2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)

2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom) | 978-1-6654-3574-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISPA-BDCloud-SocialCom-SustainCom52081.2021.00065

Exploration of the Influence on Training Deep Learning Models by Watermarked Image Dataset

1st Shiqin Liu
School of Computer Science
China University of Geosciences
Wuhan, China
Shiqin.Liu@cug.edu.cn

2nd Shiyuan Feng
School of Computer Science
China University of Geosciences
Wuhan, China
cugmonster@gmail.com

3rd Jinxia Wu
School of Computer Science
China University of Geosciences
Wuhan, China
wjxxkq2004@163.com

4th Wei Ren
School of Computer Science, China University of Geosciences, Wuhan, China
Key Laboratory of Network Assessment Technology, CAS
(Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China 100093)
Guizhou Provincial Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China
weirencs@cug.edu.cn

5th Weiqi Wang
School of Computer Science
China University of Geosciences
Wuhan, China
cugweiqiwang@163.com

6th Wenwen Zheng
School of Computer Science
China University of Geosciences
Wuhan, China
zhengwenwen@cug.edu.cn

***Abstract*—Deep learning has achieved great success in various applications with the help of a large scale of datasets. As a result, sharing the valuable big data that can be applied to training deep learning models is of essential importance currently. However, how to claim ownership and protect the copyright of image big data during the sharing process is still a vital issue that should be addressed. The application of digital watermarks can protect the copyright of image data, at the same time, it also degrades the image quality at the same time. As for invisible digital watermarks, the higher the watermark embedding intensity causes the greater the host image changes. Therefore, the performance of deep learning models may decline due to using the watermarked training set. In this paper, we evaluate the influences of different embedding intensities of various watermarking algorithms on several mainstream models and conclude how the watermarking intensity affects the model training. Besides, referring to watermarking algorithms that have been proposed, we proposed a novel discrete Fourier transform-based watermarking algorithm that can achieve image copyright protection yet maintain the utility of models.**

***Index Terms*—Deep learning, digital watermark, embedding intensity, model training, discrete Fourier transform.**

I. Introduction

However, the image dataset for training deep learning models may also be protected in data-sharing situations, especially when the dataset is hard or expensive to obtain. For example, remote sensing images, art images, professional field images, and so on. Once the data of those precious datasets have been stolen or copied on purpose or by accident, it will lead to enormous financial loss for the data owners. Thus, it is necessary to take measures to avoid the occurrence of the above situation.

The apperance of digital watermark technology brings the possible method for the data-sharing ecosystem, which is presently widely used to protect image copyright, which helps claim the ownership of image datasets. But watermarking algorithms for copyright protection degrades the quality of images, and the influence level depends on the embedding intensity and the algorithm itself. As we know, higher-intensity embedded watermarks can perform better on protecting copyright but will lead to quality declining.

To extend the above conclusion, an interesting question comes to us, that is, will the dataset with the watermark embedded images lead to the performance decrease of training the deep learning models compared to the original dataset? We observe that at present there are few pieces of research focus on the relationship between the watermarking intensity of training and test sets and the model performance. Thus, in this paper, we explored this point and implemented a series of experiments to figure it out. The contributions of this paper are listed as follows:

- We propose a watermarking algorithm based on discrete Fourier transform (DFT) for protecting the copyright of images in training deep learning.
- We explore the influence of three different watermarking algorithms, i.e., the discrete cosine transform (DCT) watermarking algorithm, the discrete wavelet transform (DWT) watermarking algorithm, and the DFT watermark-

978-1-6654-3574-1/21/$31.00 ©2021 IEEE
DOI 10.1109/ISPA-BDCloud-SocialCom-SustainCom52081.2021.00065

ing algorithm, for training classical deep learning models.
- We explore the effect of different embedding intensities on training different deep learning models.
- Two groups of practical parameters are proposed, based on the experiments, for watermark algorithms to tackle the copyright protection-related issues.

The rest of the paper is organized as follows. In Section II, the relevant background is presented and the problem is stated formally. Section III gives the details of our proposed scheme. In Section IV we deliver a summary of the experimental results with analysis. Section V presents related works, and finally in Section VI we concludes the paper.

