GIM: A Million-scale Benchmark for Generative Image Manipulation Detection and Localization

Yirui Chen[1,3,*] , Xudong Huang[3,*] , Quan Zhang [2,3,*] , Wei Li[3] , Mingjian Zhu[3], Qiangyu Yan[3], Simiao Li [3], Hanting Chen[3], Hailin Hu[3], Jie Yang[1], Wei Liu[1,†] , Jie Hu[3,†]

[1] Shanghai Jiao Tong University
[2] Tsinghua University
[3] Huawei Noah's Ark Lab
{chenyirui,weiliucv}@sjtu.edu.cn zhangqua22@mails.tsinghua.edu.cn {huangxudong9,wei.lee,hujie23}@huawei.com

Abstract

The extraordinary ability of generative models emerges as a new trend in image editing and generating realistic images, posing a serious threat to the trustworthiness of multimedia data and driving the research of image manipulation detection and location (IMDL). However, the lack of a large-scale data foundation makes the IMDL task unattainable. In this paper, we build a local manipulation data generation pipeline that integrates the powerful capabilities of SAM, LLM, and generative models. Upon this basis, we propose the GIM dataset, which has the following advantages: 1) Large scale, GIM includes over one million pairs of AI-manipulated images and real images. 2) Rich image content, GIM encompasses a broad range of image classes. 3) Diverse generative manipulation, the images are manipulated images with state-of-the-art generators and various manipulation tasks. The aforementioned advantages allow for a more comprehensive evaluation of IMDL methods, extending their applicability to diverse images. We introduce the GIM benchmark with two settings to evaluate existing IMDL methods. In addition, we propose a novel IMDL framework, termed GIMFormer, which consists of a ShadowTracer, Frequency-Spatial block (FSB), and a Multi-Window Anomalous Modeling (MWAM) module. Extensive experiments on the GIM demonstrate that GIMFormer surpasses the previous state-of-the-art approach on two different benchmarks.

Introduction

Images are one of the most essential media for information transmission in modern society and they are widely spread on public platforms such as news and social media. With the rapid advancement of generative methods (Dhariwal and Nichol 2021; Rombach et al. 2022), such natural information can be more easily manipulated for specific purposes such as tampering with an object or person. The image of this class is particularly convincing since its visual comprehensibility, leading to serious information security risks in social areas. Therefore, it is of utmost urgency to develop methods to detect whether an image is modified by generative models and identify the exact location of the manipulation. However, traditional image manipulation detection and location (IMDL) datasets (Dong, Wang, and Tan 2013; Wen et al. 2016) overlook the powerful generative models and generative IMDL (Jia et al. 2023; Guillaro et al. 2023) datasets are limited with scale, it is thus not sufficient to comprehensively evaluate the performance of IMDL methods and benefit the IMDL community.

To this end, we propose a million-level generative-based IMDL dataset, termed GIM dataset, to provide a reliable database for AI Generated Content (AIGC) security. GIM leverages the generative models (Ho, Jain, and Abbeel 2020; Yin et al. 2025) and SAM (Kirillov et al. 2023a), with the images in ImageNet (Deng et al. 2009) and VOC (Everingham et al. 2010) as the input. SAM is utilized to locate the tampering region, and generative models paint the reasonable content in the tampering area. GIM contains a total of over one million generative manipulated images. To develop an appropriate benchmark scale, we explore the impact of different amounts of training manipulated data. The final benchmark contains about 320k manipulated images with their tampering masks for training and testing. To simulate the real situation, we investigate the effect of degradation and apply random degradation to these benchmark images. Based on this GIM benchmark, the IMDL methods are evaluated and benchmarked. Overall, GIM possesses the following advantages: 1) GIM has a large and reasonable data scale, including rich image categories and contents. 2) GIM contains various generative manipulation models and tasks. 3) GIM proposes two settings for verifying the performance and generalization ability of IMDL methods.

Existing methods emphasize traditional image manipulations also known as "cheapfakes". However, generative manipulations introduce lethal alterations in content with no apparent frequency or structural inconsistency. To address the above issue, we introduce GIMFormer, a transformer-based framework for generative IMDL. The ShadowTracer is designed to embed the nuanced artifacts inherent in generative tampering and serves as the prior information. The Frequency-Spatial Block captures the manipulation clues in the frequency and spatial domains. Furthermore, the Multi Windowed Anomalous Modelling module captures local inconsistencies at different scales to refine the features. GIMFormer extracts features from both the RGB and learned

[*]These authors contributed equally. Project lead: Wei Li.
[†]Corresponding authors.

