

Article

DNN-Based Forensic Watermark Tracking System for Realistic Content Copyright Protection

Jaehyoung Park ¹, Jihye Kim ², Jiyoun Seo ³, Sangpil Kim ³ and Jong-Hyoun Lee ^{4,*} ¹ Protocol Engineering Lab., Sejong University, Seoul 143-747, Republic of Korea² Department of Computer and Information Security, Sejong University, Seoul 143-747, Republic of Korea³ Department of Artificial Intelligence, Korea University, Seoul 143-747, Republic of Korea⁴ Department of Computer and Information Security & Convergence Engineering for Intelligent Drone, Sejong University, Seoul 143-747, Republic of Korea

* Correspondence: jonghyouk@sejong.ac.kr

Abstract: The metaverse-related content market is active and the demand for immersive content is increasing. However, there is no definition for granting copyrights to the content produced using artificial intelligence and discussions are still ongoing. We expect that the need for copyright protection for immersive content used in the metaverse environment will emerge and that related copyright protection techniques will be required. In this paper, we present the idea of 3D-to-2D watermarking so that content creators can protect the copyright of immersive content available in the metaverse environment. We propose an immersive content copyright protection using a deep neural network (DNN), a neural network composed of multiple hidden layers, and a forensic watermark.

Keywords: artificial intelligence; copyright; forensic watermark; illegal 3D content; tracking system



Citation: Park, J.; Kim, J.; Seo, J.; Kim, S.; Lee, J.-H. DNN-Based Forensic Watermark Tracking System for Realistic Content Copyright Protection. *Electronics* **2023**, *12*, 553. <https://doi.org/10.3390/electronics12030553>

Academic Editor: Yu-Chen Hu

Received: 12 December 2022

Revised: 13 January 2023

Accepted: 17 January 2023

Published: 20 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the development of information, technology, and computing technology, the metaverse, which means the digital virtual world, is attracting attention. It utilizes virtual reality (VR) and augmented reality (AR) technologies to allow users to experience various contents in virtual space.

With the recent development of virtual reality and augmented reality technologies, the metaverse-related content market is expanding and the demand for immersive content such as 3D/4D media is increasing. In addition, as rendering technology using artificial intelligence technology has developed rapidly, general users can easily model 2D content into 3D models and create content through rendering technology. As a result, content authors can easily create immersive content, but there are concerns about copyright disputes and infringement of immersive content because there is currently no clear copyright definition for artificial intelligence creations.

In February 2022, in the United States, there was a case in which it was ruled that content created by artificial intelligence cannot be copyrighted because there is no human intervention [1] and, in September 2022, there was a discussion about whether content created by artificial intelligence can be copyrighted; copyright protection is for such content is in progress [2]. If the discussion on granting copyright in the production of artificial intelligence-based immersive content is completed, it is expected that the development of copyright protection technology suitable for the metaverse environment will be necessary.

In this paper, an immersive content protection system using a DNN and a forensic watermark is proposed to prevent illegal copying and the distribution of content produced with artificial intelligence [3–6] and rendering technology. Note that a DNN means a neural network used for deep learning through a multi-layer perceptron structure based on a data input layer, an output layer, and a hidden layer.

In the proposed system, when the immersive content created through the 3D mesh and rendering technology is illegally distributed or copied and used, a user other than the 3D mesh can extract a forensic watermark from the immersive content in use. In addition, through the user information stored in the forensic watermark, it is possible to track the suspect who distributed the illegal realistic content on the Internet.

In this paper, which is an extension of the paper will be published in [7], we propose ideas for building a safe metaverse environment and developing the immersive content market. Section 2 explains the background knowledge of the technologies used in the proposed system and Section 3 explains the structure and operating process of the proposed system. Section 4 analyzes the performance of the proposed system utilizing an existing one [8] and discusses the pros and cons of the system. Finally, Section 5 concludes this paper.

2. Related Work

2.1. Forensic Watermark

A watermark means a text or image that contains the information of the copyright holder and is inserted into content such as an image, video, or audio. The watermark can prove the ownership of the content by the copyright holder who created it and can protect the copyright holder from forgery and falsification. The method for verifying ownership through a watermark is divided into a private marking method that requires original data and a public marking method that does not require original data. In addition, the embedding method can be divided into a visible watermarking method that directly inserts into content and an invisible watermarking method that inserts a hidden message into the content.

Since the general watermarking method proves ownership by inserting and recording only the information of the copyright holder in the content, it has a limitation in that it is difficult to manage whether the copyrighted content is used [9]. Accordingly, since the copyright holder information inserted into the content through the watermark does not include the content user's transaction history and distribution route, illegal content distributors cannot be tracked. To solve this problem, a forensic watermarking method that inserts not only copyright holder information but also content transaction details and the distribution route into the watermark has begun to be used.

Forensic watermarking can protect the copyright of the copyright holder and identify the user of the content by inserting the content transaction history and distribution route into content such as images, video, and audio through a specific algorithm [10]. In the case of a forensic watermark, by inserting a forensic watermark in real-time while distributing the content, information on the content's user can be hidden and users who illegally copy or distribute the content can be tracked or ownership can be verified [9]. This is also called fingerprinting and is used to track illegal copying and distribution of content. The forensic watermarks can include user information such as the ID and resident registration number of the current user, including the copyright holder, first purchaser, and last purchaser, content upload and download time information, and content distribution route [11].

The forensic watermark (FW) inserts a forensic watermark containing copyright information into the embedded content when the user normally purchases the original content of the copyright holder. First, before the user purchases the content, the content must be encrypted to prevent copyright infringement and the illegal copying of content. To this end, a DRM system that manages the rights of digital content is utilized. In the DRM system, the encryption module uses the encryption key to encrypt the content of the copyright holder, transmits the encrypted content to the content distribution server, and transmits the encryption key used for encryption to the license server [12].

Afterward, when the user purchases content, the user information containing personal information, such as a user ID, is transmitted to the content distribution server, requesting a content download. The content distribution server delivers the requested content to the FW embedder. The content at this time is the encrypted content of the copyright

holder and the watermark is not inserted yet. Since the user needs a license to use the downloaded content, the content distribution server requests a necessary license from the user by transferring the user information to the license server. Values such as the license sequence number, the requested content ID, and the key used to encrypt the content are stored in the license [13]. The license server transfers the license to the FW embedded so that the user can use the content later. Next, the copyright server requests FW generation by transmitting user information to the FW server. The FW generator of the FW server creates the FW corresponding to User Info and the created FW is stored in the FW database. The generated FW is transmitted to the FW embedder and the FW embedder uses the license received encryption key to decrypt the encrypted content through the decryption module and then inserts the FW into the content through the embedding module. Through this process, the user will use the content containing their information.

