IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 15, 2021, accepted December 19, 2021, date of publication December 23, 2021,
date of current version January 7, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3138093

EYEDi: Graphical Authentication Scheme of Estimating Your Encodable Distorted Images to Prevent Screenshot Attacks

TAKAYUKI KAWAMURA[1], TADASHI EBIHARA[2], (Member, IEEE), NAOTO WAKATSUKI[2], AND KEIICHI ZEMPO[2], (Member, IEEE)

[1]Graduate School of Science and Technology, University of Tsukuba, Tsukuba 305-8577, Japan
[2]Faculty of Engineering, Information and Systems, University of Tsukuba, Tsukuba 305-8577, Japan

Corresponding author: Keiichi Zempo (zempo@iit.tsukuba.ac.jp)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Faculty of Engineering, Information and Systems, University of Tsukuba, under Application No. 2020R445, and performed in line with the Helsinki.

ABSTRACT Graphical authentication schemes have the advantage of being more memorable than conventional passwords. Although some image distortion methods have been proposed to prevent the risks of over-the-shoulder attacks (OSAs), these methods cannot prevent camera recording attacks, as the key images are the same each time. In this study, we propose a graphical authentication scheme that generates various distorted images, named Estimating Your Encodable Distorted images (EYEDi). EYEDi generates distorted images by applying several image processing filters to the original images. Moreover, EYEDi estimates the appropriate image processing filter strength based on the authentication data. To measure attack resistance, twenty participants performed three types of attacks (OSA, camera recording attack, and screenshot) 300 times, each using existing methods and EYEDi. The classification error rate of all three types of attacks showed that EYEDi had a lower classification error rate between the legitimate user and attackers. Especially for the screenshot attack, which is the most severe threat model, the existing method was completely broken through, while EYEDi prevented the attacks with a classification error rate of 10%. This result shows that EYEDi can eliminate the screenshot attacker by using the difference in authentication times and a simple improvement in defense performance.

INDEX TERMS Authentication, graphical passcode, over-the-shoulder attack, camera recording, image processing filter.

I. INTRODUCTION

Authentication schemes are used for logging into electronic accounts, such as those with banks, mobile phones, and emails. The key to authentication schemes is typically a secret word only known to a legitimate user; physical characteristics, such as fingerprints, retinas, and faces; or tokens (marks, symbols, and pictures). The requirements of any authentication passcode are memorable, easy to use, and secure [1].

The alphanumeric password is among the most popular and classical authentication systems [2], [3]. It contains numbers, alphabetical letters, and symbols to strengthen its resistance to brute force attack [4]. It is also recommended to use 6-8 characters or more [5]. However, it often proves difficult to remember a long string of letters mixed with symbols [6]. Therefore, many people use weak passwords that are easier to remember and consist of short letters [3], [5], [6], only numbers or letters [5], [7], or words that are strongly related to them [5], [8]. Such passwords are easily guessed and insecure [3], [6], [9]. The same password is not recommended for logging in to different accounts, as if one password is stolen, there is a risk of other accounts also being hacked. However, some legitimate users had to remember sixteen different passwords to log in to different accounts [10]. In short, remembering many complex passwords imposes a heavy memory burden on users.

Graphical passcodes are effective in solving the problems of alphanumeric passwords [1], [11], [12]. The most

The associate editor coordinating the review of this manuscript and approving it for publication was Ilsun You.

important feature of a graphical passcode is the ease of memorizing it. According to the picture superiority effect, image information is more easily remembered than string information due to richer encoding [13]–[16]. In addition, graphical passcodes stimulate cognitive memory, which has the advantage of a low memory burden. In the case of the most graphical passcodes, the authenticator does not need to recall the image from memory but only needs to cognitively identify the image on the authentication screen. From a physical perspective, the advantage of graphical passcodes is that they are more easily entered on touchscreen-based mobile devices than alphanumeric passwords. Since the screens of touchscreen-based mobile devices are small, there is a problem inputting alphanumeric passwords, but graphical passcodes do not require a complicated input form and only need a simple form that selects a small number of images.

Graphical authentication systems can be divided into three categories: 1. those that authenticate by clicking on cue points on the authentication screen [6], [11], [17] 2. those that authenticate by searching for a preregistered key image [1], [18]–[21], and 3. those that authenticate by searching for a combination of key images or/and symbols [22]. The system proposed in this paper, EYEDi, is a graphical authentication system classified as a 2nd category in which the user authenticates by searching a preregistered image.

Graphical passcodes are vulnerable to over-the-shoulder attacks (OSAs), as the feature of being easy to remember gives the attacker the same advantage. OSA allows hackers to memorize key images and breakthrough authentication. Therefore, a method of preventing OSAs has been proposed in which key images cannot be easily remembered. Dhamija *et al.* used geometric patterns as key images that are difficult to remember for attackers [18]. However, the problem with the geometric pattern of Déjà Vu [18] is that it may also make it difficult for legitimate users to remember key images. Hayashi *et al.* proposed a method of solving legitimate users' memory difficulties [1]. Hayashi's system, UYI, can use photos or illustrations as the key image, and the user's task is to identify the distorted key images from the authentication screen, which has key images and some dummy images. UYI is based on the fact that some distorted images can be recovered by referring to the memorized abstract images in one's mind. Zezschwitz *et al.* also show that a person with knowledge of the original image can successfully identify even a distorted image [23]. However, the Déjà Vu's or UYI's key images on the authentication screen are the same each time, so it is in danger to breakthrough by attackers with a very good memory. Therefore, the elements required for a graphical passcode are legitimate user memorability and being unmemorable to the attacker. In summary, the elements required for a graphical passcode are ease of memory for the legitimate user and difficulty of memory for the attacker.

The purpose of this research was to propose a graphical authentication scheme named *Estimating Your Encodable Distorted images* (EYEDi) that generates recognizable but sufficiently distorted images for legitimate users. EYEDi has three features: it can be used securely even in an environment where others can see the authentication screen, it estimates the appropriate distortion based on the record of distortion filter strengths and authentication result label, and only legitimate users can be authenticated sensibly, while attackers have great difficulty in the authentication.

EYEDi's use case is a personal authentication for PCs and mobile phones, frequently used in busy streets, exposing them to the risk of OSAs and recoding attacks. Similarly, there are many situations in which people work on their computers in restaurants, and in such cases, there is a risk of OSAs and recording attacks.

Our experimental threat models are OSA, camera recording attack, and screenshot attack. OSA is a threat model in which an attacker looks over the user's shoulder to look at their screen, and camera recording is the same threat model but uses a camera. A screenshot attack is a threat model in which an attacker zooms in on a camera-recorded video with graphical software and can see the key image stolen from the attack.

EYEDi's authentication procedure first registers 5 key images, as shown in Fig. 1(a), then presents a grid of 25 distorted images (5×5) on the authentication screen, and finally finds and selects the distorted key image by clicking on it, as shown in Fig. 1(e). Details are described in Chapter 3. EYEDi can generate a variety of filtered patterns from a single image. It is difficult for an attacker to break through the authentication by previously remembered memories, which are obtained by OSAs, as many different patterns are generated from a single image. The attacker will try to hack the graphical passcode by referring to the previously taken key image of the authentication system. However, the key image displayed on the authentication screen by EYEDi will have a different pattern than the previous image generated using the same key image, which makes it difficult for an attacker to find the correct key image, as shown in Fig. 1(f). Thus, a legitimate user can be identified sensibly by seeing EYEDi's distorted images, but attackers will have great difficulty breaking through authentication. Fig. 1(b) and (c) show the generating process of EYEDi's filtered patterns. EYEDi's filtered patterns are generated by trimming and applying three image processing filters to the original images. This study estimated the boundary of legitimate user identification in the image processing filter intensity space based on pretraining with authentication data.

II. RELATED WORKS

According to Hayashi *et al.*, personal authentication systems can be characterized by 1. ''something you have (Physical key)'', 2. ''something you are (Biometrics)'' or 3. ''something you know (Memory)'' [1].

