Data Hiding With Deep Learning: A Survey Unifying Digital Watermarking and Steganography

Zihan Wang, Olivia Byrnes, Hu Wang, Ruoxi Sun, Congbo Ma, Huaming Chen, Member, IEEE, Qi Wu, and Minhui Xue

Abstract—The advancement of secure communication and identity verification fields has significantly increased through the use of deep learning techniques for data hiding. By embedding information into a noise-tolerant signal, such as audio, video, or images, digital watermarking and steganography techniques can be used to protect sensitive intellectual property (IP) and enable confidential communication, ensuring that the information embedded is only accessible to authorized parties. This survey provides an overview of recent developments in deep learning techniques deployed for data hiding, categorized systematically according to model architectures and noise injection methods. In addition, potential future research directions that unite digital watermarking and steganography on software engineering to enhance security and mitigate risks are suggested and deliberated. This contribution furthers the creation of a more trustworthy digital world and advances responsible artificial intelligence (AI).

Index Terms—Artificial intelligence (AI), cybersecurity, software engineering.

I. INTRODUCTION

SERIOUS concerns are raised about the ability of artificial intelligence (AI) to make responsible decisions or behave responsibly, such as generating unfair outcomes, causing job displacement or insufficient protection of privacy and data security. In response, responsible AI aims to address these issues and create accountability for AI systems.

Data hiding is considered a promising method to achieve data security toward responsible AI. Typically, data hiding involves concealing information in a specific form within another type of media. It can take different forms, such as encoding confidential information into an existing text piece or embedding audio files into a digital image. As digital assets become more diverse and ubiquitous, the importance and scope of data hiding applications will only continue to grow [1]. In today’s digital age, as digital communication and multimedia data become increasingly prevalent, the data hiding process has become crucial. Ensuring responsible AI, such as machine learning as services, per requisite, necessitates secure communication across all media, and the accountability of responsible AI, such as digital intellectual property (IP), necessitates protection against theft and misuse. In its traditional form, the data hiding process can be categorized into three types: watermarking, steganography, and cryptography [2]. This survey focuses on the former two (watermarking and steganography), as both have demonstrated superior capability in safeguarding sensitive data for AI systems learning.

Digital watermarking utilizes data hiding techniques to embed an identification (ID) into digital media that communicates with the owners of the IP to prevent unauthorized copying or alteration. Thus, in case of any attempt to copy or modify the original media, the ID can be extracted to identify the owners.

Steganography and watermarking share similarities in that both involve embedding data into a piece of media. However, while watermarking aims to identify the creator of an artifact, steganography embeds secret messages in a way that avoids detection, interception, or decoding. Unlike cryptography, which is designed to secure data by taking advantage of complexity, steganography’s primary goal is to keep the cover media’s format readable and not distorted after data hiding. The public should still be able to see the original cover media without noticing the embedded messages. Steganography is applied in various industries, including medicine, defense, and multimedia fields, wherever confidentiality is crucial for secure communication [3].

Historically, data hiding has been accomplished using specialized algorithms that are classified based on the domain in which they operate, either spatial or frequency. However, traditional algorithms have limitations, as they are often restricted in their applications and require the creators to have an expert understanding of the embedding process.

The recent development of deep learning has had a significant impact on various industries, owing to the powerful representation capabilities of deep neural networks. In the data hiding field specifically, deep learning models offer adaptable and generalized frameworks that can be applied to a wide range of watermarking and steganography applications. Currently, most works in this area concentrate on image-based data hiding, and these are the works that will be compared in this survey. These machine learning models are able to learn advanced embedding patterns that are able to resist a much wider range of attacks with far more effectiveness than traditional watermarking or steganography algorithms [4]. Further research on [5] established the potential and criticality...
of developing generalized frameworks for data hiding that can work with a range of cover media types to robustly embed data and provide highly secure content authentication and communication services. The advantage of the deep learning approach is that networks can be retrained to become resistant to new types of attacks, or to emphasize particular goals, such as payload capacity or imperceptibility without creating specialized algorithms for each new application [5]. An additional advantage of deep data hiding techniques is that they can enhance the security of the embedded messages. The high nonlinearity of deep neural models makes it virtually impossible for an adversary to retrieve the embedded information [4]. Compared with traditional methods, deep learning-based methods are not only more secure and adaptable to different applications, but they also offer enhanced robustness to adversarial attacks and distortions. They are also able to achieve more imperceptible forms of data embedding.

The data hiding process, consisting of message embedding and extraction, maps intuitively onto the encoder–decoder network architecture, wherein the learning model is partitioned into two networks. In these models, an encoder network is trained to embed input messages to images. The images are then subjected to some forms of attack through distortion layers, and the decoder network must then extract the original message from the distorted image. These distortions can include blurring, cropping, compression, and so on. The objective of the network training is to minimize an objective function, which accounts for the differences between the cover image (i.e., the data carrier) and encoded image, as well as the differences between the embedded and extracted input message.

Since this is a relatively new area of research, current surveys on data hiding primarily concentrate on traditional algorithms. There are existing works examining deep learning-based techniques for steganography and cryptography [6], [7], but there is a lack of works examining deep watermarking techniques. There is an existing survey looking at deep learning-based watermarking and steganography [8]; however, a comprehensive survey regarding deep data hiding models unifying digital watermarking and steganography is still lacking. To the best of our knowledge, ours is the first survey to examine deep learning techniques for deep learning-based digital watermarking and steganography that includes the most extensive range of recent works. As this research area continues to expand, it is important to summarize and review the current methods. The survey aims to systematically categorize and discuss the existing deep learning models for data hiding, separating them based on their applications in either watermarking or steganography, as well as to present future research directions. In light of our findings, both digital watermarking and steganography share a common objective of embedding information in digital media, and that leveraging deep learning techniques can enhance their performance.

II. PROBLEM STATEMENT

When we evaluate the effectiveness of data hiding techniques, there are many factors that should be considered. The three most important are capacity, how much information can be embedded into the cover media, imperceptibility, how easy the data are to detect, and robustness, how resistant the data are to attacks. Here, attacks refer to any alterations made to the embedded media with the intent to degrade or remove the embedded data.

There is an implicit trade-off between these three aforementioned characteristics. For instance, if there is a high payload capacity, then the message will be easier to detect, resulting in a lower level of imperceptibility. Similarly, improving robustness against attacks can potentially decrease both payload capacity and imperceptibility, since there is added redundancy to the encoded image that allows it to resist distortions.

In digital watermarking, robustness is generally favored over secrecy, because the ability to resist attacks and distortions is more important than the watermark’s imperceptibility. Conversely, in steganography, imperceptibility is favored, since the highest priority is that the message remains a secret. This relationship is illustrated in Fig. 2. Due to the adaptable nature of deep learning-based approaches, the trade-off between these metrics can be explicitly controlled by the user, and the key properties of robustness and imperceptibility underpin the objective of the deep learning system.

![Hierarchical diagram showing different methods for classifying deep learning-based data hiding techniques. Blindness refers to the functionality of the data hiding method, further explained in Section II.](image-url)
The basic data hiding process consists of an encoding and decoding process. The encoder $E$ receives the cover media $C$ and message to be hidden $M$ and outputs the encoded media $C'$, such that: $E(C, M) = C'$. Then, the decoder receives the encoded media as input and extracts the message $M'$, such that: $D(C') = M'$. In a robust implementation, $M$ and $M'$ should be as similar as possible in an effective strategy. Similarly, maximizing the imperceptibility property is done by minimizing the difference between $C$ and $C'$. Fig. 1 provides a basis for classifying types of data hiding according to various properties, which can be further refined as follows [1].

1) **Blindness:** Related to how much information is required to extract the original data from the encoded media. Blind techniques do not require the original cover media or original data to extract the embedded data and are the most practically useful. Semiblind techniques require only the original data, whereas nonblind techniques require both the original cover media and data in order to perform data extraction.

2) **Fragility:** Related to how the embedded data reacts to attacks and distortions applied to it. Fragile data are designed to show all attacks applied to it, so that, when extracted, it is possible to verify which attacks have been applied to the media. This is useful when verifying the integrity of the media. Semifragile data are not robust against intentional distortions, such as warping and noise filtering that attempt to degrade the embedded data, but are robust against content-preserving distortions, such as compression and enhancement. Therefore, it can be used to trace any illegal distortions made to the media.

3) **Visibility:** Whether the embedded data are visible to the human eye.

4) **Invertibility:** Whether the embedded data can be removed from the cover media once embedded. Invertible data can be removed and noninvertible ones cannot.

5) **Robustness:** The ability of the embedded data to remain unchanged when attacks are applied to it.

