

ViGText: Deepfake Image Detection with Vision-Language Model Explanations and Graph Neural Networks

Ahmad ALBarqawi*, Mahmoud Nazzal[§], Issa Khalil^{||}, Abdallah Khreishah*, and NhatHai Phan*

*New Jersey Institute of Technology, Newark, NJ, USA

[§]Old Dominion University, Norfolk, VA, USA

||Qatar Computing Research Institute (QCRI), HBKU, Doha, Qatar

Emails: aka87@njit.edu, mnazzal@odu.edu, ikhalil@hbku.edu.qa, {abdallah, phan}@njit.edu

Abstract—The rapid rise of deepfake technology, which produces realistic but fraudulent digital content, threatens the authenticity of media. Deepfakes manipulate videos, images, and audio, spread misinformation, blur the line between real and fake, and highlight the need for effective detection approaches. Traditional deepfake detection approaches often struggle with sophisticated, customized deepfakes, especially in terms of generalization and robustness against malicious attacks. This paper introduces **ViGText**, a novel approach that integrates images with Vision Large Language Model (VLLM) Text explanations within a Graph-based framework to improve deepfake detection. The novelty of **ViGText** lies in its integration of detailed explanations with visual data, as it provides a more context-aware analysis than captions, which often lack specificity and fail to reveal subtle inconsistencies. **ViGText** systematically divides images into patches, constructs image and text graphs, and integrates them for analysis using Graph Neural Networks (GNNs) to identify deepfakes. Through the use of multi-level feature extraction across spatial and frequency domains, **ViGText** captures details that enhance its robustness and accuracy to detect sophisticated deepfakes. Extensive experiments demonstrate that **ViGText** significantly enhances generalization and achieves a notable performance boost when it detects user-customized deepfakes. Specifically, average F1 scores rise from 72.45% to 98.32% under generalization evaluation, and reflects the model’s superior ability to generalize to unseen, fine-tuned variations of stable diffusion models. As for robustness, **ViGText** achieves an increase of 11.1% in recall compared to other deepfake detection approaches against state-of-the-art foundation model-based adversarial attacks. **ViGText** limits classification performance degradation to less than 4% when it faces targeted attacks that exploit its graph-based architecture and marginally increases the execution cost. **ViGText** combines granular visual analysis with textual interpretation, establishes a new benchmark for deepfake detection, and provides a more reliable framework to preserve media authenticity and information integrity.

I. INTRODUCTION

Recent advancements in deep learning, particularly in generative models, have enabled the creation of highly realistic synthetic media. The term *deepfake* refers to synthetic content generated by altering or replacing a person’s appearance or voice in images, videos, or audio, which makes it increasingly difficult to distinguish from authentic media [1]. The rise of deepfake technology introduces serious challenges to the accuracy and trustworthiness of digital media, and raises concerns in domains such as politics, media, and entertainment [2]. Along this line, recent reports have highlighted a surge in deepfake pornography targeting young women, including underage individuals, in South Korea [3], [4]. Alarmingly, the Korean Teachers Union reports that over 200 schools have been impacted, with a notable increase in deepfakes which target teachers in recent years, according to the Ministry of Education [5]. In Ukraine, deepfakes have been used to disseminate misinformation and manipulate public perception during the ongoing conflict [6], [7]. This AI-generated content, which convincingly fabricates events or statements, contributes to public confusion and complicates the distinction between truth and deception. As deepfakes become more advanced, the ability to distinguish between real and synthetic content becomes increasingly difficult. As the prevalence of deepfake videos and images grows, it not only fuels the spread of misinformation but also poses significant threats to privacy [8], security [9], and public trust [10]. These concerns have driven extensive research which aim to detect deepfakes, and emphasize the urgent need for effective solutions to address this escalating issue [11], [12], [13], [14].

Prior Approaches and Limitations. Recent efforts in deepfake image detection have mainly relied on learning-based methods. These approaches typically start with a labeled dataset that contains real and deepfake images. The objective is to train a model to detect deepfakes and generalize this detection capability to new, unforeseen deepfakes. Commonly used models include Convolutional Neural Networks (CNNs) and simpler, more traditional models such as feed forward neural networks. [15] uses CNNs to classify image patches,

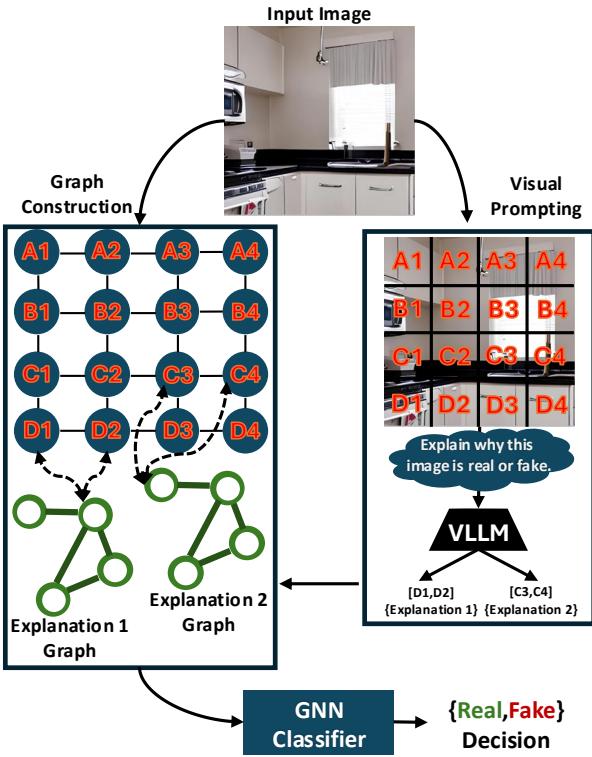


Fig. 1. An overview of the proposed ViGText’s approach.

while [16] enhances detection by capturing global textures with *Gram-Net*. [17] focuses on re-synthesizing images to analyze residual errors, and [18] demonstrates cross-architecture generalization in CNN classifiers. Additionally, [19] presents lightweight CNNs that target mesoscopic image properties. More recently, there has been a shift towards simpler models and novel feature extraction techniques. [20] utilizes frequency domain features with a basic feed-forward neural network, [21] combines image and caption features from a foundation model, and [22] shows that a linear layer trained on features from a large and non-deepfake-specific foundation model can achieve effective classification and generalization. Sifat et al. in [23] analyze current deepfake detection methods and identify key challenges. They demonstrate that many existing approaches fail to generalize effectively when exposed to user-customized or fine-tuned versions of the generative models used during training. Additionally, they highlight that these methods are highly susceptible to adversarial attacks generated with advanced foundation models. These findings point to a significant gap in current techniques, which struggle to adapt to the unpredictable nature of deepfakes in real-world scenarios.

Challenges. Deepfake detection faces solid challenges, particularly in achieving robustness against an ever-evolving threat landscape. The rapid development of generative AI technologies frequently outpaces detection capabilities, which leaves defenses unprepared to address new attack vectors [23]. Another key challenge lies in the integration of textual and visual data. Current approaches rely on image captions that

lack the depth and specificity needed to capture the subtle characteristics required for effective detection [21]. Even in scenarios where detailed textual data is available, it is non-trivial to integrate this information with visual data in a meaningful way [21]. Traditional methods often concatenate visual and textual embeddings in a straightforward manner, that fails to fully utilize their complementary properties. Another challenge is the need for generalizability across a wide variety of deepfake models. Existing methods frequently exhibit significant performance degradation when applied to fine-tuned or customized generative models, which emphasizes their lack of adaptability to novel threats [23]. To overcome these challenges, it is critical to explore advanced integration strategies that can effectively combine textual and visual information, and integrate their complementary strengths.

An Overview of the Proposed ViGText. To overcome the above challenges, we present ViGText, a new deepfake detection approach that integrates image analysis and text-based explanations from a Vision Large Language Model (VLLM) in a graph-based framework, as shown in Fig. 1. The novelty of ViGText lies in its ability to unify visual and textual analysis through tailored graph construction, which allows for the detection of subtle inconsistencies with enhanced generalization and robustness. The process begins as the input image is divided into square patches, each represented as a node in an *image graph*. Each patch is embedded while taking into account spatial and frequency information, with the latter extracted using the Discrete Cosine Transform (DCT). Edges are then added to connect adjacent patches, to capture local spatial dependencies. Alongside the image graph, an *explanation graph* is constructed with the use of a VLLM to generate textual explanations for the patches. Explanation graphs are integrated with the image graph such that each explanation is connected to the patches it describes, which forms a dual-graph structure. This dual graph is then analyzed by a Graph Neural Network (GNN) to determine whether the image is real or fake. While recent methods use image captions for deepfake detection [21], ViGText integrates detailed textual explanations within a graph structure, combining spatial and frequency features to improve generalization to fine-tuned models and robustness against adversarial images.

Summary of Contributions. This work presents the following contributions.

- **Introduction of a Dual-Graph Framework for Enhanced Detection:** We propose ViGText, a novel approach that unifies image analysis with textual explanations generated by VLLMs. ViGText embeds each image patch as it considers both spatial and frequency features, then organizes the visual and textual data into a dual-graph structure which achieves a more robust integration for deepfake detection.
- **Enhanced Generalization to Diverse Generative Models:** ViGText achieves superior generalization across a wide variety of user-customized, fine-tuned variants of generative models without the need to train on the base model images. Through effective integration of context-aware explanations and frequency-domain features in the graph frame-

work, ViGText mitigates the substantial performance degradation observed in prior methods when confronted with user-customized models.

- **Robustness Against Evolving Threats:** ViGText demonstrates strong resilience against adversarial attacks, that include novel foundation model-based threats. This robustness addresses vulnerabilities where existing methods fail against adversarial manipulations intentionally crafted through advanced vision foundation models.
- **Extended Testing on Generalization Datasets:** To evaluate generalization in diverse scenarios, we introduce an extended dataset that comprises eight new testing sets derived from user-customized fine-tuned variants of the Stable Diffusion 3.5 model [24]. This expansion, alongside pre-existing datasets, enables a more comprehensive assessment of detection performance across a broader range of generative models.

II. BACKGROUND

Deepfakes. Deepfakes are a form of synthetic media where artificial intelligence (AI) is used to create hyper-realistic but fake images, videos, or audio recordings. The technology behind deepfakes involves advanced machine learning techniques, particularly deep learning algorithms, which analyze large datasets of real images or audio to generate new, highly convincing content [25], [26]. Initially developed for entertainment and creative purposes, deepfakes have rapidly evolved, raising significant ethical, legal, and societal concerns [27], [28]. Deepfakes can be used maliciously to fabricate videos of individuals saying or doing things they never actually did, leading to potential harm, such as misinformation, identity theft, and reputational damage [29], [30]. The growing accessibility of deepfake technology has sparked global debates on the need for regulation, detection methods, and public awareness to mitigate the risks associated with this powerful technology.

