Open-Set Learning-Based Hologram Verification System Using Generative Adversarial Networks

BETUL AY
Department of Computer Engineering, Firat University, 23119 Elazig, Turkey
e-mail: betulay@firat.edu.tr
This work was supported by the Computing Resources at the Big Data and Artificial Intelligence Laboratory, Firat University.

ABSTRACT In this study, we address the hologram authenticity challenge by introducing a novel deep-learning based end-to-end hologram verification system. The system ultimately makes the decision whether the hologram image captured from a mobile application is fake or not by employing a robust classifier. We built the system by training three major deep networks; generative networks, convolutional networks and region-based convolutional networks. One major challenge in this study was the lack of negative class samples or so-called fake holograms. To the best of our knowledge there are no publicly available fake hologram datasets and it is not clear how the attackers imitate the real holograms. Therefore, the negative class in the practical hologram classification task is actually “unknown” class, as it is unknown how to imitate holograms by attackers. We hereby consider the problem of hologram classification as in a similar logic to open-set recognition. To make hologram classifier more sensitive to forgery, we generate synthetic images using generative adversarial networks (GANs) to represent negative class. We conduct extensive and comparative experiments on the closed-set and open-set using the-state-of-the-art backbone convolutional neural networks (CNNs). The proposed system gives an impressive accuracy 97.5% and 79% for closed-set and open-set samples, respectively. The reported results show the strong generalization performance of the system for unknown samples.

INDEX TERMS Deep learning, generative adversarial networks, convolutional neural networks, image classification, computer vision.

I. INTRODUCTION
Holograms are often used against fraud and imitation threats in the creation of reliable documents such as passports, brands, ID cards, banknotes and books. The most common approach used to determine the authenticity of holograms with various security features is “feel, look, tilt” method [1], which requires a person to manually check and visually control the hologram tag. According to this method, the user tries to capture a range of security features with image and color changes by holding the hologram to the light or tilting it. For example, in order to detect whether a book is not a counterfeit, it is necessary to follow holographic effects, emblem, changing numbers, writings, and glossy stripes by holding the printed hologram against the light or moving it at different angles. Automating this process, which requires significant workload and cost, is an extremely important security requirement.

A limited number of studies have been performed for the automatic detection of fake holograms through camera images. Moreover, most of these studies have used the vanilla machine learning and image processing techniques resulting in high temporal and cognitive effort with feature extraction and template or pattern matching [2]–[5]. [6] proposed an approach based on analysis of a registered stack of document images to detect holograms on security documents such as foreign passports and Euro banknotes. They use Canny edge detector for extracting the hologram area. At verification step, normalized cross correlation (NCC) is used for estimated hologram positions throughout the registered stack. [7] has presented a method based on correlation analysis of hologram patterns which uses The Wiener filter algorithm. In [8], fake currency detection has been conducted by using image processing steps including image acquisition, grey scale conversion, edge detection, image segmentation, characteristic
extraction and calculation of intensity. Another study [9] uses automated image-based technique by using Support Vector Machine (SVM) for counterfeit hologram detection on Bangladeshi banknotes.

Developments in deep learning has increased rapidly with the emergence of available large datasets, powerful computing resources and new approaches. Convolutional Neural Networks or CNNs, one of the deep learning methods, have outperformed shallow machine learning algorithms for image recognition and classification tasks [10]. The study in [11] has used CNNs to distinguish fake holograms from real holograms by learning a vector representation that captures the properties of hologram images. Although they expressed in their papers that they can reliably detect false holograms, they have not used fake holograms in the training stage, and the descriptive quality of the learned feature cannot be assessed in this way [12].

Since descriptiveness in the feature is a required quality in fraud detection, there needs to be an abnormal or unknown class (in this paper called as fake class) except for the target class for training phase. Anomaly detection methods focuses on the detection of previously unseen samples. [13] has been used GANs to generate synthetic defects similar to the patches of the test set as possible. Similarly, the hologram verification task is an anomaly detection problem where fake holograms are usually cannot distinguished by the naked eye. Furthermore, unavailability of fake holograms printed on banknotes, books or id cards, no fake hologram datasets publicly available, is an open problem that can be considered as a subset of open-set learning. [14] has performed a linear SVM to classify fake and real holograms. They have used photocopy banknotes printed with a high-resolution printer for fake class. [15] has developed a new approach named as DeepMoney using Semi Supervised Generative Adversarial Networks (SSGANs). They applied this method to generate fake Pakistani banknotes and discriminate from real ones. They achieved a promising classification result with 80% accuracy. Deep learning models are trained to classify and learn for known classes (positive data), however, it is not possible to gather data for all categories in practical scenarios. [16] have proposed a fake negative class generator for unknown class and improved the classification accuracy and capability of the known classes. Considering that it is not possible to train a classifier network for fake data generated from all possible fabrication materials, [17] proposed an open set fingerprint spoof detection based on Weibull-calibrated SVM.

