracy, when this model was tested on the dataset, 84.3% accuracy was obtained in B7, and the number of parameters was smaller than other SOTA algorithms. The model can be scaled to increase performance when designing a network model. The performance can be improved through the following three methods.

1. Depth(d)

-. As the network deepens, more and more complex features can be extracted, and it is good to generalize to other tasks, but a vanishing gradient problem occurs. And, in the case of a model with too deep layers, no further performance improvement is achieved. In order to solve this, various methods such as Skip Connection and Batch Normalization in the Resnet network are used.

2. Width(w)

-. Usually, Width(Channel) scale control is used when creating a small model through Scale Down (MobilenetV2, etc.). Since more detailed feature extraction is possible through a wide channel, the model's performance can be improved, but if the width increases, the performance quickly becomes saturated.

3. Resolution(r)

-. As the resolution of the input image increases, a more detailed pattern can be learned. Hence higher accuracy can be obtained as the input resolution increases. However, it can cause inefficiency problems such as memory problems and speed problems. Therefore, in this paper, 600x600 resolution images are currently used for learning.

In EfficientNet, composite scaling that combining Depth, Width, and Resolution performs to exploit each advantage, as shown in Figure 6(e). In this paper, ResNet50 was used as the base model for classification, and accurate classification was performed based on the parameters d=1.4, w=1.2, and r=1.3 presented in [36] during complex scaling to apply each advantage.

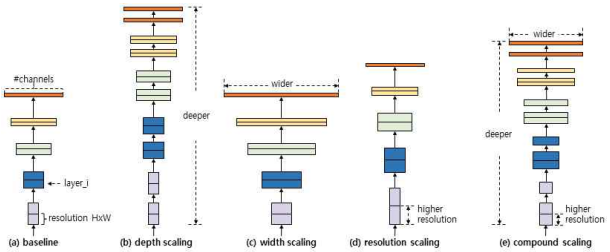

Fig. 6. Network scaling methods in EfficientNet

3-3 Selective De-Identification Linkage Design

The selective de-identification process is divided into "de-identification based on selective face recognition" and "de-identification based on selective object recognition". "de-identification based on selective facial recognition" refers to the de-identification of persons other than those who have agreed to participate in advance, and "de-identification based on selective object recognition" excludes preagreed objects and de-identifies the remaining items.

1) De-Identification Based on Selective Face Recognition

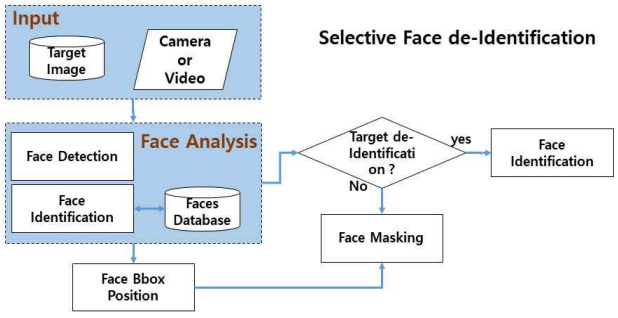

Fig. 7. Flowchart of selective face de-Identification

For interlocking de-identification, a Faces Database was constructed for N people who agreed to participate. When constructing the database, the face feature information is composed of 128-dimensional decimal values through the

MobilenetV2 network. Person information was categorized into indexes for people, and eight face images representing different angles and expression of a person were registered in the database with the same index for more accurate face recognition. The selective de-identification process based on the database of registered faces is shown in Figure 7.

Face detection is performed by applying the Centerface algorithm to the input image or video data. Extract the feature of the detected face and check whether it is the feature registered in the faces databases. At this time, the input face feature information is compared with the features of the faces databases using cosine similarity to check if it is a de-identification target.

$$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \quad (1)$$

The standard value of the similarity threshold is set, and if it is less than the threshold, de-identification proceeds.