Next, when someone illegally distributes the content, to track the illegal distributor, the copyright information inserted in the content must be extracted and analyzed and then the distributor must be tracked. To perform this, the web crawler roams the Internet, visits various links, fetches arbitrary content from the Internet, and sends it to the FW extractor. Web crawlers visit web pages with more linked links and explore content [14]. The FW extractor extracts the FW from the content through the extracting module and transmits it to the FW server. Next, after comparing and analyzing the FW received through the FW analyzer and the FW stored in the FW database, the analysis result is transmitted to the license server to report illegal distribution. To determine whether the content is illegally distributed, the content ID and user ID stored in the extracted FW are used and the illegal distributor can be tracked based on this [15,16]. Figure 1 below shows the forensic watermark embedding and verification procedure.

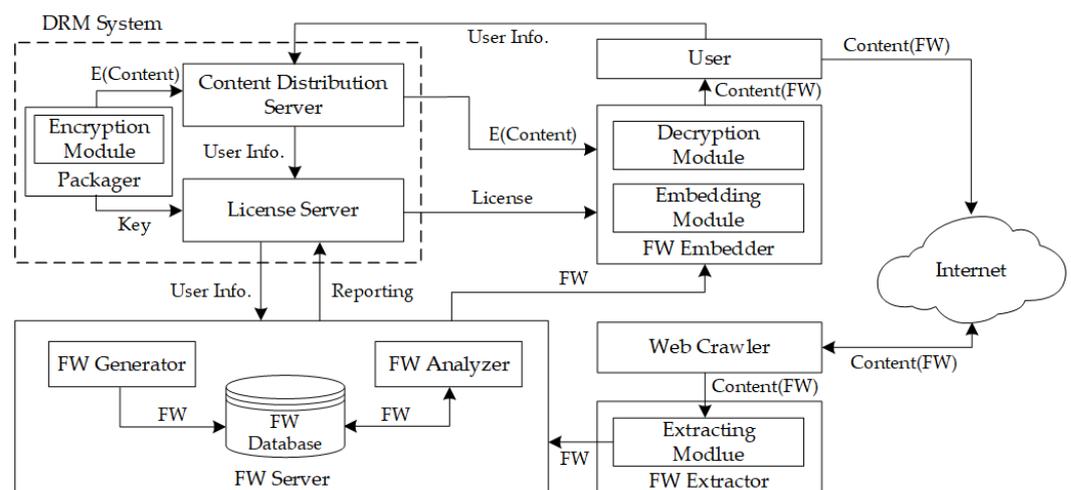


Figure 1. Forensic watermark embedding and verification procedure.

2.2. Watermark for 3D Model

3D models are used to create immersive content in fields such as AR/VR, 3D printing, games, and movies. Because of this, since the watermark for the 3D model must visually guarantee the same quality as the original content, invisibility is considered. In addition, as a general feature of the watermark, it is necessary to restore and extract the inserted watermark message as it is. At this time, even if the content is transmitted or modified, robustness must be provided so that the watermark is not removed. The same should be applied to the 3D model as well. In a general watermark, the efficiency of the watermark embedding and extraction time and the capacity of messages that can be inserted into the watermark is considered, but from the point of view of immersive content, invisibility and robustness are discussed as major requirements. In the case of robustness, the protection capability for content and watermark and the transmission accuracy may be affected. A

robust watermark will remain the same after some attacks and can provide authentication by detecting the watermark. Due to this characteristic, it is used in fields such as copyright protection, broadcasting monitoring, copy control, and fingerprinting [17]. To design a watermarking technique according to these requirements, a spatial domain or a frequency domain can be considered as the embedding domain.

2.2.1. Imperceptibility

Watermarks can be visible or invisible. A visible watermark indicates watermark ownership and is considered part of the content [18]. 3D models can be modified and adjusted by various methods such as rendering algorithms and shading algorithms and imperceptibility is considered to prevent intentional deformation by not recognizing the presence of watermarks [19]. To measure the invisibility of a watermark, peak signal to noise ratio (PSNR) and structural similarity index model (SSIM) are mainly adopted [20].

2.2.2. Robustness

Robustness is the most important requirement of most watermarking techniques and applications for copyright protection [18]. Robustness means that the watermark is not modified against intentional modification or unintentional attack by the user. However, 3D models such as polygonal meshes are capable of sophisticated and difficult-to-respond-to attacks [19]. Representative attacks can generate rotation, gaussian noise, smoothing, etc., and even attacks directed at these 3D models must be able to extract the embedded watermark [19,21]. The following is a description of representative attacks that can occur in 3D models [22].

- **Noise**
Noise means adding noise data to the vertex coordinates of the 3D model. Noise can occur during 3D model adjustments such as lossy compression or format conversion. However, in a malicious case, an attacker can weaken the watermark by inserting noise.
- **Smoothing**
Smoothing rearranges the vertices to improve the quality of the model while maintaining the topology of the model without manipulating it when deformation such as noise occurs in the 3D model.
- **Global Transformations**
Affine transforms such as transform, rotation, and non-affine transforms can be performed on the 3D model. Some watermarking techniques can extract a watermark depending on the positional coordinates and orientation of an object. In this case, if the global transform is applied to the 3D model, it may be difficult to detect the original watermark in the 3D model.
- **Cropping**
Similar to an image, geometric parts can be deleted in a 3D model. This may render the 3D model itself meaningless, but the remaining parts are still valuable and can destroy the spatial structure of the 3D model to remove the watermark.
- **Mesh simplification**
Mesh simplification is used to speed up a manipulation and rendering of 3D models. This is expressed as a 3D polygon mesh with a small number of triangles while maintaining the shape of the 3D model.

2.2.3. Embedding Domain

The embedding domain of a watermark is largely divided into a spatial domain and a frequency domain. The spatial domain approach can directly insert and extract watermark and has the advantage of being able to embed watermark more simply and quickly than the frequency domain approach. This works robustly against attacks such as cropping and noise but has the disadvantage that it is not robust against signal processing attacks such as compression. The frequency domain approach converts the original content into a

frequency signal and inserts a watermark signal [11]. The frequency domain approach has the advantage of being robust against noise, cropping, rotation, etc., by enabling most signal processing [23]. This is a method that has mainly been used recently by supplementing the disadvantages of the spatial domain approach. A discrete cosine transform (DCT) and a discrete wavelet transform (DWT) are representative signal transformation methods.