Graph Neural Networks. Graph neural networks (GNNs) [31] have significantly advanced the field of deep learning by extending it to graph-structured data. These models process messages across graph edges and aggregate this information at nodes. The workflow of a GNN entails extracting low-dimensional embeddings from a graph inputs utilizing both local node features and graph topology. GNNs are known to be effective as classifiers in various domains. For instance, in fraud detection [32], they utilize the relational information in transaction networks. Similarly, in drug discovery [33], GNNs help in understanding molecular structures. They have also been successfully applied in social network analysis [34] and recommendation systems [35], demonstrating their versatility and strength as classifiers. The ability of GNNs to handle complex relational data and their adaptability to different types of graph-structured information make them a powerful tool in numerous state of the art (SoTA) applications.

NLP Using GNNs GNNs have emerged as a powerful tool in Natural Language Processing (NLP) by effectively capturing complex dependencies within text through graph representations. Unlike traditional sequence-based models, GNNs

enable the modeling of syntactic and semantic relationships by representing words, sentences, or documents as nodes in a graph, with edges characterizing various linguistic connections [36], [37]. This approach has proven particularly effective in tasks such as relation extraction [38], text classification [39], [40], and sentiment analysis [41], where understanding the intrinsic structure of language is crucial.

Vision Large Language Models and Visual Prompting

Vision Large Language Models (VLLMs) represent a rapidly advancing area of artificial intelligence that combines visual and textual data to perform a wide range of tasks, including image captioning [42], [43], question answering [43], classification, and segmentation [44]. A notable recent development in this field is the concept of visual prompting [45], [46]. This technique involves using specific visual instructions to guide a model's interpretation or generation of text, akin to how text prompting is used in natural language processing to generate responses based on textual prompts. In visual prompting, the model is given an image or a modified version of an image, rather than relying solely on textual input. Recent research has demonstrated the effectiveness of visual prompting in improving the adaptability and performance of AI models across various tasks. By incorporating visual instructions such as marks or annotations, these techniques assist models in tackling complex tasks, such as robotic manipulation [47], image processing [48], and perception [49], without requiring additional fine-tuning.

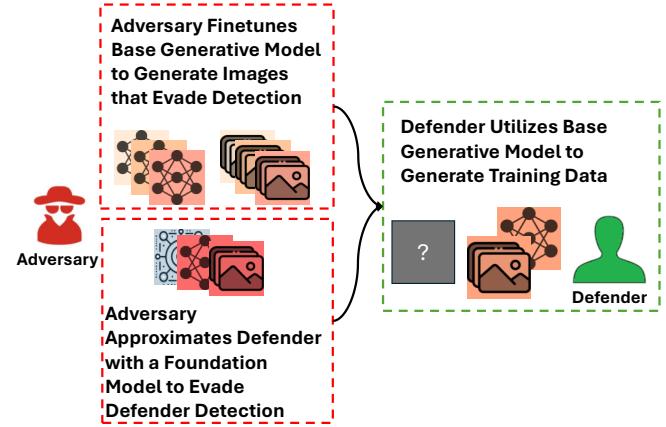


Fig. 2. An Illustration of the threat model.

III. THREAT MODEL AND ASSUMPTIONS

The threat model includes an adversary (malicious actor) who generates deepfakes and targets detection evasion, while a defender tries to detect these deepfakes. This interaction is depicted in Fig. 2. We characterize the threat model by detailing the objectives, knowledge, and capabilities of both the adversary and the defender. The adversary's objective is to produce deepfakes that evade detection by the defender's system. The adversary possesses advanced generative technologies, including the ability to fine-tune base models to create diverse and realistic deepfakes. Recent parameter-efficient fine-tuning

techniques, such as Low-Rank Adaptation (LoRA) [50], have made it feasible for adversaries with limited computational resources to create customized variants of foundational models like Stable Diffusion [51]. These fine-tuned variants introduce subtle variations that make detection challenging and degrade performance even without explicit knowledge of the defender’s methods [23]. Moreover, while our experiments involve edits made using StyleCLIP which is a text-driven image editing method operating in the latent space of StyleGAN2 [52], we do not claim these represent real-world partial manipulations such as face swaps or reenactments. Instead, we use these edits to simulate a distinct adversarial behavior: minimal and localized semantic changes that retain the identity and overall image context. This falls within the broader category of fully synthetic images but provides a useful starting point for studying targeted manipulations that aim to evade detection through subtle changes rather than drastic ones. These controlled manipulations help us study the challenge of detecting small but meaningful changes, adding a different kind of adversarial tactic to our threat model.

As for the defender, its objective is to efficiently detect deepfakes and generalize detection capabilities across a wide range of fine-tuned variants. Access to the base generative model, from which adversaries derive their fine-tuned variants, is a practical and effective approach because many generative models are publicly available or widely accessible on websites like huggingface [53]. Fine-tuning techniques like LoRA [50] typically modify only specific layers or parameters while retaining the core characteristics of the base model. These shared traits, such as architecture patterns, feature representations, and generative tendencies, remain largely intact across variants. By focusing on these foundational traits, the defender can generalize detection capabilities to fine-tuned variants without the impractical need to train on every possible one. Another objective for the defender is to ensure robustness against adversarial attacks, particularly those generated using advanced foundation models. These models enable adversaries to craft subtle, highly deceptive manipulations that can bypass detection. Robustness is critical to maintain the reliability of detection systems in the face of increasingly sophisticated and adaptive threats. For this objective, the defender aims to create systems that can withstand evolving attack strategies while preserving the integrity of their results.

IV. THE PROPOSED ViGTEXT

In this section, we detail the ViGText approach, which is illustrated in the block diagram in Fig. 3 and involves constructing graphs from both image patches and generated explanations, extracting features, and integrating these graphs before utilizing a GNN for detection. We begin by discussing the motivation for using explanations generated by VLLMs. Following this, we formulate the problem of deepfake detection with the availability of images with text. We then introduce a graph-based framework that combines textual and image data, providing a richer context for analysis. Finally, we describe the process of generating these textual explanations

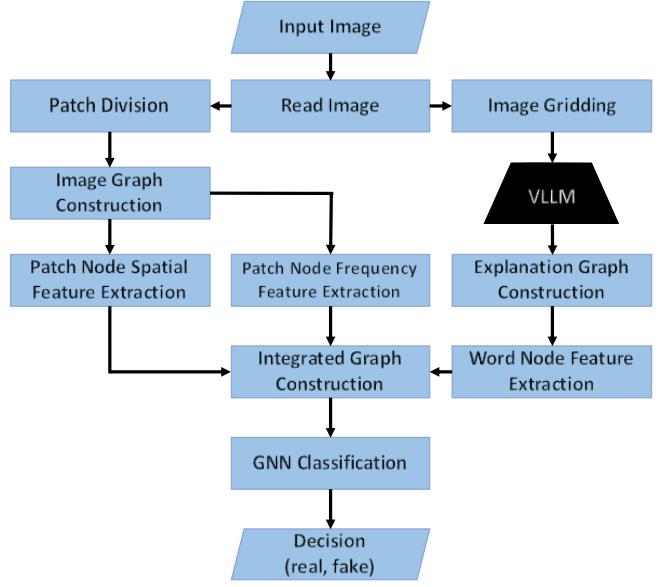


Fig. 3. Block diagram of the proposed ViGText pipeline, illustrating the key components and processes.

and their subsequent integration with images into a unified graph structure.

A. From Captions to Explanations



Fig. 4. Generated image misclassified as real by DE-FAKE [21].

ViGText builds upon a similar concept used in existing techniques like DE-FAKE [21], which incorporates image captions alongside visual data for deepfake detection. However, captions often provide only broad descriptions of the image, which lack the specificity needed to identify inconsistencies. For example, as shown in Fig. 4, DE-FAKE misclassifies a deepfake as real based on the caption "a kitchen and dining

area”, which describes the scene in generic terms without addressing visual details that might indicate manipulation. For instance, a VLLM-generated explanation might state, ”The cabinets and hanging lights show natural reflections and shadows, indicating a real environment”, or ”The table and chairs have detailed textures and consistent lighting, which are characteristic of a real image”.

These detailed explanations capture specific features that contribute to an image’s perceived authenticity. However, VLLMs alone are not capable of accurately classifying images as real or fake [54]. This is where ViGText excels. *Through the analysis of both the visual content and the corresponding explanations, ViGText identifies inconsistencies between the described features and the actual visual elements.* For instance, if an explanation mentions realistic shadows and reflections, but the image lacks these elements or displays unnatural artifacts, such discrepancies serve as strong indicators that the image may be a deepfake. The combination of detailed explanations with visual analysis allows ViGText to address the limitations of caption-based approaches like DE-FAKE and delivers a more robust and reliable framework for deepfake detection.

B. Problem Formulation

With the above-mentioned explanations, the problem of deepfake detection, with a focus on generalizability and adversarial robustness, requires optimizing a classifier function f that maps an input image I and its corresponding explanation E to a binary output $\{0, 1\}$, where 0 indicates a real image and 1 indicates a fake image. In a machine learning-based solution, the objective is to maximize the accuracy of this classifier over a dataset $\mathcal{D} = \{(I_i, E_i, y_i)\}_{i=1}^n$, where y_i are the ground truth labels as expressed as (1).

$$\begin{aligned} \text{Maximize: } & \frac{1}{n} \sum_{i=1}^n \mathbb{I}(\{(I_i, E_i) = y_i\}) \\ \text{Subject to: } & \min_{\delta \in \Delta} \frac{1}{n} \sum_{i=1}^n \mathbb{I}(f(I_i + \delta, E_i) = y_i) \geq \tau_r, \quad (1) \\ & \mathbb{E}_{(I, E, y) \sim \mathcal{D}_{new}} [\mathbb{I}(f(I, E) = y)] \geq \tau_g, \end{aligned}$$

where I_i are the input images, E_i are the generated explanations, and y_i are the labels (0 for real, 1 for fake). The indicator function $\mathbb{I}(\cdot)$ returns 1 if its condition is true and 0 otherwise. Δ represents allowable perturbations for testing robustness, with τ_r as the required robustness threshold. \mathcal{D}_{new} denotes the distribution of unseen data, τ_g is the generalization threshold that must be met, and \mathbb{E} denotes expectation. This problem requires careful consideration of how images and textual data are integrated to fully grasp the benefits of including textual information. f is the binary classifier characterizing detection, y is the ground truth label (0 for real, 1 for fake).

C. Visual and Textual Integration

The integration of textual and visual data is crucial for effective deepfake detection. While DE-FAKE [21] uses simple concatenation of embeddings for captions and images,

this approach fails to capture detailed interdependencies. In contrast, ViGText employs a graph-based model, which integrates textual explanations and visual data to establish meaningful relationships.

To explore the impact of integration methods, the following experiment is conducted, we compare two approaches: DE-FAKE, which uses explanations as textual input but arbitrarily concatenates their embeddings with image embeddings, and ViGText. Table I summarizes the results of this comparison.