II. PROBLEM STATEMENT AND HIGHLIGHTS

All over the world, banknotes, books, security documents, private and public identity cards, passports, and numerous branded products are exposed to very serious targets such as piracy and counterfeiting. The first noticeable application with an economical cost to secure these various documents and products is hologram labels. However, field experts emphasize that although the hologram is seen as a security element, the fake and reality are sometimes not easily distinguished even with the human naked eye and that fake holograms can be produced easily due to their low cost. As a natural consequence of this vulnerability, unauthorized reproduction and imitation of holograms has created a need for automatic hologram verification techniques.

Automatic hologram verification is an important requirement and potential research area in an innovation-driven global economy and national security that exposed to counterfeit documents and fake branded products. Hologram verification task is a one of the binary classification problems where only positive class is well defined, but negative class can never be measured or predicted. The biggest challenge of this task is that the samples in the positive and negative classes are so similar that they are sometimes indistinguishable even by naked eyes. Although deep learning-based recent methods has emerged successful outcomes in image classification problems, this problem cannot be solved by simply training a classifier network. The biggest challenge faced by the data scientist is that the negative class is not obvious, but must be very close to the real class.

Ultimately goal of this paper is to improve the classifier performance of known class across unseen fake hologram attacks that so close to real ones. In this paper we follow a strategy of generated-based open set learning introduced by [18]. The goal of the strategy is to generate synthetic images that are nearly indistinguishable from a given real image, and train a deep neural network using the output of the generative models. We adopted the idea of generated-based open-set recognition for our hologram verification problem.

This paper presents the following main contributions:

1) Training of a binary hologram classifier by posing the hologram verification problem as an open set learning problem
2) Conducting a detailed analysis of the baseline methods using the most promising evaluation metrics
3) Generating of fake holograms using various GANs due to the unavailability of fake holograms in public domain.
4) Developing of more robust and reliable hologram verification system using the state-of-the-art CNN methods that have proven successful in the literature.
5) Providing of the end user with a fully automated system from detection of the hologram region on the related image to the verification of whether the hologram is fake or not.

The paper is briefly organized as follows: Section 3 presents an overview of the proposed system, a comprehensive analysis of the backbone methods of the system, detailed network architectures and sample use-case scenario. Section 4 evaluates the image diversity and quality of generated fake holograms with visual inspection and shows performance results of the classifier models. Section 5 discusses potentials and limitations of the proposed system. Finally, Section 6 concludes the research.
III. PROPOSED SYSTEM

In this study, a hybrid deep learning-based hologram verification system is designed to prevent hologram counterfeiting. The proposed system is detailed in Fig. 1. The system is built with three main modules by training three major deep networks; generative networks (fake hologram generation-yellow region), convolutional networks (classification model-blue region) and region-based convolutional networks (hologram detection model-purple region). The model deployment and the use case scenario (yellow region) show end-user guide of the system.

The system is mainly divided into three major components:

1) First component is the object detection algorithm, which directly returns the coordinates of the hologram regions.
2) Second component is the generator network, which takes the detected hologram images and generates fake hologram samples.
3) Last component is the binary classification network, which discriminates fake and real holograms.

The use-case scenario of our hologram verification system is described in steps in Section C.

A. HOLOGRAM GENERATION MODULE

In this section we provide short summaries of backbone methods used in the system for fake hologram generation module.

1) GENERATIVE ADVERSARIAL NETWORK (GAN)

Generative Adversarial Networks (GANs), first introduced in 2014 by [19], are a special type of deep neural networks. Using these networks, many popular applications including image generation such as fake faces [20], audio or video generation, time series generation for medical data and stock market forecasting, visual similarity recommendation [21] and statistical inference designing clothes and shoes by analyzing photos of specific classes, text-to-image synthesis and image super-resolution has been created [22]. The GAN models consist of two neural networks: Generator $G$ and discriminator $D$. While the generator network performs the generation of new data samples, the discriminator network acts as a classifier that distinguishes whether the data given as input is real or fake. In other words, it determines whether the generated data belongs to the real training dataset. Generator does not have any randomness source, so it is a non-parametric model. Therefore, it uses random noise $z$ as randomness source. Random noise $z$ is sampled from the Gaussian distribution or uniform distribution.

2) SUPER-RESOLUTION GAN

Generative Adversarial Network for Super Resolution (SRGAN) introduced in [23] aims to generate photo-realistic images close to the natural images. SRGAN is capable of generating high-resolution images with deep CNNs. Generator network of SRGAN takes a low-resolution image as
input image and generates a super-resolution image as output image, after a series of 2D convolution, batch normalization and upsampling layers. Discriminator network of SRGAN takes a super-resolution image and try to predict the probability of the image being real or fake, after a series of 2D convolution and batch normalization layers. It minimizes the perceptual loss function in the training. loss function $l^{SR}_{VGG}$ is a weighted sum of VGG loss (as content loss) $l^{SR}_{VGG}$ and adversarial loss $l^{SR}_{GEN}$ function [23]. VGG loss introduced by [24] is based on the ReLU activation layers of convolution and batch normalization layers. It minimizes the error. BEGAN uses the Wasserstein distance [26] to compute adversarial loss $l^{SR}_{VGG}$ is calculated as the Euclidean distance between the feature representations of the generated image and real image as Eq. (1):

$$l^{SR}_{VGG/i,j} = \frac{1}{W_{ij}H_{ij}} \sum_{x=1}^{W_{ij}} \sum_{y=1}^{H_{ij}} (\Phi_{ij}(I_{HR}))_{x,y} - \Phi_{ij}(G_{θ_G}(I_{LR}))_{x,y}^2 \tag{1}$$