1. Physical Key: The most classical authentication system is a physical key, which is a key that authenticates by matching specific physical properties. In recent years, typical physical authentication systems have used combined physical shapes and magnetic patterns. However, since physical tokens

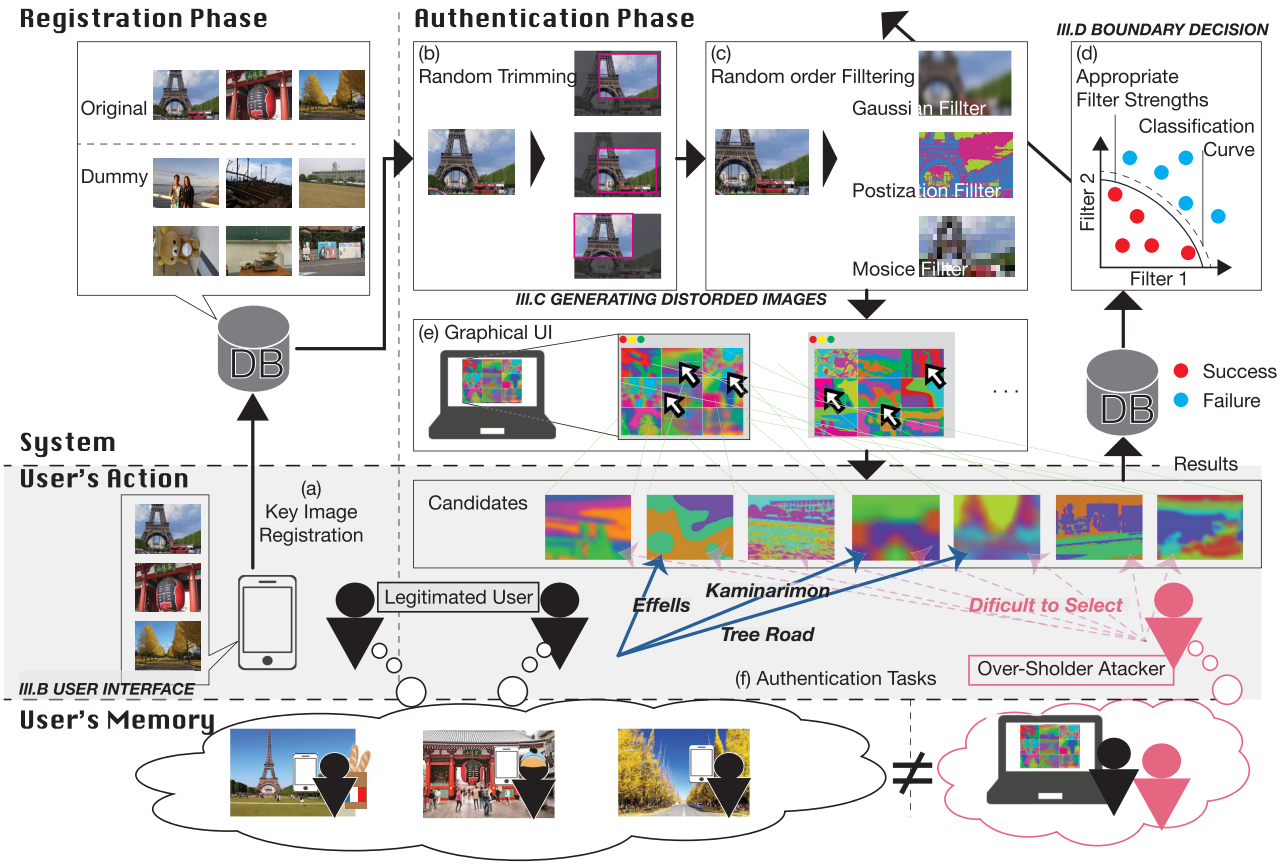

FIGURE 1. System Configuration and user interface of EYEDi: (a) First, a legitimate user registers several key images from his or her mobile device or PC. In our experiment, the number of key images was five. For simplification, this illustration shows three key images. The key image is stored in a database, and the experimenter prepares a dummy image in advance. (b) The key image is randomly cropped to a size of 4/5. Crop processing makes it difficult for the attacker to find the cue since the feature points in the key image appear and disappear with each authentication. (c) Three image processing filters, Gaussian, Posterization, and Mosaic, are randomly applied to the key image. (d) The strengths of the three filters are calculated from the classification curve estimated from past authentication data. (e) The key images and dummy images distorted by cropping and filtering are presented on the authentication screen in random order. (f) The user selects the key images from the screen to authenticate. The legitimate user will recall many key image features using familiar memories of these key images as cues. Attackers can record all the selected images by camera recording the authentication screen of a legitimate user. The attacker tries to break through the authentication based on the features contained in the recorded images but cannot find the key image due to the loss of feature points caused by cropping and various distortion changes.

can be lost or stolen, they are often used as a part of hybrid authentication schemes in conjunction with others, such as one-time passwords.

2. Biometrics: It is a system that authenticates individuals based on the characteristics of their fingerprints, faces, voiceprints, and behavioral patterns [24], [25]. These features vary between people, so we can identify a person by recognizing their appropriate biometric features. Biometrics are widely used in mobile devices, gates, and banking services.

Biometrics have become popular because they are convenient, as there is only a need for the presence of the user, and not for physical tokens. However, biometrics have some problems, such as high error rates due to noise and lighting conditions and theft of fingerprints, voiceprints, and facial features from photographs and recordings [24].

3. Knowledge-based Authentication: In the following parts of this chapter, we describe a knowledge-based authentication system (memory) that makes physical thefts difficult. A knowledge-based personal authentication system does not use physical features but uses knowledge. Typical knowledge-based personal authentication systems include alphanumeric and graphical passwords [2], [3].

3a. Nongrapical Authentication: Since the key for authentication is knowledge, the risk of physical loss or theft is low, but the loss and difficulty of memory are problems [26], [27]. An alphanumeric password is a string password that mainly consists of letters and numbers. An alphanumeric password should ideally be a complex and long string of characters to avoid an attacker guessing the password. Using both alphanumeric characters and symbols can make brute force attacks difficult [4]–[7]. However, the difficulty of remembering complex passwords has led many users to use dictionary words or short strings of characters as passwords, which is problematic [5], [8], [9]. It has been shown that a password can be easily guessed by using words from the dictionary. We also know that many people use the same password to log

into different accounts [10]. Thus, alphanumeric passwords often use inappropriate passwords due to memory difficulties, which cause various problems.

3b. Graphical Authentication: To solve the memory problem of alphanumeric passcodes, various graphical passcodes have been proposed [2], [9], [11], [12], [19], [21], [22], [28], [28]–[32]. Graphical passcodes are a set of authentication schemes that utilize the visual superiority effect that visual information is easier to remember than textual information [4]. The advantage of image information is that it can present a vast amount of information at once compared to textual information. Furthermore, although it can be difficult for humans to recognize a huge amount of information in text strings, humans can recognize the necessary features in image information [3]. Compared to character string information, graphical information can generate many patterns, making brute force attacks very difficult and could be a very effective authentication system because it is easy to remember.

However, while graphical passcodes are easy to remember, they may be susceptible to OSAs. In recent years, with the spread of mobile devices, authentication has become a common practice in many places. In busy urban areas and on trains, people must repeatedly authenticate to use their mobile devices. A rogue user can attempt to steal an authentication key by watching a legitimate user's authentication process from behind them. The main advantage of a graphical passcode is that legitimate users can easily memorize it, but the main disadvantage is that it is equally easy for illegitimate users to memorize it through OSAs. Therefore, it is essential to devise a method to prevent OSAs in graphical passcodes.

PassPoints is a graphical passcode that allows users to authenticate themselves by clicking on some points on an image in a sequence [6]. PassPoints makes use of the fact that humans can easily remember characteristic points in images. If the user selects the five or six click points in an image as keys, he or she can create the same security level as 64-character or greater alphanumeric passwords. The click point is recognized by clicking on a predefined area on an image without clicking on the exact pixel. By constructing a story for the click points on the image, they can easily remember the key points. However, there is a problem in that it is easy to detect hints from personal information and personal information in images due to the use of clear images.

Dhamija *et al.* proposed a geometric pattern-based graphical authentication system [18]. Their geometric patterns are much easier to remember for attackers than text data for legitimate users because geometric patterns are unfamiliar in everyday life. However, Déjà Vu's geometric patterns have the problem that it is difficult for legitimate users to find stories and familiar features to remember. Therefore, if no authentication is performed for a long period of time, a legitimate user may not be able to remember their key image. When attackers hack a graphical authentication system, they focus on the areas that humans tend to focus on.

Katsini *et al.* proposed a system that encourages users to create strong passwords by hiding areas where they are likely to concentrate their gaze [17]. If the area of concentrated gaze is hidden, the user pays attention to other areas and registers the key gesture, which is difficult for the attacker to estimate. Thus, there is a deep relationship between gaze and graphical authentication systems, and research on the relationship between gaze and image recognition has attracted a large amount of attention [33]–[35].