6) **Security:** Determines how difficult it is for an adversarial party to extract the data from the cover image.

III. **Deep Learning-Based Data Hiding Techniques**

Deep learning-based data hiding models utilize the encoder–decoder network structure to train models to imperceptibly and robustly hide information. They present an advantage over traditional data hiding algorithms, because they can be retrained to become resistant to a range of attacks and be applied to different end-use scenarios. Deep learning methods negate the need for expert knowledge when crafting data hiding algorithms and improve security due to the black-box nature of deep learning models.

Section II-A discusses deep learning-based data hiding techniques separated into techniques focused on watermarking and steganography. The detection and removal mechanisms are then discussed at the end of this section. The classification of deep learning-based data hiding techniques detailed in this section is outlined in Fig. 3. It should be noted that convolutional neural networks (CNNs) incorporating adversarial training are different to generative adversarial network (GAN)-based methods. Adversarial training in this instance refers to the use of trained CNNs for noise injection during the attack simulation stage, while GAN-based methods incorporate a discriminator to scrutinize encoded and cover images to improve embedding imperceptibility [9].

A. **Introduction to Deep Learning-Based Data Hiding Architectures**

Currently, the majority of new watermarking models use an encoder–decoder architecture based on CNNs. A simple diagram showing the deep learning-based data hiding process can be found in Fig. 4.

In these models, the encoder embeds data in a piece of cover media; these encoded media are subjected to attack simulation, and then, the data are extracted by the decoder network. Through the iterative learning process, the embedding strategy becomes more resistant to the attacks applied during simulation, and the extraction process improves the integrity of the extracted data. The advantage of this technique over previous traditional algorithms is that they require no expert knowledge to program and can simply be retrained for different applications and attack types instead of needing to
be designed from scratch. The system exists as a black box with high nonlinearity where the intricacies of the embedding system are unknown and impossible to ascertain. This makes deep learning-based methods highly secure, as well as adaptable to different end-use scenarios.

Some variations of the simple CNN encoder–decoder approach shown above include convolutional autoencoders and CNNs with adversarial training components. The U-Net CNN architecture is common in steganography applications due to image segmentation abilities.

Many models adopt the GAN structure [10]. The GAN framework consists of a generative model and a discriminative model. In deep data hiding, the discriminator network is given a mixture of encoded and unaltered images and must classify them as such. Throughout the learning process, the generative model improves in its data embedding capabilities, producing highly imperceptible examples, while the discriminative model improves at identifying encoded images. The endpoint of training is reached when the discriminator can only identify legitimately encoded images 50% of the time—it is making random guesses. The use of discriminative networks can greatly increase data imperceptibility and is, therefore, useful for steganography as well as watermarking applications.

There are also further variations of the GAN framework, including Wasserstein GANs (WGANs) and CycleGANs. The CycleGAN architecture is useful for image-to-image translation and includes two generative and two discriminative models. The primary benefit of CycleGANs is that the model can be trained without paired examples. Instead, the first generator generates images from domain A, and the second from domain B, where each generator takes an image from the other domain as input for the translation. Then, discriminator A takes as input both images from domain A and output images from generator A and determines whether they are real or fake (and vice versa for discriminator B). The resulting architecture is highly useful for translating between images.

**B. Deep Learning-Based Watermarking Techniques**

In this section, we categorize current deep learning models for digital watermarking based on their network architecture design.

1) **Encoder–Decoder Framework:** Due to the encoding and decoding tasks central to the data hiding process, the encoder–decoder deep learning framework is well suited for data hiding models. The encoder and decoder networks incorporate CNNs, which are used in a variety of applications, such as detection, recognition, and classification, due to their unique capabilities in representing data with limited numbers of parameters. The layers in the CNN learn nonlinear, complex feature sets, representing the inputs and outputs to the network using weight sharing mechanisms. The following deep watermarking models adopt the encoder–decoder framework without including a discriminator, which characterizes GAN-based architectures [4], [9], [11], [12], [13], [14]. A simple diagram of the encoder–decoder deep watermarking structure can be found in Fig. 4.

   a) **Autoencoder-based model:** The encoder–decoder architecture is a general framework, and autoencoder is a special case of the encoder–decoder structure. However, autoencoder is usually used in unsupervised-learning scenarios by reconstructing the inputs. The potential of the CNN-based encoder–decoder frameworks for digital image watermarking was first explored in [4], which uses two traditional convolutional autoencoders for watermark embedding and extraction. Autoencoder-based CNNs were chosen based on their uses in feature extraction and denoising in visual tasks, such as facial recognition and generation, and reconstructing handwritten digits. The intuition was that the autoencoder CNN would be able to represent the watermark input in a form that was highly imperceptible when encoded within the cover image.

   This early CNN-based method applies two deep autoencoders to rearrange the cover image pixels at the bit level to create a watermarked image. The technique was found to outperform the most robust traditional frequency domain methods in terms of both robustness and imperceptibility. Although promising for the future of deep watermarking, this technique is nonblind and, therefore, not practically useful.

   Robust and blind digital watermarking results can be achieved using a relatively shallow network, as was shown by watermarking networks (WMNet) [12]. The watermarking process is separated into three stages: watermark embedding, attack simulation, where the CNN adaptively captures the robust features of various attacks, and updating, where the model’s weights are updated in order to minimize the loss function and, thereby, correctly extract the watermark message. Embedding is achieved by increasingly changing an image block to represent a watermark bit. The model is trained to extract watermark bits from the image blocks after attack simulations have been applied.

   The backpropagation embedding technique utilized in WMNet [12] used only a single detector network, which was found to cause performance degradation if the gradient computation in the backpropagation operation was affected by batch normalization (BN). This deficiency was improved by adding an autoencoder network as well as visual masking to allow flexible control of watermark visibility and robustness. The autoencoder network was added to the encoder and subsequently shortened time taken for both embedding and detection at the encoder, because the feed-forward operation is generally much faster than backpropagation. These improvements were established in their follow-up work.
b) Robustness controls and input preprocessing: Subsequent works after the aforementioned early techniques in [4] and [12] focused on generalizing the watermarking process for multiple applications. Mechanisms, such as robustness controls to influence the robustness/imperceptibility trade-off, were introduced to gear models toward both watermarking and steganography applications, and mechanisms, such as host and watermark adaptability, were developed to preprocess inputs. A blind and robust watermarking (ROMark) technique was achieved using the CNN-based system [11]. The aim of this model is to generalize the watermarking process by training a deep neural network to learn the general rules of watermark embedding and extraction, so that it can be used for a range of applications and combat unexpected distortions. The network structure is characterized by an invariance layer that functions to tolerate distortions not seen during network training. This layer uses a regularization term to achieve sparse neuron activation, which enhances watermark robustness and computational efficiency. The layer also includes a redundancy parameter that can be adjusted to increase the levels of redundancy in the resulting image, giving the model a higher tolerance of errors and increasing robustness. The primary aim of these features is to generalize watermarking rules without succumbing to overfitting. The model was compared with two autoencoder CNN methods [4], [12] and was found to achieve greater robustness due to the new features it adopted.

Residual diffusion watermarking (ReDMark) [13] uses two full convolutional neural networks (FCNs) for embedding and extraction along with a differentiable attack layer to simulate different distortions, creating an end-to-end training scheme. ReDMark is capable of learning many embedding patterns in different transform domains and can be trained for specific attacks, or against a range of attacks. The model also includes a diffusion mechanism based on circular convolutional layers, allowing watermark data to be diffused across a wide area of an image rather than being confined to one image block. This improves robustness against heavy attacks, because if one image block is cropped out or corrupted, the watermark can still be recovered. The trade-off between robustness and imperceptibility can be controlled via a strength factor that can influence the pattern strength of the embedding network.

