Evaluation of the Privacy of Images Generated by ImageCLEFmedical GANs 2024 Based on Similarity Methods

Notebook for ImageCLEF Lab at CLEF 2024

Shitong Cao, Xiaobing Zhou*

School of Information Science and Engineering, Yunnan University, Kunming 650504, Yunnan, China

Abstract

Our team's primary contribution to the ImageCLEFmedical GANs 2024 task is as follows. This task aims to assess whether synthetic medical images generated by Generative Adversarial Networks (GANs) contain identifiable features from the training data. We employed a similarity-based classification method, categorizing real images based on their similarity to generated images. In this work, we utilized various similarity calculation methods to evaluate the similarity between real and generated images. We calculated the similarity for original images, noisy images, and features extracted through a feature network. On the validation dataset, our similarity-based approach achieved an F1-score of 0.732 and an accuracy of 0.760. In the submitted results, our best F1-score was 0.598. Our experimental results demonstrated that our method could distinguish between images that were "used" and those that were "not used".

Keywords

GANs, Medical Images, Similarity Calculation, Magnify Differences

1. Introduction

Deep learning models have significant potential in supporting medical diagnosis and treatment, achieving remarkable results in various medical image analysis tasks. However, training these models requires vast amounts of data, which is often challenging to obtain. Deep generative models, capable of generating highly realistic medical images, have been used to create large synthetic datasets to facilitate model training[1, 2].

Nevertheless, since generative models model the probability distribution of the data, synthetic images produced by these models may threaten the privacy of patient images used in training. Recent studies have shown that medical images, such as chest X-rays and MRI scans, can be used to re-identify patients, exacerbating concerns about privacy breaches.

To identify potential privacy threats associated with the use and sharing of synthetic medical data in real-world scenarios, a new challenge has been introduced as part of the ImageCLEFmedical CLEF challenge 2024[3, 4]. Our team's username is shitongcao. ImageCLEF is a multimodal challenge aimed at verifying whether images generated by Generative Adversarial Networks (GANs) are similar enough to the training data to pose a privacy risk. Specifically, given a set of synthetic images and a set of real images, the task is to identify which real images were used to train the model that generated the synthetic data. This is a binary classification task[5] where real images can be classified as "used" or "not used".

The generated images are produced by learning the data distribution of real images, meaning that the synthetic images are statistically similar to the real ones. The closer the data distribution of the generated images is to the real images, the higher the quality of the generated images, making them visually closer to real images. In this work, our task is to perform binary classification to categorize

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

*Corresponding author.

stcao@mail.ynu.edu.cn (S. Cao); zhouxb@ynu.edu.cn (X. Zhou)

0009-0003-1298-4166 (S. Cao); 0000-0003-1983-0971 (X. Zhou)

images as used or not used. To achieve this goal, we calculate similarity scores to determine the category of the images, classifying images with high similarity scores[6, 7, 8] as used and those with low similarity scores as not used.

We employed three different methods to calculate similarity. First, we directly computed the similarity between the original generated images and real images by comparing their pixel values. Second, we applied noise to the original images and then calculated the similarity between the noisy images and the real images, which enhanced the robustness of the images. Finally, we extracted features from the images using advanced deep learning models to obtain high-dimensional features and then calculated the similarity between these features. These features contain deep information about the images, providing a more accurate reflection of the similarity between images.

By using these three methods, we were able to comprehensively evaluate the similarity between generated and real images, effectively accomplishing the binary classification task. This multi-method approach not only improved the accuracy of the classification but also provided richer information and stronger guarantees for image processing and analysis.

2. Related Work

Synthetic images[9, 10, 11] offer an effective way to create representative cases, enabling researchers and clinicians to better study and understand various medical conditions, develop diagnostic tools, and explore treatment strategies. Moreover, synthetic images address privacy issues associated with patient data. Medical images often contain sensitive information, making it difficult to share or publicly release datasets. By generating synthetic images, the statistical and anatomical characteristics of real data can be preserved while removing specific patient information, thus maintaining privacy, enabling more open collaboration, and facilitating research progress. Consequently, synthetic images are an indispensable resource in the medical field, used for data augmentation, rare scenario simulation, and privacy protection. Their use helps researchers, clinicians, and technologists tackle critical challenges, enhance diagnostic accuracy, improve patient care, and advance medical imaging technologies.