Hayashi *et al.* proposed a geometric pattern-based authentication system that generates geometric patterns from photographs or pictures [1], [20]. Legitimate users can register images stored on their mobile devices or PCs as key images. An abstract geometric pattern is generated from the registered key image, and then authentication is achieved by finding the key image among the geometric images. Since legitimate users know the original image, they can easily find the geometric patterns transformed from the original image. Meanwhile, it would be difficult for an attacker to remember the geometric patterns when attempting to memorize images by OSAs.

Khamis *et al.* proposed a letter password system that protects against OSAs by distorting characters [36]. Common letter password systems use a method of replacing the input characters with asterisks to protect against OSAs. However, the problem is that many users often forget what they have typed. When using Mohamed's system, a legitimate user can mentally recover the replaced characters from one's memory. On the other hand, the attacker has difficulty identifying replaced characters, as they do not have this memory.

Thus, graphical passcodes have been studied to maintain the ease of memory and OSA resistance. However, there is a drawback that any of the abovementioned graphical passcodes can be easily broken by the difficult situation of OSAs, such as camera recordings or screenshot attacks.

III. GRAPHICAL AUTHENTICATION SYSTEM BASED ON THE SCHEME OF ENCODABLE DISTORTED IMAGES

A. OVERVIEW

EYEDi can generate appropriate distorted images that legitimate users can recognize. In this section, the authentication system of EYEDi is divided into a user interface, a distorted image generator, and appropriate filter strengths. First, we explain an overview of EYEDi in this subsection. As shown in Fig. 1(a), a legitimate user can use arbitrary images, such as photos or illustrations, stored in his or her mobile devices or PCs. A few key images in total should be registered (in our experiment, five key images were registered, but to simplify the demonstration, Fig. 1(a) shows three key images). As shown in Fig. 1(e), EYEDi's UI displays several distorted images in a gridded authentication screen. The gridded authentication screen contains all of the distorted key images, and the other images are distorted dummy images. These dummy images were prepared in advance by the experimenters. A different dummy image set is used for each user. Furthermore, the dummy images are distorted by

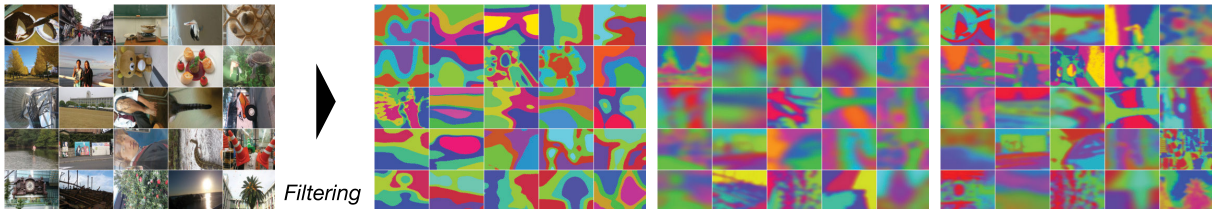

FIGURE 2. One of the original images and the images distorted by EYEDi in the same grid order: EYEDi can generate images with different impressions so that the impression of Entry UI is different each time. Attackers are distracted by the different impressions of the Entry UI. In this figure, the four image grid orders are the same, but EYEDi's authentication system generates different image grid orders at each authentication attempt. These authentication screens of EYEDi have the same arrangement of images, but the distortion of the images is the same level of the experimental authentication, so the distortion of these images is not emphasized.

filtering processing as well as the key images. The distortion processing of each dummy image is randomly chosen from the processing filter applied to the key images. The user authenticates their access by clicking on all preregistered distorted key images among the authentication screens. Thus, EYEDi can be used for touchscreen-based mobile devices since it does not require complex operations and only requires five clicks. EYEDi's method of generating distorted images applies clipping and three types of image processing filters. The requirement for filter strength is to generate appropriate distortions identifiable to the legitimate user and difficult for the attacker to recognize. To estimate the filter strength, we used the training dataset of discrimination success or failure at the filter strengths shown in Fig. 1(d). We labeled the discrimination success or failure on the space stretched by the filter strength and estimated the discrimination availability boundary. EYEDi determines a set of filter strengths on the success side near the classification boundary as the appropriate filter strengths. The distorted image generated at the filter strength determined is identifiable to the regular user and is sufficiently distorted. Figure 2 shows EYEDi's authentication screen, which was used in our experiment. These images are used with different trimming patterns and filter processing, so those that do not have the original image find it difficult to identify each image on these authentication screens. The distorted images are generated by extracting them from many filter strength sets so that EYEDi can generate various patterns of distorted images.

B. USER INTERFACE

A legitimate user can select several images from his or her PC or mobile device and register them as key images, as shown in Fig. 1(a). In this study, the legitimate users registered five key images. Each key image can be an arbitrary image, such as a photo or an illustration. The advantage of using photos or illustrations is that the images the user selects are more memorable to them than other images. In addition, when users choose the images themselves, they can devise a way of choosing images that they will find easy to remember. PassPoints and PassFaces are undesirable when using images that contain personal information, as they can lead to hints of key guessing and identity theft through OSAs, but EYEDi displays sufficiently distorted images that there is no risk of guessing or identity theft. Since there is no such restriction, various memorable images can be used as key images for the user.

Figure 2 shows an example of EYEDi's authentication screen in our experiments. These authentication screens display 25 distorted images in total in a 5×5 grid. Of the 25 total distorted images, 20 are dummy images, and five are distorted key images. The experimenters prepared the dummy images in advance. The authentication task followed two simple steps:

1) The user identifies the five distorted key images among the 25 images and clicks on them. When the user clicks the images on the authentication screen, a focus frame is drawn around their selected images. (If the user wants to deselect an image, the focus frame for the selected image will disappear when the user clicks on it again.)
2) When the user selects a total of five or fewer images and inputs the end key, EYEDi displays the authentication's success or failure.

C. DISTORTED IMAGE GENERATOR

EYEDi's distorted images are generated by trimming and three different image processing filters. Figure 1(b) shows a trimming processing. Depending on the difference in the cropping process, the arrangement of feature points, such as circles and corners, may change, disappear, or appear. These feature point changes cause effective disturbance in EYEDi. In the case of trimming a clear image, the disturbance effect of the cropping process is small, but if the filtering process distorts the image, as shown in Fig. 1(c), the disturbance effect becomes very large. In this study, the images were trimmed to $\frac{4}{5}$ size, and the trimming position was random.

Figure 1(c) shows EYEDi image filter processing. This study used three image processing filters: the Gaussian filter, Posterization filter, and Mosaic filter. The desired filter characteristic in this study is the ability to generate an abstract image from the original image. However, digitally scanned and computationally restored distorted images are a risk if

only reversible filters are used. Therefore, we used not only reversible but also irreversible filters. We used reversible Gaussian and irreversible mosaic filters to create an abstract pattern from the original image and an irreversible posterior-ization filter to reduce the boundary information in the image. Gaussian and mosaic filters reduce the features of interest to an attacker by omitting details in the image. Conversely, legitimate users can seek help in recovering their memories from abstract patterns. Combined with posterization to smooth the brightness change, it reduces the number of characteristic figures in the image. By applying these three filters in sequence, we can generate abstract patterns that are difficult to recover informationally but still retain the general shape of the image.

The distorted image, $g(x, y)$, is generated by applying the three image processing filters to the original image, $f(x, y)$, in random order as follows:

$$g(x, y) = \left(\prod_{i \in \mathrm{G,P,M}} \hat{F}_i \right) \cdot f(x, y),$$

where $\hat{F}_\mathrm{G}$, $\hat{F}_\mathrm{P}$, and $\hat{F}_\mathrm{M}$ denote the operator of the Gaussian filter, posteriorization filter, and mosaic filter, respectively. In this study, to generate appropriate distorted images, EYEDi estimated the standard deviation σ of the Gaussian filter, the number of colors c of the Posterization, and the filter width w of the Mosaic filter using the authentication data and generated the images as shown in the following equation.

For a given $f(x, y)$, the image filtered by Gaussian, $\hat{F}_\mathrm{G} \cdot f(x, y)$, is defined as

$$\hat{F}_\mathrm{G} \cdot f(x, y) \stackrel{\mathrm{def}}{=} \sum_{n=-\sigma}^{\sigma} \sum_{m=-\sigma}^{\sigma} f(x+m, y+n) G(m, n),$$

where m and n denote the horizontal and vertical widths of the Gaussian filter, respectively; σ denotes the standard deviation of the Gaussian filter defined as follows:

$$G(m, n) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{m^2 + n^2}{2\sigma^2} \right),$$

respectively.