The watermarking model developed by Lee et al. [14] uses a simple CNN for both embedding and extraction, without using any resolution-dependent layers. This allows for host image resolution adaptability—meaning that the images of any resolution can be used as input to the system to be watermarked. There is an image preprocessing network that can adapt images of any resolution for the watermarking process. There is also watermark preprocessing, meaning the system can handle user-defined watermark data. This is achieved by using random binary data as the watermark that is updated at each iteration of training. The model also adopts a strength scaling factor, which allows for the controllability of the trade-off between robustness and imperceptibility. The method showed comparable, if not better, performance compared with ReDMark [13] and two generative adversarial-based models [5], [15].

c) Adversarial training: A further improvement to the CNN-based encoder–decoder framework was made by adopting trained CNNs for attack simulation. While many other works use a fixed pool of attacks or a differentiable attack layer, using a trained CNN to generate attacks can greatly improve robustness and introduces an adversarial component to model training. The focus of the distortion agnostic (DA) model [9] was to directly improve the hiding data with deep networks (HiDDeN) model [5], primarily through adding robustness to the watermarking system in situations where the model is trained on a combination of distortions rather than one predetermined type. Instead of explicitly modeling different distortions during training from a fixed pool, the distortions are generated via adversarial training by a trained CNN. This technique was found to perform better in terms of robustness than HiDDeN [5] when distortions not seen during training were applied to images. The DA framework also incorporates channel coding, a means of detecting and correcting errors during signal transmission, to add an additional layer of robustness to images by injecting extra redundancy. The watermark message is initially fed through the channel encoder to add redundancy before being input into the encoding network. Similarly, prior to extraction, the redundant watermark message is input to a channel decoder to retrieve the final message.

2) Generative Adversarial Networks: The second primary approach for deep watermarking uses GANs, building upon the aforementioned techniques. Current models that adopt the GAN framework include [5], [15], [16], [17], and [18]. Many of the following models also use CNNs within their network architecture, but their use of the generative and discriminative components of the GAN framework set them apart from the aforementioned implementations.

The first end-to-end trainable framework for data hiding was a model called HiDDeN [5], which uses an adversarial discriminator to improve performance. It was a highly influential paper that has informed the development of deep watermarking models since its release. The model consists of an encoder network, trained to embed an encoded bit string in a cover image while minimizing perceptual perturbations, a decoder network, which receives the encoded image and attempts to extract the information, and an adversary network, which predicts whether or not an image has been encoded. HiDDeN [5] uses a novel embedding strategy based on adversarial examples. When neural networks classify image examples to a particular target class, invisible perturbations in the image can fool the network into misclassifying that example [19]. These perturbations have been shown to remain preserved when exposed to a variety of image transformations [20]. Adversarial examples are ordinarily a deficiency in neural networks, since they reduce classification accuracy. However, since meaningful information can be extracted from imperceptible image perturbations, it was theorized that in a similar fashion, meaningful information could be encoded in adversarial distortions and used as a watermark embedding strategy. This embedding technique, paired with the GAN framework, is able to achieve a higher payload capacity [measured in bits per pixel (BPP)]
than other common data hiding mechanisms, such as highly undetectable steganography (HUGO) [21], wavelet obtained weights (WOWs) [22], and spatial-universal wavelet relative distortion (S-UNIWARD) [23]. One drawback of HiDDeN [5] was that loss between encoded and decoded messages was minimized when trained only on a specific kind of attack compared with a combination of different attack types. This shows that the model is best when trained specifically to combat one type of attack, but not as effective when trained on a variety.

This shortcoming was improved in the model ROMark [16], which builds upon the framework from HiDDeN [5] by using a min–max formulation for robust optimization. This was done by addressing two main goals; first, to obtain the worst-case watermarked images with the largest decoding error, and second, to optimize the model’s parameters when dealing with the worst-case scenario, so that decoding loss is minimized. The idea of this technique is to minimize decoding loss across a range of attacks, rather than training the model to resist specialized attacks, creating a more versatile and adaptable framework. Due to the optimization for worst-case distortions in ROMark [16], it performed better when trained on a combination of attacks, particularly on those that had not been seen during training. ROMark [16] was also more robust in some specialized attack categories, though HiDDeN [5] had higher accuracy in these categories.

Additional improvements were made to the framework from HiDDeN [5] by Hamamoto and Kawamura [24]. This work uses a neural network for attack simulation rather than a single differentiable noise layer. There is a rotation layer followed by an additive noise layer, allowing the model to learn robustness against geometric rotation attacks. It also features a noise strength factor to control the robustness/imperceptibility trade-off. It was tested against HiDDeN [5] and found to achieve superior robustness for all but one attack category (crop). However, the two-stage training method is also robust against black-box noise attacks that are encapsulated in image processing software, which have not been tested in previous works.

a) Wasserstein GAN: A popular variation of the traditional GAN technique is the WGAN [27]. This technique improves the model stability during training as well as decreases the sensitivity of the training process to model architecture and hyperparameter configurations. The WGAN framework also provides a loss function that correlates with the quality of generated images. This is particularly useful for watermarking and steganography in the image domain, since image quality must be effectively optimized. Instead of the discriminator component of the network, WGANs include a critic. Rather than predicting the probability that a given image is real or fake, as is the discriminator’s goal, the critic outputs a score denoting the “realness” of the input image. In a data hiding scenario, the encoder’s aim is to maximize the score given by the critic for real instances—which corresponds to an encoded image. Current papers adopting the WGAN framework for their watermarking models include [26], [28], [29], and [30].

Zheng et al. [26] introduced SteganoGAN. Three variants of encoder architecture are explored, each with different connectivity patterns. The basic variant applies two convolution blocks, where the encoded image is the output of the second block. Residual connections have been shown to improve model stability [31], and in the residual variant, the cover image is added to the encoder outputs, so that it learns to produce a residual image. The third variant uses a method inspired by DenseNet [32], in which there are additional feature connections between convolutional blocks that allow the feature maps generated by earlier blocks to be concatenated to those generated by subsequent blocks.

Using adversarial training, this model achieves a relative payload of 4.4 BPP, ten times higher than competing deep learning methods. Although both works [5], [26] have mechanisms in place for handling arbitrarily sized cover images as input, the higher payload capability of [26] means that it can support a greater range of watermark data. This article also proposes a new metric, Reed Solomon BPP (RS-BPP), to measure the payload capacity of deep learning-based data
hiding techniques, so that results can be compared with traditional data hiding methods. Although the primary focus for SteganoGAN is steganography, the high payload capacity and low detection rate of SteganoGAN produced images can also be applied to watermarking.

Two further works by the same authors also use the WGAN framework. Plata and Syga [28] introduced a new embedding technique where the watermark is spread over the spatial domain of the image. The watermark message is converted into a sequence of tuples, where the first element of each is converted to a binary representation. The spatial message is created by randomly assigning binary converted tuples to sections of the message, including redundant data for increased robustness. This article also introduces a new technique for differentiable noise approximation of nondifferentiable distortions, which allows the simulation of subsampling attacks. The attack training pool is expanded from previous works to include subsampling and resizing attacks, and this wide range of attacks used during training increases general robustness. However, the spatial spread embedding technique reduces the embedding capacity, so it is only useful for applications where capacity is not a priority. Furthermore, the training framework proposed requires half as much time as prior methods [5], [9], [13]. The authors expand upon this work in a follow-up paper [29], which introduces the double discriminator-detector architecture. The discriminator is placed after the noise layer and receives both noised cover images and noised encoded images, and thus, the discriminator learns to distinguish watermarked and nonwatermarked images with attacks already applied. In practical contexts, this is useful, because it reduces the likelihood of false accusations being made of IP theft. If the image has already been attacked and must be proven to contain a watermark, this training technique is useful. Crucially, it does not degrade the overall robustness of the encoded images.

A technique for encoded image quality improvement is introduced by Wang et al. [30] based on texture analysis. The cover image texture features are analyzed by a gray occurrence matrix, which divides the image into complex and flat regions. This article utilizes the StegaStamp network [33] for embedding the watermark in the flat texture regions. This reduces the degree of image modification and improves the quality, and hence imperceptibility, of encoded images. The network in StegaStamp [33] produces higher quality images from low contrast examples; therefore, the contrast value is used to calculate the texture complexity of the image.

Table I shows a summary of the deep watermarking models reviewed in Section III-B. It shows whether the embedding strategy operates in the spatial or frequency domain, the nature of the watermark embedding and extracting networks, whether the technique supports host resolution adaptability (so that any cover image resolution can be used), and whether it includes controls for influencing the trade-off between imperceptibility and robustness. The remarks column describes any important information or novel contributions of this article.

### C. Deep Learning-Based Steganography Techniques

This section classifies deep steganography techniques based on model architecture. Most techniques adopt the encoder–decoder structure shown in Fig. 4 and are based on CNNs. Many implementations differ from watermarking approaches in their use of the U-Net structure for image segmentation [34]. Table II shows a summary of deep learning-based steganography methods reviewed in this section.