II. Preliminaries

A. Principle of Two-Dimensional DCT

For a $M \times N$ matrix A, the two-dimensional DCT is defined as follows:

$$B_{pq} = a_p a_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2m+1)q}{2N}, \tag{1}$$

where $0 \leq p \leq M-1, 0 \leq q \leq N-1$. Then:

$$a_p = \begin{cases} \frac{1}{\sqrt{M}} & , p=0 \\ \sqrt{\frac{2}{M}} & , 1 \leq p \leq M-1 \end{cases}, \tag{2}$$

$$a_q = \begin{cases} \frac{1}{\sqrt{N}} & , p=0 \\ \sqrt{\frac{2}{N}} & , 1 \leq q \leq N-1 \end{cases}. \tag{3}$$

While the inverse two-dimensional DCT is defined as:

$$A_{mn} = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} a_p a_q B_{pq} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2m+1)q}{2N}, \tag{4}$$

where $0 \leq m \leq M-1, 0 \leq n \leq N-1$.

B. Principle of Two-Dimensional DWT

Two-dimensional DWT of function $f(x,y)$ of size $M \times N$ is defined as:

$$W_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\varphi_{j_0,m,n}(x,y), \tag{5}$$

$$W_\Psi^i(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\Psi_{j,m,n}^i(x,y), \quad i=\{H,V,D\} \tag{6}$$

where j_0 is an arbitrary starting scale, $\varphi_{j_0,m,n}(x,y)$ and $\Psi_{j,m,n}^i(x,y)$ are the scaled and translated basis functions, the $W_\varphi(j_0,m,n)$ coefficients define an approximation of $f(x,y)$ at scale j_0 and the $W_\Psi^i(j,m,n)$ coefficients add horizontal, vertical, and diagonal details for scale $j \geq j_0$.

while the inverse discrete wavelet transform is described as:

$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_m \sum_n W_\varphi(j_0,m,n)\varphi_{j_0,m,n}(x,y) + \frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{\infty} \sum_m \sum_n W_\Psi^i(j,m,n)\Psi_{j,m,n}^i(x,y) \tag{7}$$

C. Principle of Two-Dimensional DFT

The two-dimensional DFT is defined below:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{[-j2\pi(\frac{ux}{M}+\frac{vy}{N})]}, \tag{8}$$

where $f(x,y)$ is a digital image of size $M \times N$, $u = 0,1,2,...,M-1$ and $v = 0,1,2,...,N-1$.

while its inverse transform is described as:

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{[j2\pi(\frac{ux}{M}+\frac{vy}{N})]}. \tag{9}$$

for $x = 0,1,2,...,M-1$ and $y = 0,1,2,...,N-1$.

D. Problem Formulation

Assume that the deep learning model is D, the watermarking algorithm is A, the watermark image is I, the watermarking intensity is P, the training set is $S_{train_{(A,P)}}$, the test set is S_{test}. In addition, the result of model training is R, i.e., the performance of the model on the test set. Moreover, it is necessary to mention that $Strain_{(0,0)}$ is the initial training set without watermarks.

First, D is trained by $Strain_{(0,0)}$ and tested by S_{test}, then recording R and analyzed it. Then, I is embedded into each image in $Strain_{(0,0)}$ by the algorithm A with intensity P, the dataset $S_{train_{(A,P)}}$ is generated in this step. After we train D with $S_{train_{(A,P)}}$ and test it with S_{test}, R' is analyzed. The influence that watermarks cause on model training is described by the comparison with R and R'.

III. Proposed Scheme

We will first briefly introduce two kinds of watermarking algorithms based on DCT and DWT proposed by Xiang et al. [1]. Referring to thier proposed scheme, we embed watermarks to each image of a dataset several times with various intensities, and we train different deep learning models with the watermarked datasets. It is found that the watermarked images are different on the basis of different embedding algorithms and different intensities each time. Furthermore, based on the above work, we proposed a novel DFT-based algorithm and we will illustrate it in the following section in detail.

A. Introduction to DCT, DWT and DFT Watermarking Algorithms

According to the code implemented by Xiang et al. [1] on their GitHub[1], We proposed a novel DFT-based watermarking algorithm, which is described in Algorithm 1.