Since the Posterization filter was applied together with standardization, the Posterization filter performance in this study is as follows. For $f(x, y)$, the image filtered by posterization, $\hat{F}_\mathrm{P} \cdot f(x, y)$, is

$$\hat{F}_\mathrm{P} \cdot f(x, y) \stackrel{\mathrm{def}}{=} \left\lfloor \frac{c}{2^{\mathrm{n}_b}} \left(\hat{F}_\mathrm{S} \cdot f(x, y) \right) \right\rfloor,$$

where c and $\hat{F}_\mathrm{S}$ denote the number of colors and the standardization filtering operator, respectively. The colors were randomly applied as the $\left\lfloor \frac{360}{c} \right\rfloor$ degree for the Hue difference between the colors, with a fixed value for the saturation of 200 and value 200 in the HSV color space.

For $f(x, y)$, the image filtered by standardization, $\hat{F}_\mathrm{S} \cdot f(x, y)$, is defined as

$$\hat{F}_\mathrm{S} \cdot f(x, y) \stackrel{\mathrm{def}}{=} (f(x, y) - \mu) \frac{s}{s'} + 2^{\mathrm{n_b}-1},$$

where μ denotes the input image's average value, s, s' denote the input and output image's standard deviation, n_b denote the input image's bit number, and $2^{n_\mathrm{b}-1}$ is the average value of the output image.

For a given $f(x, y)$, the image filtered by Mosaic, $\hat{F}_\mathrm{M} \cdot f(x, y)$, is

$$\hat{F}_\mathrm{M} \cdot f(x, y) \stackrel{\mathrm{def}}{=} \sum_{n=-w}^{w} \sum_{m=-w}^{w} \frac{f(\lfloor \frac{w}{255} x \rfloor + n, \lfloor \frac{w}{255} y \rfloor + m)}{nm},$$

where w denotes the Mosaic filter width.

Figure 1(c) shows how the Gaussian filter makes the image's outline fuzzy; how the posteriorization filter reduces the color gradation and changes colors at the same time; and how the mosaic filter gives the image an angular appearance. Since Gaussian and posteriorization filters irreversibly delete information from the image, it is difficult to recover information from them.

It was also found that when the order of application for these three filters and trimming area was different, the distorted images produced differed significantly, as shown in Fig. 2, even if each filter had the same intensity. Therefore, we treated the three filter application orders $3! = 6$ as different data to determine the filter intensity values described below.

D. APPROPRIATE FILTER STRENGTHS

When generating distorted images with EYEDi, it is necessary to estimate the filter strength that can be authenticated for legitimate users but not for attackers in each filter order. Therefore, we estimated the three filter variables, σ, c, and w, by applying machine learning to each user's authentication data. During authentication, the filters and success labels of the distorted images are obtained, and machine learning is performed using these data.

The appropriate filter strength, S, is calculated by

$$S = \lfloor E_p \times p \times D \rfloor, \tag{1}$$

where E_p denotes the random point on the classification curve obtained by machine learning; p denotes the most recent correct answer rate; D denotes the attenuation rate, D; and S denotes the applying filter strength. In this experiment, we applied $D = 0.75$ based on our preliminary experiments.

In generating the distorted image, one of the six orders of filter processing was selected in each attempt and image. The data used for the estimation were three types of image processing filter intensity values and a discrimination label of success or failure.

Table 1 shows the initial values of the data. The default values are labeled success for filter intensity values that result in a clear image and failure for the settings that produce unrecognizable distortions. Initially, $6 \times 3 \times 3 = 54$ success labels and $6 \times 3 \times 4 = 72$ failure labels were prepared. In this regard, success labels were set for all initial data so that the image distortion was small and close to the original image. In contrast, the failure labels were set up such that

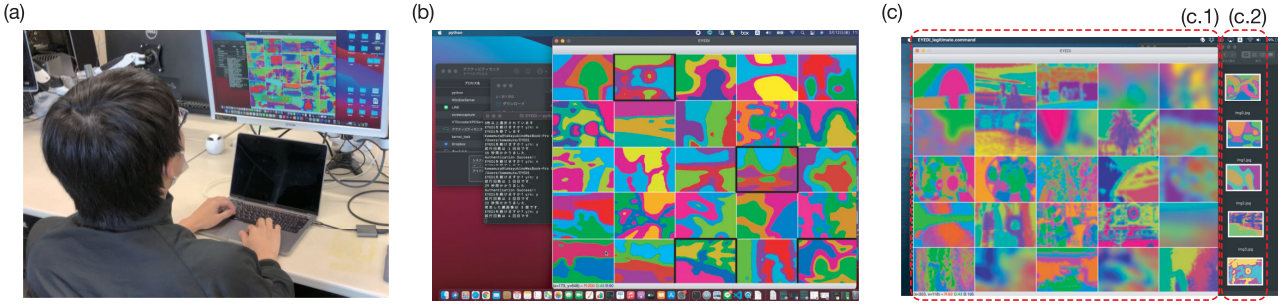

FIGURE 3. (a) Over-the-shoulder attacker's view, (b) camera recording attacker's view, and (c) screenshot attacker's view (c.1) The five key images obtained from the screenshot. The attacker obtained all five key images shown in (c.2) in the screenshot attack. The attacker tried to hack the authentication of (a) by checking the key images of (c.2).

TABLE 1. Initial values of learning data set.

σ	c	w	answer label
2, 4, ..., 12	3, 4, 5	94, 96, 98	1
90, 92, ..., 100	2, 3, 4	4, 6, 8, 10	0

the images were heavily distorted and far from the original images. The reason for inputting these initial data was that the amount of data required to estimate the support vector machine (SVM) had to be greater than a certain amount. We estimated the classification curve in the space stretched by the filter strength, and SVM was used for estimation.

The distorted image generated by the determined filter strength was presented to the authentication user interface, as shown in Fig. 2. After authentication, the key image ID data, the filtering order, three filter intensity values, and the label of success or failure were stored in the database. After the second time, the filter strength is estimated using the updated data.

IV. USER TEST

A. USE CASE

EYEDi's use case authenticates with a PC or mobile device in a crowded, busy street or restaurant. The attacker will try to identify the key image by recording the authentication screen in the crowd. Therefore, the threat model in this paper is a screenshot attack, which assumes that a camera records the authentication screen. Since the captured video is clear, the attacker can identify and extract all the distorted key images selected by enlarging and editing the video. Therefore, it is possible to perform an attack by referring to the identified images. However, in this study, we created a simple system that achieves the same effect to reduce the actual video recording and editing work with a camera. We built a threat model system that stores five distorted key images selected by a legitimate user and presents the stored images to the attacker.

B. EVALUATION METRICS

The metrics of this study are the number of correct key images and authentication time. In addition, this authentication system identifies between attackers and legitimate users based on the limits of the number of correct key images and the authentication time. Therefore, even if the number of correct key images was sufficient, the user would be rejected if the authentication time was longer than the limit, and even if the authentication time was shorter than the limit, the user would be rejected if the number of correct key images was small.

1) USABILITY METRICS

The usability metrics are the classification error rate of the false rejection rate (FRR) and how short the authentication time is compared with the attackers. The longer the authentication time is, the worse the usability and the more tired legitimate users are. Ideally, legitimate users should be able to authenticate for a short period. The larger the FRR, the worse the usability. Ideally, the FRR should be a low rate.

2) SECURITY METRICS

The security metrics are the classification error rate of the false acceptance rate (FAR) and how long the authentication time is compared with the attackers. The longer the authentication time is, the worse the security and the greater the chances for attack. Ideally, legitimate users should be able to authenticate for a short period, and attackers should be forced to take a long time to do so. The larger the FAR, the worse the security. Ideally, the FAR should be a low rate.

To identify the attacker and the legitimate user, we calculated some regions for classifying the attacker and the legitimate user based on the number of correct key images and the authentication time.

3) LONG-TERM MEMORY METRICS

Since the images in EYEDi are distorted images, there is a possibility that the key images will be forgotten. Therefore, it is necessary to verify the long-term memory of legitimate users of EYEDi. In addition, in a real environment, most users do not consciously memorize the images. In this study, we conducted the experiment 2–3 weeks later without prior notice to collaborators.