- **DCT**
DCT is one of the digital orthogonal conversion methods; it converts spatial domain data into frequency domain [11,23]. This is expressed as a sum of sine waves (sine and cosine functions) with various sizes and frequencies. The watermark is inserted by adding the watermark to the existing frequency coefficient. Representatively, the DCT compression method is used in MPEG and JPEG. It is powerful for general image processing tasks and has the advantage that, even if attacked, the watermark is not easily removed and the image pixels do not affect each other. On the other hand, it requires a large amount of calculation, is difficult to implement, and is weak against geometric deformation attacks such as scaling, rotation, and cropping [17].
- **DWT**
DWT is a transform method that decomposes a signal into wavelets. A wavelet means an oscillation that repeats increases and decreases based on 0 and is a function used as a basic function representing a signal [17]. When DWT is applied to content, it has a low–low band (LL), a high–low band (HL), a low–high band (LH), and a high–high band (HH). LL represents the coarse scale DWT coefficients and the sub bands LH, HL, and HH represent the fine scale of DWT. In this technology, it is possible to satisfy the characteristics of imperceptibility and robustness by first embedding a watermark in LL [24]. This has the advantage of being robust even when attacks such as watermark compression and cropping are applied but has the disadvantage that the compression time is long and noise shapes may occur.

Figure 2 below shows the watermark embedding process of the general frequency domain approach. A means original content for embedding a watermark and w means a watermark. The watermark embedding module encrypts the watermark message (w) using the key value (k) and inserts it into A . A_w denotes the content in which a watermark is inserted through the frequency domain approach.

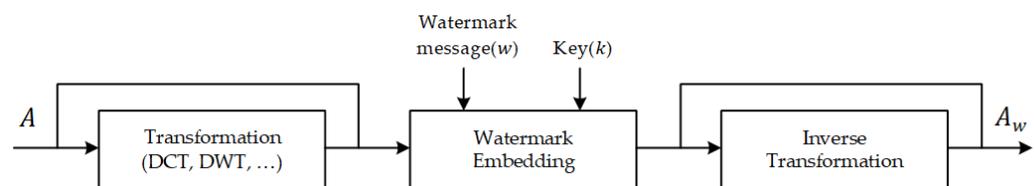


Figure 2. Frequency domain watermarking procedure.

2.3. Deep Learning-Based Watermarking

Watermark can be altered by malicious or non-malicious manipulation. Malicious manipulation is aimed at damaging or removing watermarks by users with bad intentions. Non-malicious manipulation means that the watermark is unintentionally damaged by content compression and transmission, etc. However, the watermark must be strong against normal distortion and malicious distortion. If the embedded watermark is corrupted or distorted, the watermark cannot be recovered and thus cannot provide traditional security features. Therefore, most deep learning-based watermarking technologies aim for watermark that can withstand at least fine tuning by an application [25].

Deep learning-based watermarking technology starts with a 2D image watermark. As in [26] proposed HiDDeN, a framework that can watermark digital images using deep learning. HiDDeN consists of an encoder and decoder for embedding and extracting the watermark. In addition, a noise layer was constructed to be robust against watermark attacks by simulating attacks that may occur in 2D images. Through this, a watermark

was created that satisfies the watermark requirements and can withstand image distortion without any visual difference from the original image. Since then, follow up studies to design a noise layer based on HiDDeN or to extend the framework have emerged. As in [27] proposed ReDMark, which adopts an approach of stochastically specifying the image attack type at every iteration in a similar way to HiDDeN. This approach has been shown to be effective in achieving reasonable watermark robustness. As in [28] to deceive the DNN, they proposed a universal secret adversarial perturbation that can attack and hide at the same time by utilizing universal adversarial perturbation, steganography, and universal secret perturbation to cause it to be faster. This involves inducing convergence and performance improvement. Furthermore, ref. [29] proposed a new framework using an adversarial training model and channel coding to complement the robustness of existing methods against unknown attacks. Deep learning-based watermarking techniques generally consist of an encoder, decoder, and attack layer. The structure is shown in Figure 3 below.

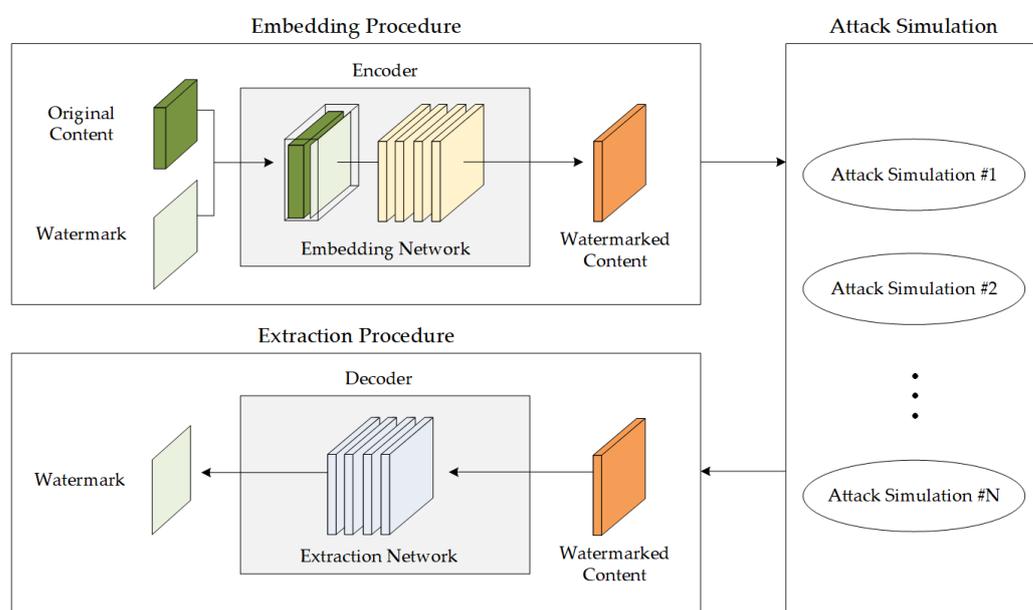


Figure 3. Deep learning watermark embedding and extracting procedure.

The encoder and decoder are responsible for embedding and extracting the watermark using deep learning and the attack layer simulates possible attacks on the 3D model, such as noise and smoothing. Through this, it is possible to ensure the robustness of the watermark and satisfy the visual quality for copyright protection of the contents.

3. Proposed DNN-Based Forensic Watermark Tracking System

In this section, we propose a DNN-based forensic watermark tracking system to prevent the illegal distribution of 3D content created based on artificial intelligence. The system utilizes DNN technology to construct a forensic watermark insertion and extraction process and prevent watermarks from being altered by artificial intelligence algorithms. It also adds a layer of attack between the forensic watermark insertion process to ensure that the forensic watermark is robust against both malicious and non-malicious distortion. Therefore, even if 3D contents with forensic watermarks are unintentionally modified while being used by ordinary users or by malicious users on intentionally attack pirated copies, the unmodified forensic watermarks are extracted from the 3D contents. Forensic watermarks are used to track piracy and distributors of 3D content. To this end, during the forensic watermark insertion process, the forensic watermark is inserted to include transaction details (e.g., buyer, purchase details, purchase price, etc.) for each content ID. Afterward, a strong uncensored watermark is extracted from the 3D content that is illegally reported or monitored and, based on the content ID, the user transaction history

is retrieved to track the users who have copied or distributed the content illegally. This system is designed with the coming metaverse environment in mind and is a tracking model for determining and protecting from copyright infringement of 3D content. The system was inspired by the 3D-to-2D watermarking technology proposed by I. Yoo. the components of their thesis were referred to [8] and the overall structure of the proposed system is shown in Figure 4. The proposed system consists of a total of 11 modules and the functions of each module are shown in Table 1 below.