Here, $G_{θ_G}(I_{LR})$ describes a high-resolution image generated by the Generator $G$ and $I_{HR}$ indicates a reference image sampled from the real dataset. $Φ_{ij}$ represents the feature map obtained by VGG19 network. Adversarial loss $l^{SR}_{GEN}$ is calculated on the probabilities of the Discriminator $D$ over all training samples, as follows:

$$l^{SR}_{GEN} = \sum_{n=1}^{N} -\log D_{θ_D}(G_{θ_G}(I_{LR})) \tag{2}$$

Here, $D_{θ_D}(G_{θ_G}(I_{LR}))$ describes the probability that the generated image $G_{θ_G}(I_{LR})$ is a real image $I_{HR}$.

3) BOUNDARY EQUILIBRIUM GAN

Other GAN framework utilized in this study is BEGAN [25] which employs auto-encoder instead of discriminator used as classifier in vanilla GANs. While vanilla GANs uses discriminator as a classifier that gives probability score of fake and real images, discriminator of BEGAN uses auto-encoders that extract features from input image and generated image (reconstructed image) by computing the reconstruction error. BEGAN uses the Wasserstein distance [26] to compute a lower bound between the reconstruction loss of real and generated samples. Rather than matching data distributions directly, it matches the distribution of reconstruction losses, in other words, loss distributions of auto-encoders. More formally, let $μ_1$ and $μ_2$ be the loss distribution of auto-encoders and let the set all of couplings of $μ_1$ and $μ_2$ be denoted by $γ ∈ Γ(μ_1, μ_2)$. The Wasserstein distance is computed in Eq. (3):

$$W_1(μ_1, μ_2) = \inf_{γ ∈ Γ(μ_1, μ_2)} \mathbb{E}(x_1, x_2)_{γ} \[|x_1 - x_2|\] \tag{3}$$

Lower bound of $W_1(μ_1, μ_2)$ can be derived by applying Jensen’s inequality to Eq. (3), as following Eq. (4):

$$\inf \mathbb{E}[|x_1 - x_2|] ≥ |\mathbb{E}[x_1 - x_2]| = |m_1 - m_2| \tag{4}$$

where $m_1, m_2 ∈ \mathbb{R}^+$ there are two possible solutions to maximize the distance between real and generated samples: $m_1 → ∞, m_2 → 0$ and $m_1 → 0, m_2 → ∞$. BEGAN selects the second solution considering as minimizing $m_1$ leads to reconstructing real samples. The objective function of BEGAN can be formulated as [25]:

$$\begin{align*}
L_D(x, z) &= L(x) - k_t * L(G(z_D)) \\
L_G(z) &= L(G(z_G)) \\
k_{t+1} &= k_t + λ_k (γ_L(x) - L(G(z_G)))
\end{align*} \tag{5}$$

In Eq. (5), $L(x)$ and $L(G(z))$ are the auto-encoder losses of real data and generated data, where $x$ are real samples, $z ∈ [-1, 1]^N_z$ are uniform random samples, $G : \mathbb{R}^{N_z} → \mathbb{R}^{N_x}$ is the generator function, $z_D$ and $z_G$ are samples from $z$, and $λ_k$ is the learning rate of $k$. The variable $k_t ∈ [0, 1]$ is implemented to control the emphasis of generator losses when discriminator is being trained. It uses proportional control theory to keep the equilibrium $E[L(G(z))] = γ E[L(x)]$ by using $k_t$. At the equilibrium, a new hyper-parameter $γ ∈ [0, 1]$ introduced in the paper [25] maintains the balance between the auto-encoder loss of real and generated samples. $γ$ is also called as the diversity ratio because of an indicator of image diversity.

B. HOLOMGRAM RECOGNITION MODULE

This section provides short summaries of CNN-based backbone methods used for our hologram recognition module.

1) CONVOLUTIONAL NEURAL NETWORK (CNN)
Convolutional Neural Networks (CNNs) have performed superhuman performance in many areas of image recognition, object recognition, automatic video classification, and computer vision. A typical CNN architecture consists of three important layers: The convolutional layer, the pooling layer, and the fully connected layer. The feature maps are passed through the pooling layer, which reduce the dimension of the input and also reduce the number of parameters in the network. The output of the last pooling layer is fed into a fully connected layer that predict a classification label. Before the fully connected layer, a CNN architecture can be built with multiple convolutional layers followed by a pooling layer or can be built with only convolutional layers. After invention of the first CNN named as LeNet-5, many CNN architectures have been developed with different number and types of layers, which stacked on top of each other such as AlexNet, VGG, ResNet, Inception and more detailed in [27].