C. EXPERIMENTAL TEST SUBJECTS

The participants of the study were 20 university students between the ages of 20 and 25 years old, with 12 males and 8 females in total. The number of attackers was 13, seven attackers attacked one account with 15 trials, five attackers attacked two accounts with 15 trials, and one attacker attacked three accounts with 15 trials. All attack trials consisted of 15 trials each of OSAs, as well as camera and screenshot attacks, and the total number of attack attempts for each threat model was 300. We conducted the experiment 2-3 weeks later for the long-term memory experiment without prior notice to the collaborators. Because the long-term memory experiment was conducted without prior notice, the number of collaborators for the long-term memory experiment was 5 due to the schedule of the collaborators. Five long-term memory experimental collaborators attempted to log in to their account 10 trials after 2–3 weeks of the legitimate user experiment. We experimented during August and September 2020. All collaborators were briefed on how to use the graphical password and practiced before the experiment. Legitimate users were given an explanation of the proposed method and practiced with it before the experiment. The role of the attacker was performed by someone skilled in the manipulation of the proposed method. Specifically, the attacker participated in the experiment as a legitimate user before, and those who were sufficiently familiar with the operation of EYEDi participated in the experiment as the attacker.

The experimental design was based on ethical considerations[1] If a collaborator feels tired, he or she can stop the experiment and take a break during the experiment to reduce the physical burden.

D. EXPERIMENTAL PROCEDURE

To prepare for the experiment, the collaborators submitted five key images. The collaborators could submit any images from their PC or mobile devices. EYEDi has no user authentication data at the beginning of the experiment, and the filter strength cannot be estimated by SVM. Therefore, we initialized the authentication data by the pseudo authentication data at the beginning of the experiment. We initialized the authentication data with the values shown in Table 1, assuming that the extremely small filter strength is successful as there is almost no distortion, and the extremely large filter strength is unsuccessful as the distortion is too large.

The user completed the authentication by finding five key images from the authentication screen. A legitimate user needed to try a total of 100 authentication attempts, but they could take a break if they felt tired. The attacker's authentication screen is shown in Fig. 3(a), and the five distorted key images identified by recording are shown in Fig. 3(b). The attacker attempted to break through the authentication screen in Fig. 3(a) while referring to Fig. 3(b). The attacker could attempt this 15 times.

In the OSA, camera, and screenshot attack resistance experiment, we measured EYEDi's attack resistance by providing the number of correct key images to the attacker in a favorable setting for the attackers.

E. LOG-IN TIME AND FEATURE POINT CHANGES IN EACH TRIAL RANGE

The authentication time was measured from when the authentication screen was displayed, the end time was set to when all the images were selected, and the end button was entered. Fig. 4(a) shows the legitimate user's authentication time for an authorized user to log in when using EYEDi in each trial range. We detected a Shapiro–Wilk test in each trial range of authentication time and confirmed that all groups did not follow a normal distribution. Therefore, we detected the Wilcoxon signed-rank test and found significant differences in the trial number ranges {(1,10], (51,60]}, {(11,20], (51,60]}, {(11,20], (61,70]}, {(21,30], (51,60]}, {(21,30], (61,70]}, {(31,40], (61,70]}. The authentication time for legitimate users remained constant at approximately 34 seconds in the trial number range of 1-10 for the first and 91-100 for the last log-in attempts. It is important to note that the system is set up such that the distortion of the image displayed on EYEDi is small at the first log-in attempt and the distortion increases as the number of log-in attempts increases. In this regard, we can see that legitimate users can be authenticated in approximately 34 seconds, regardless of the presence or absence of image distortion; The distorted images generated by EYEDi neither overburden the identification process for legitimate users nor affect the authentication time.

We used AKAZE to calculate image features, which is robust to object scaling, rotation, blurring, and brightness changes, making it suitable for use on images distorted by EYEDi [37]. We computed the feature points with the best feature match in brute force for the feature points calculated from each image. The feature distance was treated as the Hamming distance of the 61-dimensional features, and for the feature points that disappeared by EYEDi, all 61-dimensional features were treated as $61 \times N_b$, where N_b is the feature bit number. We calculated the feature matching rate R as follows:

$$R = \frac{\sum_{n=1}^{N_p} D_n}{61 \times N_b \times N_p}, \tag{2}$$

where N_b and N_p denote the feature distance of the n-th feature point and the number of feature points, respectively. Matching rate R as $R = \frac{\sum_{n=1}^{N_p} D_n}{61 \times N_b \times N_p}$, We calculated the feature matching rate for the key and distorted images and used it as an image change index. Fig. 4(b) shows the feature matching rate between the key image and the EYEDi's distorted image in each trial range of legitimate users. We performed a Shapiro– Wilk test in each feature matching rate trial range and confirmed that all groups did not follow a normal distribution. Therefore, we detected the

[1]The study was conducted according to the guidelines of the Declaration of Helsinki and approved by the Ethics Review Committee of the Faculty of Engineering, Information and Systems, University of Tsukuba (2020R445).

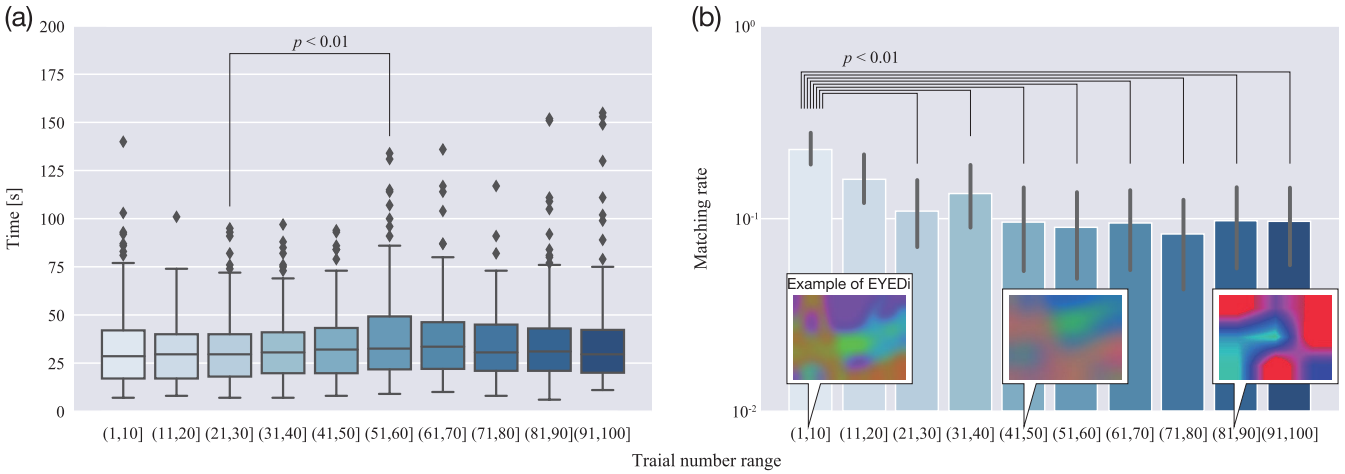

FIGURE 4. Authentication time and feature matching rate between the key image and EYEDi's distorted images in each trial number range. (a) The authentication time using box plots. (b) The feature matching rate between the key image and the EYEDi's distorted image using the box plot. AKAZE calculated the image features. We calculated the feature matching rate to the feature point distances with the best match in a brute force fashion for the 61-dimensional features obtained by AKAZE. We treated all 61-dimensional features of feature points lost by EYEDi's distortion as zero. For distorted images for which no feature points could be obtained, the feature distance was calculated as 300. The authentication time in each trial range was approximately 30 seconds. The feature distance between the key image and the distorted image increased with the number of authentication attempts, and the number of cases where the feature points of the distorted image could not be obtained increased. For outliers, the length limit of the whisker was set to 1.5 times the length of the box.

Wilcoxon signed-rank test and found significant differences in the trial number ranges {(1,10], (21,30]}, {(1,10], (31,40]}, {1,10], (41,50]}, {(1,10], (51,60]}, {(1,10], (61,70]}, {(1,10], (71,80]}, {(1,10], (81,90]}, {(1,10], (91,100]}, {(11,20], (51,60]}. As the number of trials increases, the average feature matching rate decreases, the learning progresses, and the change in images becomes larger. In the range of 71-80 trials, the average feature matching rate was the smallest, and then it increased in the range of 81-90 trials and 91-100 trials. The distortion was too large in the 71-80 trials range, so the learning reduced the distortion. Figure 4(b) shows that EYEDi gives changes to the image as the number of trials increases. In summary, Fig. 4(a) and (b) show that EYEDi can distort the image by increasing the number of trials, but it does not significantly affect the authentication time for legitimate users.