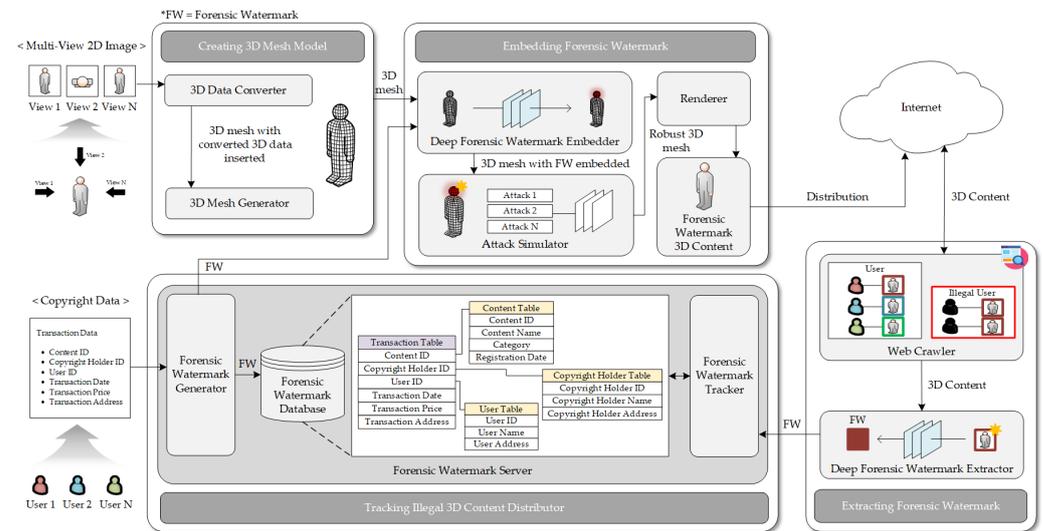


Figure 4. Proposed architecture of the DNN-based forensic watermark tracking system.

Table 1. System modules.

Module	Description
3D Data Converter	A module that converts multi-view 2D Images into 3D data
3D Mesh Generator	A module that creates 3D mesh based on 3D data
Forensic Watermark Generator	A module that creates a forensic watermark based on the user’s copyright data
Deep Forensic Watermark Embedder	A module that embeds forensic watermark into 3D mesh
Attack Simulator	A module that trains to respond to attacks of 3D mesh
Renderer	A module that creates 3D content by rendering based on a robust 3D mesh
Forensic Watermark Server	A server that creates, stores, and analyzes a forensic watermark to track illegal users
Forensic Watermark Database	A database that stores a forensic watermark
Web Crawler	A module that monitoring 3D contents in Internet
Deep Forensic Watermark Extractor	A module that extracts a forensic watermark on 3D mesh
Forensic Watermark Tracker	A module that analyzes and tracks a forensic watermark

Based on the modules defined in Table 1, the proposed system is described by two operational processes. The first is a procedure for inserting a forensic watermark with guaranteed robustness and the second is a procedure for extracting the forensic watermark

from illegally distributed 3D content and tracking an illegal distributor. Table 2 is a notation for expressing the flow of messages between the modules in each procedure.

Table 2. Used notations.

Notation	Description
FW	Forensic Watermark
C_{ID}	Content ID
U_{ID}	User ID
$Image_{2D}$	2D Image
$User_{U_{ID}}$	User with User ID
$3D_{Mesh}$	3D Mesh Data to Create 3D Content
$3D_{Mesh}(FW)$	3D Mesh with Forensic Watermark
$3D_{Mesh}_{Robust}(FW)$	Robust 3D Mesh with Forensic Watermark
$3D_{Content}(FW)$	3D Content with Forensic Watermark
$Gen_A(x, y)$	Generate A using x, y
$Ins_A(x)$	Insert x into A
$Ext_A(x)$	Extract A from x
$Req(x)$	Request x
$Res(x)$	Response x
$Exist_x$	Exist x
$Rendering(x)$	Rendering x
$Monitoring(x)$	Monitoring x
$Analysis_A(x)$	Analyze the A using x
$Tracking_A(x)$	Tracking the A using x
$Training_A(x)$	Training the A for x

3.1. Forensic Watermark Embedding Procedure

To create 3D content with forensic watermarking, the forensic watermark generator creates information templates related to purchasing 3D content. The template includes the purchased 3D content ID, user ID, copyright holder ID, etc., and records the transaction date and transaction price. Based on this, a forensic watermark is created through a forensic watermark generator and the created forensic watermark is transmitted to the forensic watermark database. A forensic watermark is a textual representation of a content ID. Forensic watermarks are stored and managed in the form of a table by the forensic watermark database and are later used to detect illegal 3D content. Furthermore, the generated forensic watermark is transferred as an input to the deep forensic watermark embedder. A deep forensic watermark embedder trains the embedding procedure and assigns weights to the embedding procedure to embed the forensic watermark. Next, after inserting the trained forensic watermark into the 3D mesh, robustness is added by training the attack process. 3D content is then created by rendering the 3D mesh. Figure 5 below shows the process of embedding a DNN-based forensic watermark in 3D content.

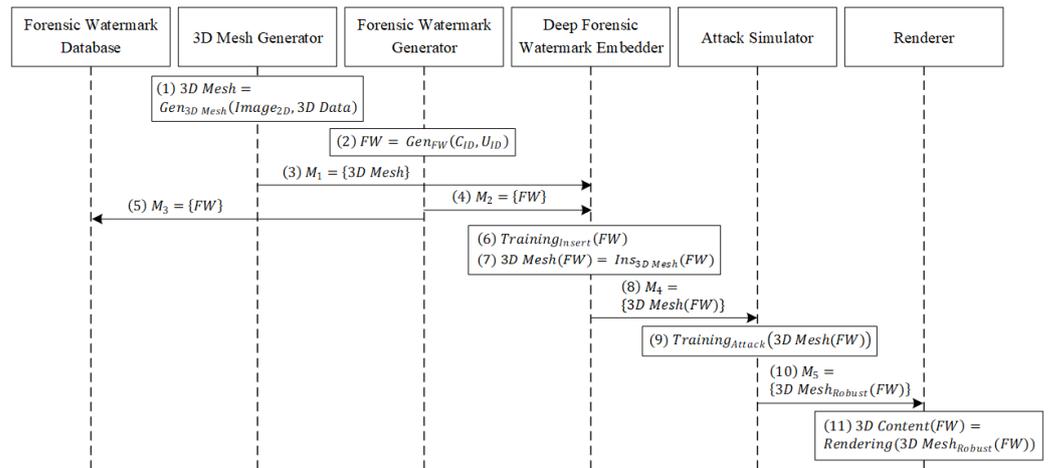


Figure 5. DNN-based forensic watermark embedding procedure on the proposed system.