[1]https://github.com/Y-Xiang-hub/Generate-Adversarial-Examples-By-Digital-Watermarking

Algorithm 1 Improved DFT-based Watermarking

Input: host images $Image_{host}$, watermark image $Image_{water}$, Embeding intensity P, total number of the host images $Totel_{image}$, Discrete Forier Transform DFT, low frequency coefficient shift DFT_{shift} and the inverse operations $IDFT$, $IDFT_{shift}$. Spectrum matrix C_{DT}, shifted spectrum matrix C_{DF}

Output: dataset $STrain_{DFT,P}$, with $Total_{image}$ watermarked images $Candidate_{image}$ in it.

set P
read $Image_{water}$ as wm
resize wm
for i=1:$Total_{image}$ **do**
    read $Image_{host}[i]$ as img
    resize img
    separate R,G and B channels of img
    $C_{DT} = DFT(img)$
    $C_{DF} = DFT_{shift}(C_{DT})$
    $M_R = img_R C_{DF} + P \times wm_R$
    $M_{RF} = IDFT(IDFT_{shift}(M_R))$
    $M_G = img_G C_{DF} + P \times wm_G$
    $M_{GF} = IDFT(IDFT_{shift}(M_G))$
    $M_B = img_B C_{DF} + P \times wm_B$
    $M_{BF} = IDFT(IDFT_{shift}(M_B))$
    $Candidate_{image} = (M_{RF}, M_{GF}, M_{BF})$
    save $Candidate_{image}$
end for
return

Our experimental operations of the above three watermarking algorithms are similar, thus, we use the DFT-based watermarking algorithm as the example to show how we apply the algorithms for our experiment.

In this DFT-based improved watermarking method, we embed the watermark image into the DFT domain of the host image channel by channel. First, the watermark image is transformed into the DFT domain. Then perform the DFT operation for each host image in the training set $STrain_{(0,0)}$. Finally, a DFT domain watermark has been embedded into the host image. We use P to represent the watermarking intensity. As P gets bigger, the watermark gets easier to be noticed as shown in Fig. 1.

In the following section, we will illustrate the datasets and the deep learning models we apply.

Fig. 1. Different results when P increase

B. Datasets and Deep Learning Models

The dataset "Dogs vs. Cats" from Kaggle is adopted in our experiment, as shown in Fig. 2. The purpose of our experiment is to figure out how watermarked data affect model training. With this end in view, we generate 33 datasets composed of watermarked images with different watermarking methods and different intensities. The methods are the 3 ones above. As for the watermarking intensity, Fig. 1 shows a cat image with a dog watermark by intensities from 0.5 to 5. As can be seen, when the intensity rises to 5, the cat is already disappeared, replaced by the dog in the watermark image. Therefore, on the basis of a plenty of tests, in our formal experiment, we choose the intensity P from 0 to 0.12, step 0.01. Up to now, the 36 datasets for the experiment are ready. Each one of them is composed of images watermarked by one of the 3 algorithms and an embedding intensity in range 0 to 0.12, step 0.01. Besides, during each training in the experiment, the training and testing sets are from the same dataset mentioned above, in order to simulate the real situation. However, when we evaluate the results of model trainings, the original datasets without watermarks is used.

Moreover, we select the following four typical models to launch the experiment: Alexnet, VGG16, LeNet-5, and ResNet50 respectively. It is noted that the transfer learning method is adopted when we train ResNet50, which may reduce the influence of watermarked data. This point will be illustrated detailedly in Section IV.

IV. Experiments and Analysis

Our experiment consists of three parts as follows:

- Watermarked datasets with different algorithms and embedding intensities are built, and the intensities are ranged from 0 to 0.12 with step 0.01. Total number: 36.
- Four classic deep learning models, AlexNet, VGG16, LeNet-5 and ResNet50, are trained with each watermarked dataset. Additionally, there is a total of 13 experimental results will be shown below. Total training times: 144.
- The results of the models trained by different watermarked datasets are compared with a baseline model (i.e., a model trained by a normal training set), and we evaluate their accuracy, precision, recall, and $F1$-score.

In the following four sections, we will analyze and illustrate our experimental results in detail. For each result of the trained model, a set of bar graphs show the models' accuracy, precision, recall, and $F1$-score under watermarking intensities from 0 to 0.12. Next, for each given watermarking algorithm, we horizontal compare models' $F1$-score to explore the influences.