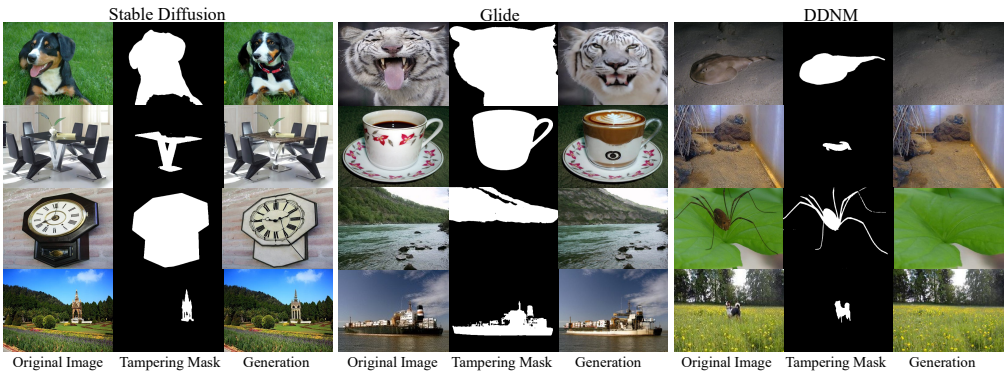

Figure 1: Example images from the GIM dataset. Our dataset includes images manipulated by three state-of-the-art generators: Stable-Diffusion, GLIDE, and DDNM. Three columns display authentic images, manipulation masks and manipulated images.

tampering trace maps, capturing details from both the frequency and spatial domains while modeling inconsistencies at different scales for precise manipulation detection and localization. We conduct experiments on our proposed GIM. Both the qualitative and quantitative results demonstrate that GIMFormer can outperform previous state-of-the-art methods. In summary, our main contributions are as follows:

- We build a data generation pipeline and construct a large-scale dataset for IMDL tasks.
- We investigate the impact of data scales and degradation, constructing a comprehensive benchmark with two evaluation settings for IMDL methods evaluation.
- We propose a framework named GIMFormer for generative IMDL task. Extensive experiments demonstrate that GIMFormer achieves state-of-the-art performance.

Related Work

Image Forensic Datasets

Early datasets (Ng, Chang, and Sun 2004) primarily focus on one type of manipulation. CASIA (Dong, Wang, and Tan 2013) first incorporates multiple types of manipulations with forged images manually crafted using Adobe Photoshop. The Wild Web dataset (Zampoglou, Papadopoulos, and Kompatsiaris 2015) collects forged images from the Internet, surpassing previous datasets in the scale. NIST (Guan et al. 2019) dataset provides an extensive collection of datasets serving as a crucial standard for assessing media tampering detection methods. IMD2020 (Novozamsky, Mahdian, and Saic 2020) provides locally manipulated images generated through manual operations or random slicing and online images without obvious manipulation traces. Recently, HiFi-IFDL (Guo et al. 2023) constructs a hierarchical fine-grained dataset containing some representative forgery methods. AutoSplice (Jia et al. 2023) leverages large-scale language-image models like DALL-E2 (Ramesh et al. 2021) to facilitate automatic image editing and generation. CocoGlide (Guillaro et al. 2023) contains images manipulated by the diffusion model. Besides, other datasets focus on facial manipulations (Rossler et al. 2019; Guarnera et al. 2022) or entirely synthesized images(Zhu et al. 2023; Yan et al. 2024; Verdoliva and Cozzolino 2022).

The datasets mentioned above have limitations such as small data sizes and limited manipulation techniques. Recent advances in generative models have demonstrated remarkable manipulation abilities. Leveraging these models, we introduce GIM, a large-scale dataset that incorporates various recent generative manipulation techniques.

Image Manipulation Detection and Localization

Early studies on natural image manipulation localization mainly focus on detecting a specific type of manipulation (Cozzolino, Poggi, and Verdoliva 2015; Yin et al. 2023). Due to the exact manipulation type is unknown in real-world scenarios, most state-of-the-art methods (Liu et al. 2022; Ying et al. 2023) primarily concentrate on general manipulation. MVSS-Net (Dong et al. 2022) uses a two-stream CNN to extract noise features and employs dual attention to fuse their outputs. PSCC-Net (Liu et al. 2022) extracts hierarchical features in a top-down manner and it detects manipulations in a bottom-up manner. Motivated by the power of transformer(Zhang et al. 2024; Zhang and Qi 2024; Lu et al. 2024a,b; Chen et al. 2023), ObjectFormer (Wang et al. 2022) uses object prototypes to model object-level consistencies and find patch-level inconsistencies to detect the manipulation. However, these methods focus on "cheapfake" detection and encounter challenges when applied to generative manipulation. Certain works (Huang et al. 2022; Chai et al. 2020; Liu et al. 2024) localize manipulations using generative models but mainly focus on human faces.

In this work, we focus on the generative IMDL task. We leverage a deep network to embed the subtle artifacts inherent in generative tampering as a prior trace map. A dual network fuses the trace map and RGB image, combining frequency and spatial information and capturing pixel-level inconsistencies to detect and locate the manipulation.