2) De-Identification Based on Selective Object Recognition

For object detection, 8 classes (person, car, truck, motorcycle, bicycle, electric sign, symbol, traffic light) are used for learning. About 3000 images for each object were used for learning, and YOLOv5 was used as the learning algorithm. This time, epoch=30, batch=32, resolution=512, lr=1e-2, weight_decay=0.001 momentum=0.98 were used as learning parameters. In addition, fine-grained classification performs on the extracted objects, and about 1000 images for each subclass were used for learning with the EfficientNet algorithm. After processing both object training data and classification training data, the selective object de-identification process can proceed as shown in Figure 8.

Suppose the probability of an object class extracted by object detection is greater than the threshold value. The object's Bbox Position and Class Object Index information transfer to the Object Classification layer. Also, fine-grained classification is done when an emblem object is detected. The emblem class has 6 sub-class objects(Benz, BMW, Audi, Hyundai, Kia, Volvo) and as learning factor values, epoch=30, volume=32, resolution=224, lr =0.0125, weight_decay=0. Momentum = 0.9 was used, the sample size was set to B0 considering the momentum, and the pre-trained model was used for learning. When determining an object to be de-identified in Sub-Class Object, de-identification was performed using a Gaussian filter in the Bbox only when the object was less than the threshold value.

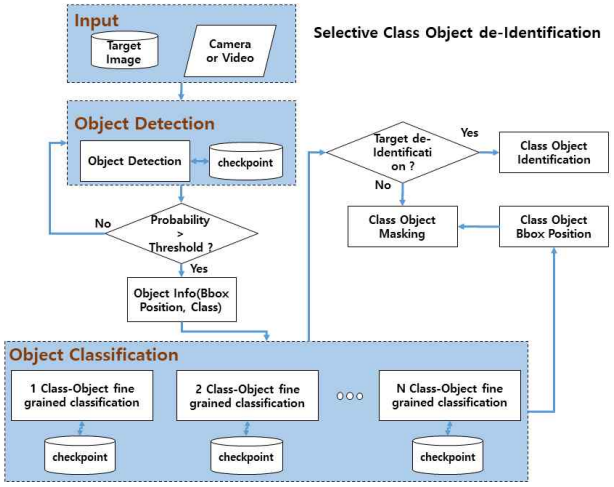

Fig. 8. Flowchart of selective object de-Identification

Ⅳ. Prototype Implementation

4-1 Experiment Setup

(a) (b)

Fig. 9. This is selective de-identification results. (a) de-identification example based on selective face recognition; (b) de-identification example based on selective object recognition.

As a result of applying selective de-identification, Figure 9(a) shows face based selective de-identification, and Figure 9(b) shows object based selective de-identification result screens. For the selective de-identification test based on face recognition, only the face of the main character 'Lee Jeong-Jae' in the movie 'The Squid Game'[37] was identified, and the rest of the characters were de-identified. In addition, for the selective de-identification test based on object recognition, the 'Benz' emblem was identified as the target for recognizing images of multiple vehicles, and the emblems of the remaining vehicle models were de-identified.

For de-identification equipment, OS is 64bit Windows10, Intel(R) core(TM) i5-7500 CPU@3.40GHz, and memory was 64Gbyte. YOLOv5 and EfficientNet, were the corresponding algorithms used, and an RTX 3080TI graphic card GPU was used for learning. In order to implement it as a service, it was

configured in WAS (Web Application Server), and HTTP was used for the upload and control protocol of input data.

4-2 Analysis of Experiment Results

F1-Score and Accuracy were used as measurement criteria information to measure the performance of de-identification.