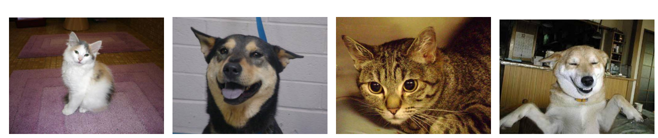

Fig. 2. "Dogs vs. Cats" dataset

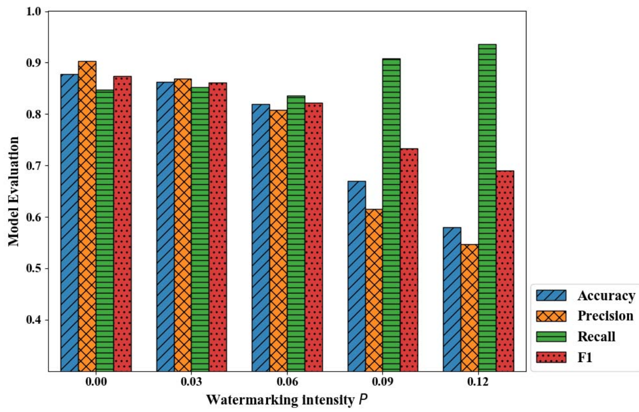

Fig. 3. AlexNet performances under DCT-based watermarking

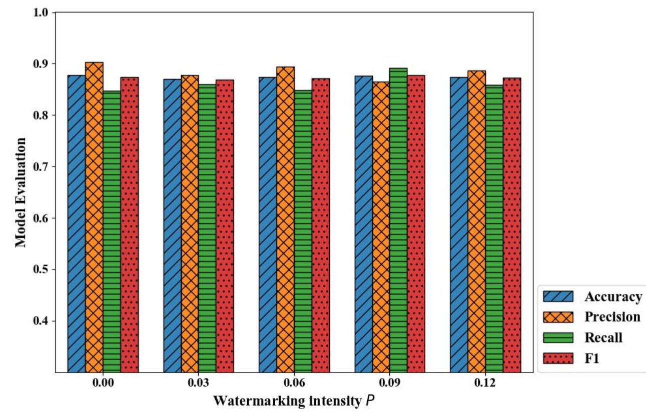

Fig. 5. AlexNet performances under DFT-based watermarking

A. *Experimental Results of AlexNet*

- **DCT-based Watermarking Algorithm.** Fig. 3 below shows the performance of AlexNet trained with images watermarked by the DCT-based watermarking algorithm, using intensities from 0 to 0.12. As can be seen from Fig. 3, the model's accuracy and precision gradually decrease as the embedding intensity goes up. However, the recall increases to 0.93, which means training with DCT-watermarked images has no negative impacts on the model's recall. Besides, the drop of the $F1$-score comprehensively tells that the model is affected negatively.
- **DWT-based Watermarking Algorithm.** Fig. 4 below shows the performance of AlexNet trained with training set watermarked by the DWT-based watermarking algorithm, using intensities from 0 to 0.12. From Fig. 4, it can be seen that compared with the DCT-based watermarking algorithm, the DWT-based one has a greater influence on AlexNet. Actually, it is obvious that the $F1$-score sharply dropped when the intensity comes to 0.02. Moreover, it is easy to find that recall remains at a high level as Fig. 3, which means DWT-based watermarking does not influence recall either.
- **DFT-based Watermarking Algorithm.** Furthermore, when we have the images in the training set watermarked with DFT-based watermarking algorithm shown in Algorithm 1, using intensities from 0 to 0.12, the model's performance is showed in Fig. 5.

B. *Experimental Results of VGG16*

- **DCT-based Watermarking Algorithm.** Fig. 6 below shows the performance of VGG16 trained with the training set watermarked by the DCT-based watermarking algorithm, using intensities from 0 to 0.12. Besides, Fig. 6 also illustrates that the DCT-based patchwork watermarking algorithm has a great influence on the accuracy of VGG16. Specifically, when the intensity comes to 0.03, the accuracy dramatically drops from 0.89 to 0.51. Likewise, the precision falls too, however, the recall has fluctuated at a high level.
- **DWT-based Watermarking Algorithm.** Fig. 7 below shows the performance of VGG trained with the train-