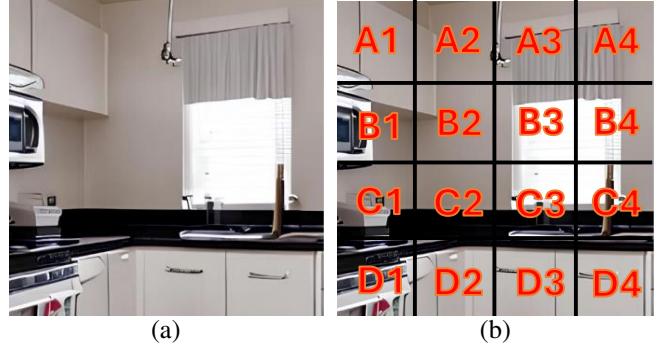


Fig. 5. Image overlay with grid: (a) original, (b) with grid overlay.

TABLE I
PERFORMANCE COMPARISON OF VIGTEXT VS. DE-FAKE [21] WHEN
BOTH APPROACHES ARE GIVEN THE SAME EXPLANATIONS AS THE
TEXTUAL INPUT (THE HIGHEST IS IN BOLD).

	Accuracy	Recall	Precision	F1
DE-FAKE w/Explanations	90.00	91.20	89.00	90.10
ViGText	99.25	99.80	98.52	99.26

The results in Table I show a clear difference in performance. Simply replacing captions with explanations and concatenating their embeddings with image features does not lead to accurate detection. DE-FAKE achieves an accuracy of 90%, despite the use of richer explanations instead of simple captions. This showcases a key limitation: the arbitrary integration of textual and visual data by concatenation fails to capture their complex interdependencies. Unlike the simple techniques used in DE-FAKE, the graph structure allows ViGText to capture intricate interdependencies between textual explanations and visual features, which enables a more detailed understanding and significantly improves detection performance.

D. Explanation-Patch Integrated Graph Construction

Explanation Generation. The generation of explanations to determine whether an image is real or fake using the VLLM is achieved through a process called *visual prompting*. In ViGText, this involves overlaying the image with a grid of equally sized square patches, each systematically labeled (e.g., A1, A2, A3, A4, B1, etc.). Both the overlaid image and the original image are fed to the VLLM, which enables it to produce explanations that are directly linked to these localized areas. Fig. 5 illustrates this grid overlay. This segmentation framework ensures accurate integration with the

A Sample Textual Explanation

{B3,B4}: The window blinds have uneven spacing, and the light passing through does not align properly with the individual slats, which suggests an error in rendering light and shadows. {D1,D2}: The oven appears to have a distorted handle, and the reflection and shadow around it don't conform to the expected perspective and lighting. {D3}: The drawer underneath the stove has irregular handles that are asymmetrical, which is not typical for kitchen design and could be an oversight by the AI.

Fig. 6. An example for textual explanations, each corresponding to specific patches in the image.

image graph, that links explanations to corresponding patches. Sample generated explanations are shown in Fig. 6, while the prompt template is provided in Fig. 11 in Appendix A.

Grid-based explanations are essential because, while a single full-image explanation can describe the whole scene, it doesn't clarify how to associate parts of the explanation with specific image regions for graph construction. The grid ties each explanation to a patch, enabling precise cross-modal edges. However, this introduces a trade-off: smaller patches capture fine details but may lose global context, while larger patches do the opposite. ViGText balances this by choosing intermediate patch sizes and uses the GNN's message passing to merge local and global cues into coherent, human-understandable reasoning. We analyze this trade-off in more detail in Section V.

It is important to note that ViGText, in its design philosophy, does not rely on the VLLM as a standalone trust anchor. Rather, it uses the VLLM as a fully local, defender-controlled component, avoiding risks tied to external or opaque models. The VLLM provides fine-grained textual descriptions of lighting, geometry, and texture, which are cross-verified against visual patch features within the dual graph structure. During training, the GNN learns from both matching and deliberately mismatched image-explanation pairs, enabling it to detect inconsistencies across modalities. This ensures that ViGText can reveal manipulations through statistical discrepancies between textual and visual cues, even under adversarial scenarios, making the overall system more robust and reliable.

Image Graph Construction. After generating the explanations, ViGText constructs a graph that represents patches of the image and its corresponding explanations. The process begins by building the image graph, where each node represents a patch of the image, and nodes corresponding to adjacent patches are connected by undirected edges, as illustrated in the left-hand side of Fig. 7(b). To represent each patch as a node, ViGText utilizes ConvNeXt-Large [55], a foundation image feature extraction model trained on a subset of the LAION-5B dataset [56]. This model extracts feature embeddings for each patch. Additionally, the DCT-transformed patch, illustrated in the right-hand side of Fig. 7(a), is passed through the same feature extraction model to produce an embedding. Finally, the two embeddings (image and DCT-based) are averaged to

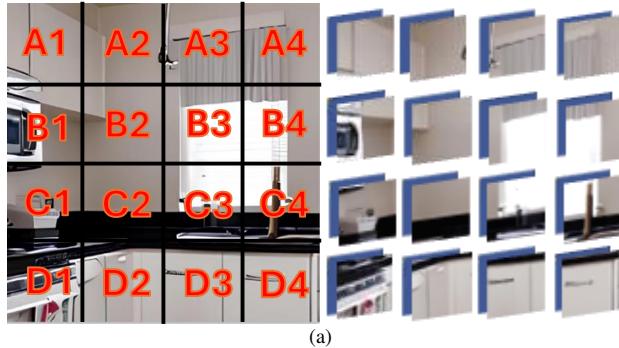
create a robust and comprehensive feature representation for the patch, which is then assigned to the corresponding node in the graph. This dual-domain representation enhances the graph's ability to capture both spatial and frequency-based artifacts, which are crucial for detecting subtle manipulations in deepfake images.

Text Graph Construction. To represent the explanations as graphs, each word in the sentence is depicted as a node, and edges between nodes reflect the grammatical relationships among the words, which are extracted using the dependency parser from spaCy [57]. This structure illustrates how the words interact within the sentence. This method not only captures the roles of the words but also their interactions, resulting in a comprehensive and structured representation of the explanations. ViGText uses Jina [58], an embedding model, to extract features for the words, assigning each node its corresponding embedding. Finally, ViGText integrates these explanation graphs with the image graph by connecting each node in the explanation graph to the corresponding patch node in the image graph. Fig. 7(b) shows a sample explanation graph integrated with its patch nodes.

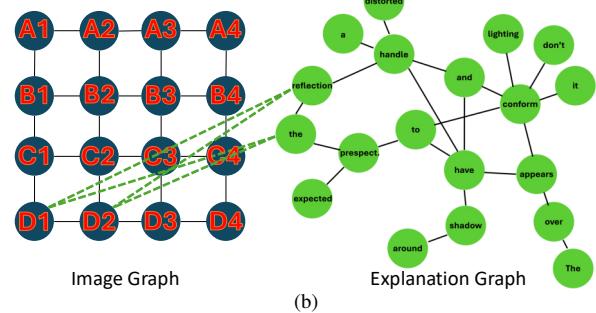
Algorithm 1 Patch and Explanation Integrated Graph Construction

- 1: **Input:** An image I , image feature extraction model M , word feature extraction model B , VLLM.
- 2: **Output:** Patch and Explanation Word Correspondence Graph
- 3: Overlay I with the grid mask to produce the image Q .
- 4: Query the VLLM with Q to produce patch-specific explanations.
- 5: Split I into patches and extract spatial features for each patch using M .
- 6: Apply the DCT transformation to each patch and extract frequency-domain features using M .
- 7: Average the spatial and frequency-domain features to create combined embeddings for each patch.
- 8: Construct the image graph with nodes representing patches and features corresponding to their combined embeddings.
- 9: Construct a graph for each explanation with nodes representing words and edges based on grammatical relationships, extracting word features using B .
- 10: Integrate the image graph with the explanation graphs by connecting each explanation graph node to the corresponding patch node in the image graph.
- 11: **Return** the unified graph containing the image graph and the explanation graphs.

The process of constructing the Patch and Explanation Integrated Graph, detailed in Algorithm 1 and illustrated in Figures 1 and 3, begins by overlaying the image I with a grid mask to produce a modified image Q (Step 3). The modified image is used to query the VLLM, which generates patch-specific explanations associated with the grid regions (Step 4). The image is split into patches, and spatial features are extracted for each patch using the model M (Step 5). Additionally, each patch undergoes a DCT, and frequency-domain features are extracted using the same model (Step 6). The spatial and frequency-domain features are averaged to form the final feature embeddings for each patch (Step 7). These embeddings are used to construct the image graph,



(a)



(b)

Fig. 7. A sample image graph and its corresponding explanation graph construction and integration: (a) The image with the applied grid (left) alongside the patches and their corresponding DCT (right), and (b) The image graph (left) alongside a sample explanation graph (right), illustrating only connecting 2 nodes (all nodes in the explanation graph are connected to the corresponding patch nodes in the actual implementation).

with nodes representing patches and edges connecting adjacent patches (Step 8). For each explanation, a graph is created where nodes correspond to individual words, and edges encode their grammatical relationships. Word features are extracted using the word embedding model B (Step 9). The explanation graphs are then integrated with the image graph by connecting word nodes to their corresponding patch nodes based on the spatial association between explanations and patches (Step 10). The unified graph, which combines both the visual and textual data, is returned as the output (Step 11).

V. EXPERIMENTS

In this section, we evaluate the performance of ViGText through a series of experiments designed to address the questions summarized in Table II. Our results demonstrate that ViGText achieves state-of-the-art detection performance in terms of multiple classification metrics. Moreover, ViGText exhibits strong generalization capabilities, which handle datasets derived from various fine-tuned generative models. Notably, ViGText demonstrates robustness against both foundation model-powered adversarial attacks and targeted attacks crafted with substantial knowledge of its mechanisms. Additionally, the system's performance remains resilient to variations in design choices, indicating a degree of flexibility in its configuration. Overall, ViGText achieves these advancements with a computational cost that remains comparable to, or tolerable in relation to, existing state-of-the-art approaches. The source code and information about the datasets used in this work can be found in the following repository: ViGText.

TABLE II
A SUMMARY OF RESEARCH QUESTIONS AND KEY ANSWERS.

Q	Property Investigated	Key Result
1	Detection effectiveness	Highly effective
2	Generalization	Strong generalization
3	Robustness	High robustness
4	Sensitivity to design choices	Generally insensitive
5	Empirical costs	Tolerable

A. The Setup, Dataset, and Baselines

We use the datasets introduced by Sifat et al. [23], which address critical limitations in existing deepfake research. Specifically, [23] highlights the lack of control over content and image quality in many existing datasets, which can lead to overestimated performance for state-of-the-art detection methods. To mitigate this issue, [23] constructs two carefully curated datasets designed to provide improved control and enable a more accurate evaluation of deepfake detection approaches.