Nataraj et al. proposed a novel method combining co-occurrence matrices and deep learning techniques to detect GAN-generated fake images. The authors extracted co-occurrence matrices from the three color channels in the pixel domain and trained a deep convolutional neural network (CNN) model. The method demonstrated good generalization capability when trained on one dataset and tested on another.

GANs[9] have also been widely applied in medical image-to-image translation tasks. For example, Zhu et al. introduced CycleGAN[12], a method capable of translating images from one domain to another without the need for paired training data.

In conclusion, previous studies have demonstrated the potential of GANs in generating synthetic medical images and performing image-to-image translation tasks. However, distinguishing between synthetic and real medical images remains an active area of research, requiring robust methods to ensure the reliability and integrity of generated data.

3. Method

3.1. Data Similarity Statistics

To conduct a statistical analysis of the images, we categorized them into three groups: generated images, used images, and not used images. It was observed that the similarity between the images was significantly high. To validate the similarity between the data, we employed the Three-Component Weighted Structural Similarity Index (3-SSIM) to calculate the similarity. 3-SSIM is an improved version of SSIM (Structural Similarity Index). Unlike SSIM, which compares the entire image as a whole, 3-SSIM evaluates the similarity in edge, texture, and smooth regions separately, assigning different weights to these components to obtain the final assessment result.

The Three-Component Weighted Structural Similarity Index calculates the similarity between images by separately evaluating the edge information, texture areas, and smooth regions, and then assigns different weights to these components. The final similarity score is obtained by summing these weighted values. We performed statistical analysis on three sets of data: the similarity between real images (real-real), the similarity between generated and real images (generated-real), and the similarity between generated images (generated-generated).

Table 1
Data analysis was conducted on the datasets for both tasks. The mean, minimum, and maximum 3-SSIM values between images in the development and test datasets were calculated, and the results for the two datasets were averaged.

Dataset	Subsets	Average SSIM	Minimum SSIM	Maximum SSIM
Development	Real-Real	12.9	5.5	19.0
	Generated-Real	13.4	7.9	21.5
	Generated-Used	15.2	9.1	21.5
	Generated-Not Used	13.4	6.4	18.6
Test	Real-Real	12.8	5.6	18.8
	Generated-Real	13.2	7.8	21.3

The experimental results presented in Table 1 reveal that the similarity between generated images is higher than the similarity between generated and real images, which in turn is higher than the similarity between real images. This is because the generated images are produced by the same model, resulting in relatively higher similarity. The similarity between generated and real images is slightly lower due to the inclusion of two types of images; the used images likely have a higher similarity to the generated images, while the not used images have a lower similarity. The similarity values for real-real are slightly lower, but the difference is not significant. When calculating the data in the table, the self-comparison similarity was excluded as it is identical and has no practical significance. Therefore, the maximum similarity in real-real and gen-gen is not 30.

3.2. Similarity Calculation Methods

Similarity calculation methods play a crucial role in image processing, machine learning, and information retrieval. This paper provides a detailed introduction to several common similarity calculation methods, including Euclidean distance, cosine similarity, and Structural Similarity Index (SSIM). Similarity calculation is used to measure the degree of similarity between two objects, such as images, feature vectors, or strings. These methods are widely applied in image classification, clustering, retrieval, and recommendation systems. Each method has its own applicable scenarios and advantages, making the choice of an appropriate similarity calculation method critical to the performance and effectiveness of algorithms. As shown in Figure 1, this paper presents an illustration of different image similarity calculations.

When calculating image similarity, Euclidean distance is a commonly used method that measures the difference between two images at the pixel level. This method assumes that images can be represented as points in a high-dimensional space, where each pixel's color value (typically RGB values) is considered a dimension in this space. Euclidean distance calculates the straight-line distance between these two points (i.e., the two images), with a smaller distance indicating higher similarity between the images. First, it is essential to ensure that the two images have the same dimensions; in this task, the generated images and the real images are of identical sizes. The formula for calculating the Euclidean distance is as follows.

$$d(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2}$$

Where $\mathbf{A}$ and $\mathbf{B}$ are two n-dimensional vectors.