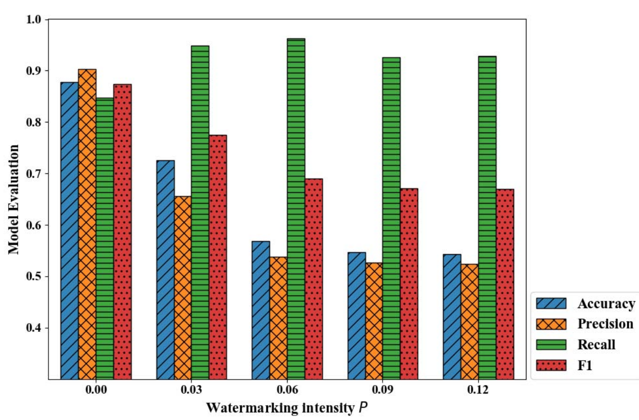

Fig. 4. AlexNet performances under DWT-based watermarking

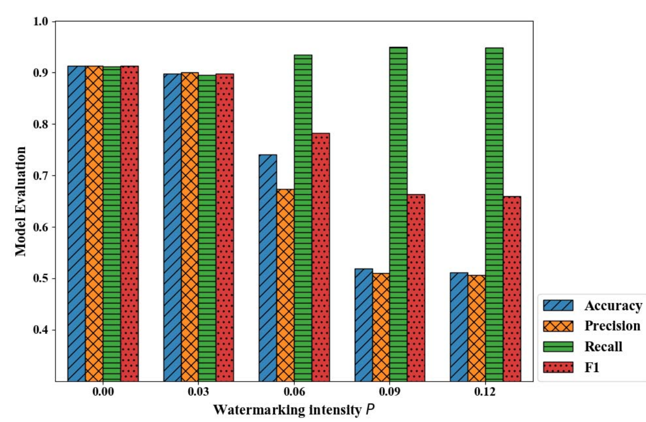

Fig. 6. VGG16 performances under DCT-based watermarking

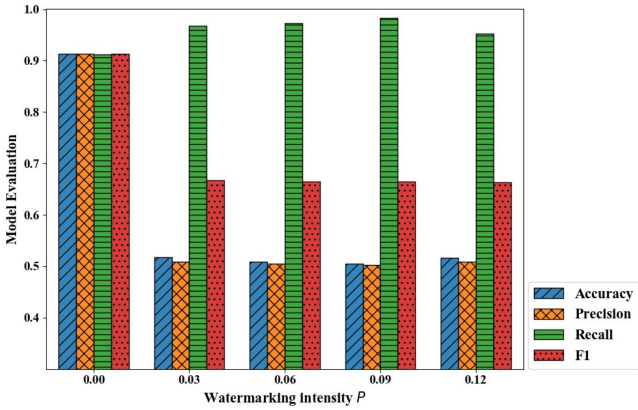

Fig. 7. VGG16 performances under DWT-based watermarking

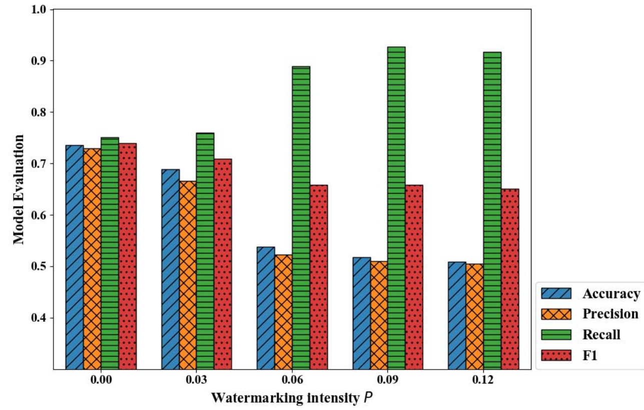

Fig. 9. LeNet-5 performances under DCT-based watermarking

ing set watermarked by the DWT-based watermarking algorithm, using intensities from 0 to 0.12. This figure illustrates the influence of DWT-based watermarking on VGG16's training. As we can see, the accuracy and precision decrease even when the intensity is not that high, about 0.02, which leads to the massive drop of $F1$. However, the recall still reaches a high stage.

- **DFT-based Watermarking Algorithm.** Fig. 8 below shows the performance of VGG trained with training set watermarked by the DFT-based watermarking algorithm applying intensities from 0 to 0.12. From the chart below it is obvious that the DFT-based watermarking algorithm has little impact on this model compared to the DCT-based and DWT-based watermarking algorithms.

C. Experimental Results of LeNet-5

Fig. 9, Fig. 10, and Fig. 11 below shows the performance of LeNet-5 trained with images watermarked by the DCT-based, DWT-based and DFT-based algorithms, using intensities from 0 to 0.12. Fig. 9, Fig. 10, and Fig. 11 illustrate almost the same result as the AlexNet and VGG16. There is no doubt that the watermarking methods can degrade the model's performance, i.e., the accuracy, precision, and $F1$-score drop with the rising of the embedding intensity. Similarly, the recall has a rising trend too. However, LenNet-5 has a poor performance on recognizing cats and dogs itself compared with the former deep learning models.