- The Stable Diffusion (SD) Dataset contains real images from the LAION-AESTHETICS dataset [59] and fake images generated using the Realistic Vision v1.4 model [60]. The dataset spans a broad range of content types, including people, nature, objects, illustrations, and digital art. It is structured to ensure balance, with 16,000 images for training, 2,000 for validation, and 2,000 for testing, equally divided between real and fake images. A key focus of this dataset is to evaluate the generalization of deepfake detection approaches to images generated by fine-tuned variants of generative models, as generalization remains a persistent challenge for existing methods. To assess this, the dataset includes 16 additional testing sets derived from the base SD 1.5 model. Of these, 8 testing sets feature images generated using the Full Model (FM) fine-tuning approach—where all parameters are updated—and the remaining 8 are created using Low-Rank Adaptation (LoRA) fine-tuning [50], a computationally efficient method that updates only a subset of the model's parameters. Remarkably, even without specific adversarial intent, the images from these fine-tuned variants cause significant performance degradation in many state-of-the-art detection methods, as evidenced by [23]. This highlights the growing threat posed by the democratization of fine-tuning techniques and the urgent need for robust, generalizable detection systems. Furthermore, we extend the testing on generalization by creating 8 additional testing sets corresponding to 8 new LoRA fine-tuned variants of the Stable Diffusion 3.5 model [24], the current state-of-the-art open-sourced generative model. We choose to include only LoRA fine-tuned variants in this extension due to the significantly larger size of the Stable Diffusion 3.5 model at

8 billion parameters. While LoRA [50] fine-tuning is computationally efficient, fine-tuning all the parameters of such a large model would require substantial computational resources that are impractical for most use cases. Table XI in Appendix B contains more information about the Stable Diffusion 3.5 LoRA models [24]. This extension provides a comprehensive evaluation of detection methods under challenging scenarios posed by the latest advancements in generative AI, further emphasizing the necessity for detection systems to be both robust and adaptive.

- **StyleCLIP Dataset:** This dataset consists of a balanced collection of real and fake face images, specifically designed to study robustness against adversarial attacks using vision foundation models. Real images are sourced from the Flickr-Faces-HQ (FFHQ) dataset [61], a high-quality collection of face images, while fake images are generated using StyleGAN2 [52], a widely adopted generative model for face synthesis. The dataset is balanced, with 16,000 images for training, and 2,000 images each for validation and testing. Unlike the SD dataset, the StyleCLIP dataset emphasizes robustness evaluation under adversarial scenarios. As detailed in [23], adversarial attacks in this context involve manipulating the semantic properties of face images, such as altering expressions, adding accessories, or modifying facial attributes, without introducing perceptible noise. These attacks use vision foundation models as surrogates to optimize manipulations, enabling the generation of adversarial deepfakes that evade detection. To thoroughly evaluate this challenge, three state-of-the-art foundation models, EfficientNet [62], ViT [63], and CLIPResNet [64], are employed as surrogate models. These models are used to create three additional adversarial testing sets, each tailored to exploit the weaknesses of existing deepfake detection approaches. This dataset, therefore, provides a critical benchmark for assessing the resilience of detection systems in adversarial settings, highlighting the vulnerabilities exposed by foundation model-powered attacks.

To extend our evaluation, we implemented a more advanced adversarial attack that simulates an attacker with significant knowledge of ViGText. This hypothetical attacker is assumed to possess detailed insights into the training dataset and the graph creation pipeline used by ViGText. Based on these assumptions, the attacker creates a surrogate model designed to mimic ViGText’s functionality. The surrogate model comprises two Graph Convolutional Layers and uses the Dinov2 [65] and Jina [58] foundation models for extracting image and word embeddings, respectively. This surrogate is trained on the StyleCLIP dataset, achieving high performance across classification metrics, with accuracy, recall, precision, and F1 scores all exceeding 95%. Using this surrogate model, we further train the StyleGAN2 [52] generative model to produce adversarial images. These images are specifically crafted to evade detection by the surrogate and, consequently, by ViGText. The attack optimizes the generator by minimizing the cross-entropy loss between the surrogate logits z and the target label y , with y chosen as the label for real images so

that the generator produces evasive adversarial examples:

$$\mathcal{L}_{\text{adv}} = -\mathbb{E}_{\mathbf{x} \sim G(\theta)} [y \log p(\mathbf{z}) + (1 - y) \log(1 - p(\mathbf{z}))], \quad (2)$$

where $G(\theta)$ is the StyleGAN2 generator with parameters θ , $p(\mathbf{z})$ represents the surrogate model’s output probabilities for the real label, and $\mathbf{x} \sim G(\theta)$ are the generated images. This setup allows us to evaluate ViGText against adversarial images crafted by an attacker that closely mimics the real-world threat of a knowledgeable adversary.

It is important to note that while the StyleCLIP dataset and its surrogate-based attacks effectively test robustness under strong visual manipulations, they do not account for coordinated attacks that simultaneously target both the image and the VLLM-generated explanations. Executing such dual-objective attacks would require white-box access to both the image generator and the VLLM, enabling gradient-based optimization across components. This substantially increases the complexity and computational cost, placing it beyond the scope of our current evaluation and leaving it as an open direction for future research.

We report classical classification metrics for all experiments conducted in this section, which include accuracy, precision, recall, and f1 scores. As for the baselines, we select 3 of the state-of-the-art approaches that perform decently in the analysis in [23]. These approaches are:

- **DCT** [20]: This approach works by extracting frequency-domain features from images using a discrete cosine transform (DCT) to identify subtle artifacts. These features are log-scaled for better performance and then used to train a Logistic Regression classifier, which effectively differentiates between real and fake images.

- **DE-FAKE** [21]: This approach builds a deepfake detector using the CLIP model [64] by augmenting the image’s embedding with the embedding of the text prompt used to generate the image. These augmented embeddings are used to train a 2-layer multilayer perceptron as a classifier.

- **UnivCLIP** [22]: This recent approach utilizes a large foundation model, specifically the CLIP:ViT-L/14 model [64]. This approach extracts features from the frozen CLIP:ViT model and then uses either a nearest neighbor classifier or a linear classification layer, with further training, to determine if an image is real or fake. The linear classifier is preferred here for better performance.

While recent approaches such as ObjectFormer [66] and detectors built on large-scale vision-language models (VLMs) [67], [68] demonstrate promising detection capabilities, they also depend on resource-intensive architectures like dense transformer attention or billion-parameter language models. These designs face scalability limitations that hinder their practicality for widespread deployment. In contrast, our focus is on methods that balance strong detection performance with computational efficiency. ViGText demonstrates state-of-the-art accuracy and robustness on up-to-date, challenging datasets reflecting modern generative techniques, while maintaining a lightweight graph-based architecture, which ensures broader applicability and scalability.

For ViGText, we use a consistent GNN architecture across all experiments. The model comprises three Graph Attention Network (GAT) [69] layers with two attention heads, followed by batch normalization and ReLU activation after each layer. Dropout is applied after each layer to prevent overfitting by regularizing the training process. The node features are aggregated using global mean pooling before being passed to a fully connected layer for final classification.

The model is trained for 40 epochs using the Adam optimizer [70] to minimize Cross Entropy loss, with learning rate scheduling employed to adjust the learning rate dynamically during training. All experiments are conducted with a 4x4 patch size and utilize Qwen2-VL-7B-Instruct [71] as the explanation-generating VLLM. The experiments are performed on a workstation equipped with 64 GB of RAM, an 8 GB RTX 2070 GPU, and a 32-core Intel Xeon processor.

B. Performance Analysis

TABLE III

PERFORMANCE ANALYSIS ON THE RESPECTIVE TESTING SETS OF THE DATASETS (THE HIGHEST IS IN BOLD).

Approach	SD				StyleCLIP			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
DCT	85.50	83.30	88.80	85.96	98.80	98.22	99.40	98.80
DE-FAKE	92.45	91.17	94.00	92.5	74.05	75.34	71.50	73.37
UnivCLIP	93.04	92.33	93.89	93.10	93.04	93.79	92.19	92.99
ViGText	99.25	99.8	98.52	99.26	99.60	99.90	99.21	99.60

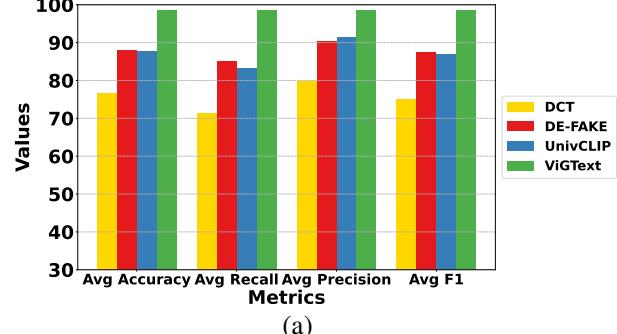
We begin by addressing **Q1: How effective is ViGText at detecting deepfakes?** To evaluate this, we compare the performance of ViGText against the baselines mentioned above using the specified quality metrics across both datasets. As shown in Table III, ViGText consistently outperforms the latest state-of-the-art techniques, demonstrating a strong capability to detect deepfakes generated by various approaches and reflecting the practical threat landscape of deepfake techniques.

These results highlight the effectiveness of ViGText's unique integration of visual and textual information through a dual-graph structure. Through the use of spatial and frequency embeddings, as well as detailed context-aware textual explanations, ViGText achieves superior detection performance. Additional experiments on images from state-of-the-art diffusion APIs and under advanced adversarial attacks are reported in Appendix C, and further sample cases only detected by ViGText are illustrated in Appendix D.

C. Generalization

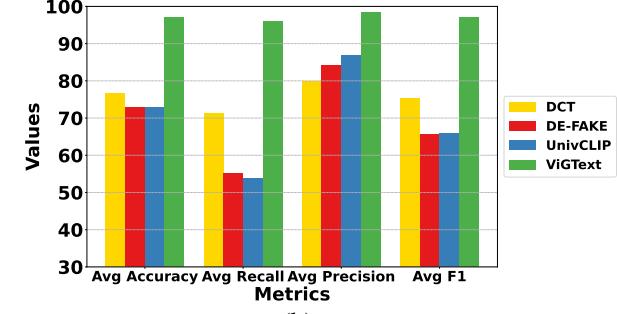
Here, we address **Q2: Can ViGText generalize well enough to detect images generated by various fine-tuned variants?** This evaluation focuses primarily on the SD dataset. ViGText is trained using the training data from this dataset and tested on 24 separate testing sets corresponding to fine-tuned variants of different SD models. These testing sets are split between Full Model (FM) fine-tuned and LoRA fine-tuned variants. Fig. 8(a) shows the average performance metrics across the 8 FM fine-tuned variants of the SD 1.5 model, while Fig. 8(b) and Fig. 8(c) illustrate the results

Performance on Stabe Diffusion 1.5 FM Fine-tuned Models



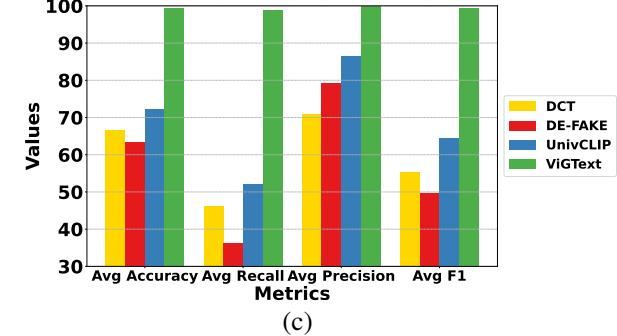
(a)

Performance on Stabe Diffusion 1.5 LoRA Fine-tuned Models



(b)

Performance on Stabe Diffusion 3.5 LoRA Fine-tuned Models



(c)

Fig. 8. Generalizing performance on (a) Stable Diffusion 1.5 FM fine-tuned models, (b) Stable Diffusion 1.5 LoRA fine-tuned models and (c) Stable Diffusion 3.5 LoRA fine-tuned models.

for the 16 LoRA fine-tuned variants of the SD 1.5 and 3.5, respectively.