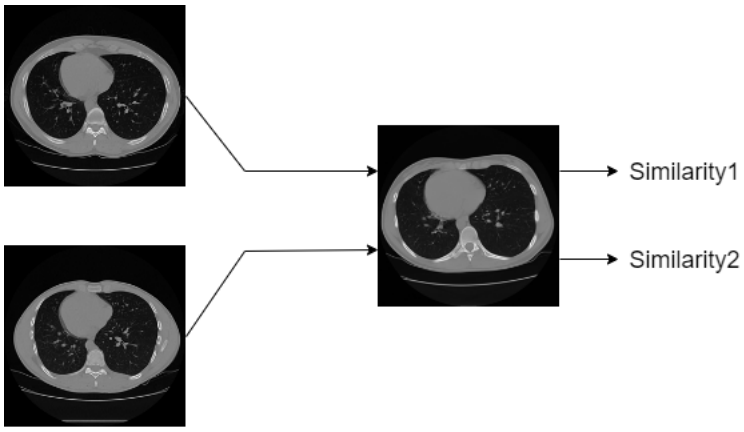

Figure 1: Schematic diagram of triplet loss. The generated image serves as the anchor, the used image serves as positive sample, and the not used image serves as negative sample.

Cosine similarity is another commonly used method for measuring image similarity, particularly in content-based image retrieval (CBIR) systems. Cosine similarity assesses the similarity between two vectors by measuring the cosine of the angle between them. The core idea is that the closer the directions of the two vectors, the more similar they are, regardless of their magnitudes. Calculating cosine similarity involves converting each image into a vector form. This typically entails flattening the pixel values of the image or features extracted from the image (such as color histograms, texture descriptors, shape features, etc.) into a one-dimensional vector. The cosine similarity value ranges from -1 to 1, where 1 indicates identical directions (very similar), 0 indicates orthogonality (no similarity), and -1 indicates completely opposite directions. Cosine similarity focuses on directional similarity, ignoring magnitude. In some cases, two images might be very similar in terms of certain feature ratios, but the absolute differences in actual pixel values could be significant.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Where A and B are two vectors. $\mathbf{A} \cdot \mathbf{B}$ represents the dot product of vectors A and B. ||A|| and ||B|| represent the magnitudes of vectors A and B.

Structural Similarity (SSIM) is a more intuitive and effective method for calculating image similarity. SSIM considers the luminance, contrast, and structural information of images, which allows it to more accurately reflect the human visual system's perception of image quality. SSIM first calculates the luminance difference between two images. The luminance comparison is achieved by calculating the mean values of the images, which reflects the overall brightness levels of the images. Next, SSIM calculates the contrast difference. The contrast comparison is achieved by calculating the standard deviation of the images; the greater the standard deviation, the higher the image contrast. Finally, SSIM compares the structural information of the two images. This step is achieved by calculating the covariance of the images. Covariance reflects the linear relationship between the pixels of the images, capturing the structural characteristics of the images.

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Where x and y are corresponding blocks of the two images. μ_x and μ_y are the mean values of image blocks x and y. σ_x and σ_y are the standard deviations of image blocks x and y. σ_{xy} is the covariance of image blocks x and y.

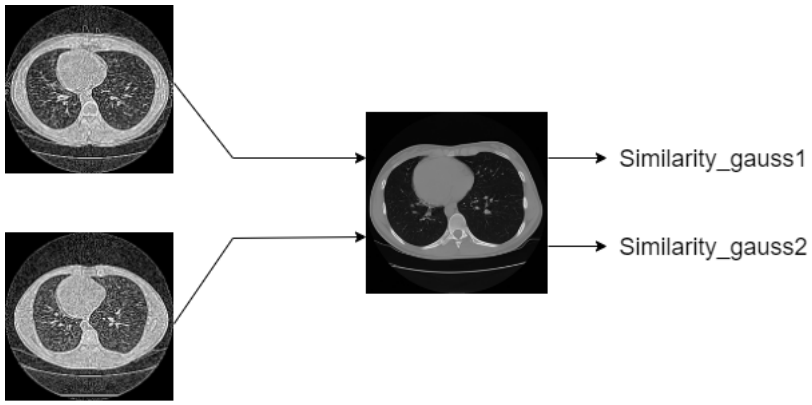

Figure 2: Schematic diagram of triplet loss. The generated image serves as the anchor, the used image serves as positive sample, and the not used image serves as negative sample.