F. ATTACK TEST

In this experiment, we used a more robust attack configuration for OSAs than in previous studies. Compared to OSAs, screenshot attacks and camera recording allow us to have perfect memory and more powerfully hack the image authentication system. Since modern mobile devices are equipped with standard cameras and have a widespread filming function, it is more reasonable to assume they are exposed to screenshotting or other camera recording attacks rather than OSAs. The videos captured by a high-performance camera are very clear, and the attacker can magnify and edit the video to identify all the distorted key images selected by a legitimate user. Therefore, we assumed that all five sets of distorted key images had been stolen from the authentication screen of a legitimate user. This study automatically stored distorted key images selected by a legitimate user and presented them to the attacker to simplify the recording and editing process. In other words, the attacker can see all five distorted key images.

Figure 3 is an example of the interface of EYEDi's authentication screen. The attacker tried to break through EYEDi while checking the five distorted key images prepared, as shown in Fig. 3. The attacker tried to break through the authentication by using lines, circles, and patterns in given distorted key images as clues. The attackers had previously participated in the EYEDi experiments as legitimate users and had sufficient knowledge of EYEDi. The attack process involved a trial to find and click the distorted key images on the EYEDi authentication screen, as shown in Fig. 3(a) while checking the distorted key images shown in Fig. 3(b).

Each authentication trial was completed, and success or failure was presented to the attacker. In addition, the attacker was shown the success or failure and how many of the five key images they identified correctly. The attacker made 15 of the above attack attempts. Note that the distorted key images shown in Fig. 3(b), which were presented during the 15 attack trials, did not change in each trial. Moreover, during this experiment, the data from the authentication results were not updated to ensure that the difficulty level of each attacker did not change due to changes in the EYEDi strain estimation results.

To compare the performance of EYEDi, we used the UYI authentication system as an existing method in our experiments [1]. A total of $25 = 5 \times 5$ images are shown for both EYEDi and UYI, and we can see that the images are distorted. The most significant difference between EYEDi and UYI is that UYI can only produce one type of distorted image, whereas EYEDi can produce various distorted images.

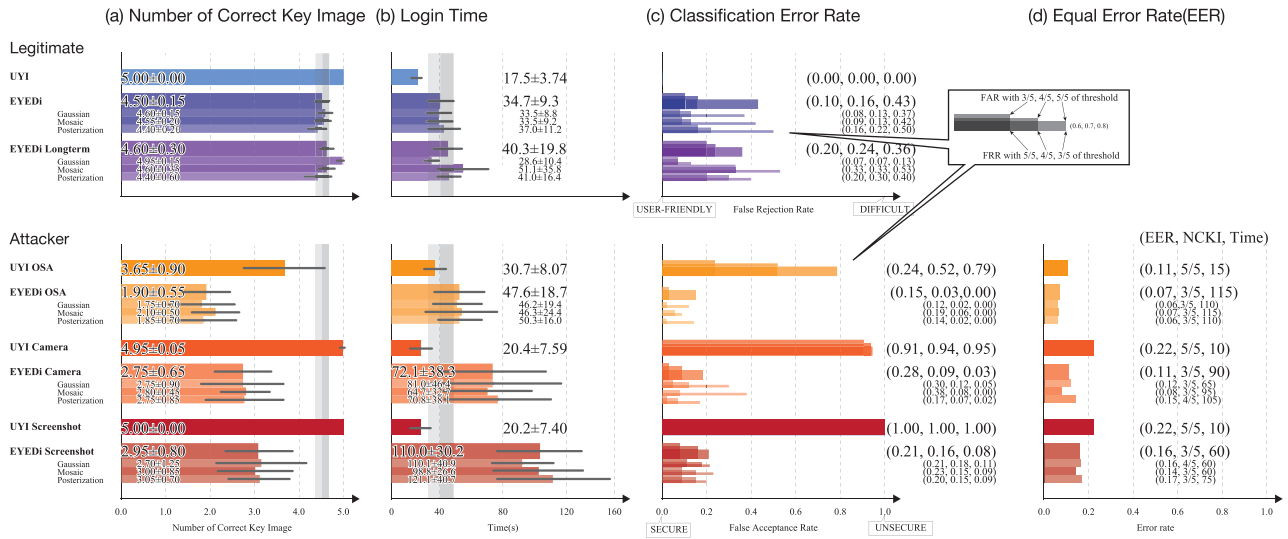

FIGURE 5. **We compared the number of correct key images, time, and the classification error rate of both EYEDi and UYI (an existing method). (a, b) The bars for the number of correct key images and time show the mean values, while the whisker line and the numbers behind the ± symbols show the standard deviations, and the gray area shows the standard deviation area for EYEDi's legitimate users. (c) The classification error rate indicates the false rejection rate (FRR) for a legitimate user and the false acceptance rate (FAR) for an attacker when the response time is 60 seconds. There are three, four, or five correct key images. (d) The equal false rejection rate shows the value obtained when the parameters of the number of correct key images and the authentication time are varied; the UYI data show the results for legitimate users and three attacks: screenshot, camera, and OSA. In the EYEDi data, the auxiliary bar graphs for each filter applied at the end are also shown together.**

The method for generating the oil-colored image was set to 20×20 applied pixels; the frequency of the luminance mode of the filtered range was set to b_{mf}; and the integrated value of the rgb of the luminance mode was set to R_{mt}, G_{mt}, and B_{mt}, respectively, so that the applied values were $R = \frac{R_{mt}}{b_{mf}}$, $G = \frac{G_{mt}}{b_{mf}}$, and $B = \frac{B_{mt}}{b_{mf}}$. The attacker was presented with a set of five distorted key images and authenticated while the distorted images were at hand.

1) NUMBER OF CORRECT KEY IMAGES

Figure 5 shows a bar graph of the number of key images, authentication time, classification error rate, and equal error rate for legitimate users and attackers. In the case of the UYI system (existing method), the number of correct key images was 5 for legitimate users, but the screenshot attack completely breached the UYI system. Similarly, in the case of the camera recording attack, the number of correct key images was 4.95, which was almost completely breached by the attack. However, the number of correct key images for the OSA was 3.65, indicating that the attack had a protective effect.

On the other hand, the number of correct key images for EYEDi is 4.50 for legitimate users, which is inferior to UYI. However, the results for screenshot attacks, camera attacks, and OSA are 2.95, 2.75, and 1.90, respectively, indicating that EYEDi provides better protection than existing methods in all attack methods.

Since the distorted image impression of EYEDi tends to be determined by the last image processing filter, we also analyzed the details of the correct answer by the last filter. However, there was no significant difference in the order of any of the filters.

We also found no significant difference in the number of correct key images for regular users two weeks after registration. In other words, the regular users of EYEDi retained their long-term memory even after two weeks.

2) AUTHENTICATION TIME

The authentication times of UYI (existing method) were 17.5 s for the legitimate user, 20.2 s for the screenshot attack, and 20.4 s for the camera recording attack, which were close to one another, but only the OSA was slightly longer at 30.7 s.

The authentication time of EYEDi was 34.7 s for legitimate users, which is slightly longer than the existing methods. However, the screenshot attack took 110.0 s, the camera recording attack took 72.4 s, and the OSA took 47.6 s. The more advantageous the settings were for the attacker, the longer it took. Two weeks after registration, the authentication time of the legitimate users was found to be not significantly different.

From the usability viewpoint, the authentication time for a legitimate user is approximately twice as long in EYEDi, indicating that usability is reduced. However, from the security viewpoint, the authentication time for attackers is sufficiently longer than that of legitimate users. This result shows that EYEDi improves security performance.

3) CLASSIFICATION ERROR RATE

We used the number of correct key images and the authentication time as the classification criteria for legitimate users and attackers. The classification error rate in Fig. 5 clearly

shows the results when the authentication time was less than 60 s, and the number of correct key images was more than 3, 4, or 5.

The existing method (UYI) has a false rejection rate (FRR) of 0% for legitimate users in all conditions. However, the OSA classified the attacker with a false acceptance rate (FAR) of 79% for 3 correct key images, 52% for 4 correct key images, and 24% for 5 correct key images, and the camera recording attack had a very low FAR of 91-95%. In contrast, the screenshot attack of the existing method never succeeded in correctly classifying the attacker.