2) REGION WITH CNN

CNNs are not only used for image classification but also for object detection. Regions with CNN (R-CNN) is one of the first breakthroughs, which uses CNNs for object detection [28]. A typical object detection system consists of four modules including region proposal network (RPN), a deep CNN, bounding box regression and classifier module. RPN module generates a series of region proposals per image.
TABLE 1. Proposed generator network architecture inspired by SRGAN.

| Block Name   | Layer (type)                       | Input Shape          | Output Shape         | Param # |
|--------------|------------------------------------|----------------------|----------------------|---------|
| Pre-residual | 2D convolution (Conv2D)            | (None, 64, 64, 3)    | (None, 64, 64, 64)   | 15616   |
|              | 2D convolution (Conv2D)            | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 36928   |
|              | Batch normalization                | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 256     |
| Residual (x16) | 2D convolution (Conv2D)           | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 36928   |
|              | Batch normalization                | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 0       |
|              | Addition layer (Add)              | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 0       |
|              | 2D convolution (Conv2D)            | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 0       |
| Post-residual | Batch normalization               | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 256     |
|              | Addition layer (Add)              | (None, 64, 64, 64)   | (None, 64, 64, 64)   | 0       |
| Upsampling-1  | 2D upsampling (UpSampling2D)       | (None, 64, 64, 64)   | (None, 128, 128, 64) | 0       |
| Upsampling-2  | 2D convolution (Conv2D)            | (None, 128, 128, 256)| (None, 128, 128, 256)| 147712  |
| Final Layer   | 2D convolution (Conv2D)            | (None, 256, 256, 256)| (None, 256, 256, 256)| 590080  |

CNN module extracts a fixed-size feature vector for each region. Bounding box regression module is used for reducing many localization errors or mis-localizations by taking the location of the proposed region as input and ground truth labels of the region as target. Classifier module trained for each class scores each extracted feature vector to classify the observed object. Many object detection methods based on CNN have been developed including Fast R-CNN, Faster R-CNN, R-FCN and more detailed in [29].

C. PROPOSED GENERATIVE MODEL ARCHITECTURES

This study follows a strategy of generated-based open set learning using two generative backbone architectures based on super resolution and boundary equilibrium approaches for the generation of counterfeit holograms. We generate fake holograms from 3-channels hologram images with the size of 64 × 64 × 3 pixels (input shape).

The first generative model, SRGAN, used in this study consists of the discriminator and generator architectures. The generator network architecture, presented in Table 1, is created with final layer and 4 different blocks contain pre-residual, residual, post-residual and upsampling layers. All residual blocks use ReLU activation function and final convolution layer uses tanh activation function. The discriminator network architecture, is built with 8 convolution layers and 2 dense (fully connected) layers. All convolution layers use LeakyReLU and last dense layer use sigmoid activation function to predict the probability of real and fake classes.

The second generative model, BEGAN, used in this study consists of encoder (discriminator) and decoder (generator) architectures including 2D convolution layers. It is applied LeakyReLU to the output of the convolution layers instead of Exponential Linear Units (ELU). Generator uses 3 convolution blocks which contains 2D convolution layers, 2 upsampling layers and last convolution layer. Discriminator uses first convolution layer, 3 convolution blocks contains 2D convolution layers and 2 subsampling layers. Each convolution block is followed by an upsampling layer in the generator and the blocks are followed by a subsampling layer in the discriminator.

D. MODEL DEPLOYMENT AND SERVING

The use-case scenario of our hologram verification system depicted in Fig. 1 (yellow color) works as follows:

1) End-user takes a photo of book cover containing the hologram region image using the mobile application.
2) The mobile application uploads the captured image to the server through Tensorflow serving.
3) The regional CNN based hologram detection model is invoked on the received book cover image.
4) Hologram region is cropped and resized with bounding box coordinates.
5) Pre-processed 3-channel hologram image is provided to the CNN based hologram verification model as the input.
6) The hologram verification model analyzes the input image and outputs the classification scores.
7) The system displays the verification result.

IV. EXPERIMENTS AND RESULTS

A. SETUP

We recorded original hologram videos from Turkish book covers on Samsung Galaxy Note 9 cell phone which has 6 GB Ram, 1.7GHz CPU, 12 MB Camera and Android Oreo OS. Afterwards, video frames are extracted and saved using OpenCV library in Python. The experiments have been conducted on a server which has 24-core Intel Xeon E5-2628L CPU, 256 GB RAM and Ubuntu Server 16.04 OS. We implemented the modified BEGAN and SRGAN models on Tensorflow framework using 8 NVidia GTX 1080-Ti GPUs.

For detection model of hologram regions on book covers, we utilize TensorFlow object detection API. The API offers many pre-trained models, which are trained on COCO dataset. Based on our former experiments, we retrained region-based fully convolutional network (R-FCN [30]) using the 101-layer Residual Net as the backbone and fine-tuned for our custom dataset. To create labelled training data, hologram
regions on the book images have been manually annotated by drawing the bounding boxes with LabelImg annotation tool.