3.3. Expanding the Differences between Images

Noise can significantly affect the quality of images, thereby impacting the results of similarity calculations. Common types of noise include Gaussian white noise and salt-and-pepper noise.

Gaussian White Noise: This type of noise follows a normal distribution, typically with a mean of zero, and the standard deviation can be set according to the actual situation. Gaussian noise adds a random value to each pixel of the image, resulting in an overall blurring effect.

Salt-and-Pepper Noise: This type of noise randomly changes image pixels to either white (255) or black (0) with a certain probability, commonly occurring during image transmission. Salt-and-pepper noise creates random white or black spots in the image, making it appear speckled and unclear. As shown in Figure 2, similarity calculation is performed on the two images with added noise

Experiments show that noise significantly impacts image similarity values. After adding noise, the similarity between images decreases notably, effectively expanding the differences between images. Gaussian white noise and salt-and-pepper noise degrade image quality in different ways, thus affecting the results of similarity calculations. As a method for measuring image similarity, it is crucial to consider the impact of noise on the results and to apply appropriate noise reduction measures in practice to improve the accuracy of similarity calculations.

4. Experiments

4.1. Evaluation Metrics

We carried out assessments across two distinct experiments. Initially, we split the validation set into two equal portions, designating one half as the test set to streamline our experimental analysis. Subsequently, we submitted our findings to the IMAGECLEFMED GANS 2024: IDENTIFY TRAINING DATA FINGERPRINTS competition. This challenge was addressed as a binary classification issue, and its evaluation criteria encompassed several critical performance metrics: F1-score, accuracy, precision, recall, and specificity. Notably, the F1-score has been chosen as the principal metric for this year's evaluation. The definitions of these metrics are as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{3}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$\text{F1-score} = \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

4.2. Experimental Results

In this experiment, we conducted tests on both the original images and images with added noise. We used Euclidean distance, cosine similarity, and Structural Similarity Index (SSIM) to calculate the similarity between images. Based on the similarity scores, we performed classification to obtain accuracy, precision, recall, and F1-score.

To highlight the differences between images, we introduced varying degrees of noise into the images for the experiments. Specifically, we added different levels of noise, including Gaussian noise and salt-and-pepper noise, to assess their impact on similarity scores and classification performance. This approach helped us understand how noise degrades image quality and affects similarity calculations.

By systematically introducing different noise levels, we were able to evaluate the effectiveness and stability of different similarity calculation methods under various noise conditions. This comprehensive experimental design allowed us to more accurately measure the impact of noise on image similarity calculations, providing valuable insights for future image processing and analysis improvements. The experimental results, summarized in the table 2 illustrate the performance metrics of each similarity calculation method under different noise levels.

Table 2
Similarity scores between the original images and the noise-added images were calculated using Euclidean distance, cosine similarity, and SSIM methods. The same procedures were applied to both tasks, and the results for the two tasks were averaged.

Gaussian noise	Metric	Accuracy	Precision	Specificity	Recall	F1-score
w/o	Eucledian Distanc	0.675	0.663	0.638	0.713	0.687
w/o	Cosine Similarity	0.613	0.576	0.375	0.850	0.687
w/o	SSIM	0.678	0.682	0.702	0.689	0.704
w/	Eucledian Distanc	0.650	0.650	0.650	0.650	0.650
w/	Cosine Similarity	0.705	0.683	0.720	0.696	0.717
w/	SSIM	0.731	0.740	0.750	0.713	0.726

We made predictions for all 4,000 images generated by each model and submitted these prediction results. To evaluate the performance of the models, we used the F1-score as the primary evaluation metric, as it comprehensively considers both precision and recall, providing a more holistic assessment of performance. Additionally, we used accuracy as a secondary metric to measure the correctness of the model across all predictions. This evaluation process ensured that we could fully understand the performance of the models under different conditions. We submitted a total of eight different results. The detailed scores are summarized in the table 3, showing the specific performance and corresponding evaluation scores for each submission. These results help us to further analyze and improve the models, enhancing their effectiveness in practical applications.