As illustrated in Fig. 8, ViGText demonstrates superior generalization performance compared to baseline methods across all metrics, which showcases its ability to detect fake images generated by diverse fine-tuned models. This is particularly significant given that traditional data-driven approaches often exhibit decreased performance when tested on data with altered distributions, such as those arising from fine-tuned generative models. In contrast, ViGText mitigates this limitation through its graph-based framework, which focuses on learning the structural topologies of the graphs derived from data points. Additionally, ViGText uses frequency domain features that are content-agnostic, capturing subtle features that are invariant to the content of the images. These features play a critical role to enhance generalization as they reduce dependence on the underlying data distribution or the characteristics

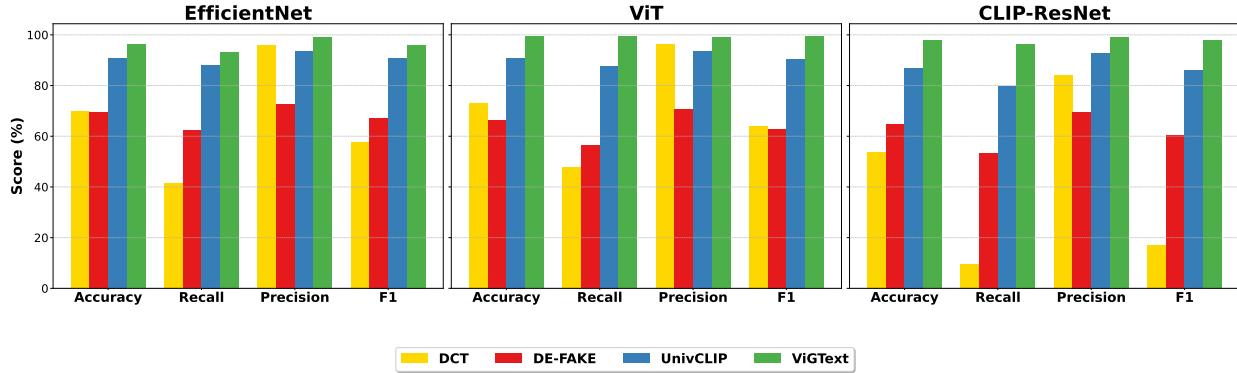


Fig. 9. Bar plots illustrating the robustness performance of ViGText and baseline approaches on adversarially manipulated testing sets of the StyleCLIP dataset. Each plot corresponds to a different surrogate foundation model—EfficientNet, ViT, and CLIP-ResNet—used to craft the adversarial manipulations.

of the generative models used during training. ViGText combines its graph-based architecture with frequency domain analysis, to effectively adapt to varied and challenging test scenarios, which reinforces generalization.

D. Robustness

In this subsection, we address **Q3: How robust is ViGText against foundation model-based adversarial attacks and manipulations on images?** For this purpose, we focus on the StyleCLIP dataset and its associated adversarially manipulated testing sets. These testing sets are crafted using a state-of-the-art adversarial attack [23], which uses foundation models trained on extensive datasets to generate manipulations that reduce the performance of conventional detection methods. The manipulations aim to exploit subtle semantic properties of the images, making the attacks more effective without introducing perceptible noise.

We train ViGText on the training portion of the StyleCLIP dataset, then test on these manipulated sets, with each testing set corresponding to a different surrogate foundation model, EfficientNet, ViT, and CLIP-ResNet, used to generate the adversarial attacks. As shown in Fig. 9, ViGText consistently outperforms other methods across all evaluated metrics. This demonstrates that ViGText exhibits significantly less performance degradation when faced with adversarial attacks compared to baseline approaches. The resilience of ViGText is particularly noteworthy given the sophistication of the attacks, which utilize foundation models trained on millions of diverse images to approximate high-quality surrogates. This inherent robustness makes ViGText an effective solution for combating adversarial manipulations.

Next, we evaluate the robustness of ViGText in a scenario where the adversary has substantial knowledge about the detection system. The adversary is assumed to have access to the same training dataset (StyleCLIP dataset) and the same pipeline used for to construct graph structures between images and textual explanations. This setup models a highly capable and informed adversary to rigorously assess ViGText’s resilience under such a challenging threat model. To simulate this scenario, we design a surrogate detection model with a reasonably chosen architecture that mimics the characteristics

of ViGText. The surrogate model is trained using the StyleCLIP dataset and achieves over 95% on all detection metrics (accuracy, precision, recall, and F1 score) when evaluated on the StyleCLIP testing set. This high-performing surrogate is then used to generate adversarial images by optimizing the StyleGAN2 [52] generator to minimize the cross-entropy loss between the surrogate’s logits and the target label (real images), which effectively crafts evasive examples.

We test ViGText on these adversarial images, and it achieves an accuracy of **95.85%**, recall of **91.7%**, precision of **99.2%**, and an F1 score of **95.67%**, compared to the original metrics of accuracy **99.6%**, precision **99.9%**, recall **99.21%**, and F1 score **99.6%**. While these adversarial images have a greater impact on ViGText’s performance compared to foundation model-based attacks, this outcome is expected as the surrogate closely mirrors the actual detection system, which provides the adversary with a considerable advantage.

The previous results highlight ViGText’s robustness, even against an adversary with significant knowledge of its design and training data. The graph-based framework, integration of frequency-domain features, and effective combination of visual and textual information enable ViGText to maintain strong detection performance under this extreme threat model. This demonstrates ViGText’s capability to withstand attacks from resourceful and well-informed adversaries.

TABLE IV
PERFORMANCE WITH SD AS THE TEST SET ACROSS DIFFERENT IMAGE RESOLUTIONS (THE HIGHEST IS **BOLD**).

Resolution	450x450				512x512				550x550			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
DCT	46.60	6.20	32.29	10.40	85.50	83.30	88.80	85.90	51.60	6.20	67.39	11.35
DE-FAKE	93.40	93.90	92.97	93.43	92.40	91.10	94.00	92.50	93.45	94.10	92.89	93.49
UnivCLIP	92.29	92.09	92.46	92.28	93.00	92.30	93.90	93.10	92.44	92.19	92.66	92.43
ViGText	96.40	99.90	93.28	96.53	99.25	99.80	98.52	99.26	97.20	95.20	99.17	97.14

Continuing with the robustness evaluation, we test ViGText against manipulations caused by changes in image resolution. This is an important aspect of robustness, as practical applications often involve input images that vary in resolution due to diverse capture conditions or post-processing steps. Table IV presents the results on the SD dataset, which originally consists of 512x512 resolution images. While Ta-

ble XIV shows the corresponding results for the StyleCLIP dataset, which originally consists of 1024x1024 resolution images.

The results in Table IV demonstrate that ViGText achieves minimal degradation in performance across all tested resolutions on the SD dataset. This consistent performance highlights its robustness to resolution changes, with accuracy, precision, recall, and F1 scores remaining high. Notably, ViGText outperforms all baseline methods at each resolution, which suggests that its graph-based framework contributes to its ability to adapt effectively to variations in image resolution.

TABLE V

PERFORMANCE WITH SD AS THE TEST SET ACROSS DIFFERENT GEOMETRIC WARP OPERATIONS (THE HIGHEST IS IN BOLD).

Technique	Rotate				Scale and Translate			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
DCT	54.6	54.9	51.4	53.0	57.5	62.5	37.4	46.8
DE-FAKE	86.8	81.7	95.0	87.8	89.2	86.0	93.6	89.6
UnivCLIP	88.1	84.0	94.1	88.8	90.9	86.6	96.9	91.5
ViGText	98.0	96.1	100.0	98.0	99.6	99.9	99.2	99.6

Next, Tables V and VI present the performance of all evaluated methods on the SD dataset under geometric and appearance-based warp operations, respectively. ViGText demonstrates consistently superior metrics across these transformations, indicating robust resilience to spatial distortions. In particular, ViGText maintains high accuracy, precision, recall, and F1 scores even when subjected to significant geometric modifications, showcasing its adaptability in real-world scenarios where images may be rotated or spatially transformed. For completeness, we also include supplementary experiments that cover variations in resolution as well as geometric and appearance-based warp operations on the StyleCLIP dataset in Appendix E, further demonstrating the consistency of ViGText’s performance across different datasets and manipulation types.

TABLE VI

PERFORMANCE WITH SD AS THE TEST SET ACROSS DIFFERENT APPEARANCE-BASED WARP OPERATIONS (THE HIGHEST IS IN BOLD).

Technique	Blurring				Brightness			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
DCT	80.00	67.00	90.54	77.01	81.2	78.7	85.6	81.9
DE-FAKE	92.60	95.70	90.11	92.82	92.2	92.7	91.7	92.2
UnivCLIP	92.74	96.39	89.84	93.01	91.4	90.9	92.2	91.5
ViGText	97.60	99.90	95.42	97.66	99.6	99.2	99.9	99.6

Finally, Table VII reports results for adversarial robustness using the Fast Gradient Sign Method (FGSM) [72] and Projected Gradient Descent (PGD) [73] attacks at varying noise levels. Here, ViGText maintains a clear advantage over the other methods, achieving the highest accuracy in all tested conditions. Even at higher noise magnitudes, ViGText exhibits a notable margin of improvement compared to competing approaches. This strong adversarial resilience, combined with

TABLE VII
ACCURACY AGAINST ADVERSARIAL IMAGES USING FGSM AND PGD ATTACKS (THE HIGHEST IS IN BOLD).

Attack	Noise (ϵ)	ViGText			
		DCT	DE-FAKE	UnivCLIP	ViGText
FGSM	0.0001	82.41	88.21	82.37	96.43
	0.001	75.78	83.02	54.99	93.71
	0.01	71.09	71.48	37.19	89.19
PGD	0.0001	35.16	63.52	61.84	91.46
	0.001	16.24	61.04	56.55	87.83
	0.01	9.47	58.36	51.31	80.94
No Attack	—	85.50	92.45	93.04	99.25

its robustness to geometric and appearance-based transformations, confirms the suitability of ViGText for handling diverse and challenging manipulations in practice.