Our class-balanced dataset consists of four subsets named as train, validation, unseen known test and unseen unknown test groups, comprising of 2800 (70%), 800 (20%), 400 (10%) and 400 images respectively. The dataset has two classes named as real and fake. Real class has been harvested using over 100 different books in different light, shadow and angles from real world environment. Fake class consists of three major data types. Two of which contains fake hologram images generated by two different GAN architectures for open-set, and the third data consists of printed hologram photographs for closed-set.

B. EXPERIMENTS ON HOLOGRAM GENERATION

For optimization of SRGAN [23], we have used Adam optimizer [31] with the learning rate of $2 \times 10^{-4}$ and $\beta_1 = 0.5$. The perceptual loss value has been written following each epoch and visualized on TensorBoard. As the value of the perceptual loss decreases, the generator network has started generating more realistic holograms. But to decide whether to continue or stop the training, we have saved the generated images after every 1000 epochs. Although the loss value is a strong indicator for deep neural networks, the decrease in loss does not always mean that the image quality improves for GANs. There is no significant correlation between image quality and loss value [26]. That is the reason why researchers rely upon visual inspection of the generated images [32]. The training has been stopped after 30K epochs or when the quality of the generated high-resolution images is good. The original (real hologram) data and the generated (fake hologram) sample outputs of the SRGAN model trained in this study are given in Fig. 2.

![Generated vs Original images](image)

**FIGURE 2.** Visual inspection of original holograms and generated fake holograms.

The visual inspection of generated and original images of same holograms shows that the two sample outputs are very similar. For optimization of BEGAN [25], Adam optimizer has been used with the learning rate of $10^{-4}$, batch size of 16, $\beta_1 = 0.5$, $\beta_2 = 0.99$ and $\gamma = 0.5$. Image quality of samples generated by Generator is an important evaluation metric for GANs. Blurry images are not desirable outputs. A quick way of qualitative measure is to visualize fake samples produced by generator. This solution helps us detect the network’s problems early and decide when to stop the training.

Fig. 3 shows the fake holograms generated from first step to last step of the training process for BEGAN. When we look at intermediary steps of the training process in Fig. 3, we observe blurry images. We also observe that generated samples start resembling hologram images as the number of steps increase. Eventually at 6000 steps, the model starts learning the image type very well and generating more realistic holograms. When performed visual inspection of the fake samples, it is clear that there is a high similarity between real and fake holograms in terms of pattern and color. In literature, many researchers have used the visual inspection of fake samples [33]–[35].

The biggest barrier to overcome for successfully using the GANs for real-world problems is the lack of diversity in generated images. Although GANs are capable of learning the non-linear mapping between two discrete domains, they often face significant challenges such as poor image quality and diversity resulted with low accuracy [36]. In this study, we try to select correct hyper-parameter values for both GANs paying attention to the trade-off between image quality and diversity. Although we change the diversity ratio $\gamma$ between the range of 0.3 and 0.7 as stated in the original paper [25], we encountered the diversity challenge of generated images with BEGAN. While BEGAN samples have good quality, when compared with SRGAN, it has generated uniform hologram images in terms of pattern and color. When we evaluate relative performances of SRGAN and BEGAN, we observed that SRGAN samples outperform BEGAN with high image quality and image diversity.

In order to quantify the effectiveness of GANs, we employ Learned Perceptual Image Patch Similarity (LPIPS) [37] and Frechet Inception Distance (FID) [38] quality metrics which were shown to correlate well to human visual inspection. LPIPS and FID indicate the distance between two distributions of original and generated images and measure the diversity of generated images (lower is better for both metrics). Both metrics are frequently used to assess the performance of GANs in terms of the quality and diversity of the samples generated for various application scenarios [39]–[41]. The generative models are used to generate 2000 fake hologram images, and the evaluation metrics are used for quantitative comparisons by using these generated images. Fig. 4 shows samples generated by SRGAN and BEGAN as well as the similarity results from LPIPS and FID metrics. It can be seen that the improved SRGAN achieves LPIPS of 0.087, 0.049, 0.052, 0.066 and 0.043 and FID of 126, 83.24, 95.43, 120.31, 75.59 on randomly selected five sample hologram images, better than BEGAN. The proposed SRGAN model has an
average LPIPS score of 0.07 and FID score of 112.54 on 2000 generated images. It can be observed from Fig. 2 and Fig. 4 that SRGAN presents visually appealing results, where most of the structural features such as holographic effect, emblem, glossy stripe and color in generated images are close to those in original images. On the other hand, BEGAN generates uniform hologram images in terms of the aforementioned structural features and produces relatively poor results in terms of the quality metrics. When we evaluate relative performances of SRGAN and BEGAN, we observed that SRGAN samples outperform BEGAN with high image quality and image diversity.

The overarching objective of the hologram generation module is to assess whether or not GANs offer a reliable framework for synthesizing realistic, high quality and diverse hologram images. The improved SRGAN (proposed generative network in this study) is used to generate the synthetic images for negative class of the training set, which guarantees the network to generate high quality hologram images with realistic hologram effects. In the next step, the synthetic images and the original hologram images are combined to train the state-of-the-art CNN classification models.