1. A 3D mesh generator creates a 3D mesh using 2D images and 3D data.
2. A forensic watermark generator creates a forensic watermark using a content ID and a user ID.
3. The 3D mesh generator transfers the generated 3D mesh to a deep forensic watermark embedder.
4. The forensic watermark generator transfers the forensic watermark to the deep forensic watermark embedder.
5. The forensic watermark generator stores the generated forensic watermark in a forensic watermark database.
6. The deep forensic watermark embedder trains a procedure to embed the forensic watermark.
7. The deep forensic watermark embedder embeds the forensic watermark into the 3D mesh.
8. The deep forensic watermark embedder transfers the 3D mesh with the forensic watermark to an attack simulator.
9. The attack simulator trains possible attacks on the 3D mesh.
10. The attack simulator provides a the robust 3D mesh to a renderer.
11. The renderer creates a 3D content by robustly rendering the 3D mesh.

3.1.1. Forensic Watermark Embedding

For forensic watermark embedding of the system proposed in this paper, the deep forensic watermark embedder utilizes I. Yoo’s encoder [8] to embed messages into the vertex components of the 3D mesh. In the deep forensic watermark embedder, the secret message, which is a forensic watermark, is replicated N_v times and $N_v \times FW_b$ tensors are constructed according to the dimension of the vertex. Here, N_v means the number of vertices in the 3D mesh and FW_b means the binary message inserted as a forensic watermark. A 3D mesh is expressed as $M(V, F, T, P)$ and the vertices are expressed as $V \in R^{N_v \times C_v}$. C_v refers to factors such as 3D positions, normal, and vertex colors. We define $V_m \in R^{N_v \times (C_v + FW_b)}$ by concatenating the vertices with the tensor composed of a watermarked vertex. V_e is obtained by configuring $E_G(V_m) \in R^{N_v \times C_v}$, which is a vertex message embedding neural network. In the proposed system, Equation (1) below can be expressed through $Training_{insert}(FW)$ and $Ins_{3DMesh}(FW)$. Figure 6 below shows the DNN-based forensic watermark embedding architecture.

$$V_e = E_G(V_m) \in R^{N_v \times C_v} \tag{1}$$

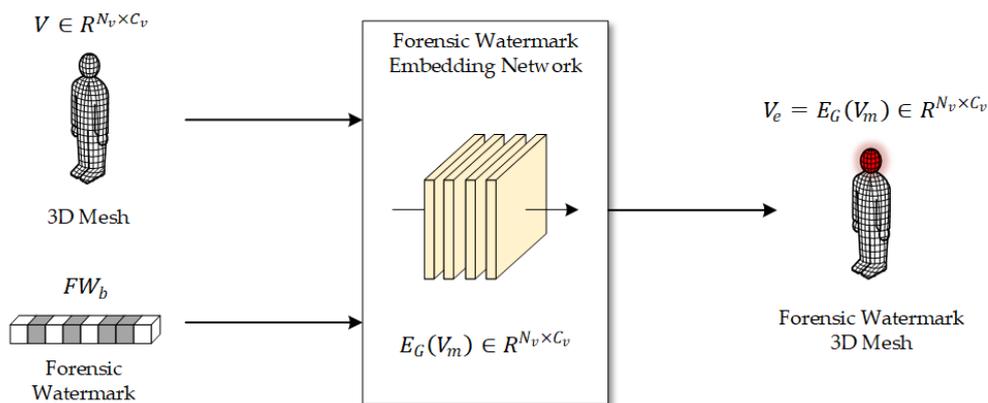


Figure 6. DNN-based forensic watermark embedding architecture.

3.1.2. Attack Simulation and Rendering

The attack simulator iterates and simulates the procedure of performing and extracting possible attacks from a 3D mesh to create a forensic watermarked 3D mesh while ensuring robustness against forensic watermark. We consider gaussian noise, rotation, random scaling, and 3D vertex cropping as simulated attacks and add them to the forensic watermark embedding pipeline. To create 3D content, the 3D mesh is used as an input to the renderer and, finally, 3D content with forensic watermark is created. This corresponds to the $Training_{Attack}(3DMesh(FW))$ procedure in the proposed system.

Additionally, a differentiable rendering layer is required to extract forensic watermark from 2D rendered image for immersive content. For immersive content, a rendered 2D image is generated through 3D mesh M , lighting camera matrix K , and lighting parameter L . This means the $Rendering(3DMesh_{Robust}(FW))$ procedure. In this case, H_r in Equation (2) means the height of the rendered image and W_r means the width of the rendered image. Figure 7 shows an architecture that ensures the robustness of forensic watermark.

$$I = R_D(M, K, L) \in \mathbb{R}^{H_r \times W_r \times 3} \tag{2}$$

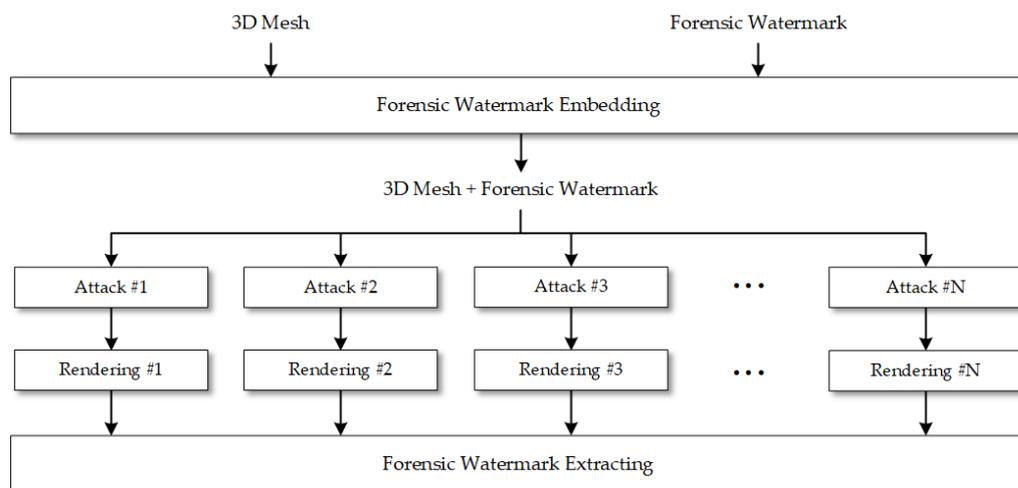


Figure 7. DNN-based forensic watermark attack simulation architecture.