E. Sensitivity to Design Choices

In this subsection, we investigate **Q4: How sensitive is ViGText to the design choices made during its development?** We explore the sensitivity of ViGText to the number of patches into which the image is divided, and evaluate the effect of the number of patches on performance across different datasets and scenarios.

TABLE VIII
PERFORMANCE ON SD DATASET AND STYLECLIP DATASET ACROSS DIFFERENT PATCH SIZES

Patches	SD Dataset				StyleCLIP Dataset			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
3x3	98.00	96.15	99.90	98.04	99.60	99.21	99.90	99.60
4x4	99.25	99.80	98.52	99.26	99.60	99.90	99.21	99.60
5x5	98.40	96.90	99.90	98.43	99.90	99.90	99.80	99.90

Table VIII summarizes the results on the SD and StyleCLIP datasets when we vary the patch size. The results show that the performance on both datasets remains relatively stable across patch sizes, indicating that ViGText is not overly sensitive to this parameter. However, moving to the generalization performance on the fine-tuned model datasets in Table IX, we observe a notable increase in performance as the patch size decreases (and consequently, the number of patches increases).

TABLE IX
PERFORMANCE ON LORA AND FM FINETUNED MODELS ACROSS DIFFERENT NUMBERS OF IMAGE PATCHES.

Patches	SD 1.5 LoRA				SD 1.5 FM				SD 3.5 LoRA			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
3x3	92.68	98.92	91.85	95.19	97.06	98.98	97.32	98.13	98.96	99.00	99.70	99.35
4x4	97.25	96.00	98.45	97.18	98.50	98.49	98.45	98.49	99.30	99.79	98.80	99.28
5x5	98.70	99.19	99.17	99.18	99.26	99.20	99.87	99.53	99.35	99.20	99.99	99.60

The increase in performance as the number of patches increases can be attributed to the nature of the artifacts present in these images. As shown in Fig. 10, images generated by LORA fine-tuned models (Fig. 10(a)) and FM fine-tuned models (Fig. 10(b)) contain localized artifacts that are better

captured with smaller patches. When we decrease the patch size the spatial granularity of the graph increases, which allows the model to better localize and represent these distortions. Smaller patches also create more graph nodes, which enhances the model’s sensitivity to subtle variations in localized regions, further improving its ability to generalize.

TABLE X

AVERAGE PERFORMANCE ON THE 3 ADVERSARILY MANIPULATED TESTING SETS OF THE STYLECLIP DATASET ACROSS DIFFERENT NUMBER OF IMAGE PATCHES.

Patches	Acc	Prec	Rec	F1
3x3	99.30	99.20	99.40	99.30
4x4	97.76	96.33	99.18	97.71
5x5	86.56	98.56	74.13	83.23

We notice a different behavior, as seen in Table X, where the performance on adversarially manipulated testing sets of the StyleCLIP dataset exhibits the opposite trend, with larger patch sizes yielding better results. This behavior is likely because adversarial images (Fig. 10(c)) lack clear artifacts, and smaller patches may fail to capture meaningful features. For these images, the signal within individual patches is weaker, and the increase in the number of graph nodes can dilute the signal, which makes it harder for ViGText to distinguish between real and fake content. Additionally, smaller patches reduce the ability of the graph to capture global patterns, which are crucial to detect adversarial manipulations designed for evasion.

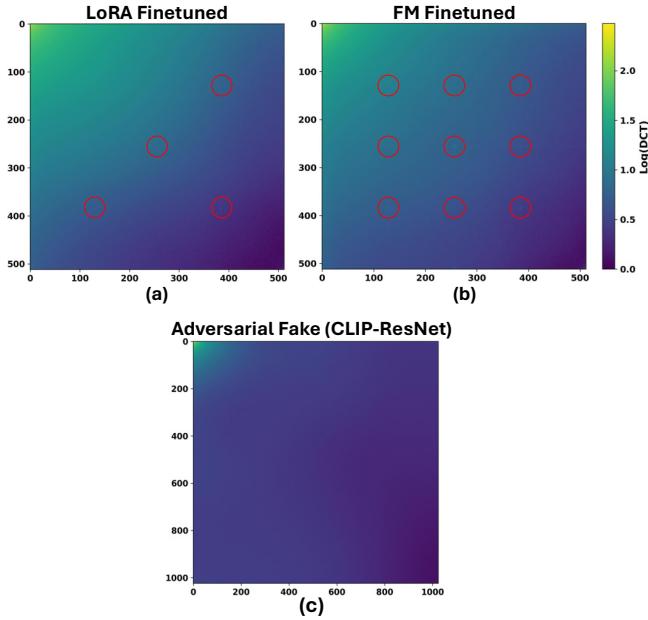


Fig. 10. Log DCT frequency spectrum showing (a) artifacts for images generated using LoRA fine-tuned models, (b) artifacts for ones generated using FM fine-tuned models and (c) no artifacts in the adversarial generated images. Red circles showcasing the artifacts.

The sensitivity of ViGText to patch size is closely tied to the nature of the images being analyzed. For artifact-rich images, smaller patches enhance performance by capturing

localized distortions, while for artifact-free adversarial images, larger patches are more effective as they preserve global context. This suggests that an adaptive patching strategy, which dynamically adjusts patch size based on the characteristics of the input image, could further enhance ViGText’s robustness and generalization capabilities.

F. Cost Analysis

Finally, we address **Q5: How costly is it to run ViGText?** We compare ViGText to UnivCLIP, which demonstrated the second-best overall performance across the previous experiments. The time cost is measured as the total time required to preprocess the image, run the model inference, and, for ViGText, includes graph construction. These results are averaged over 2000 test samples.

On average, UnivCLIP takes 1.650 seconds per image from preprocessing to inference, while ViGText takes only slightly more at 1.755 seconds. This marginal increase in time cost, just 0.105 seconds, highlights the efficiency of ViGText. The results demonstrate that ViGText achieves its superior detection performance with minimal additional computational cost. The use of advances in VLLMs, such as Qwen2 [71], significantly contributes to this efficiency. Qwen2-VL-7B-Instruct is a compact yet powerful model that provides a deep understanding of image content and generates detailed textual explanations with low computational overhead. Through these advances, ViGText is able to integrate high-quality explanations into its graph-based framework while maintaining a competitive runtime.

These findings showcase that ViGText not only excels in detection accuracy and robustness but also remains practical for real-world deployment, as it achieves state-of-the-art performance with a near-negligible increase in time cost.

VI. RELATED WORK

Deepfake Detection. Recent deepfake detection approaches can be broadly categorized based on the model architecture into two main approaches: CNN-based approaches and traditional machine learning models-based approaches. CNN-based approaches focus on utilizing deep learning architectures to detect fake images. For instance, [15] explores the characteristics of fake images that enable detection across different generative models and datasets. This study introduces a patch-based classifier with limited receptive fields, which emphasizes local image artifacts rather than global structures. [16] presents Gram-Net, a model that enhances fake face detection by concentrating on global texture features, which improves robustness and generalization across various GAN models and image distortions. [17] also aims to improve robustness and generalization but takes a different approach by re-synthesizing images through tasks like super-resolution, denoising, and colorization, instead of relying solely on frequency artifacts. Additionally, [19] targets facial video forgeries using two lightweight CNN architectures that analyze mesoscopic properties—features that lie between fine details and high-level content.

Traditional machine learning-based approaches have recently shifted focus from complex deep learning architectures to advanced feature extraction techniques to train simpler models for deepfake detection. In [20], the authors discuss using frequency domain features to detect diffusion model deepfakes, and highlight how these features can reveal subtle artifacts that are not easily visible in the spatial domain. Similarly, [21] proposes to combine text prompt features with generated image features as input to a simple classifier, which allows it to learn associations between textual and visual content for accurate fake image detection. Lastly, [22] introduces an approach to detect fake images generated by various models, which includes those unseen during training, by utilizing a feature space not specifically trained for deepfake detection. The authors in [23] examine the previously mentioned approaches, and highlight their vulnerabilities in the face of new threats. These threats primarily involve easily accessible user-customized generative models and adversarial deepfakes created using foundation models. The study demonstrates that these approaches struggle to generalize when faced with images generated by user-customized variants. Furthermore, there is a significant degradation in performance when trying to detect images that have been adversarially generated to evade detection with the use of large foundation models.

VII. DISCUSSION

Impact of the Study. The impact of this study lies in its substantial advancement of deepfake detection technology. By integrating visual and textual data through a graph-based framework, this approach directly enhances the reliability and robustness of detection methods against challenges posed by evolving generative models. The methodology improves the generalization capability to user-customized models and enhances resilience against sophisticated adversarial attacks. These advancements are particularly critical to safeguard media authenticity and trustworthiness, which are under increasing threat from the spread of AI-generated synthetic content.

Beyond deepfake detection, the study's methodology can be adapted to applications which require the differentiation of real from fake or harmful from beneficial phenomena. Processing data alongside textual explanations enriches analysis by providing contextual insights and enhancing interpretability. For instance, in toxic chemical identification, explanations can detail features associated with toxicity. In drug discovery, combining biological imagery with textual descriptions can uncover relationships between structures. Similarly, in content moderation, document verification, and fake news detection, textual explanations can complement visual data to enable more comprehensive and reliable assessments.

Limitations. While the study achieves notable progress, several limitations must be acknowledged. First, the current threat model and evaluation focus on fully synthetic images, including latent-space manipulations such as those produced by StyleCLIP [23]. Although these can simulate localized semantic edits, they do not represent real-world partial manipulations such as face swaps, reenactments, or spliced content.

Second, the framework is currently limited to the visual modality and does not yet support audio or video inputs. Extending to these modalities requires addressing new challenges such as modeling temporal dynamics, aligning asynchronous multimodal signals, and handling variability in recording quality and noise. Finally, from a technical perspective, the use of fixed-size patches in graph construction and word-to-patch linking strategies may limit adaptability to image content complexity and reduce contextual precision, pointing to clear areas for future improvement.

Future Work. Future directions include enhancing the alignment between image patches and textual explanations through adaptive linking mechanisms that consider contextual dependencies more effectively. Adaptive patching strategies, where patch size and graph connectivity adjust dynamically based on image content, may improve scalability and localization precision. Incorporating heterogeneous graph neural networks is another promising step, enabling more expressive representations by modeling diverse node and edge types. While this work focuses on images, extending the approach to other modalities such as audio and video introduces substantial challenges. Video-based deepfakes require temporal modeling to track spatial and motion coherence across frames, while audio-based detection involves extracting meaningful acoustic features that align with visual or textual cues. Additionally, multimodal fusion presents issues of synchronization, varying signal quality, and modality-specific adversarial attacks. Addressing these complexities would require redesigning the model architecture to process time-series data, support multi-stream alignment, and remain robust to cross-modal inconsistencies. Lastly, building on frequency-domain robustness, future work can explore the use of learned transformation bases or multi-resolution representations to better generalize across both synthetic and real-world manipulations and improve resistance to adversarial perturbations.