### C. CLASSIFICATION RESULTS

For competitive benchmarking we evaluate our dataset by retraining the state-of-the-art CNN models including VGGNet (VGG19 architecture), DenseNet (DenseNet121 architecture), ResNet (ResNet152 architecture) and MobileNet (MobileNetV2 architecture), which have been pre-trained on ImageNet dataset. We have leveraged the power of transfer learning by transferring the learned features of the backbone models to our hologram domain. The classification accuracy on the validation data has been obtained in the range between 94% and 99% (detailed in Table 2). Backbone-ResNet model has demonstrated superior classification performance on hologram verification. All models were designed with the goal of achieving the best accuracy over model parameters and optimizations. In order
TABLE 2. Hologram classification performance evaluated with golden metrics.

| Backbone Network | # of Layer | Class | Precision | Recall | AUC | Accuracy |
|------------------|------------|-------|-----------|--------|-----|----------|
| MobileNet        | 53         | Fake  | 0.975     | 0.948  | 0.988 | 0.96     |
|                  |            | Real  | 0.948     | 0.975  | 0.988 | 0.96     |
| VGGNet           | 19         | Fake  | 0.982     | 0.979  | 0.998 | 0.98     |
|                  |            | Real  | 0.978     | 0.981  | 0.998 | 0.98     |
| DenseNet         | 121        | Fake  | 0.953     | 0.933  | 0.982 | 0.94     |
|                  |            | Real  | 0.933     | 0.952  | 0.982 | 0.94     |
| ResNet           | 152        | Fake  | 0.982     | 0.991  | 1.0   | 0.99     |
|                  |            | Real  | 0.991     | 0.981  | 1.0   | 0.99     |

TABLE 3. Open-set and closed-set recognition performance on unseen test images.

| Metric (%)    | Open-set | Closed-set | Data Type | Result | Data Type |
|---------------|----------|------------|-----------|--------|-----------|
| Precision     | 90.85    | Printed    | 98.48     | Generated |
|               | 72.48    | Genuine    | 97.04     | Genuine |
| Recall        | 64.50    | Printed    | 97.00     | Generated |
|               | 93.50    | Genuine    | 98.50     | Genuine |
| F1-score      | 75.45    | Printed    | 97.73     | Generated |
|               | 81.66    | Genuine    | 97.76     | Genuine |
| Accuracy      | 79.00    | Printed    | 97.75     | Generated |

to prevent overfitting, some checkpoints have been defined in model training. The first is the early stopping control, which monitors the performance of the validation loss and stops the training when the validation loss increases or does not improve with respect to the previous loss, thus determining the number of epochs. The second is the model control that records the best validation results as a copy of the model at each epoch time. These controls have been automated with Keras library. The dropout technique [37] has been also used to overcome overfitting before dense layer.

The experiments also provide ROC (receiver operating characteristic) curves plotted in Fig. 5, to evaluate our classifiers in the hologram verification application. It can be noted from the smooth ROC curves that all models fit the data well. The best promising open-set performance obtained through the Backbone-ResNet experiments has been conducted on the unseen holograms (See Table 3). We have used photocopy holograms printed with a high-resolution printer for unknown fake class. New printed holograms (unknown material and unseen in training) have been performed to learn generalization capability and open-set recognition performance of the classifier model. In Fig. 6, results of classifier experiments have been presented as confusion matrices. The test classification report in Table 3 shows the summary of the precision, recall and F1 score for each set. Outcome of open-set recognition task shows that for the fake class the precision is higher than the recall. Which means that most of the samples classified as fakes are actually fakes. On the other hand, for the real class the recall is higher than the precision. Which means that most of the real samples are correctly identified. We achieved a remarkable test accuracy of 79% on the unseen and unknown images in training and validation experiments. It can be noted from Table 3 that the best classifier model has achieved an impressive accuracy of 97.75%.

V. DISCUSSION: POTENTIALS AND LIMITATIONS

The biggest barrier to overcome the hologram verification challenge is unavailability of fake hologram images in public domain, because this domain is dominated by organized crime groups benefitting significantly from counterfeiting and piracy. However, in this classification problem, it is important to know the reliability of a classifier’s prediction, which means the network must effectively learn the feature vectors of both classes. Considering all these potentials and limitations, we treated this problem as an open-set learning problem because of unpredictable of fake class. We have designed an applicable system for verification of hologram images captured with a digital camera.

There are few studies related to deep learning-based hologram verification task in the literature for a comparative performance analysis. A comparison between studies in the literature and our study in terms of data type, dataset size, method, accuracy and usage area is provided in Table 4. Our system has similar focus on our data generation task with the proposed system called DeepMoney [15] that generates fake banknotes to discriminate fake notes from genuine ones. DeepMoney achieved the classification accuracy of 80% by the proposed GANs framework for counterfeit money detection. Unlike this study, we have generated fake holograms using different GAN architectures. The biggest obstacle to successfully generating fake holograms using GANs was the lack of diversity and quality of generated images. We tried to select correct hyper-parameter values for both GANs paying attention to the trade-off between image quality and diversity. Although we change the diversity ratio $\gamma$ between the range of 0.3 and 0.7 as stated in the original paper [25], we encountered the diversity challenge of generated images with BEGAN. We observed that the samples from the trained SRGAN architecture outperform BEGAN for this task. Our study achieved the classification accuracy of 97.75%.