1) Encoder–Decoder Framework: CNN-based encoder–decoder structures have been adopted in the following papers [33], [35], [36], [37], [38], [39], [40], [41]. A simple diagram of the encoder–decoder deep watermarking structure can be found in Fig. 4. A paper from Google research [35], published in 2017, presented a deep steganography technique for hiding images inside other images. The structure contains three components in total. The first being the prep network, which serves two purposes; to adjust the size of a smaller secret image to fit into the cover image, and to transform
the color-based pixels to recognizable features to be encoded into the cover image. The second layer of the encoder is called the hiding network. As the name suggests, this layer creates the final stego-image. The final layer is the reveal network, which is used to decode the output from the second layer. The experiments in the work [35] were mainly conducted to show that it was possible to completely encode a large amount of data with limited visual disturbances in the cover media. However, such a technique lacked robustness, security, and was not of high quality. It was possible for attackers to recover both the cover and secret image with a trained network.

Another image encoding technique called StegNet [36] was released in 2018. StegNet used structures from both autoencoders and GANs to set up the encoding network. The cover image and the secret image were concatenated by a channel prior to the CNN encoding structure. Variance loss was included in loss calculations for the encoder and the decoder. It was found that including the variance loss helped the neural network distribute the loss throughout the image rather than having concentrated areas of perceptual loss, improving the overall imperceptibility of embedding. The presented technique was highly robust against statistical analysis, and when used against StegExpose [42], a commonly used steganalysis tool, it was also resistant against those attacks. Although robust, there were still some limitations to this method. Secret images and cover images must match in size, and noise is still somewhat prominent in smoother regions.

Comparing the method of Baluja [35] and StegNet [36], there has been a vast improvement in image hiding. The inclusion of components adapted from GANs and autoencoders in StegNet [36] increased the robustness of the stego-images and was, therefore, more resistant to StegExpose [42]. Though StegNet [36] is quite robust, it is still lacking in some areas, such as quality, image size restrictions, and noise.

A faster region-based CNN (R-CNN) method was introduced in [37]. First, the cover image is passed through a region proposal network, which makes the selection for feature extraction faster. Softmax loss is used to box these regions, and then, the specific existing steganographic algorithms are selected and assigned to the boxed regions. Since [37] uses a technique of selecting different steganography algorithms, using a combination of HUGO [21], S-UNIWARD [23], and WOW [22] algorithms, it is able to achieve highly imperceptible embedding. Being able to select effective areas on the cover image also allows Meng et al. [37] to maintain a high level of robustness.

The adaptation of the fusion technique [37] allows for minimal distortion in the extracted stego-image. When looking at StegNet [36], there is a concern for noise in smoother areas of the cover photograph. Although [36] has been shown to be robust, it could be further improved in this area with the box selection the fusion method [37] has. This does lead to a capacity dilemma if there was a method combining both StegNet [36] and the fusion method [37], since the latter would naturally be using less cover image area and, therefore, has a smaller capacity compared with StegNet [36].

A unique approach for steganography is shown by Sharma et al. [38]. In this article, both the encoder and decoder consist of two layers. In the first layer, the prep layer, smaller images are increased in size to correctly fit the cover image; there is a reconstruction of color-based pixels to create more useful features, and the pixels are scrambled and then permuted. The second layer, the hiding layer, produces the stego-image with the output of the first layer and the cover image inputted. The decoder consists of the reveal layer and the decrypt layer. The reveal layer removes the cover image, and the decrypt layer decrypts the output of the reveal layer. The advantage of this technique is that the first layer and its encryption method are similar to cryptography practices, allowing for a more secure embedded stego-image. Even when the cover media is known to the attacker, it is far more secure and difficult for the attacker to decode the secret image. This technique can also be applied to audio.

A reversible image hiding method was introduced by Chang [41]. The structure relies on a concept called long short-term memory. To encode, the cover image goes through a neural network in order to get a prediction, called the reference image. By subtracting the cover image from the reference image, the cover residuals (prediction errors) are calculated. Using histogram shifting (HS) on the cover residuals then produces stego-residuals, along with an overflow map that is later used for the decoder. The stego-image is then created by adding the stego-residual to the reference image. Where there is a pixel intensity flow, the overflow map is precalculated to flag these pixels. The decoder is essentially the reverse of the encoder, where the stego-image goes through a neural network to get a reference image, and the rest follows in reverse. This technique creates high-quality, high-capacity images that contain minimal noise. Its invertible feature to recover the cover image is unique to many other CNN-based steganography.

Tang et al. [43] produced a steganography technique that uses adversarial embedding, called ADV-EMB. The network is able to hide a stego-message while being able to fool a CNN-based steganalyzer. The encoding network consists of a distortion minimization framework that adjusts and minimizes the costs of the image according to features (gradients backpropagation) from the CNN steganalyzer. The focus of ADV-EMB [43] is to prevent steganalyzers from being able to detect the stego-image. This shows in their results, with a high security rate and also increased imperceptibility. They are also able to train the system to counter unknown steganalyzers by using a local well-performing CNN steganalyzer as the target analyzer, allowing for diverse applications. Although ADV-EMB [43] is able to decrease the effectiveness of adversary-aware steganalyzers, it has a weak pixel domain. However, an increase in payload capacity would increase the detection rate of steganalyzers.

1) U-Net CNN: U-Net CNNs are used to facilitate more nuanced feature mapping through image segmentation. This is useful in image steganography applications, because a cover image can be broken up into distinct segments based on a certain property (for example, [39] uses a heatmap to score suitable embedding areas).
A U-Net CNN technique for reversible steganography was developed by Duan et al. [39]. This technique is capable of directly encoding the secret image into the cover image by concatenating the secret image into a six-channel tensor. Its decoder is formed from six convolution layers, each of which is followed by a BN and a rectified linear unit (ReLU) activation layer. The embedding technique is not easily affected by excessively high- or low-frequency areas. It is also able to produce a high-quality image with a capacity that is, in general, better than other cover-selection and cover-synthesis-based steganography techniques. Even with its high-capacity capabilities, such as other steganographic techniques that do not focus on robustness, too high of an embedding rate will increase the distortion rate more dramatically compared with robustness-based steganography.

Universal deep hiding (UDH) is a model proposed for uses in digital watermarking, steganography, and light field messaging [40]. The encoder uses the simplified U-Net from Cycle-GAN [48] and a dense CNN for the decoder. The encoder hides the image in a cover-agnostic manner, meaning that it is not dependent on the cover image. UDH is an effective method due to the high-frequency discrepancy between the encoded image and the cover image. This discrepancy makes embedding robust in low-frequency cover images. UDH is also less sensitive to pixel intensity shifts on the cover image. The UDH method was compared with cover-dependent deep hiding (DDH). Since the encoding of the secret image was independent of the cover image, there was no method to adapt the encoding mechanism according to the cover image. The cover image may have some smoother areas, which may not be ideal to embed data into, but with UDH there was no method of finding out this type of information. UDH is also less able to work well with severe uniform random noise.

b) CNN with adversarial training: The inclusion of adversarial attack networks during training can be used to help improve against steganalysis and promote robustness. Adversarial training helps the system distinguish small perturbations that an untrained steganography method may bypass. This is an important feature to have in steganography, since its main focus is to protect message security. Chen et al. [44] developed a model that incorporates a trained CNN-based attack network that generates distortions, similar to the technique used in the watermarking framework used by Luo et al. in DA Watermarking [9]. The encoding structure is based off of a simple model that hides a secret gray-scale image into channel B (blue) of a colored cover image, where both must have the same resolution. From this basic model, this article was able to add two other enhanced models, a secure model and a secure robust model. The secure model inserts a steganalysis network into the basic model where its goal is to increase security against steganalysis. The secure and robust model uses the secure model and inserts an attack network to increase the robustness of the system. The separation of each model allows the framework by Chen et al. [44] to be used in several different scenarios, allowing the user to adjust to their needs. A downside is that users are unable to send red, green, and blue (RGB) pictures and are limited to just gray-scale images.

The secure model was shown to have the best invisibility against all models and was still the best when compared with the following methods [35], [49]. The secure and robust model did not perform as well as the secure model but was still able to improve. The basic model and the secure model showed increased visual results in this comparison, with the secure model having the best visual results. The visual results of the secure and robust model were similar to invisible steganography via GAN (ISGAN) [45].

2) GANs-Based Models: GANs are used extensively in deep steganography. Various new structures have allowed the simple GAN structure to be improved, increasing the effectiveness of steganography. It has brought up interesting techniques, such as coverless steganography, and the ability to generate cover images. GAN-based architectures have been used in the following papers [45], [46], [47], [50], [51].

ISGAN is a steganography technique that uses the GAN structure [45]. Similar to the secure technique developed by Chen et al. [44], ISGAN is only able to send a gray-scale secret image. The encoder first converts the cover image into the YCrCb color space where only the Y channel is used to hide the secret image, since it holds luminance but no color information. This color space conversion does not affect backpropagation. The encoder also uses an inception module, which helps to fuse feature maps with different receptive field sizes. To aid the speed of training, the technique also adds a residual module and BN. A CNN is used to decode with BN added after every convolutional layer excluding the final...
layer. The results of this model showed that it was able to achieve a high level of robustness, with low detectability when scrutinized using steganalysis tools.