TP is the result of the correct classification of the positive sample, and TN is the correct result after negative sample comparison. FN is the result of incorrect classification of the positive sample, and the last FP is the incorrect result after negative sample comparison. The recall is the number of correct predictions out of the total number of samples with true results. The precision represents the predicted performance of the positive sample. The Accuracy is the percentage of all samples answered correctly and is suitable when the positive and negative samples are relatively average. However, F1-Score is a more suitable measurement method if the number of positive and negative samples is not balanced. Therefore, F1-Score can be viewed as the overall grade of precision and recall.

$$recall = \frac{TP}{TP+FN} \tag{2}$$

$$precision = \frac{TP}{TP+FP} \tag{3}$$

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP} \tag{4}$$

$$F1-Score = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

In the video, a large number of people appear, and a large number of objects exist. Therefore, in de-identification, it is necessary to accurately distinguish all of these. At this time, it was judged that it was accurately predicted when all identification targets were non-masked and all de-identification targets were masked in one frame, and it was judged that it was incorrectly predicted if even one object of identification target was masked or even if even one object of de-identification target was non-masked.

In the experiment, the following restrictions were placed on the selection of target faces as follows:

1. Even if the head is lowered or only one side of the face is seen, if more than three keypoints are seen, it is judged as a target face for de-identification;

2. When the face size is 75x75 or more, it is judged as a target face for de-identification.

And, the following restrictions were placed on the selection of target objects as follows:

3. When more than two thirds of the total shape of the object is seen, it is judged as a target object for de-identification;

4. When the object size is 75x75 or more, it is judged as a target object for de-identification;

5. When the aspect ratio of the object is more than 1/3 according to the photographing angle, it is judged as the target object.

Based on the above criteria, ‘de-identification based on selective face recognition’ and ‘de-identification based on selective object recognition’ were performed, and in the case of ‘de-identification based on selective object recognition’, sub-object class was de-identified for each object. As a test set, 1,000 frames with target faces or objects among video frames were randomly extracted, and performance was measured using F1-Score and Accuracy. At this time, the threshold for face detection was set to IOU Threshold=0.5, Probability Threshold=0.7, and the threshold for face identification was set to Similarity Threshold=0.7, and the target for identification/non-identification was selected when the threshold was above the threshold. In addition, the thresholds for object detection were set to IOU threshold = 0.5, probability threshold = 0.7, and probability threshold = 0.7 for object identification, and they were selected as identification/non-identification targets if the threshold was higher. The specified threshold value is determined by experimental values, and may vary depending on the state of the image. If you do factor analysis for accuracy measurements, TP is predicted to be the face/object of the identification after detecting the face/object of the identification target and is non-masked, and TN is predicted to be the face/object of the de-identification after detecting the face/object of the de-identification target and is masked. FP is predicted to be the face/object of the identification after detecting the face/object of the de-identification target and is non-masked, and FN is predicted to be the face/object of the de-identification after detecting the face/object of the identification target and is masked.

For the test, five 1000 frame datasets for faces/objects were configured in a number of videos(Movie, Drama, Youtube etc.), and Table 1 shows the masking accuracy of faces detected in five face datasets, and Table 2 shows the masking accuracy of objects detected in five object datasets. Through Tables 1 and 2, it is possible to see how well selective de-identification has been performed for each target object.

Table 1. Masking accuracy of faces detected within 1000 frames (based on 1 face to be identified)

Face Testset (each 1000ea.)	Accuracy	F1-Score
Dataset1(face)	0.947	0.822
Dataset2(face)	0.940	0.842
Dataset3(face)	0.965	0.849
Dataset4(face)	0.893	0.724
Dataset5(face)	0.919	0.739
Average	**0.933**	**0.795**

Table 2. Masking accuracy of objects detected within 1000 frames (based on 1 object to be identified)

Face Testset (each 1000ea.)	Accuracy	F1-Score
Dataset1(object)	0.967	0.937
Dataset2(object)	0.951	0.904
Dataset3(object)	0.943	0.887
Dataset4(object)	0.928	0.862
Dataset5(object)	0.954	0.876
Average	**0.949**	**0.893**

Table 3. FP+FN rate of faces detected within 1000 frames (based on 1 face to be identified)