Another study [6] has similar focus with our hologram verification task from a mobile phone. For images captured with
a smartphone, the study presented a real-time hologram detection method using different map segmentation approaches. Similar to this study, we also presented a hologram verification system that can be used with mobile devices. However, our models include hologram region detection and classification models which have been developed using CNNs. Our classifier model built by transferring the learned features of Backbone-ResNet models achieved an overall test accuracy of 97.6% on unseen real and generated-fake samples. This is an expected performance result. Even though the samples generated with GANs are seen as indistinguishable to the naked eye and to the discriminator network, it is not possible
We emphasize that we focused on the methods of generating fake holograms that ensure the balance of quality and diversity of synthetic data, however the ultimately goal of this study was not to generate the best holograms that even the machine cannot distinguish from the real one. We constructed a negative class generated from fake holograms; it enables the classifier to better learn the feature vectors of real holograms.

We have comparatively evaluated our verification models using both closed-set and open-set images on the hologram dataset. For closed-set recognition performance, the CNN benchmark models have been trained and tested on the subsets of same dataset, which contains real holograms and GAN-based generated-fake holograms. For open-set recognition performance, the trained CNN models have been tested on the printed holograms unseen in the training that can be considered as a new material for fake class. The model has achieved a remarkable test accuracy of 79% on unseen real and printed-fake hologram images. The open-set experiments indicate that there is still plenty room for further improvements on unknown hologram attacks.

VI. CONCLUSION AND FUTURE WORK

This study presents reliable and deployable models that resist counterfeit hologram threats in an effective way with a novel hybrid roadmap. The developed system has been conducted on a series of experiments on generative networks, convolutional networks and open-set learning. We validated our findings of proposed generative network with LPIPS and FID scores and also provided classification results of CNN models by reporting classification accuracy, precision, recall, confusion matrices and ROC curves. The system is able to perform binary classification tasks with an accuracy 97.5% and 79%, for known holograms unseen in training and unknown holograms unseen in training, respectively. The classification results have depicted that the proposed system works effectively with a commendable and applicable performance; however, there is still scope for improvement with new GANs for preventing unknown imitation threats.

Remote biometric ID verification methods are being used as extra security layers to prevent fraud and to help with any account-related issues. Hologram verification methods have become an important challenge to be solved in order to detect whether the ID card presented to the mobile app is real or not. In future, we will focus on the further development and implementation of the proposed approach to detect holograms in security documents such as identity cards and passports for customer verification and authentication solutions. Next study will also track the visual changes of holograms over time using time series.

REFERENCES

[1] H. de Heij, “User experience functions (UXFs) and user interface functions (UIFs) of optical authenticity features,” in Proc. Conf. Opt. Document Secur. Page, vol. 1, 2016, p. 30.
[2] A. D. Hartl, W. A. Isop, C. Arth, and D. Schmalstieg, “Towards mobile recognition and verification of holograms using orthogonal sampling,” in Proc. IEEE Int. Symp. Mixed Augmented Reality Workshops, Sep. 2015, pp. 75–80.
[3] H.-J. Kwon and T.-H. Park, “An automatic inspection system for hologram with multiple patterns,” in Proc. SICE Annu. Conf., Sep. 2007, pp. 2663–2666.
[4] R. Shah, M. Champaneri, P. Sheth, and V. Gaikwad. (Oct. 30, 2018). Currency Counting Fake Note Detection. [Online]. Available: https://ssrn.com/abstract=3275068
[5] D. Alekhya, G. D. S. Prabha, and G. V. D. Rao, “Fake currency detection using image processing and other standard methods,” Int. J. Res. Comput. Commun. Technol., vol. 3, no. 1, pp. 128–131, 2014.
I. Jo, J. Kim, H. Kang, Y.-D. Kim, and S. Choi, “Open set recognition on security documents using a mobile phone,” in Proc. Int. Symp. Vis. Comput., 2014, pp. 335–346.

J. Janucki and J. Owsiak, “A Wiener filter based correlation method intended to evaluate effectiveness of holographic security devices,” Opt. Commun., vol. 218, nos. 4–6, pp. 221–228, Apr. 2003.

T. Agasti, G. Burand, P. Wade, and P. Chitra, “Fake currency detection using image processing,” in Proc. IOP Conf., Mater. Sci. Eng., 2017, vol. 263, no. 5, Art. no. 052047.

M. S. Uddin, P. P. Das, and M. S. A. Roney, “Image-based approach for the detection of counterfeit banknotes of Bangladesh,” in Proc. 5th Int. Conf. Informat., Electron. Vis. (ICIEV), May 2016, pp. 1067–1072.

A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

D. Soukop and R. Huber-Mörk, “Mobile hologram verification with deep learning,” IPSJ Trans. Comput. Vis. Appl., vol. 9, no. 1, pp. 1–6, Dec. 2017.