As shown in Table 3, our three best experimental results are presented. Through our experiments, we found that adding noise significantly increases the differences between images. We performed similarity calculations on the images with added noise and classified them based on these similarity scores. Notably, using the SSIM similarity calculation method yielded the best results on the test set.

Table 3
Presentation of experimental results

Metric	F1	Acc	Prec	Rcall	F1	Acc	Prec	Rcall	F1
SSIM	0.598	0.614	0.599	0.689	0.641	0.504	0.503	0.622	0.556
Eucledian Distanc	0.598	0.615	0.600	0.688	0.641	0.504	0.503	0.619	0.555
Cosine Similarity	0.614	0.711	0.824	0.538	0.651	0.593	0.751	0.279	0.407

5. Conclusion

In this study, we employed various similarity calculation methods to classify images. By setting similarity thresholds, we classified the images based on the similarity between real images and generated images. Different similarity calculation methods were used to evaluate the similarity between real and generated images. We performed similarity calculations on both the original images and the images with added noise. Next, we will investigate calculating similarity based on extracted features. When calculating feature similarity, it is crucial to ensure that the features contain more detailed information to capture the subtle differences between different images.

References

[1] N. K. Singh, K. Raza, Medical image generation using generative adversarial networks: A review, Health informatics: A computational perspective in healthcare (2021) 77–96.

[2] C. Han, H. Hayashi, L. Rundo, R. Araki, W. Shimoda, S. Muramatsu, Y. Furukawa, G. Mauri, H. Nakayama, Gan-based synthetic brain mr image generation, in: 2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018), IEEE, 2018, pp. 734–738.

[3] A. Andrei, A. Radzhabov, D. Karpenka, Y. Prokopchuk, V. Kovalev, B. Ionescu, H. Müller, Overview of 2024 ImageCLEFmedical GANs Task – Investigating Generative Models' Impact on Biomedical Synthetic Images, in: CLEF2024 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, Grenoble, France, 2024.

[4] B. Ionescu, H. Müller, A. Drăgulinescu, J. Rückert, A. Ben Abacha, A. Garcıa Seco de Herrera, L. Bloch, R. Brüngel, A. Idrissi-Yaghir, H. Schäfer, C. S. Schmidt, T. M. Pakull, H. Damm, B. Bracke, C. M. Friedrich, A. Andrei, Y. Prokopchuk, D. Karpenka, A. Radzhabov, V. Kovalev, C. Macaire, D. Schwab, B. Lecouteux, E. Esperança-Rodier, W. Yim, Y. Fu, Z. Sun, M. Yetisgen, F. Xia, S. A. Hicks, M. A. Riegler, V. Thambawita, A. Storås, P. Halvorsen, M. Heinrich, J. Kiesel, M. Potthast, B. Stein, Overview of ImageCLEF 2024: Multimedia retrieval in medical applications, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the 15th International Conference of the CLEF Association (CLEF 2024), Springer Lecture Notes in Computer Science LNCS, Grenoble, France, 2024.

[5] Y. Tokozume, Y. Ushiku, T. Harada, Between-class learning for image classification, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5486–5494.

[6] G. Palubinskas, Image similarity/distance measures: what is really behind mse and ssim?, International Journal of Image and Data Fusion 8 (2017) 32–53.

[7] P.-E. Danielsson, Euclidean distance mapping, Computer Graphics and image processing 14 (1980) 227–248.

[8] P. Xia, L. Zhang, F. Li, Learning similarity with cosine similarity ensemble, Information sciences 307 (2015) 39–52.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Communications of the ACM 63 (2020) 139–144.

[10] D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013).

[11] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution image synthesis with

latent diffusion models, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 10684–10695.

[12] J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2223–2232.