Face Testset (each 1000ea.)	FP+FN
Dataset1(face)	32.4%
Dataset2(face)	24.5%
Dataset3(face)	21.4%
Dataset4(face)	34.8%
Dataset5(face)	26.0%
Average	**27.8%**

Table 4. FP+FN rate of objects detected within 1000 frames (based on 1 object to be identified)

Face Testset (each 1000ea.)	FP+FN
Dataset1(object)	14.54%
Dataset2(object)	16.2%
Dataset3(object)	14.8%
Dataset4(object)	13.5%
Dataset5(object)	12.1%
Average	**14.2%**

Multiple faces may exist in one frame, and if any one FP or FN appears in the frame, manual work is required later. Table 3 shows the ratio of FP+FN in five face datasets, and Table 4 shows the ratio of FP+FN in five object datasets. This represents the percentage of frames requiring manual work after selective de-identification processing, with an average of 27.84% for faces and 14.22% for objects. In other words, based on 1000 frames, about 278 frames with faces are manually targeted, and about 142 frames with objects are manually targeted. The reason for this difference is that in the case of identification objects, the shape of the identification object remains constant, while the facial expression continues to change and the angle continues to change, resulting in an average difference of 13.62%. This part is an element that can be affected by the state of the image. However, it makes meaningful that a significant part of the workload can be reduced by selective de-identification based on vision AI.

V. Conclusion

As both data stability and usability become important, the need for personal information protection is increasing, and video data can be easily shared on the Internet, and portrait rights or privacy issues can arise in which individuals don't want and information are disclosed without realizing it. To prevent this, the person of video is being de-identified, but de-identified of the manually progressed image is an essential element, and it takes a lot of time and money to perform de-identification. Instead of manual de-identification, de-identification may be performed collectively through face recognition and object recognition. In the case of batch processing, all objects are de-identified. However, in this case, de-identification must be carried out manually when we want to identify specific objects. It is inefficient because it takes a lot of time, and in the case of video content that has to deal with a large amount of data, a lot of costs are incurred. Therefore, in order to solve the problem, this problem may be solved through automated selective de-identification.

In this paper, we present a method of identifying only targeting people or objects with prior consent using selective de-identification techniques and de-identifying the rest. Selective de-identification technology can be used for marketing according to the business model while ensuring the usability and stability of personal information, so it can be meaningful not only to automatically perform selective protection but also to save a lot of time and money.

Reference

[1] A. Fitwi, Y. Chen, S. Zhu, E. Blasch, and G. Chen, “Privacy-Preserving Surveillance as an Edge Service Based on Lightweight Video Protection Schemes Using Face De-Identification and Window Masking,” *Electronics*, Vol. 10, No. 236, 2021. https://doi.org/10.3390/electronics10030236

[2] S. Mosaddegh, L. Simon, and F. Jurie, “Photorealistic Face De-Identification by Aggregating Donors’ Face Components,” in *Proceedings of Asian Conference on Computer Vision(ACCV) 2014*, Singapore, pp. 159-174, November 2014. https://doi.org/10.1007/978-3-319-16811-1_11

[3] M. Boyle, C. Edwards, and S. Greenberg, “The effects of

filtered video on awareness and privacy," in *Proceedings of ACM Conference on Computer Supported Cooperative Work*, Philadelphia, PA, pp. 1-10, December 2000. https://doi.org /10.1145/358916.358935

[4] J. Crowley, J. Coutaz, and F. Berard, "Perceptual user interfaces: Things that see," *Communications of the ACM* Vol. 43, No. 3, pp. 54-64, 2000. https://doi.org/10.1145/330534.330540

[5] C. Neustaedter, Balancing privacy and awareness in home media spaces, Master's thesis, University of Calgary, Calgary, AB, 2003. https://doi.org/10.11575/PRISM/21505