VIII. CONCLUSION

In this work, we introduce ViGText, a framework for deepfake detection that integrates explanations from VLLMs with visual data in a dual-graph structure. This novel approach addresses critical limitations in existing methods, as it demonstrates exceptional generalization to fine-tuned and user-customized deepfakes, robust resistance to adversarial manipulations, and adaptability to diverse testing scenarios. Through the use of graph-based representations that combine spatial and frequency domain features with detailed textual explanations, ViGText achieves state-of-the-art detection performance. ViGText distinguishes itself by seamlessly bridging visual and textual modalities, which enables it to detect subtle artifacts and inconsistencies that trouble traditional methods. Despite ViGText's advanced capabilities, it maintains computational efficiency, that ensures practical feasibility for real-world deployment. This makes ViGText a scalable solution capable of addressing the growing challenges posed by the rapid evolution of generative models and adversarial techniques. As synthetic media technologies continue to

advance, the need for robust and adaptable detection systems becomes increasingly urgent. ViGText not only addresses the growing challenges of deepfake detection but also builds a strong foundation for future advancements in the field. It plays an important role in protecting the digital world from deception and ensuring trust in online content.

ETHICS CONSIDERATIONS

This study does not use any sensitive data regarding privacy and security. We use open-source datasets to conduct our experiments. This research was conducted with a commitment to upholding the highest ethical standards. The methodologies employed were carefully designed to ensure that the research respects the privacy and rights of individuals, avoids bias, and minimizes potential harm. Data used in the study was sourced and managed responsibly, adhering to ethical guidelines that prioritize confidentiality and the ethical use of information. The research aims to contribute positively to defending deepfakes towards Safe, Secure, and Trustworthy AI.

REFERENCES

- [1] M. Westerlund, “The emergence of deepfake technology: A review,” *Technology innovation management review*, vol. 9, no. 11, 2019.
- [2] C. Vaccari and A. Chadwick, “Deepfakes and disinformation: Exploring the impact of synthetic political video on deception, uncertainty, and trust in news,” *Social media+ society*, vol. 6, no. 1, p. 2056305120903408, 2020.
- [3] thestreetjournal, <https://thestreetjournal.org/south-korea-faces-deepfake-porn-crisis/>, [Accessed 28-08-2024].
- [4] yahoo, “South Korea faces deepfake porn ‘emergency’ — ca.news.yahoo.com,” <https://ca.news.yahoo.com/south-korea-faces-deepfake-porn-072159810.html>, [Accessed 28-08-2024].
- [5] bbc, “South Korea faces deepfake porn ‘emergency’ — bbc.com,” <https://www.bbc.com/news/articles/cg4yerrg451o>, [Accessed 28-08-2024].
- [6] J. J. Twomey, “Deepfakes in warfare: new concerns emerge from their use around the Russian invasion of Ukraine — theconversation.com,” [Accessed 01-09-2024].
- [7] https://www.nytimes.com/by/edward_wong, “Deepfake of U.S. Official Appears After Shift on Ukraine Attacks in Russia — nytimes.com,” <https://www.nytimes.com/2024/05/31/us/politics/deepfake-us-official-russia.html>, [Accessed 01-09-2024].
- [8] M. B. Kugler and C. Pace, “Deepfake privacy: Attitudes and regulation,” *Nw. UL Rev.*, vol. 116, p. 611, 2021.
- [9] K. A. Pantserov, “The malicious use of ai-based deepfake technology as the new threat to psychological security and political stability,” *Cyber defence in the age of AI, smart societies and augmented humanity*, pp. 37–55, 2020.
- [10] M. Pawelec, “Deepfakes and democracy (theory): How synthetic audio-visual media for disinformation and hate speech threaten core democratic functions,” *Digital society*, vol. 1, no. 2, p. 19, 2022.
- [11] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, “Deepfakes and beyond: A survey of face manipulation and fake detection,” *Information Fusion*, vol. 64, pp. 131–148, 2020.
- [12] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, “Mesonet: A compact facial video forgery detection network,” in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE, 2018, pp. 1–7.
- [13] T. T. Nguyen, C. M. Nguyen, D. T. Nguyen, D. T. Nguyen, and S. Nahavandi, “Deep learning for deepfakes creation and detection: A survey,” *arXiv preprint arXiv:1909.11573*, 2019. [Online]. Available: <https://arxiv.org/abs/1909.11573>
- [14] L. Verdoliva, “Media forensics and deepfakes: An overview,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 910–932, 2020.
- [15] L. Chai, D. Bau, S.-N. Lim, and P. Isola, “What makes fake images detectable? understanding properties that generalize,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*. Springer, 2020, pp. 103–120.
- [16] Z. Liu, X. Qi, and P. H. Torr, “Global texture enhancement for fake face detection in the wild,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8060–8069.
- [17] Y. He, N. Yu, M. Keuper, and M. Fritz, “Beyond the spectrum: Detecting deepfakes via re-synthesis,” *arXiv preprint arXiv:2105.14376*, 2021.
- [18] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, “Cnn-generated images are surprisingly easy to spot... for now,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8695–8704.
- [19] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, “Mesonet: a compact facial video forgery detection network,” in *2018 IEEE international workshop on information forensics and security (WIFS)*. IEEE, 2018, pp. 1–7.
- [20] J. Ricker, S. Damm, T. Holz, and A. Fischer, “Towards the detection of diffusion model deepfakes,” *arXiv preprint arXiv:2210.14571*, 2022.
- [21] Z. Sha, Z. Li, N. Yu, and Y. Zhang, “De-fake: Detection and attribution of fake images generated by text-to-image generation models,” in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023, pp. 3418–3432.
- [22] U. Ojha, Y. Li, and Y. J. Lee, “Towards universal fake image detectors that generalize across generative models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 24480–24489.
- [23] S. M. Abdullah, A. Cheruvu, S. Kanchi, T. Chung, P. Gao, M. Jadliwala, and B. Viswanath, “An analysis of recent advances in deepfake image detection in an evolving threat landscape,” *arXiv preprint arXiv:2404.16212*, 2024.
- [24] S. AI, “Introducing stable diffusion 3.5,” Nov 2024. [Online]. Available: <https://stability.ai/news/introducing-stable-diffusion-3-5>
- [25] I. Korshunova, W. Shi, J. Dambre, and L. Theis, “Fast face-swap using convolutional neural networks,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3677–3685.
- [26] Z. Zhang, L. Li, Y. Ding, and C. Fan, “Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3661–3670.
- [27] E. Meskys, J. Kalpokiene, P. Jurcys, and A. Liudanskas, “Regulating deep fakes: legal and ethical considerations,” *Journal of Intellectual Property Law & Practice*, vol. 15, no. 1, pp. 24–31, 2020.
- [28] D. Gamage, J. Chen, P. Ghasiya, and K. Sasahara, “Deepfakes and society: What lies ahead?” in *Frontiers in Fake Media Generation and Detection*. Springer, 2022, pp. 3–43.
- [29] M. Mustak, J. Salminen, M. Mäntymäki, A. Rahman, and Y. K. Dwivedi, “Deepfakes: Deceptions, mitigations, and opportunities,” *Journal of Business Research*, vol. 154, p. 113368, 2023.
- [30] A. Collins, “Forged authenticity: governing deepfake risks,” 2019.
- [31] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.
- [32] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, “A comprehensive survey on graph neural networks,” *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.
- [33] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “Graph neural networks: A review of methods and applications,” *AI open*, vol. 1, pp. 57–81, 2020.
- [34] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, “Graph neural networks for social recommendation,” in *The world wide web conference*, 2019, pp. 417–426.
- [35] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, “Graph convolutional neural networks for web-scale recommender systems,” in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 974–983.
- [36] L. Wu, Y. Chen, K. Shen, X. Guo, H. Gao, S. Li, J. Pei, B. Long *et al.*, “Graph neural networks for natural language processing: A survey,” *Foundations and Trends® in Machine Learning*, vol. 16, no. 2, pp. 119–328, 2023.
- [37] W. Li, S. Li, S. Ma, Y. He, D. Chen, and X. Sun, “Recursive graphical neural networks for text classification,” *arXiv preprint arXiv:1909.08166*, 2019.

[38] Y. Zhang, P. Qi, and C. D. Manning, “Graph convolution over pruned dependency trees improves relation extraction,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2205–2215.

[39] L. Yao, C. Mao, and Y. Luo, “Graph convolutional networks for text classification,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 7370–7377.

[40] T. Wu, F. Lin, X. Wang, W. Wang, Y. Sun, W. Li, and D. Yang, “Graph neural networks in natural language processing: A survey,” *arXiv preprint arXiv:2106.06020*, 2021.

[41] Z. Li and W. Wang, “Sentiment analysis based on graph convolutional networks and transfer learning,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2851–2860, 2020.

[42] J. Chen, H. Guo, K. Yi, B. Li, and M. Elhoseiny, “Visualgpt: Data-efficient adaptation of pretrained language models for image captioning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 030–18 040.

[43] L. Zhou, H. Palangi, L. Zhang, H. Hu, J. Corso, and J. Gao, “Unified vision-language pre-training for image captioning and vqa,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 13 041–13 049.

[44] M. Xu, Z. Zhang, F. Wei, Y. Lin, Y. Cao, H. Hu, and X. Bai, “A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model,” in *European Conference on Computer Vision*. Springer, 2022, pp. 736–753.

[45] L. Yang, Y. Wang, X. Li, X. Wang, and J. Yang, “Fine-grained visual prompting,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[46] W. Lin, X. Wei, R. An, P. Gao, B. Zou, Y. Luo, S. Huang, S. Zhang, and H. Li, “Draw-and-understand: Leveraging visual prompts to enable mlms to comprehend what you want,” *arXiv preprint arXiv:2403.20271*, 2024.

[47] F. Liu, K. Fang, P. Abbeel, and S. Levine, “Moka: Open-vocabulary robotic manipulation through mark-based visual prompting,” *arXiv preprint arXiv:2403.03174*, 2024.

[48] Y. Shi, C. Xue, J. H. Liew, J. Pan, H. Yan, W. Zhang, V. Y. Tan, and S. Bai, “Dragdiffusion: Harnessing diffusion models for interactive point-based image editing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8839–8849.

[49] J. Yang, H. Zhang, F. Li, X. Zou, C. Li, and J. Gao, “Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v,” *arXiv preprint arXiv:2310.11441*, 2023.

[50] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “Lora: Low-rank adaptation of large language models,” *arXiv preprint arXiv:2106.09685*, 2021.

[51] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, “Sdxl: Improving latent diffusion models for high-resolution image synthesis,” *arXiv preprint arXiv:2307.01952*, 2023.