D. Experimental Results of ResNet50

Fig. 12 shows the performance of ResNet50 trained with images watermarked by the DFT-based watermarking algorithm, which also applies the transfer learning method for enhancement. Due to the transfer learning, the evaluations stay at a high level. The chart tells that the impact of watermarked training set on a deep learning model is negligible when the transfer learning method is utilized.

As for another two watermarking algorithms (i.e., DCT-based and DWT-based watermarking algorithms), the results are almost the same. The reason may be the images used for the model training in the training set are not all watermarked generally. In contrast, we find out that when adopting transfer learning, we can pay less attention to worrying about the

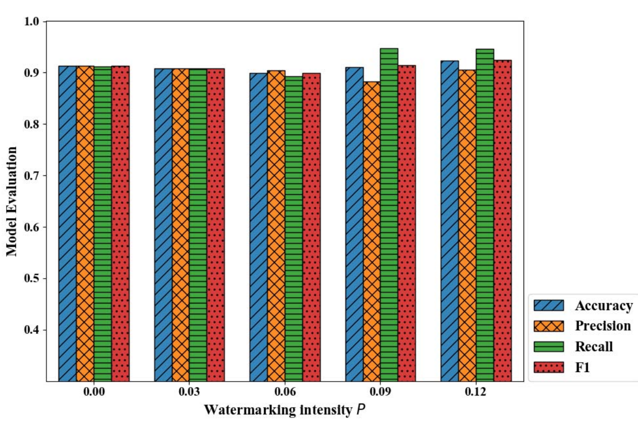

Fig. 8. VGG16 performances under DFT-based watermarking

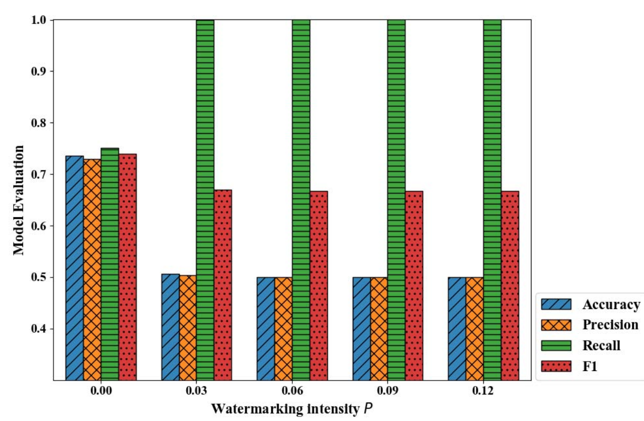

Fig. 10. LeNet-5 performances under DWT-based watermarking

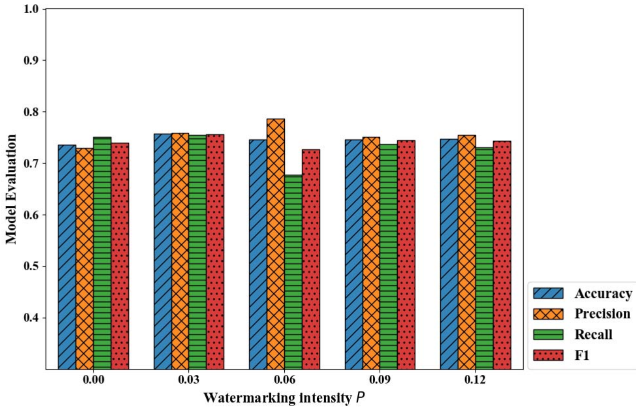

Fig. 11. LeNet-5 performances under DFT-based watermarking

negative effect that watermarked training images have on models.

E. Horizontal Comparison

In order to show the influence of digital watermarking on each model more intuitively, we hereby give three line charts as follows.

Fig. 13 shows that unlike the other two, the DFT-based watermarking method we propose has little effect on each model conversely. By comparison, DFT-based method may be the best way of watermarking images in the datasets for deep learning model training.

Fig. 14 shows that there exists a clear inflection point on the VGG16 line, which indicates that when the model is trained using images with our DCT-based watermark, the embedding intensity should better not exceed 0.04. At the same time, this watermarking method has a greater impact on the other models when P rises except for the ResNet50 which uses transfer learning.