3.2. Track Down Suspects of Illegal 3D Content Procedure

Illegal 3D content can be flagged by others or monitored by web crawlers that monitor 3D content on the Internet. When illegal 3D content distribution is detected in the 3D content randomly fetched by the web crawler, the system’s process of tracking the distributor is initiated. If the illegal 3D content is detected through monitoring, the detected illegal

3D content is transmitted to the forensic watermark extractor. The forensic watermark extractor extracts the forensic watermark and transfers it to the forensic watermark tracker. Next, the forensic watermark tracker tracks the illegal 3D content based on the extracted forensic watermark. Figure 8 below shows the process of tracking illegal 3D content based on forensic watermarking.

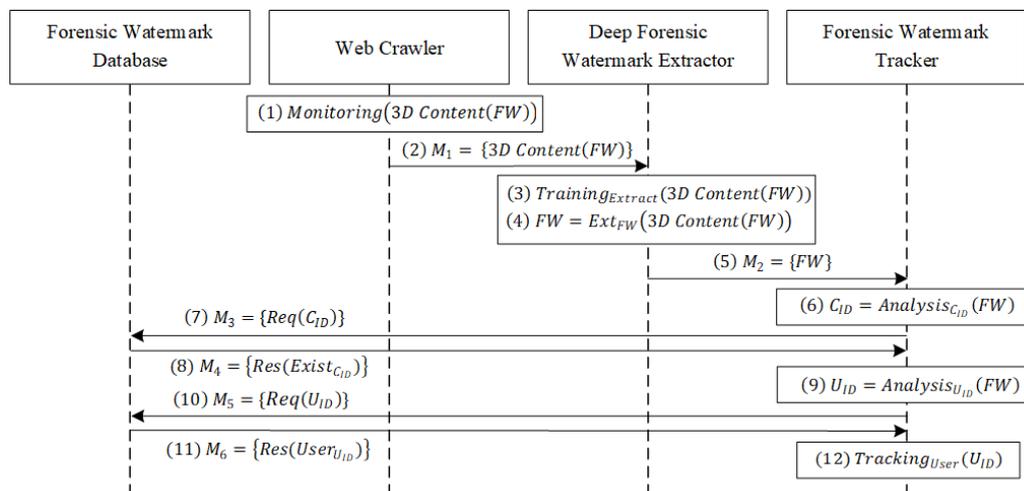


Figure 8. Illegal 3D content distributor tracking procedure on the proposed system.

1. A web crawler is monitoring 3D content with a forensic watermark.
2. The web crawler randomly transfers the 3D content uploaded on the Internet to a deep forensic watermark extractor for verification.
3. The deep forensic watermark extractor trains a forensic watermark extraction procedure.
4. The deep forensic watermark extractor extracts the forensic watermark from the 3D content.
5. The deep forensic watermark extractor transfers the forensic watermark to a forensic watermark tracker.
6. The forensic watermark tracker analyzes a content ID using the forensic watermarks.
7. The forensic watermark tracker requests the content ID to a forensic watermark database.
8. When the content ID in the forensic watermark database is the same as the requested content ID, a response regarding whether the requested content ID exists is transferred to the forensic watermark tracker.
9. The forensic watermark tracker, receiving the response about the content ID, analyzes a user ID using the forensic watermark.
10. The forensic watermark tracker requests the user ID from the forensic watermark database.
11. The forensic watermark database transfers user information corresponding to the user ID to the forensic watermark tracker as a response.
12. The forensic watermark tracker tracks users using the response from the forensic watermark database.

3.2.1. Forensic Watermark Extraction

For the forensic watermark extraction of the system proposed in this paper, the deep forensic watermark extractor extracts the forensic watermark from the 2D rendered image using I. Yoo’s decoder [8]. The deep forensic watermark extractor extracts message M_r from the 2D rendered image I using neural network D and is expressed as Equation (3). Here, neural network D uses multiple convolution layers and global pooling layers for input 2D images of various sizes. The final message bit M_{rb} is extracted through Equation

(4) and M_r can be used to calculate the message loss and M_{rb} to calculate the bit accuracy. Figure 9 below shows the DNN-based forensic watermark extraction architecture.

$$M_r = D(I) \quad (3)$$

$$M_{rb} = \text{clamp}(\text{sign}(M_r - 0.5), 0.1) \quad (4)$$

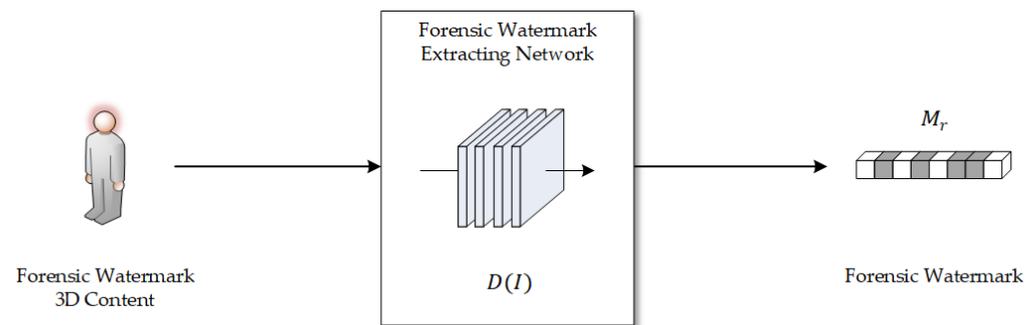


Figure 9. DNN-based forensic watermark extraction architecture.

3.2.2. Forensic Watermark-Based Tracking

Figure 10 is the process of tracking users who illegally distributed content using forensic watermarks. The information contained in forensic watermarks can be utilized to track users. The forensic watermark consists of a transaction table containing content transaction information and three master tables containing information on the content, copyright holders, and users included in the transaction information. In the case of the transaction table, it has information on content ID, copyright holder ID, user ID, transaction date, transaction price, and transaction address. When a user purchases content, the content purchased, the copyright holder of the content, the user who purchased the content, and the date, price, and location of the purchase are recorded. In the case of the content table, the content information is listed in detail in connection with the content ID in the transaction table. The content information is organized by content ID, content name, category, and registration date. It also organizes the name of each content and the field of the content by category and records the date the content was registered as a copyright. In the case of the copyright holder table, the copyright holder information is listed in detail in connection with the copyright holder ID in the transaction table. The copyright holder information records the copyright holder's ID, name, and address. In the case of the user table, detailed user information is listed in connection with the user ID in the transaction table. The user information records the user ID, username, and address.