For legitimate users of EYEDi, the classification error rate was an FRR of 10% for 3 correct key images, 16% for 4 correct key images, and 43% for 5 correct key images. Especially when the number of correct key images was 5 and the authentication time was within 60 s, the FRR was large at 43%. The classification error rates of the OSA, camera recording attack, and screenshot attack were 0-9%, 2-28%, and 3-16%, respectively, showing lower classification error rates than existing methods for all attack methods and classification conditions. For the OSA, EYEDi showed a small classification error rate of FAR, in particular, 0% for 5 correct key images. Similarly, existing methods showed a classification error rate larger than 90% for camera attacks, but EYEDi showed a small error rate with a 28% FAR for 3 correct key images, 9% for 4 correct images, and 2% for 5 correct images. In particular, while the screenshot attack completely broke through the existing methods, EYEDi showed a lower error rate with a FAR of 16% for 3 correct key images, 9% for 4 correct key images, and 3% for 5 correct key images. Two weeks after registration, the EYEDi's legitimate users had a lower error rate with an FRR of 20-36%. The legitimate users after two weeks had an increased authentication time compared to the short-term memory group, which may have resulted in an increased FRR. As described above, we showed that EYEDi could classify legitimate users and attackers with better accuracy than UYI (existing method) by limiting the number of correct key images and the authentication time.

From the viewpoint of usability, the rejection rate of legitimate users in EYEDi is 10–43% (depending on the configuration of how many correct key images are required in the authentication phase), which indicates that usability is reduced. However, from the security viewpoint, EYEDi shows high protection performance even in the threat model, where the situation in which attackers almost always break UYI. This result shows that EYEDi improves security performance.

4) EQUAL ERROR RATE

The equal error rate is calculated by equaling FRR and FAR with adjusting the parameters, such as the number of correct key images and authentication time. The bars shown in Fig. 5 are the equal error rate, and the value shown next to the bar is (equal error rate, number of correct key images, authentication time). The equal error rate of the existing methods was 11% for OSA and 22% for camera recording attacks and screenshot attacks. On the other hand, the equal error rate of EYEDi was 7% for OSA, 11% for camera recording attacks, and 16% for screenshot attacks, and the equal error rate was small for all attacks. The number of correct key images to answer where FRR and FAR become equal was five for the existing method and three for EYEDi. In other words, it was not necessary to find all the key images when authenticating by EYEDi. The authentication time of the existing methods is 15 s for OSA, 10 s for camera recording attacks, and 10 s for screenshot attacks, while the time of EYEDi is 115 s for OSA, 90 s for camera recording attacks, and 60 s for screenshot attacks. In other words, the existing methods require a short authentication time, while EYEDi requires a longer authentication time.

V. DISCUSSION

A possible future direction for this work is on applications in personal authentication systems that use both images and voices with appropriate filter strength estimates. In the future, we believe that it will be possible to realize a system that does not require prior training or authentication by using deep learning and other methods by accumulating images and authentication data. The inverse estimation of appropriate distortions from key image features allows us to present a sufficiently distorted image even during initial authentication. The realization of EYEDi's system of inverse estimation would avoid the risk of a recording attack on authentication in the early stages of EYEDi.

For the authentication time in Fig. 5, we focused on the three types of attacks, OSA, camera, and screenshot on EYEDi. We hypothesized that screenshot was the most advantageous for the attacker and OSA was the most disadvantageous. However, the authentication time was the longest for screenshots and the shortest for OSA. This result was probably because OSA was too difficult for the attacker, and they quickly gave up, while screenshots took more time as the attackers thought they could answer correctly. The authentication time of long-term memory users increased slightly compared to two weeks ago. This may be because the long-term memory users needed more time to recall the key image. Therefore, the authentication time will decrease again after the user recalls the key image.

Table 2 shows a comparison between existing authentication systems and EYEDi. The feature of EYEDi is resistant to recording attacks, which are difficult for other authentication systems to defend against. However, the table also shows that EYEDi has the disadvantage that the authentication time needs to be improved. Although EYEDi has the disadvantage of a longer authentication time, it is resistant to various attacks and can be used for personal authentication in companies that need to be protected even with a long authentication time. Focusing on the graphical passwords in Table 2, we can see that PassPoints is vulnerable to OSAs and recording attacks. Déjà Vu has improved OSAs, which is the problem with PassPoints. UYI solved the problem of long-term memory, which is an issue of Déjà Vu, and EYEDi

TABLE 2. The characteristics shown in the table are G for good, F for fair, and B for bad.

	Password	Biometrics	PassPoints [6]	Déjà Vu [18]	UYI [1]	EYEDi
Ease of short memory	G	G	G	G	G	G
Ease of long memory	B	G	G	F	G	G
Ease of quick login	F	G	G	G	G	F
Resist of OSA attack	B	G	B	G	G	G
Resist of Camera recording attack	B	B	B	B	B	G

solved the problem of recording attacks, which had not been solved to date. Conceivably, we believe it is also possible to create passcodes by extracting features in the images, such as GhostNet [38].

VI. CONCLUSION

In this study, we proposed a graphical authentication scheme that generates various distorted images, Estimating Your Encodable Distorted images (EYEDi). The distorted image of EYEDi is generated by applying several image processing filters to the original image. Through 20-subject experiments and 300 screenshot attacks, we confirmed that EYEDi estimated the appropriate filter strength based on the authentication records and that the attacker had difficulty estimating the key image. Through these experiments, EYEDi prevented OSA, camera recording attacks, and screenshot attacks better than the existing method. Thus, EYEDi makes it possible to authenticate individuals with their own memories rather than easily forgotten keywords.

Necessary improvements for EYEDi include estimation, a stronger threat model, and difficulties in continuous authentication for the attacker. First, let us consider how to improve the estimation. In this paper, the data used in EYEDi are filter strength and discrimination success or failure. However, the time taken to discriminate will be an important factor in estimating discrimination ability; therefore, it would be possible to extend the weighting of the discrimination success or failure labels to the time spent on each image discovery. Next, we consider the powerful threat model configuration. Nowadays, threat models are familiar with memory based on actual measurements, but in practice, we should also expect multiple recordings by cameras. As shown in this study, normal authentication systems are typically breached by recording hand-held images and log-in screens at the time of entry. Since EYEDi displays different impressions of images each time, we can show that it is significantly more difficult to break through authentication when assuming a single screenshot attack than existing methods. However, we expect that attackers can learn and gradually become able to identify key images by repeating multiple recordings.

We also have to assume an attack where the key image is mechanically inferred through multiple recordings. EYEDi's distorted images are difficult to restore to their original images due to the multiple filter processes. However, with the recent remarkable development of image processing technology, it is expected that this danger can be resolved by collecting a large amount of data. The difficulty of continuous authentication was confirmed in the screenshot attack experiment. During the EYEDi screenshot attack attempts, we found many cases in which authentication was not possible after a successful attempt. In many previous studies, once an attacker had succeeded in authentication, the attacker would remember the key combination of the successful authentication attempt so that their subsequent attempt would also succeed. In contrast, our EYEDi system is difficult to use for continuous authentication, as the impression of the image presented on the next authentication screen changes significantly even if the authentication is successful once.

REFERENCES

[1] E. Hayashi, R. Dhamija, N. Christin, and A. Perrig, "Use your illusion: Secure authentication usable anywhere," in *Proc. 4th Symp. Usable Privacy Secur. (SOUPS)*, 2008, pp. 35–45.

[2] J. Nicholson, L. Coventry, and P. Briggs, "Faces and pictures: Understanding age differences in two types of graphical authentications," *Int. J. Hum.-Comput. Stud.*, vol. 71, no. 10, pp. 966–995, 2013.

[3] X. Suo, Y. Zhu, and G. S. Owen, "Graphical passwords: A survey," in *Proc. 21st Annu. Comput. Secur. Appl. Conf. (ACSAC)*, 2005, p. 10.

[4] A. M. Eljetlawi and N. Ithnin, "Graphical password: Comprehensive study of the usability features of the recognition base graphical password methods," in *Proc. 3rd Int. Conf. Converg. Hybrid Inf. Technol.*, Nov. 2008, pp. 1137–1143.

[5] M. Zviran and W. J. Haga, "Password security: An empirical study," *J. Manage. Inf. Syst.*, vol. 15, no. 4, pp. 161–185, Mar. 1999.

[6] S. Wiedenbeck, J. Waters, J.-C. Birget, A. Brodskiy, and N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system," *Int. J. Hum. Comput. Stud.*, vol. 63, nos. 1–2, pp. 102–127, Jul. 2005.

[7] J. M. Stanton, K. R. Stam, P. Mastrangelo, and J. Jolton, "Analysis of end user security behaviors," *Comput. Secur.*, vol. 24, no. 2, pp. 124–133, Mar. 2005.

[8] B. Menkus, "Understanding the use of passwords," *Comput. Secur.*, vol. 7, no. 2, pp. 132–136, Apr. 1988.