In comparison with [49], ISGAN residuals were less obvious, showing that the extracted secret image of ISGAN is much closer to the original than [49]. ISGAN is also able to achieve higher levels of invisibility, since ISGAN uses a gray-scale image. Thus, there is a trade-off between the complexity of the image (i.e., RGB values) and the imperceptibility of embedded information.

Two separate designs are introduced by Li et al. in [46], which studies embedding data into texture images. The first model separates the texture image generation and secret image embedding processes. The texture image is generated using a deep convolutional generative neural network. The output of this is then used as the input for the concealing network for image hiding. The second model integrates the concealing network with the deep convolutional generative network. This second network should be able to generate the texture image while simultaneously embedding another image. The first model is easier to train and can be used in more diverse applications than the second model. Detection rates from steganalysis tools are almost 0 in both cases. These models provide high security, but with a few limitations. First, the cover images generated are only textures, and other subjects are not considered. Color distortions also occur in the second model when the cover and secret images differ too much.

Coverless steganography is possible because of the features of GAN. The general encoding idea of the proposed coverless method [47] was that, at first, convolutional blocks were used to process the cover image to get a tensor, \( a \). The secret message was then concatenated to \( a \) and processed through another convolutional block to get \( b \), which was the same size as \( a \). This article details two different models, one called basic model and the dense model. The basic model uses the aforementioned encoding scheme, whereas the dense model includes a skip connection to increase the embedding rate. The decoder for both models uses Reed Solomon algorithms on the tensor produced from the stego-image. The aim with this model was to improve the capacity and quality that other coverless steganography has not been able to achieve. Encoded image quality and payload capacity of the stego-images were improved when compared with [26]. The basic model introduced by Qin et al. [47] was able to perform significantly better in these aspects, while the dense model was able only to match SteganoGAN [26]. The decoding network used in the coverless method [47] was also more accurate than the one proposed for SteganoGAN [26]. But, there is a difference in the media encoded, where the coverless method [47] encodes a string of binary message, while SteganoGAN [26] is able to encode an image.

Many steganography techniques have not been invertible and leave the cover image distorted after removing the secret piece of media. With the method proposed by Chang [51], the model was able to achieve invertible steganography. The method used is based on the regular-singular (RS) method. RS realizes lossless data embedding through invertible noise hiding. There are three discriminate blocks used in this technique—regular, singular, and unusable—in order to get the RS map. The adversarial learning component serves to capture the regularity of natural images. The model uses conditional GAN to synthesize, where the generator uses a U-Net structure and a Markovian discriminator is used. The use of GAN in conjunction with the RS method greatly improved upon the results of previous RS-based models.

Currently, the paper [39] uses a basic GAN structure and could be further improved or diversified with the adaption of other GAN structures. This could lead to further improvements in cover media recovery in terms of quality. Overall, the adversarial learning adopted in GAN-based models leads to greater robustness in steganography applications.

a) CycleGAN: CycleGAN is a variation of the GAN architecture for image-to-image translation. The CycleGAN framework was adopted for steganography in S-CycleGAN [50]. Within the model, there were three discriminators, two of which used the same function as the original CycleGAN framework. The third was an increased steganalysis module used to distinguish the stego-image from the generated images. The training cycle consists of three stages. First is the translation of the image from the X-domain to the style of the Y-domain. Second is the use of an LSB matching algorithm to embed the secret message into the output of the first stage. The third stage is where the stego-image is reconstructed to the input image of the first generator to the second generator. The full objective function included adversarial loss for all discriminators, as well as cycle consistency loss for the generative models. The advantage of S-CycleGAN is its ability to produce high-quality images that are also robust against steganalysis. Overall S-CycleGAN [50] showed results that were more resistant to detection, increasing the invisibility of embedded stego-images.

When S-CycleGAN [50] is compared against steganographic GAN (SGAN) [52], the quality of the image is much higher, with 2.6× the inception score and 7× the Frechet inception distance of SGAN. In general, the results of S-CycleGAN [50] showed that it was much more robust than SGAN, and the combined use of the original CycleGAN framework and traditional steganography algorithm S-UNIWARD [23].

D. Discussion

From the state-of-the-art deep learning methods discussed above, it seems that the GAN framework is the most promising in terms of robustness and secrecy optimization due to the inclusion of adversarial loss in the objective calculations. To illustrate this, in HiDDeN [5], tests were conducted without including the adversary network and found to include perceptible alterations, whereas including the adversary greatly improved performance to produce an invisible watermarking technique. In tests comparing robustness, GAN-based models performed well, but were improved by techniques to target robust pixels for watermark embedding, such as the attention mechanisms used in attention-based data hiding [17] and the IGA method [18].

It is also important for models to be robust against a range of attacks. Papers, such as those by Hamamoto and
Kawamura [24] and Plata and Syga [28] incorporate geometric rotation techniques, and subsampling and resizing attacks, respectively. Having a wider range of attack types during training increases general robustness. In addition, using true nondifferentiable noise, as shown in the two-stage training technique developed by Liu et al. [15], provides better results for JPEG compression than models that use differentiable approximations. Using a trained CNN to generate attacks, DA watermarking [9] is also a promising approach for diversifying the attacks encountered during training.

To produce an adaptable, generalized framework, it is important to include features, such as host and watermark resolution adaptability, as well as robustness controls to influence the robustness/imperceptibility trade-off. Robustness controls make these models suitable for both watermarking and steganography, which relies more heavily on imperceptibility. The ideal digital watermarking model would include preprocessing networks for any watermark data or cover image to be input, as the framework was developed by Lee et al. [14], while also taking advantage of an adversarial discriminator or critic. The techniques used in [16] to improve robustness by obtaining and optimizing parameters for ROMark worst-case examples are also a promising technique for improving robustness.

If a future model could combine these features; attention mechanisms to improve embedding, robustness against geometric attacks, host and watermark resolution controls, robustness controls, and handling worst case distortions, it could result in a highly robust and adaptable framework. However, the added overhead of all these added features could be an issue in practice.

IV. OPEN QUESTIONS AND FUTURE WORK

Deep learning for data hiding is a new and evolving research field, and there remain many different avenues to consider moving forward. This section will discuss some open questions that we believe warrant further research and consideration, including expanding the applications of digital watermarking to other media domains, pursuing deep learning-based language watermarking, improving robustness against deep learning-based watermark removal attacks, watermarking machine learning models, combating the use of watermarking to launch backdoor attacks on machine learning models, and exploring the applications of watermarking for detecting and identifying synthetic media. Finally, future directions for steganography are discussed, including its potential use for spreading malware.

A. Expanding Applications for Deep Learning Digital Watermarking Models

While the deep watermarking models discussed in this survey were primarily focused on image watermarking, there is significant potential for applying watermarking to other types of media. Although there are many traditional algorithms focused on watermarking video, audio, 3-D models, and electronics, there is yet to be any deep learning models focusing on these areas. Furthermore, a recent paper from Google [53] details a promising deep learning technique of embedding watermark messages into simple 3-D meshes, which can then be decoded from 2-D rendered images of the model from multiple angles and lighting configurations. It is noted in this work that robustness and capacity will need to be improved before practical application, particularly robustness to nondifferentiable 3-D attacks. More complex models and lighting arrangements could be explored with a better-quality renderer. A paper from Google research by Yoo et al. [53] sets the precedent for watermarking 3-D models using deep learning, but more work is required before a generalized, practically applicable framework for 3-D watermarking can be developed.

Similarly, audio watermarking faces a comparable situation. While there are existing traditional methods for audio watermarking and GAN-based frameworks for audio generation [54], [55], there is currently a lack of exploration of deep learning frameworks specifically designed for audio watermarking. This suggests that machine learning models have the potential to learn audio embedding techniques and apply them to audio databases. As far as we are aware, there are currently no works that investigate deep learning frameworks for audio watermarking. However, given the growing interest in this field, it is likely that there will be forthcoming contributions.