[6] M. Nishiyama, H. Takeshima, J. Shotton, T. Kozakaya, O. Yamaguchi, "Facial deblur inference to improve recognition of blurred faces," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition 2009*, June 2009. https://doi.org/10.1109/CVPR.2009.5206750

[7] J. Kim and N. Park, "De-Identification Mechanism of User Data in Video Systems According to Risk Level for Preventing Leakage of Personal Healthcare Information," *Sensors*, Vol. 22, No. 7, 2589, March 2022. https://doi.org/10.3390/s22072589

[8] J. Cichowski and A. Czyzewski, "Reversible video stream anonymization for video surveillance systems based on pixels reloca-tion and watermarking," in *Proceedings of 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 1971-1977, November 2011. https://doi.org/10.1109/ICCVW.2011.6130490

[9] H. A. Rashwan, M. A. García, A. Martínez-Ballesté, and D. Puig, "Defeating face de-identification methods based on DCT-block scrambling," *Machine Vision and Applications*, Vol. 27, No. 2, pp. 251-262, 2016. https://doi.org/10.1007/s00138-015-0743-5

[10] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proceedings of the 33rd International Conference on Machine Learning*, pp. 201-210, June 2016. https://proceedings.mlr.press/v48/gilad-bachrach16.html

[11] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-Preserving Deep Learning via Additively Homomorphic Encryption," *IEEE Transactions on Information Forensics and Security*, Vol. 13, No. 5, pp. 1333-1345, December 2017. https://doi.org/10.1109/TIFS.2017.2787987

[12] A. A. Badawi, J. Chao, J. Lin, C. F. Mun, J. J. Sim, B. H. M. Tan, ... and, V. R. Chandrasekhar, "Towards the AlexNet Moment for Homomorphic Encryption: HCNN, the First Homomorphic CNN on Encrypted Data with GPUs," *IEEE Transactions on Emerging Topics in Computing,* Vol. 9, No. 3, pp. 1330-1343, August 2020. https://doi.org/10.1109/T ETC.2020.3014636

[13] M. Tanaka, "Learnable image encryption," in *Proceedings of 2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW),* May 2018. https://doi.org/10.1109/ICCE-China.2018.8448772

[14] W. Sirichotedumrong, T. Maekawa, Y. Kinoshita, and H. Kiya, "Privacy-Preserving Deep Neural Networks with Pixel-Based Image Encryption Considering Data Augmentation in the Encrypted Domain," in *Proceedings of 2019 IEEE International Conference on Image Processing (ICIP)*, pp. 674-678, September 2019. https://doi.org/10.1109/ICIP.2019.8804201

[15] D. H. Ko, S. H. Choi, J. M. Shin, P. Liu, and Y. H. Choi, "Structural Image De-Identification for Privacy-Preserving Deep Learning," *IEEE Access*, Vol. 8, pp. 119848-119862, June 2020. https://doi.org/10.1109/ACCESS.2020.3005911

[16] R. Gross, E. Airoldi, B. Malin, and L. Sweeney, "Integrating Utility into Face De-identification," in *Proceedings of International Workshop on Privacy Enhancing Technologies*, pp. 227-242, 2006. https://doi.org/10.1007/11767831_15

[17] R. Gross, L. Sweeney, F. de la Torre, and S. Baker, "Semi-supervised learning of multi-factor models for face de-identification," in *Proceedings of 2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, June 2008. https://doi.org/10.1109/CVPR.2008.4587369

[18] E. M. Newton, L. Sweeney, and B. Malin, "Preserving privacy by de-identifying face images," in *Proceedings of IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 2, pp. 232-243, January 2005. https://doi.org/10.1109/TKDE.2005.32

[19] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," in *Proceedings of IEEE Transactions on Image Processing*, Vol. 12, No. 8, pp. 882-889, August 2003. https://doi.org/10.1109/TIP.2003.815261