P. Perera and V. M. Patel, “Learning deep features for one-class classification,” IEEE Trans. Image Process., vol. 28, no. 11, pp. 5450–5463, Nov. 2019.

M. Haselmann and D. P. Gruber, “Pixel-wise defect detection by CNNs without manually labeled training data,” Appl. Artif. Intell., vol. 33, no. 6, pp. 548–566, May 2019.

A. Berenguel, O. R. Terrades, J. Lladós, and C. Canero, “Banknote counterfeit detection through background texture printing analysis,” in Proc. 12th IAPR Workshop Document Anal. Syst. (DAS), Apr. 2016, pp. 66–71.

T. Ali, S. Jan, A. Alkhodre, M. Nauman, M. Amin, and M. S. Siddiqui, “DeepMoney: Counterfeit money detection using generative adversarial networks,” PeerJ Comput. Sci., vol. 5, p. e216, Sep. 2019.

I. Jo, J. Kim, H. Kang, Y.-D. Kim, and S. Choi, “Open set recognition by regularising classifier with fake data generated by generative adversarial networks,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Apr. 2018, pp. 2686–2690.

A. Rattani, W. J. Scheirer, and A. Ross, “Open set fingerprint spoof detection across novel fabrication materials,” IEEE Trans. Inf. Forensics Security, vol. 10, no. 11, pp. 2447–2460, Nov. 2015.

L. Neal, M. Olson, X. Fern, W.-K. Wang, and F. Li, “Open set learning with counterfactual images,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 613–628.

I. Goodfellow and J. Pouget-Abadie, “Generative adversarial nets,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2672–2680.

X. Ning et al., “Multi-view frontal face image generation: A survey,”Concurrency Comput., Pract. Exper., p. e6147, 2020, doi: 10.1002/cpe.6147.

B. Ay, G. Aydin, Z. Koyun, and M. Demir, “A visual similarity recommendation system using generative adversarial networks,” in Proc. Int. Conf. Deep Learn. Mach. Learn. Emerg. Appl. (Deep-ML), Aug. 2019, pp. 44–48.

A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative adversarial networks: An overview,” IEEE Signal Process., vol. 35, no. 1, pp. 53–65, Jan. 2017.

C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4681–4690.

J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016.

D. Berthelot, T. Schumm, and L. Metz, “BEGAN: Boundary equilibrium generative adversarial networks,” 2017, arXiv:1703.10717.

M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in Proc. Int. Conf. Mach. Learn. PMLR, 2017, pp. 214–223.

A. Canziani, A. Paszke, and E. Culurciello, “An analysis of deep neural network models for practical applications,” 2016, arXiv:1605.07678.

R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.

W. Zhiquang and L. Jun, “A review of object detection based on convolutional neural network,” in Proc. 36th Chin. Control Conf. (CCC), Jul. 2017, pp. 11104–11109.

J. Dai, Y. Li, K. He, and J. Sun, “R-FCN: Object detection via region-based fully convolutional networks,” in Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 379–387.

D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

M. M. Joshi, R. P. Bajpai, G. Mitra, H. K. Sardana, H. N. Bhargav, and S. Batra, “Fake currency detector using visual and reflective spectral response,” U.S. Patent 7 684 607 B2, Mar. 23, 2010.

R. Valle, W. Cai, and A. Doshi, “TequilaGAN: How to easily identify GAN samples,” 2018, arXiv:1807.04919.

A. Radford, L. Metz, and S. Chintalia, “Unsupervised representation learning with deep convolutional generative adversarial networks,” 2015, arXiv:1511.06434.

M. Rosca, B. Lakshminarayanan, D. Warde-Farley, and S. Mohamed, “Variational approaches for auto-encoding generative adversarial networks,” 2017, arXiv:1706.04987.

X. Ning, S. Xu, W. Li, and S.Nie, “FEGAN: Flexible and efficient face editing with pre-trained generator,” IEEE Access, vol. 8, pp. 65340–65350, 2020.

M. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” J. Mach. Learn. Res., vol. 15, pp. 1929–1958, Jun. 2014.

S. Muhammad and T. Ahmed, “Counterfeit currency recognition using deep learning: A review,” Technium, vol. 3, pp. 14–26, Mar. 2021.

R. Valle, Hands-on Generative Adversarial Networks With Keras: Your Guide to Implementing Next-Generation Generative Adversarial Networks. Birmingham, U.K.: Packt Publishing, 2019.

BETUL AY received the bachelor’s degree in computer engineering from Istanbul University, and the M.Sc. and Ph.D. degrees from the Computer Engineering Department, Firat University. During her M.Sc. studies, she has studied with the Photogrammetric Computer Vision (PCV) Laboratory, The Ohio State University, Columbus, USA. She is currently an Assistant Professor with the Big data & Artificial Intelligence Laboratory, Department of Computer Engineering, Firat University, Turkey. She is also the Chief Data Scientist at Deepsis Technology, which provides AI-driven applications for cross-vertical solutions, including defense, security, retail, and other ready-to-use visual recognition solutions. Her research interests include big data, cloud computing, deep learning, NLP, and computer vision.