[9] E. Stobert and R. Biddle, "Memory retrieval and graphical passwords," in *Proc. 9th Symp. Usable Privacy Secur. (SOUPS)*, 2013, pp. 1–14.

[10] M. A. Sasse, S. Brostoff, and D. Weirich, "Transforming the 'weakest link'—A human/computer interaction approach to usable and effective security," *BT Technol. J.*, vol. 19, no. 3, pp. 122–131, 2001.

[11] K. Bicakci, N. B. Atalay, M. Yuceel, H. Gurbaslar, and B. Erdeniz, "Towards usable solutions to graphical password hotspot problem," in *Proc. 33rd Annu. IEEE Int. Comput. Softw. Appl. Conf.*, Jul. 2009, pp. 318–323.

[12] D. Davis, F. Monrose, and M. K. Reiter, "On user choice in graphical password schemes," in *Proc. 13th Conf. USENIX Secur. Symp.*, vol. 13, 2004, p. 11.

[13] A. Paivio, *Imagery and Verbal Processes*. Hove, U.K.: Psychology Press, 2013.

[14] D. L. Nelson, V. S. Reed, and J. R. Walling, "Pictorial superiority effect," *J. Experim. Psychol., Hum. Learn. Memory*, vol. 2, no. 5, p. 523, 1976.

[15] R. S. Nickerson, "Short-term memory for complex meaningful visual configurations: A demonstration of capacity," *Can. J. Psychol./Revue Canadienne de Psycholog.*, vol. 19, no. 2, p. 155, 1965.

[16] R. N. Shepard, "Recognition memory for words, sentences, and pictures," *J. Verbal Learn. Verbal Behav.*, vol. 6, no. 1, pp. 156–163, 1967.

[17] C. Katsini, N. Avouris, and C. Fidas, "CogniPGA: Longitudinal evaluation of picture gesture authentication with cognition-based intervention," *I-Com*, vol. 18, no. 3, pp. 237–257, Nov. 2019.
[18] R. Dhamija and A. Perrig, "Déjà Vu: A user study: Using images for authentication," in *Proc. USENIX Secur. Symp.*, vol. 9, 2000, p. 4.
[19] T. Takada, T. Onuki, and H. Koike, "A user evaluation study about security and usability of Awase-E," *IPSJ J.*, vol. 47, no. 8, pp. 2602–2612, 2006.
[20] L. N. Potter, J. D. Still, L. N. Tiller, and A. A. Cain, "Graphical authentication schemes: Balancing amount of image distortion," in *Advances in Human Factors in Cybersecurity* (Advances in Intelligent Systems and Computing), vol. 782, 2019.
[21] T. Takada and M. Yoshida, *Pict-Place Authentication: Recognition-Based Graphical Password Using Image Layout for Better Balance of Security and Operation Time*. New York, NY, USA: Association for Computing Machinery, 2021.
[22] W. A. J. van Eekelen, J. van den Elst, and V.-J. Khan, "Picassopass: A password scheme using a dynamically layered combination of graphical elements," in *Proc. CHI Extended Abstr. Hum. Factors Comput. Syst. (CHI EA)*, 2013, pp. 1857–1862.
[23] E. von Zezschwitz, S. Ebbinghaus, H. Hussmann, and A. D. Luca, "You Can't watch this!: Privacy-respectful photo browsing on smartphones," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, May 2016, pp. 4320–4324.
[24] S. Ghouzali, M. Lafkih, W. Abdul, M. Mikram, M. E. Haziti, and D. Aboutajdine, "Trace attack against biometric mobile applications," *Mobile Inf. Syst.*, vol. 2016, pp. 1–15, Mar. 2016.
[25] P. J. Phillips, A. Martin, C. L. Wilson, and M. Przybocki, "An introduction evaluating biometric systems," *Computer*, vol. 33, no. 2, pp. 56–63, Feb. 2000.
[26] M. Keith, B. Shao, and P. J. Steinbart, "The usability of passphrases for authentication: An empirical field study," *Int. J. Hum.-Comput. Stud.*, vol. 65, no. 1, pp. 17–28, Jan. 2007.
[27] D. Adams, L. Morales, and S. Kurniawan, "A qualitative study to support a blind photography mobile application," in *Proc. 6th Int. Conf. Pervasive Technol. Rel. Assistive Environ. (PETRA)*, 2013, pp. 1–8.
[28] R. Biddle, S. Chiasson, and P. C. Van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Comput. Surv.*, vol. 44, no. 4, pp. 1–41, Aug. 2012.
[29] S. Chiasson, A. Forget, R. Biddle, and P. C. V. Oorschot, "User interface design affects security: Patterns in click-based graphical passwords," *Int. J. Inf. Secur.*, vol. 8, no. 6, p. 387, 2009.
[30] W. Jansen, S. I. Gavrila, V. Korolev, R. P. Ayers, and R. Swanstrom, *Picture Password: A Visual Login Technique for Mobile Devices*. Baltimore, MD, USA: UMBC Student Collection, 2003.
[31] W. Moncur and G. Leplâtre, "Pictures at the ATM: Exploring the usability of multiple graphical passwords," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, Apr. 2007, pp. 887–894.
[32] J. Thorpe, M. Al-Badawi, B. MacRae, and A. Salehi-Abari, "The presentation effect on graphical passwords," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, Apr. 2014, pp. 2947–2950.
[33] C. Katsini, C. Fidas, M. Belk, G. Samaras, and N. Avouris, "A human-cognitive perspective of users' password choices in recognition-based graphical authentication," *Int. J. Hum. Comput. Interact.*, vol. 35, no. 19, pp. 1800–1812, Nov. 2019.
[34] A. Constantinides, M. Belk, C. Fidas, and A. Pitsillides, "An eye gaze-driven metric for estimating the strength of graphical passwords based on image hotspots," in *Proc. 25th Int. Conf. Intell. User Interface*, Mar. 2020, pp. 33–37.
[35] G. Hadjidemetriou, M. Belk, C. Fidas, and A. Pitsillides, "Picture passwords in mixed reality: Implementation and evaluation," in *Proc. Extended Abstr. CHI Conf. Hum. Factors Comput. Syst.*, May 2019, pp. 1–6.
[36] M. Khamis, T. Seitz, L. Mertl, A. Nguyen, M. Schneller, and Z. Li, "Passquerade: Improving error correction of text passwords on mobile devices by using graphic filters for password masking," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, May 2019, pp. 1–8.
[37] P. F. Alcantarilla and T. Solutions, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1281–1298, Oct. 2011.
[38] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1580–1589.

TAKAYUKI KAWAMURA was born in Osaka, Japan, in 1996. He received the B.Eng. degree from the University of Tsukuba, in 2020. He is currently pursuing the master's degree with the Graduate School of Science and Technology. His research interests include password security and computer science.

TADASHI EBIHARA (Member, IEEE) was born in Tokyo, Japan, in 1986. He received the Ph.D. degree from the University of Tsukuba, Tsukuba, Japan, in 2010. From September 2013 to December 2013, he was a Visiting Professor with the Delft University of Technology, The Netherlands. He is currently an Associate Professor with the Faculty of Engineering, Information and Systems, University of Tsukuba. His research interests include mobile communications and their applications to underwater acoustic communication systems. He received the Research Fellowship for Young Scientists (DC1) from the Japan Society for the Promotion of Science (JSPS), for the years 2009 and 2010. He received the 2017 IEEE Oceanic Engineering Society Japan Chapter Young Researcher Award.

NAOTO WAKATSUKI received the B.Eng., M.Eng., and D.Eng. degrees from the University of Tsukuba, in 1993, 1995, and 2004, respectively. He was with Okayama University, from 1995 to 2001, and Akita Prefectural University, from 2001 to 2006. He is currently an Associate Professor with the University of Tsukuba. His research interests include acoustic instrumentation, simulation-based visualization, vibration sensors and actuators, acoustical engineering, musical acoustics, and inverse problems. His affiliated academic societies include the Acoustical Society of Japan, Acoustical Society of America, the Society of Agricultural Structures, and Japan Society for Simulation Technology.

KEIICHI ZEMPO (Member, IEEE) received the B.S. degree in engineering, the M.B.A. degree in business administration and public policy, and the Ph.D. degree in engineering from the Department of Intelligent Interaction Technologies, College of Natural Science, University of Tsukuba, in 2008, 2010, and 2013, respectively. He worked with the Center for Service Engineering, National Institute of Advanced Industrial Science and Technology (AIST), from 2013 to 2014. He is currently an Assistant Professor with the University of Tsukuba. His research interests include human augmentation, sense substitution, service engineering, telepresence, and xR.