[20] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein, "The generalized patchmatch correspondence algorithm," in *Proceedings of the 11th European Conference on Computer Vision: Part III*, September 2010, pp. 29-43. https://dl.acm.org/doi/10.5555/1927006.1927010

[21] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by

inpainting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2536-2544, 2016. https://doi.org/10.48550/arXiv.1604.07379

[22] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-Form Image Inpainting with Gated Convolution," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4471-4480, 2019. https://doi.org/10.48550/arXiv.1806.03589

[23] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "Edgeconnect: Generative Image Inpainting with Adversarial Edge Learning," *Computer Vision and Pattern Recognition*, 2019. https://doi.org/10.48550/arXiv.1901.00212

[24] U. Demir and G. Unal, "Patch-Based Image Inpainting with Generative Adversarial Networks," *Computer Vision and Pattern Recognition*, 2018. https://doi.org/10.48550/arXiv.1803.07422

[25] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks.," *Computer Vision and Pattern Recognition*, 2019. https://doi.org/10.48550/arXiv.1812.04948

[26] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks," *IEEE Signal Processing Letters*, Vol. 23, No. 10, pp. 1499-1503, 2016. https://doi.org/10.48550/arXiv.1604.02878

[27] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4690-4699, 2019. https://doi.org/10.48550/arXiv.1801.07698

[28] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, "Retinaface: Single-Shot Multi-Level Face Localisation in the Wild," In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5203-5212, June 2020. https://doi.org/10.1109/CVPR42600.2020.00525

[29] Y. Xu, W. Yan, H. Sun, G. Yang, and J. Luo, "CenterFace: Joint Face Detection and Alignment Using Face as Point," *Computer Vision and Pattern Recognition*, 2019. https://doi.org/10.48550/arXiv.1911.03599

[30] Yolov5 [Internet]. Available: https://github.com/ultralytics/yolov5

[31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot Multibox Detector," in *Proceedings of European Conference on Computer Vision*, pp. 21-37, October 2016. https://doi.org/10.1007/978-3-319-46448-0_2

[32] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440-1448, 2015. https://doi.org/10.48550/arXiv.1504.08083

[33] WIDER FACE: A Face Detection Benchmark [Internet]. Available : http://shuoyang1213.me/WIDERFACE/

[34] casia dataset [Internet]. Available: https://www.kaggle.com/datasets/sophatvathana/casia-dataset

[35] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-Celeb-1M: A Dataset and Benchmark for Large-Scale Face Recognition," in *Proceedings of European Conference on Computer Vision*, pp. 87-102, October 2016. https://doi.org/10.1007/978-3-319-46487-9_6

[36] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of International Conference on Machine Learning*, pp. 6105-6114, 2019. https://doi.org/10.48550/arXiv.1905.119 46

[37] Netflix. Squid Game [Internet]. Available: https://www.netflix.com/kr/title/81040344

김대진(Dae-Jin Kim)

1998년 : 대진대학교 전자공학과(공학사)
2000년 : 동국대학교 대학원 전자공학과(공학석사)
2010년 : 대진대학교 대학원 전자공학과(공학박사)
2017년～현 재: 동국대학교 영상문화콘텐츠연구원 조교수
※관심분야 : 동작인식, 이상탐지, 얼굴인식, 콘텐츠 DNA, 워터마크, 딥러닝, 번호인식, 자율주행 등

전윤걸(Youn-Girl Jeon)

2004년 : 서울과학기술대학교 매체공학과(공학사)
2014년 : 서울과학기술대학교 대학원(체육학 석사)
2020년 : 성균관대학교 스포츠과학(체육학 박사)
2021년～현 재: 경북대학교 생태환경대학 강사
2020년～현 재: 성균관대학교 체력과학연구소 선임연구원
※관심분야 : 인간동작분석, 자세 평형성, 인공지능, 디지털 헬스케어, 스포츠 경기력, 이상동작 탐지, 노인 건강, 인지장애 등