When the forensic watermark is inserted, the user can use the content in which the transaction table information is inserted. At this time, if a user who normally purchases and downloads content illegally copies and uploads or distributes content on the Internet, this is regarded as copyright infringement. When content is copied and distributed illegally, the illegal distributor can be tracked by utilizing the content ID and user ID among the forensic watermark information. First, a web crawler visits various sites on the Internet and fetches arbitrary content to verify forensic watermarks. Web crawler usually browse content by visiting sites with many links, such as illegal sites. The content subject to randomly search by the web crawler is delivered to the forensic watermark extractor and the forensic watermark extractor extracts the forensic watermark of the corresponding content. By comparing the content ID stored in the extracted forensic watermark with the content ID stored in the forensic watermark database, it is determined whether the currently uploaded content is the content already registered in the database. If the extracted content ID is an ID that exists in the database, it is regarded as a case of illegally copying and re-uploading content for which copyright has already been registered and it is possible

to determine that the content is illegally copied content. Next, to track the user who illegally copied and distributed the content, the user ID stored in the forensic watermark is used. The corresponding user ID can be retrieved from the database to identify the user that purchased or downloaded the content last; a suspect who copied and distributed illegal content can be identified by searching for the user information in the user table mapped with the user ID.

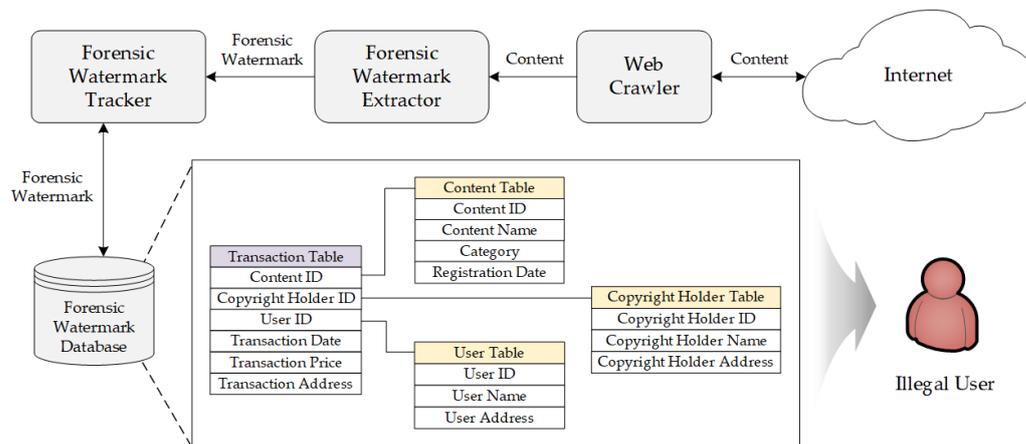


Figure 10. Illegal user tracking procedure.

4. Discussion

This paper aims to track illegal piracy and the illegal distribution of immersive content using forensic watermarking. The paper proposed software that has not been developed and is in the stage of presenting an idea. Several papers on 3D watermarking have been proposed, but 3D-to-2D watermarking is the first to embed a watermark in a 3D model and extract the watermark from a rendered 2D image accessible from a real application or user perspective. Therefore, since this study intends to utilize I. Yoo’s 3D-to-2D watermarking method [8], its performance is summarized and provided. In I. Yoo’s experiment [8], a ModelNet 40 data set was used to evaluate the performance of the encoder, decoder, and distortion. The input mesh has parameters of 5000 vertices, 5000 faces, 5 vertex elements, and 3 mesh indexes. For the evaluation of the encoder, PointNet and fully convolutional PointNet (PointNet v2) were used to measure the best accuracy, average (μ), and standard deviation (σ) of bit accuracy and Normal and Texcoord were measured for the difference in geometry L_1 . Finally, PSNR and SSIM of the rendered 2D image were measured. The 3D-to-2D watermark encoder performance measurement results are provided in Table 3. In the decoder, four-layer Simple CNN, Residual CNN, and HiDDeN, a deep learning based image watermarking technology, were compared in terms of bit accuracy. The 3D-to-2D watermark decoder performance measurement results are provided in Table 4. As an evaluation of distortion, a well-known distortion was applied and the bit accuracy was measured and shown. This means robustness against attacks, and is shown in Table 5.

Table 3. 3D-to-2D watermark encoder performance.

Architecture	Bit Accuracy			Geometry L_1 Diff		Rendered Image	
	Best	μ	σ	Normal	Texcoord	PSNR	SSIM
PointNet	0.6837	0.6151	0.0366	0.1873	0.0546	28.47	0.9563
PointNet v2	0.6616	0.6255	0.0216	0.1506	0.0430	28.83	0.9557

Table 4. 3D-to-2D watermark decoder performance.

Architecture	Bit Accuracy		
	Best	μ	σ
Simple CNN (four-layers)	0.6396	0.6148	0.0415
Residual CNN (four-layers)	0.7390	0.4538	0.0332
CNN (HiDDeN)	0.8269	0.7614	0.0411

Table 5. Effect on distortions.

Distortion Type	Bit Accuracy
No Distortion	0.9046
Noise ($\sigma = 0.01$)	0.9036
Rotation ($\pm\pi/6$)	0.9028
Scaling (< 25%)	0.8953
Cropping (< 20%)	0.8840

As far as we know, papers on technologies that can insert and extract watermarks in 3D models have been presented, but papers presenting systems or ideas using extracted watermarks are insufficient. In the future, it seems that discussions on methodologies that can utilize 3D watermarks should be continued. Since the idea we propose extracts a forensic watermark from a rendered 2D image that can be used for immersive content, there is a possibility that it can be used for the copyright protection of immersive content such as AR/VR. In addition, by tracking a suspect who illegally copies and distributes immersive content through forensic watermarking, it is possible to contribute to copyright protection technology by preventing illegal acts on immersive content. However, the currently proposed system needs to check the requirements and detailed design requirements for applying the technology and forensic watermark as a technology for extracting a watermark from a 2D rendered image that has recently been developed. In addition, for the verification of the proposed system, the performance should be measured through actual developments and analyzed by comparing with the existing deep learning-based watermarking system. To this end, future research plans to produce actual implementations of the proposed system.

5. Conclusions

As the current metaverse industry develops and the demand for 3D content increases, it is expected that copyright disputes over 3D content will arise. Therefore, in this paper, we propose a system that can track suspects of illegal 3D content distribution by using DNN and forensic watermark technology to build a fair metaverse environment that is safe from copyright. Through the proposed system, forensic watermarks with guaranteed robustness can be inserted into 3D content and illegal suspects can be tracked through the inserted forensic watermarks. In future research, we plan to implement the proposed system to track the actual illegal 3D content suspects by supplementing them.