In addition to image and audio watermarking, video watermarking is another promising direction for research. There are existing traditional algorithms for video watermarking [56], and deep learning techniques are also being considered, for example, in [57]. This article introduces robust invisible video watermarking with attention (RIVAGAN), a new architecture for robust video watermarking. The attention-based architecture is robust against common video processing attacks, such as scaling, cropping, and compression. In the framework, a 32-bit watermark is embedded into a sequence of frames, and the watermark can be extracted from any individual or collection of frames. The framework uses an attention mechanism to identify robust areas for embedding and produces watermarked footage that is nearly indistinguishable from human observers (approximately 52% detection accuracy). There is also deep multiscale framework for video watermarking (DVM) [58] from Google research, which employs a multiscale design where the watermark is distributed across multiple spatial–temporal scales. It was found to be more robust than traditional video watermarking algorithms [3-D discrete wavelet transforms (DWT)] and the deep learning-based image watermarking framework from HiDeN [5] against video distortions, while retaining a high level of quality. A 3-D CNN that simulates video compression attacks was used during training to achieve high levels of robustness. The framework in DVM [58] also includes a watermark detector that can analyze a long video and locate multiple short instances of copyrighted content. This could be useful for reliably identifying copyright infringement on online video platforms, such as YouTube.

B. Text Watermarking for Combating Misinformation

Another promising application for deep watermarking is in watermarking text. As natural language generation
technology improves, machine learning models are capable of generating highly fluent language that can fool human detectors [59]. There is growing concern that such models will be used to spread misinformation and “fake news” online. The adversarial watermarking transformer (AWT) developed by Abdelnabi et al. [60] is the first end-to-end deep learning model for language watermarking. Language watermarking is inherently more complex than image, video, and audio watermarking, because the language itself must be altered, which can cause drastic syntactic and semantic changes. Previous techniques include synonym replacement and altering sentence structure, which rely on fixed rule-based techniques. The aim of such techniques is to achieve high effectiveness, secrecy, and robustness, while sustaining only subtle changes to the text, preserving correct structure and grammar, as well as language statistics. The deep learning model AWT [60] uses an attention mechanism and adversarial training to achieve robustness against attacks, such as denoising, random changes, and re-watermarking. The model undergoes human evaluation and achieves better results than the state-of-the-art synonym substitution baseline. This technique only works effectively for long pieces of text, such as news articles, whereas shorter pieces would require longer relative watermarks, thereby noticeably degrading the original text. For practical scenarios, it is suggested to combine this AWT technique with automated or human fact checking to reduce the likelihood of false positives. Further research into deep learning-based models for language watermarking is highly important, as text-generating models continue to improve and become widely available to potentially malicious actors. This will help to identify misinformation and differentiate generated from genuine text.

C. Mitigating Data Hiding Removal Attacks

As technology for data hiding embedding improves, so too does technology for attacking and removing those hidden information. Thus, the process can be regarded as an ever-evolving, adversarial game between content owners and attackers. Currently, deep learning-based techniques for data hiding removal exist that are able to remove information robustly embedded using the top frequency domain algorithms. This technique uses a simple CNN to perform denoising removal attacks and is able to not only remove the hidden information, but also recover the original cover images without significant quality degradation. It was tested on a dataset of images with hidden information created using state-of-the-art discrete cosine transform (DCT), DWT, and discrete Fourier transform (DFT) traditional algorithms in a black-box setting. Although this technique is focused on traditional algorithms, as deep learning techniques for digital data hiding evolve, it is inevitable that adversarial techniques will continue to be developed that aim to remove these hidden information. Therefore, it is important to continue to improve the robustness of data hiding techniques, so they can resist these emerging, deep learning-based removal methods.

Current deep data hiding strategies have been tested against a range of attacks, including cropping, pixel dropout, compression, and blurring. However, in most models, these attacks are encapsulated by a differentiable attack layer, meaning that they must support backpropagation, which is not representative of many real-world attack scenarios. However, it should be noted that these models can still simulate JPEG compression, which is nondifferentiable, by using differentiable approximations. Promising techniques for improving the scope of attacks used during training include generating attacks using adversarial examples from a trained CNN [9], and the use of blackbox noise [15], which are from algorithms encapsulated in the image processing software that is difficult to simulate. To achieve optimal robustness, it is important to train models on attacks generated from an adversarial network to improve results, rather than generating attacks from a fixed pool of differentiable attacks, as was done in earlier implementations.

D. Watermarking for Protecting Machine Learning Models

Another important application for digital watermarking is protecting machine learning models as intellectual property. Although the survey has discussed watermarking digital media, such as images, audio, and video, machine learning models themselves require increasingly large amounts of computational resources and private training data to train and operate. Therefore, there is a growing need to protect machine learning models as IP. Digital watermarking is just one of many tasks that were once done using traditional algorithms, but are now being offloaded to the effectiveness of machine learning strategies. As this happens, it is important to protect these models from theft and misuse, not only because of the resource expenditure by the owners in creating these models, but also because their immense computational capabilities could be used for malicious activities.

There are many techniques currently being researched for watermarking machine learning models, most of which rely on embedded identifying information into training datasets. However, as was learned through the concept of adversarial examples, even small perturbations in training instances can cause extreme degradation in the model’s performance. Therefore, many watermarking strategies also sacrifice the model’s classification accuracy.

A famous example is DeepSigns [61], an end-to-end IP protection framework that inserts digital watermarks into deep learning models by embedding the watermark into the probability density function of the activation sets in different layers of the network. It is robust against a range of attacks, including model compression, fine-tuning, and watermark overwriting, which proved challenging for previous model watermarking techniques.

One recent and promising technique for watermarking training datasets is entangled watermarking embeddings [62]. Instead of only learning features sampled from the task distribution, the defending model also samples data that encode watermarks when classifying data. Therefore, watermark data are entangled with legitimate training data, so an adversary attempting to remove the watermarks cannot do this without damaging the training data itself and, thereby, sacrificing performance. The method uses soft nearest neighbor loss to increase entanglement, a new loss function. Also, the method is evaluated in the image and audio domains, showing that with this method, an owner can claim with 95% confidence
that model extraction has taken place to produce a similarly performing model by extracting prediction vectors. This technique notably shows robustness against adaptive adversaries, meaning the adversary has knowledge of the watermarking technique being used.

As machine learning models become more ubiquitous across a range of industries, it is important to implement digital watermarking strategies within the models themselves, so that the owner’s private information remains secure. However, there will inevitably be technology developed to remove watermarks, even from machine learning models. For example, REFIT is a recent unified watermark removal framework based on fine-tuning. It does not require knowledge of the watermarks being removed and is effective against a wide range of current watermarking techniques.

E. Data Hiding for Launching Backdoor Attacks

Deep neural networks have been proven vulnerable to backdoor attacks, where triggers can be embedded into DNNs through data hiding, and they can trick the model into producing unexpected behavior with crafted triggers. For instance, watermarks can be embedded into the training examples of machine learning models in order to cause inaccurate classifications when the model is deployed. Many third-party cloud computing providers, such as Google, Microsoft, Azure, and Amazon, provide machine learning as a service (MLaaS) to train machine learning models. A malicious party can embed watermarks into training data images and train the model to misclassify such examples, either randomly (random target attack) or mislabel as a different example (single target attack). As more third-party providers offer MLaaS and machine learning models become more ubiquitous, backdoor attacks on neural networks are certain to become more common. Therefore, it is important to be able to detect triggers embedded in training examples. As data hiding technologies become more advanced through deep learning as discussed in this survey, this will become more difficult.

Although the techniques presented in this survey are promising for digital IP protection purposes, they will inevitably be utilized for malicious purposes, such as embedding undetectable, invisible backdoors in machine learning models.

F. Deepfake Detection and ID

Synthetic media technologies are rapidly advancing, and it is more painless to generate media, such as images, audios, and videos, that look and sound increasingly realistic. Since deepfakes often present a person saying or doing something they have not done or said, people need to be able to identify the original source of the media that has been manipulated. Similarly, it is important to be able to identify a piece of media as synthetic in the first place, without malicious parties removing this identifying tag and presenting the synthetic media as genuine. To this end, the ID and detection of synthetic media become a promising application for digital watermarking.

A recent paper presents DeepTag, an end-to-end deep watermarking framework that includes a GAN simulator that applies common distortions to facial images [63]. The watermark can be recovered to identify the original unaltered facial image. In future, as regulations surrounding deepfakes arise, watermarking techniques that are robust to GAN-based distortions will become increasingly important. A connected application is embedding watermarks into synthetic media, so that they can be easily identified as such.

G. Malware Using Steganography

Digital steganography can be used maliciously to spread malware to victim technology. The ability to hide an executable file within an image or audio file gives attackers an easy attack vector to target unaware users. There have also been known cases of attackers using steganography to pass data through unsuspecting platforms. They can easily set a time for the receiver and either upload or update an image temporarily. At this set time, the receiver can download and save the photograph and decode the message, and then, the attacker can restore the image to the original or delete the image. This could be almost impossible to detect when the third parties are unaware of where the attack will take place and would be even more unlikely to catch the act if the stego-image used was highly imperceptible. These high levels of imperceptibility can now be achieved easily through modern deep learning approaches.