Fig. 15 shows that there exists a clear inflection point on both the VGG16 line and AlexNet line, which indicates that

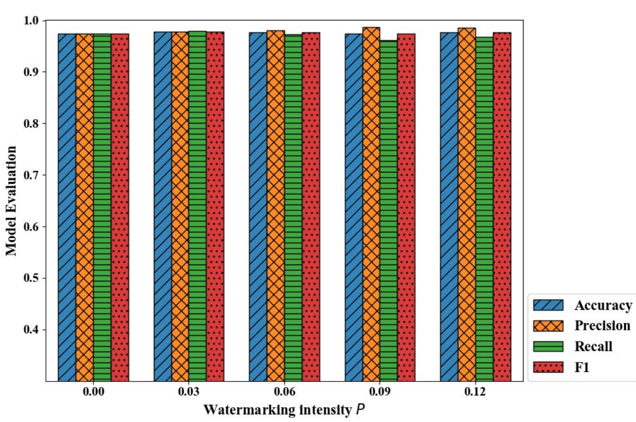

Fig. 12. ResNet50 performances under DFT-based watermarking

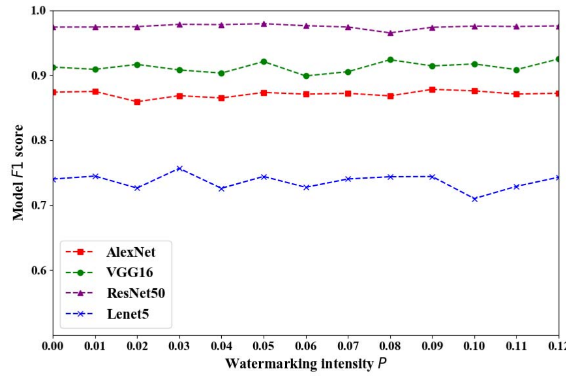

Fig. 13. Models' performance under DFT-watermarking

when we use images with our DWT-based watermark as the training set, the embedding intensity should better not exceed 0.02. Compared with Fig. 14, DWT has a greater influence. In addition, this watermarking method has little impact on the ResNet50 which uses transfer learning.

To sum up, compared with the DFT-based algorithms, the DCT-based and DWT-based methods have more influence on model training, specifically the DWT-based one. Thus, we suggest when adopting the two methods in dataset protecting, the embedding intensities should be under 0.04 and 0.02 respectively. Luckily, to avoid the negative effects, the DFT-based watermarking algorithm we proposed can be a better choice. Besides, using the transfer learning method or "shallow" watermarks based on DCT-based or DWT-based methods is also able to prevent the negative effects.

V. Related Works

A. Deep Learning

Krizhevsky et al. [2] proposed a deep convolutional neural network, named AlexNet. The model uses the ReLU activation function to accelerate convergence, GPU parallel to accelerate

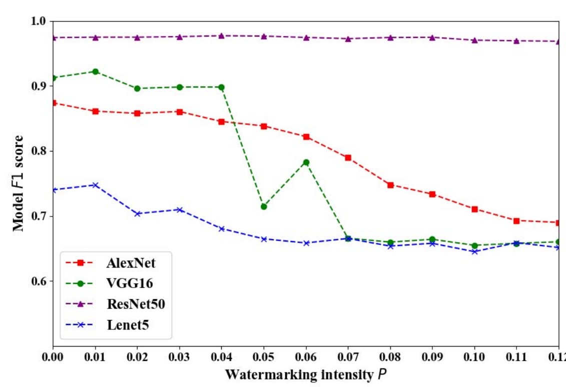

Fig. 14. Models' performance under DCT-watermarking

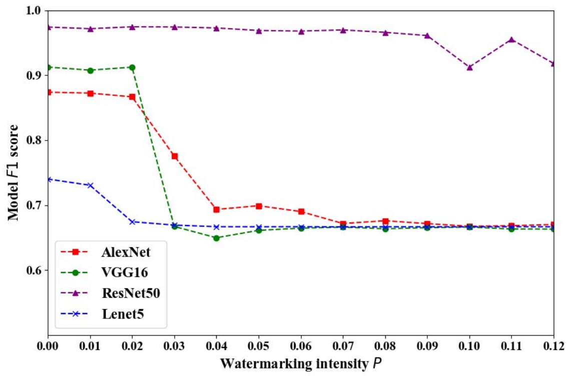

Fig. 15. Models' performance under DWT-watermarking

training, and uses overlap pooling, dropout, data enhancement, and other means to prevent overfitting; Simonyan and Zisserman [3] proposed the VGG deep learning model, which uses a very small convolution filter architecture to comprehensively evaluate the deep network, multi-layer stack to enhance the nonlinear network to enhance the ability of feature expression, and uses a smaller and even number of pooled cores to enhance the ability to capture small feature changes; He et al. [4] proposed a network called deep residual learning framework. In this structure, each stacked layer is fitted with residual mapping instead of directly fitting the underlying mapping expected by the whole building block to solve the "gradient vanishing" and "gradient exploding" problems at an extent.