Author Contributions: J.P. and J.K. wrote most of this paper using ideas from their research on DNNs and forensic watermarks. J.S. and S.K. reviewed this thesis and presented their opinions on 3D content watermarking. J.-H.L. revised the paper and supervised all work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2022. (Project Name: 4D Content Generation and Copyright Protection with Artificial Intelligence, Project Number: R2022020068).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. U.S. Copyright Off. Re: Second Request for Reconsideration for Refusal to Register A Recent Entrance to Paradise (Correspondence ID 13ZPC6C3; SR # 17100387071). Available online: <https://www.copyright.gov/rulings-filings/review-board/docs/a-recent-entrance-to-paradise.pdf> (accessed on 7 January 2023).
2. AI-Generated Artwork Is Copyrighted for the First Time. Available online: <https://petapixel.com/2022/09/27/ai-generated-artwork-is-copyrighted-for-the-first-time/> (accessed on 7 January 2023).
3. Madani, M.; Lin K.; Tarakanova, A. DSResSol: A sequence-based solubility predictor created with Dilated Squeeze Excitation Residual Networks. *Int. J. Mol. Sci.* **2021**, *22*, 13555. [CrossRef] [PubMed]
4. Roshani, G.H.; Hannus, R.; Khazaei, A.; Zych, M.; Nazemi, E.; Mosorov, V. Density and velocity determination for single-phase flow based on radiotracer technique and neural networks. *Flow Meas. Instrum.* **2018**, *61*, 9–14. [CrossRef]
5. Mahmoodi, F.; Darvishi, P.; Vaferi, B. Prediction of coefficients of the Langmuir adsorption isotherm using various artificial intelligence (AI) techniques. *J. Iran. Chem. Soc.* **2018**, *15*, 2747–2757. [CrossRef]
6. Roshani, G.H.; Nazemi, E.; Roshani, M.M. Intelligent recognition of gas-oil-water three-phase flow regime and determination of volume fraction using radial basis function. *Flow Meas. Instrum.* **2017**, *54*, 39–45. [CrossRef]
7. Park, J.; Kim, J.; Seo, J.; Kim, S.; Lee, J.-H. Illegal 3D Content Distribution Tracking System based on DNN Forensic Watermarking. In proceedings of the 2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), Bali, Indonesia, 20–23 February 2023.
8. Yoo, I.; Chang, H.; Luo, X.; Stava, O.; Liu, C.; Milanfar, P.; Yang, F. Deep 3D-to-2D Watermarking: Embedding Messages in 3D Meshes and Extracting Them from 2D Renderings. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 10031–10040.
9. Kim, D.-H. Copyright Protection and Ownership Authentication of Video Using Watermarking. Ph.D. Thesis, Chosun University, Gwangju, Republic of Korea, 2006.
10. Singh, P.; Chadha, R.S. A survey of digital watermarking techniques, applications and attacks. *Int. J. Engi Neering. Innov. Technol.* **2013**, *2*, 165–175.
11. Liu, F.; Liu, Y. A Watermarking Algorithm for Digital Image Based on DCT and SVD. In Proceedings of the Congress on Image and Signal Processing, Sanya, China, 27–30 May 2008; pp. 27–30.
12. Obimbo, C.; Salami, B. Using digital watermarking for copyright protection. In *Watermarking*; IntechOpen: London, UK, 2012; Volume 2.
13. Zhang, J.; Li, B.; Zhao, L.; Yang, S.-Q. License management scheme with anonymous trust for digital rights management. In Proceedings of the 2005 IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 6–8 July 2005; p. 4.
14. Jang, H.W.; Kim, W.G.; Lee, S.H. An illegal contents tracing system based on web robot and fingerprinting scheme. In Proceedings of the Fifth IEEE Workshop on Mobile Computing Systems and Applications, Monterey, CA, USA, 9–10 October 2003; pp. 415–419.
15. Lee, J.-S.; Yoon, K.-S. The system integration of DRM and fingerprinting. In Proceedings of the 2006 8th International Conference Advanced Communication Technology, Pyeongchang, Republic of Korea, 20–22 February 2006; p. 2183.
16. Dustin, W.E. Circulation gatekeepers: Unbundling the platform politics of YouTube’s content ID. *Comput. Compos.* **2018**, *47*, 61–74.
17. Mahbuba, B.; Mohammad, S.U. Digital image watermarking techniques: A review. *Information* **2020**, *11*, 110.
18. Wan, W.; Wang, J.; Zhang, Y.; Li, J.; Yu, H.; Sun, J. A comprehensive survey on robust image watermarking. *Neurocomputing* **2022**, *488*, 226–247. [CrossRef]
19. Francesca, U.; Massimiliano, C.; Mauro, B. Wavelet-based blind watermarking of 3D models. In Proceedings of the 2004 Workshop on Multimedia and Security, Magdeburg, Germany, 20–21 September 2004; pp. 143–154.
20. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
21. Wang, F.; Zhou, H.; Fang, H.; Zhang, W.; Yu, N. Deep 3D mesh watermarking with self-adaptive robustness. *Cybersecurity* **2022**, *5*, 1–14. [CrossRef]
22. Bennour, J.; Dugelay, J. Toward a 3D watermarking benchmark. In Proceedings of the 2007 IEEE 9th Workshop on Multimedia Signal Processing, Crete, Greece, 1–3 October 2007; pp. 369–372.
23. Wang, B.; Ding, J.; Wen, Q.; Liao, X.; Liu, C. An image watermarking algorithm based on DWT DCT and SVD. In Proceedings of the 2009 IEEE International Conference on Network Infrastructure and Digital Content, Beijing, China, 6–8 November 2009; pp. 1034–1038.
24. Haj, A. Combined DWT-DCT digital image watermarking. *J. Comput. Sci.* **2007**, *3*, 740–746.

25. Li, Y.; Wang, H.; Barni, M. A survey of Deep Neural Network watermarking techniques. *Neurocomputing* **2021**, *461*, 171–193. [[CrossRef](#)]
26. Zhu, J.; Kaplan, R.; Johnso, J.; Fei, L. Hidden: Hiding data with deep networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 657–672.
27. Ahmadi, M.; Norouzi, A.; Karimi, N.; Samavi, S. Redmark: Framework for residual diffusion watermarking based on deep networks. *Expert Syst. Appl.* **2020**, *146*, 113157. [[CrossRef](#)]
28. Chaoning, Z.; Philipp, B.; Adil, K.; In-So, K. Universal adversarial perturbations through the lens of deep steganography: Towards a fourier perspective. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; pp. 3296–3304.
29. Luo, X.; Zhan, R.; Chang, H.; Yang, F.; Milanfar, P. Distortion agnostic deep watermarking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13548–13557.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.