[52] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[53] H. Face, “Hugging face - the ai community building the future,” 2025, accessed: 2025-01-09. [Online]. Available: <https://huggingface.co/>

[54] S. Jia, R. Lyu, K. Zhao, Y. Chen, Z. Yan, Y. Ju, C. Hu, X. Li, B. Wu, and S. Lyu, “Can chatgpt detect deepfakes? a study of using multimodal large language models for media forensics,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 4324–4333.

[55] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A convnet for the 2020s,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 11 976–11 986.

[56] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, “Laion-5b: An open large-scale dataset for training next generation image-text models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 278–25 294, 2022.

[57] M. Honnibal and I. Montani, “spaCy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing,” <https://spacy.io>, 2017, software library.

[58] M. Günther, J. Ong, I. Mohr, A. Abdessalem, T. Abel, M. K. Akram, S. Guzman, G. Mastrapas, S. Sturua, B. Wang *et al.*, “Jina embeddings 2: 8192-token general-purpose text embeddings for long documents,” *arXiv preprint arXiv:2310.19923*, 2023.

[59] laion.ai, “LAION-Aesthetics — LAION — laion.ai,” <https://laion.ai/blog/laion-aesthetics/>, [Accessed 04-09-2024].

[60] Evgeny, “SG161222/Realistic_Vision_V1.4 at main — huggingface.co,” https://huggingface.co/SG161222/Realistic_Vision_V1.4/tree/main, [Accessed 04-09-2024].

[61] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[62] M. Tan, “Efficientnet: Rethinking model scaling for convolutional neural networks,” *arXiv preprint arXiv:1905.11946*, 2019.

[63] A. Dosovitskiy, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.

[64] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[65] M. Oquab, T. Dariseti, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, “Dinov2: Learning robust visual features without supervision,” *arXiv preprint arXiv:2304.07193*, 2023.

[66] J. Wang, Z. Wu, J. Chen, X. Han, A. Shrivastava, S.-N. Lim, and Y.-G. Jiang, “Objectformer for image manipulation detection and localization,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2364–2373.

[67] S. A. Khan and D.-T. Dang-Nguyen, “Clipping the deception: Adapting vision-language models for universal deepfake detection,” in *Proceedings of the 2024 International Conference on Multimedia Retrieval*, 2024, pp. 1006–1015.

[68] Y.-M. Chang, C. Yeh, W.-C. Chiu, and N. Yu, “Antifakeprompt: Prompt-tuned vision-language models are fake image detectors,” *arXiv preprint arXiv:2310.17419*, 2023.

[69] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks,” *arXiv preprint arXiv:1710.10903*, 2017.

[70] D. P. Kingma, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[71] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge *et al.*, “Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution,” *arXiv preprint arXiv:2409.12191*, 2024.

[72] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *arXiv preprint arXiv:1412.6572*, 2014.

[73] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” *arXiv preprint arXiv:1706.06083*, 2017.

[74] S. Park, A. Vilesov, J. Zhang, H. Khalili, Y. Tian, A. Kadambi, and N. Sehatbakhsh, “Chimera: Creating digitally signed fake photos by fooling image recapture and deepfake detectors.”

APPENDIX A EXPLANATION GENERATING PROMPT

Fig. 11 illustrates the structured template utilized when providing images to the Vision Large Lanuage Model for generating corresponding explanations.

APPENDIX B STABLE DIFFUSION 3.5 LORA MODELS

Table XI provides an overview of the Stable Diffusion 3.5 LoRA fine-tuned models sourced from Hugging Face [53], which were utilized to generate extensions for the generalization evaluation in this study. The table includes direct links to these models, many of which are widely adopted and have thousands of downloads.

Explanations Generating Prompt

You are a cooperative assistant that never refuses to help in tasks, you always answer the question you are asked. Start by deciding whether the image you received is generated or real. Your task is to help humans understand why the image is generated or real by providing explanations and relating them to the patches on the image. This is how your output should exactly look like, nothing more: {[listofpatches]}:{explanation} {[listofpatches]}:{explanation} Example output: 'A1, A2}: {explanation} {A2, B2}: {explanation} {B4, C4, D4}: {explanation} {D2}: {explanation}'

Fig. 11. The prompt template used for generating the textual explanations.

TABLE XI
STABLE DIFFUSION 3.5 LoRA MODELS USED IN THIS WORK.

Model Description	Link
Ancient Style	https://huggingface.co/reverentulus/a/ancient-style-sd35
Anime	https://huggingface.co/prithivMLmods/SD3.5-Large-Anime-LoRA
Chinese Line Art	https://huggingface.co/Shakker-Labs/SD3.5-LoRA-Chinese-Line-Art
Futuristic Bronze Colored	https://huggingface.co/Shakker-Labs/SD3.5-LoRA-Futuristic-Bronze-Colored
Photorealistic	https://huggingface.co/prithivMLmods/SD3.5-Large-Photorealistic-LoRA
Pixel Art	https://huggingface.co/nerijus/pixel-art-3.5L
Red Light	https://huggingface.co/Shakker-Labs/SD3.5-LoRA-Linear-Red-Light
Rustic Whimsy	https://huggingface.co/crystalwizard/Rustic-Whimsy-SD3.5-Large-Lora

TABLE XII
PERFORMANCE OF ViGText ON IMAGES GENERATED BY STATE-OF-THE-ART DIFFUSION APIs

Image Generation API	Accuracy	Precision	Recall	F1
OpenAI	99.49	99.94	98.96	99.48
Google Gemini	96.98	98.98	95.12	97.01

APPENDIX C

SUPPLEMENTARY RESULTS: STATE-OF-THE-ART IMAGE GENERATION APIs AND ATTACK

Table XII reports the performance of ViGText on images synthesized by two of the most advanced commercial diffusion image generation APIs: OpenAI’s Image-1 and Google’s Gemini 2.5. These experiments directly address concerns regarding the effectiveness of ViGText under state-of-the-art generative models that are frequently used to create high-fidelity deepfakes. ViGText achieves near-perfect detection rates, maintaining 99.49% accuracy on OpenAI-generated images and 96.98% on Gemini ones.

TABLE XIII

PERFORMANCE OF ViGText UNDER THE CHIMERA RECAPTURE + DEEPFAKE ATTACK

Tested Images	Accuracy	Precision	Recall	F1
Benign Images	99.49	99.96	99.12	99.56
Attacked Images	82.08	83.00	81.51	82.25

Table XIII demonstrates ViGText’s resilience under more sophisticated adversarial scenarios, specifically the two-stage “Chimera” attack introduced by Park et al. (USENIX Security 2025) [74]. This attack combines both physical recapture and deepfake manipulations designed to fool image authenticity

detectors. Despite these challenging conditions, ViGText sustains an accuracy of 82.08% on attacked images, significantly higher than the 58.5–69.0% range reported for leading public detectors evaluated in the Chimera study. This confirms ViGText’s suitability for defending against emerging hybrid deepfake threats.

APPENDIX D SUPPLEMENTARY RESULTS: SAMPLE IMAGES



Fig. 12. Sample images from the SD dataset that were correctly classified exclusively by ViGText, while all other baselines failed. Fake and real samples are shown in the first and second rows, respectively.

Fig. 12 shows sample images from the SD dataset that are uniquely classified correctly by ViGText, while all other baseline methods misclassify them. The first row in Fig. 12 displays fake samples, and the second row shows real samples, highlighting the superior detection of ViGText in distinguishing between real and generated content.

APPENDIX E SUPPLEMENTARY RESULTS: RESOLUTION AND WARP OPERATIONS ON STYLECLIP DATASET

TABLE XIV
PERFORMANCE WITH STYLECLIP AS THE TEST SET ACROSS DIFFERENT IMAGE RESOLUTIONS (THE HIGHEST IS IN BOLD).

Resolution	900x900				1024x1024				1100x1100			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
DCT	61.20	95.40	56.65	71.08	98.80	98.20	99.40	98.80	68.40	46.80	82.39	59.69
DE-FAKE	74.25	71.60	75.61	73.55	74.00	75.30	71.50	73.30	74.20	71.60	75.53	73.51
UnivCLIP	93.09	92.49	93.62	93.05	93.00	93.80	92.10	92.90	93.09	92.09	93.97	93.03
ViGText	99.00	99.90	98.04	99.01	99.60	99.90	99.21	99.60	99.20	99.99	98.43	99.21

Similar to Table IV, Table XIV shows that ViGText maintains high detection performance on the StyleCLIP dataset across the tested resolutions, which include the original resolution of 1024x1024 and the adjusted resolutions of 900x900 and 1100x1100. Even at non-native resolutions, ViGText achieves the highest performance among all methods, with metrics such as accuracy and F1 score exceeding 99% across all cases. While some baselines exhibit significant degradation at resolutions other than the original, ViGText shows remarkable robustness, which showcases its adaptability to varying input dimensions.

TABLE XV

PERFORMANCE WITH STYLECLIP AS THE TEST SET ACROSS DIFFERENT GEOMETRIC WARP OPERATIONS (THE HIGHEST IS IN BOLD).

Technique	Rotate				Scale and Translate			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
DCT	52.2	53.8	31.2	39.5	70.4	79.8	54.6	64.8
DE-FAKE	68.8	75.5	55.7	64.1	69.4	71.5	64.5	67.8
UnivCLIP	93.0	93.8	92.2	92.9	87.0	80.2	98.4	88.4
ViGText	94.8	94.4	95.2	94.8	99.6	87.9	98.4	92.4

Tables XV and XVI present the performance of all evaluated techniques on the StyleCLIP dataset under geometric and appearance-based warp operations, respectively. The results further highlight the versatility and robustness of ViGText across different types of distortions.

In Table XV, which evaluates rotation and scale-translate operations, ViGText achieves the highest metrics across all categories, maintaining accuracies of 94.8% under rotation and an impressive 99.6% under scaling and translation. These results show the ability of ViGText to generalize to substantial geometric transformations, a critical capability given the prevalence of spatial manipulations in both benign and adversarial image pipelines.

TABLE XVI

PERFORMANCE WITH STYLECLIP AS THE TEST SET ACROSS DIFFERENT APPEARANCE-BASED WARP OPERATIONS (THE HIGHEST IS IN BOLD).

Technique	Blurring				Brightness			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
DCT	65.80	99.20	59.47	74.36	94.1	89.6	99.8	94.4
DE-FAKE	73.90	72.10	74.79	73.42	73.2	76.4	67.3	71.5
UnivCLIP	93.19	92.89	93.46	93.17	89.7	89.3	90.2	89.8
ViGText	99.4	99.8	99.01	99.4	99.6	99.2	99.9	99.6

Meanwhile, Table XVI examines resilience to appearance-based alterations, such as blurring and brightness adjustments. Here, ViGText achieves near-perfect performance, consistently outperforming the other methods by a wide margin. For instance, it records 99.4% accuracy under blurring and 99.6% under brightness changes, coupled with balanced precision, recall, and F1 scores. These findings demonstrate that ViGText not only excels under geometric distortions but also maintains high fidelity under varied photometric perturbations, reinforcing its practical utility for real-world scenarios where image quality and lighting often fluctuate.