B. Digital Watermarking

Kii et al. [5] proposed a DCT patchwork watermarking method. Watermarking techniques are utilized in the study of Xiang et al. [1] when they proposed a method to generate adversarial examples. However, the classical watermark embedding algorithm is easy to cause visual changes in the image, resulting in the algorithm's transparency decline. To solve this problem, Zhou et al. [6] proposed a patchwork watermarking algorithm in the transform domain. The algorithm applies the characteristics of DCT transform and the human visual system. By superimposing the watermark information on the DCT components of R, G, and B channels and R, G, and B channels after DCT transform, the transparency of the algorithm is greatly improved.

C. Dataset Copyright

Protecting data's copyright is becoming a common issue. Hong et al. [7] used the visible watermark to protect image dadta, but the visible watermark is not transparent to users. Zope-Chaudhari et al. [8] proposed a digital watermarking scheme that can be used for copyright protection of geospatial data using vector data as a watermark.

VI. Conclusion and Future Work

For the common deep learning models in our experiment, the DFT-based watermarking algorithm has the smallest impact, followed by the DCT-based and the DWT-based methods. The DWT-based watermark influences the model training the most under the same embedding intensity. We suggest that among the above three algorithms, the DFT-based algorithm is more suitable for protecting deep learning image copyright because when using DCT-based and DWT-based methods, we need to determine the appropriate embedding parameters to limit the watermark's negative effect on the model. Taking our proposed algorithms as examples, the thresholds are 0.02 for DWT and 0.04 for the DCT algorithm.

Compared with AlexNet, training with watermarked images has more influence on VGG16 and LeNet models. The ResNet50 model, trained by transfer learning, is the least sensitive to watermark. According to the proposed experiment, ResNet50 is the most suitable for protecting copyright and maintaining the effectiveness of images in deep learning. However, it cannot rule out the impact of the transfer learning method, which is another issue we are about to study. Through the analysis of the performance of the same model under different embedding coefficients, it can be concluded that the image with watermarks has a certain impact on the model.

For future work, we will implement more experiments on different multi-type datasets such as CIFAR-10 and CIFAR-100 datasets to improve our current work. In addition, we will also explore the influence of extracting the watermark image from the host image in different situations, which is essential for digital watermark application on copyright protection.

Acknowledgment

The research was financially supported by National Natural Science Foundation of China (No. 61972366), the Foundation of Key Laboratory of Network Assessment Technology, Chinese Academy of Sciences (No. KFKT2019-003), Major Scientific and Technological Special Project of Guizhou Province (No. 20183001), and the Foundation of Guizhou Provincial Key Laboratory of Public Big Data (No. 2018BDKFJJ009, No. 2019BDKFJJ003, No. 2019BDKFJJ011).

References

[1] Y. Xiang, W. Ren, T. Li, X. Zheng, T. Zhu, and K.-K. R. Choo, "Efficiently constructing adversarial examples by feature watermarking," *arXiv preprint arXiv:2009.05107*, 2020.
[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.
[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[5] H. Kii, J. Onishi, and S. Ozawa, "The digital watermarking method by using both patchwork and dct," in *Proceedings IEEE International Conference on Multimedia Computing and Systems*, vol. 1. IEEE, 1999, pp. 895–899.
[6] Y. Zhou and J. Liu, "Blind watermarking algorithm based on dct for color images," in *2009 2nd International Congress on Image and Signal Processing*. IEEE, 2009, pp. 1–3.

[7] S. Hong, T.-h. Kim, T. Dumitraş, and J. Choi, "Poster: On the feasibility of training neural networks with visibly watermarked dataset," *arXiv preprint arXiv:1902.10854*, 2019.
[8] S. Zope-Chaudhari, P. Venkatachalam, and K. M. Buddhiraju, "Copyright protection of vector data using vector watermark," in *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2017, pp. 6110–6113.