Steganalysis could be implemented into antivirus software that not only scans images but scans sites before entering. Though this would be a difficult task due to the large amount of power it would require, constantly scanning almost every website or item on the web page just in case there is malware present. There could be a filter that decides when steganalysis could be used such as situations when users decide to continue onto an already scanned suspicious website. Currently, the possibilities are limited, but the implementation of steganalysis in antivirus software may be essential in the future as steganography techniques both improve in performance and become more widely accessible to the public.

V. Conclusion

This survey has provided an extensive overview of current deep learning techniques for data hiding, encompassing watermarking and steganography methods. Through analysis of network architecture and model performance, the survey has demonstrated how digital watermarking and steganography share a common goal of embedding information in digital media, and how both can benefit from deep learning techniques. In addition, the survey explored future research directions and highlighted the potential for this field to revolutionize the protection of digital IP and communication security in responsible AI software industries. As deep learning techniques continue to advance, they are expected to surpass traditional algorithms in all types of media, ultimately enhancing the accountability and safety of AI. This promising field holds great potential and is expected to have a significant impact on digital security.
ACKNOWLEDGMENT

The authors would like to thank Wendy La for the discussion and feedback on this article.

REFERENCES

[1] C. Kumar, A. K. Singh, and P. Kumar, “A recent survey on image watermarking techniques and its application in e-governance,” Multimedia Tools Appl., vol. 77, no. 3, pp. 3597–3622, 2018.
[2] A. Rasmi, B. Arunkumar, and V. M. Anees, “A comprehensive review of digital data hiding techniques,” Pattern Recognit. Image Anal., vol. 29, no. 4, pp. 639–646, Oct. 2019.
[3] I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Hallorana, “Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research,” Neurocomputing, vol. 335, pp. 299–326, Mar. 2019.
[4] H. Kandi, D. Mishra, and S. R. K. S. Gorre, “Exploring the learning capabilities of convolutional neural networks for robust image watermarking,” Comput. Secur., vol. 65, pp. 247–268, Mar. 2017.
[5] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, “Hidden: Hiding data with deep networks,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 657–672.
[6] K. Sharma, A. Aggarwal, T. Singhania, D. Gupta, and A. Khanna, “Hiding data in images using cryptography and deep neural networks,” J. Artif. Intell. Syst., vol. 1, no. 1, pp. 143–162, 2019.
[7] S. T. I. Hussain and J. Zeng, “Survey on deep learning based data hiding, steganography and steganalysis,” KSII Trans. Internet Inf. Syst., vol. 14, pp. 1228–1248, 2020.
[8] C. Zhang, C. Lin, P. Benz, K. Chen, W. Zhang, and I. S. Kweon, “A brief survey on deep learning based data hiding, steganography and watermarking,” IEEE Trans. Multimedia, vol. 23, pp. 1951–1961, 2021.
[9] X. Luo, R. Zhan, H. Chang, F. Yang, and P. Milanfar, “Distortion agnostic deep watermarking,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 13548–13557.
[10] I. J. Goodfellow et al., “Generative adversarial networks,” in Proc. Adv. Neural Inf. Process. Syst., vol. 3, no. 11, 2014, pp. 1–6.
[11] X. Zhong, P.-C. Huang, S. Mastorakis, and F. Y. Shih, “An automated and improved steganography embedding scheme based on deep neural networks,” IEEE Trans. Multimedia, vol. 23, pp. 951–961, 2021.
[12] S.-M. Mun, S.-H. Nam, H. Jang, D. Kim, and H.-K. Lee, “Finding robust domain from attacks: A learning framework for blind watermarking,” Neurocomputing, vol. 337, pp. 191–202, Apr. 2019.
[13] M. Ahmadi, A. Norouzi, N. Karimi, S. Samavi, and A. Emami, “ReDMark: Framework for residual diffusion watermarking based on deep networks,” Exp. Syst. Appl., vol. 146, May 2020, Art. no. 113157.
[14] J.-E. Lee, Y.-H. Seo, and D.-W. Kim, “Survey of image steganography based on generative adversarial network,” IEEE Trans. Inf. Forensics Secur., vol. 60, no. 2, pp. 91–110, 2004.
[15] H. Zhang, H. Wang, Y. Cao, C. Shen, and Y. Li, “Robust data hiding using inverse gradient attention,” 2020, arXiv:2011.10850.
[16] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explainable and harnessing adversarial examples,” 2014, arXiv:1412.6572.
[17] A. Kurakin, I. J. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” in Adversarial Examples in the Physical World, Boca Raton, FL, USA: Chapman & Hall/CRC, 2018, pp. 99–112.
[18] G. Gul and F. Kurugollu, “A new methodology in steganalysis: Breaking highly undetectable steganography (HUGO),” in Proc. 13th Int. Workshop Inf. Hiding, vol. 6938, May 2011, pp. 71–84.
[19] V. Holub and J. Fini, “Designing steganographic distortion using directional filters,” in Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS), Dec. 2012, pp. 234–239.
[20] V. Holub, J. Fridrich, and T. Denemark, “Universal distortion function for steganography in an arbitrary domain,” EURASIP J. Inf. Secur., vol. 2014, no. 1, Dec. 2014.
[21] I. Hamamoto and M. Kawamura, “Neural watermarking method including an attack simulator against rotation and compression attacks,” IEICE Trans. Inf. Syst., vol. 103, no. 1, pp. 33–41, 2020.
[22] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” Int. J. Comput. Vis., vol. 60, no. 2, pp. 91–110, 2004.
[23] K. Alex Zheng, A. Cuesta-Infante, L. Xu, and K. Veeramachaneni, “SteganoGAN: High capacity image steganography with GANs,” 2019, arXiv:1901.03892.
[24] M. Azajosky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 214–223.
[25] M. Plata and P. Syga, “Robust spatial-spread deep neural image watermarking,” in Proc. IEEE 19th Int. Conf. Trust. Secur. Privacy Comput. Commun. (TrustCom), Dec. 2020, pp. 62–70.
[26] M. Plata and P. Syga, “Robust watermarking with double detector-discriminator approach,” 2020, arXiv:2006.03921.
[27] K. Wang, L. Li, T. Luo, and C.-C. Chang, “Deep neural network watermarking based on texture analysis,” in Artificial Intelligence and Security, X. Sun, J. Wang, and E. Bertino, Eds: Singapore: Springer, 2020, pp. 555–569.
[28] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[29] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4770–4778.
[30] M. Tancik, B. Mildenhall, D. Mar, H. Phong, N. Max, and J. T. Dunning, “NeRF: Hidden: Invertible hidden layers in neural networks,” 2020, arXiv:2003.08934.
[31] C.-C. Chang, “Neural reversible steganography with long short-term memory,” Secur. Commun. Netw., vol. 2014, no. 1, Dec. 2014.
[32] B. Boehm, “StegExpose—A tool for detecting LSB steganography,” 2014, arXiv:1410.6650.
[33] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang, “CNN-based adversarial embedding for image steganography,” IEEE Trans. Inf. Forensics Security, vol. 14, no. 8, pp. 2074–2087, Aug. 2019.
[34] B. Chen, J. Wang, Y. Chen, Z. Jin, H. J. Shim, and Y.-Q. Shi, “High-capacity robust image steganography via adversarial network,” KSII Trans. Internet Inf. Syst., vol. 14, no. 1, pp. 366–381, 2020.
[35] R. Zhang, S. Dong, and J. Liu, “Invisible steganography via generative adversarial networks,” Multimedia Tools Appl., vol. 78, no. 7, pp. 8559–8575, Apr. 2019.
[36] C. Li, Y. Jiang, and M. Chesley, “Embedding image through generated intermediate medium using deep convolutional generative adversarial network,” Comput. Mater. Continua, vol. 60, no. 2, pp. 313–324, 2018.
[37] J. Qin, J. Wang, Y. Tan, H. Huang, X. Xiang, and Z. He, “Coverless image steganography based on generative adversarial network,” Mathem. Inform. Sci., vol. 86, pp. 1394–1400, Aug. 2020.
[38] J.-Y. Zhu, T. Park, P. Isola, and A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proc. IEEE Int. Conf. Comput. Vis. Oct. 2020, pp. 2223–2232.
[39] R. Rahim et al., “End-to-end trained CNN encoder–decoder networks for image steganography,” in Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops, 2018, pp. 1–6.
[40] R. Meng, Q. Cui, Z. Zhou, Z. Fu, and X. Sun, “A steganography algorithm based on CycleGAN for covert communication in the Internet of Things,” IEEE Access, vol. 7, pp. 90874–90884, 2019.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
I. Yoo et al., “Deep 3D-to-2D watermarking: Embedding messages in 3D meshes and extracting them from 2D renderings,” 2021, arXiv:2104.13450.

J.-Y. Liu, Y.-H. Chen, Y.-C. Yeh, and Y.-H. Yang, “Unconditional audio generation with generative adversarial networks and cycle regularization,” 2020, arXiv:2005.08526.

S. Ji, J. Luo, and X. Yang, “A comprehensive survey on deep music generation: Multi-level representations, algorithms, evaluations, and future directions,” 2020, arXiv:2011.06801.

D. I. Adelani, H. Mai, F. Fang, H. H. Nguyen, J. Yamagishi, and S. Abdelnabi and M. Fritz, “Adversarial watermarking transformer: Towards tracing text provenance with data hiding,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2021, pp. 121–140.

B. Darvish Rouhani, H. Chen, and F. Koushanfar, “DeepSigns: A generic watermarking framework for IP protection of deep learning models,” 2018, arXiv:1804.00750.

D. V olkhonskiy, I. Nazarov, and E. Burnaev, “Steganographic generative adversarial networks,” in Proc. 12th Int. Conf. Mach. Vis. (ICMV), Jan. 2020, Art. no. 114333.

I. Echizen, “Generating sentiment-preserving fake online reviews using neural language models and their human- and machine-based detection,” in Proc. Adv. Inf. Netw. Appl., 34th Int. Conf. Adv. Inf. Netw. Appl. (AINA-2020), Cham, Switzerland: Springer, 2020, pp. 1341–1354.

X. Luo, Y. Li, H. Chang, C. Liu, P. Milanfar, and F. Yang, “Robust invisible video watermarking with attention,” 2021, arXiv:2104.12734.

Zihan Wang is currently pursuing the Ph.D. degree with The University of Queensland, Brisbane, QLD, Australia. He is a Visiting Researcher with CSIRO’s Data61, Marsfield, NSW, Australia. His current research interests include artificial intelligence (AI), cybersecurity, software engineering, and computer vision.

Olivia Byrnes is currently pursuing the bachelor’s degree with The University of Adelaide, Adelaide, SA, Australia. Her research interests include artificial intelligence (AI) and cybersecurity.

Ruoxi Sun is currently pursuing the Ph.D. degree with The University of Adelaide, Adelaide, SA, Australia. He is a Research Fellow with CSIRO’s Data61, Marsfield, NSW, Australia. His work in the cybersecurity and privacy domains has led to the publication of over ten papers in leading conferences and journals, such as the IEEE Symposium on Security and Privacy (S&P), ACM Conference on Computer and Communications Security (CCS), Network and Distributed System Security Symposium (NDSS), and the Web Conference (WWW), International Conference on Software Engineering (ICSE), The ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (FSE), The IEEE/ACM Automated Software Engineering Conference (ASE), Annual Conference on Neural Information Processing Systems (NeurIPS), and others. His research interests include mobile security and privacy, the INTERNET OF THINGS JOURNAL (IoT) security, and machine learning security.

Olivia Byrnes is currently pursuing the bachelor’s degree with The University of Adelaide, Adelaide, SA, Australia. Her research interests include artificial intelligence (AI) and cybersecurity.

Qi Wu was a Discovery Early Career Researcher Award (DECRA) Fellow with The University of Adelaide (UoA), Adelaide, SA, Australia, where he is currently a Senior Lecturer and also the Director of vision-and-language methods with the Australian Institute for Machine Learning (AIML). He has been in the computer vision field for ten years, and he has a strong track record, having pioneered the field of vision-and-language, one of the most interesting and technically challenging areas of computer vision.

Dr. Sun is currently a researcher with CSIRO’s Data61, Marsfield, NSW, Australia. He is also an Honorary Lecturer with the Macquarie University, Sydney, NSW, Australia. His current research interests include machine learning security and privacy, system security and privacy, and Internet measurement.

Zihan Wang is currently pursuing the Ph.D. degree with The University of Queensland, Brisbane, QLD, Australia. He is a Visiting Researcher with CSIRO’s Data61, Marsfield, NSW, Australia. His current research interests include artificial intelligence (AI), cybersecurity, software engineering, and computer vision.

Olivia Byrnes is currently pursuing the bachelor’s degree with The University of Adelaide, Adelaide, SA, Australia. Her research interests include artificial intelligence (AI) and cybersecurity.

Qi Wu was a Discovery Early Career Researcher Award (DECRA) Fellow with The University of Adelaide (UoA), Adelaide, SA, Australia, where he is currently a Senior Lecturer and also the Director of vision-and-language methods with the Australian Institute for Machine Learning (AIML). He has been in the computer vision field for ten years, and he has a strong track record, having pioneered the field of vision-and-language, one of the most interesting and technically challenging areas of computer vision.

Dr. Wu has made breakthroughs in methods and conceptual understanding to advance the field and is recognized as an international leader in the discipline. His achievements have been recognized with the Australian Academy of Science J G Russel Award in 2019, one of four awards to Early Career Researcher (ECR) across Australia, and an NVIDIA Pioneer Research Award.

Olivia Byrnes is currently pursuing the bachelor’s degree with The University of Adelaide, Adelaide, SA, Australia. Her research interests include artificial intelligence (AI) and cybersecurity.

Qi Wu was a Discovery Early Career Researcher Award (DECRA) Fellow with The University of Adelaide (UoA), Adelaide, SA, Australia, where he is currently a Senior Lecturer and also the Director of vision-and-language methods with the Australian Institute for Machine Learning (AIML). He has been in the computer vision field for ten years, and he has a strong track record, having pioneered the field of vision-and-language, one of the most interesting and technically challenging areas of computer vision.

Dr. Wu has made breakthroughs in methods and conceptual understanding to advance the field and is recognized as an international leader in the discipline. His achievements have been recognized with the Australian Academy of Science J G Russel Award in 2019, one of four awards to Early Career Researcher (ECR) across Australia, and an NVIDIA Pioneer Research Award.

Olivia Byrnes is currently pursuing the bachelor’s degree with The University of Adelaide, Adelaide, SA, Australia. Her research interests include artificial intelligence (AI) and cybersecurity.

Qi Wu was a Discovery Early Career Researcher Award (DECRA) Fellow with The University of Adelaide (UoA), Adelaide, SA, Australia, where he is currently a Senior Lecturer and also the Director of vision-and-language methods with the Australian Institute for Machine Learning (AIML). He has been in the computer vision field for ten years, and he has a strong track record, having pioneered the field of vision-and-language, one of the most interesting and technically challenging areas of computer vision.

Dr. Wu has made breakthroughs in methods and conceptual understanding to advance the field and is recognized as an international leader in the discipline. His achievements have been recognized with the Australian Academy of Science J G Russel Award in 2019, one of four awards to Early Career Researcher (ECR) across Australia, and an NVIDIA Pioneer Research Award.

Qi Wu was a Discovery Early Career Researcher Award (DECRA) Fellow with The University of Adelaide (UoA), Adelaide, SA, Australia, where he is currently a Senior Lecturer and also the Director of vision-and-language methods with the Australian Institute for Machine Learning (AIML). He has been in the computer vision field for ten years, and he has a strong track record, having pioneered the field of vision-and-language, one of the most interesting and technically challenging areas of computer vision.

Dr. Wu has made breakthroughs in methods and conceptual understanding to advance the field and is recognized as an international leader in the discipline. His achievements have been recognized with the Australian Academy of Science J G Russel Award in 2019, one of four awards to Early Career Researcher (ECR) across Australia, and an NVIDIA Pioneer Research Award.

Qi Wu was a Discovery Early Career Researcher Award (DECRA) Fellow with The University of Adelaide (UoA), Adelaide, SA, Australia, where he is currently a Senior Lecturer and also the Director of vision-and-language methods with the Australian Institute for Machine Learning (AIML). He has been in the computer vision field for ten years, and he has a strong track record, having pioneered the field of vision-and-language, one of the most interesting and technically challenging areas of computer vision.

Dr. Wu has made breakthroughs in methods and conceptual understanding to advance the field and is recognized as an international leader in the discipline. His achievements have been recognized with the Australian Academy of Science J G Russel Award in 2019, one of four awards to Early Career Researcher (ECR) across Australia, and an NVIDIA Pioneer Research Award.