Dataset and Benchmark Construction

In this section, We propose an automatic pipeline to generate manipulated images from unannotated data. Leveraging this pipeline, we construct the comprehensive large-scale

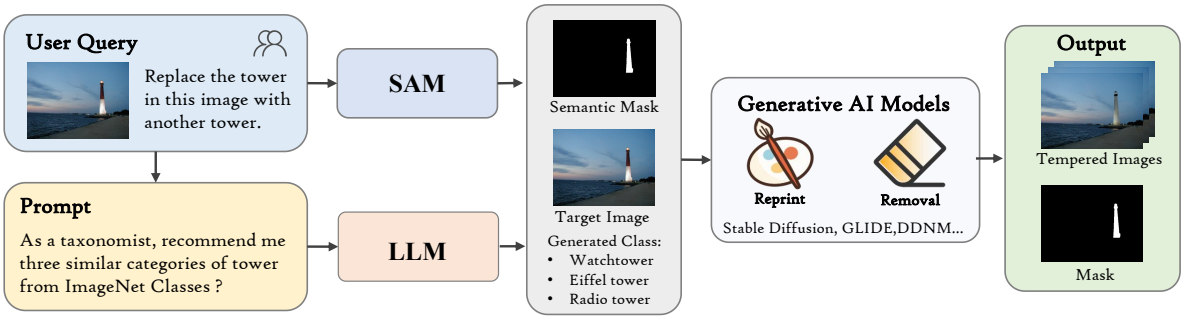

Figure 2: An overview of the dataset generation. Given the original image and a user query (classification attribution or mouse input), the manipulation mask is extracted using SAM. Tampering prompts are then organized with LLM by combining replacement classes. The final generations are produced by generative models with the image, tampering mask and prompts.

GIM dataset. To build a reasonable benchmark, we conduct preliminary experiments focusing on two key aspects: data scale and image degradation. First, we investigate the impact of the training data scale to determine an appropriate size for the GIM benchmark. Second, to reveal real-world scenarios, the manipulated and original images are subjected to three random degradations. Finally, the GIM benchmark comprises over 320k manipulated images paired with authentic images for algorithm training and evaluation. The example images are shown in Figure 1 with their original images and tampering masks. Finally, we introduce the criteria and settings used to evaluate the IMDL methods.

Data Generation Pipeline

Benefiting from open-source projects (Kirillov et al. 2023b; Ren et al. 2024), we develop our data generation pipeline. Figure 2 illustrates the overall process. Our pipeline includes two types of manipulation: reprinting the target class using a generative inpainting method Stable Diffusion (Rombach et al. 2022) and GLIDE (Nichol et al. 2021) or removing the destination region using a specific removal method DDNM (Wang, Yu, and Zhang 2022). Given an arbitrary image, the process begins by extracting the local manipulation mask using a zero-shot segmentation network SAM, guided by either classification attributes or user queries. For reprint tampering, the image category is embedded in the replacement prompt and interacts with LLM (Achiam et al. 2023), which returns an approximate category. The approximate category is then embedded into the inpainting prompt. By combing with the original image, manipulation mask and inpainting prompt, generative models generate the reprint tampered result. For removal tampering, only the original image and manipulation mask are required for the model.

The GIM dataset utilizes images from the ImageNet dataset (Deng et al. 2009) and VOC dataset (Everingham et al. 2010). These two datasets contain a large number of diverse images with wide category coverage, providing a reliable database and laying the foundation for future research.

Analysis of the Benchmark Scale

With the data generation pipeline, we are qualified to generate large amounts of manipulated images. Nonetheless, blindly increasing the amount of data does not improve the algorithm performance, while it may lead to data redundancy. Therefore, taking data generated by the Stable Diffusion as an example, we explore the influence of data volume and category volume on baseline classification (He et al. 2016) and segmentation algorithms (Xie et al. 2021). Training sets vary in the scale, while the validation uses the same test set. As shown in Table 2, the metrics of classification and segmentation are improved as the dataset scale increases. When the scale reaches 180K, the algorithm performance almost saturates. All the metrics remain almost unchanged with either the image class number or the data scale increased. Experiments demonstrate that increasing the amount of data or category brings negligible benefits when the data tends to be saturated. According to the analysis, the GIM benchmark uses 100 labels from ImageNet to generate tampered images for training and employs all the test sets from ImageNet and VOC for evaluation.

Post-processing of Degradation Method

When uploaded to the Internet, images will encounter various post-processing. These transformations can pose challenges for image forensics methods. Table 3 investigates the impact of degradation. The baseline models are trained on the clean data and tested on the test set with a single degradation. Experimental results show that these degradations make the identification difficult. To reveal the real-world scenarios, random degradation (JPEG compression, downsampling and Gaussian blur) is performed on the dataset.

Benchmark Settings

Benchmark Description: GIM consists of four subsets. GIM-SD, GIM-GLIDE and GIM-DDNM which contain data from ImageNet manipulated by Stable Diffusion, GLIDE and DDNM, respectively. Additionally, there is a cross-distribution subset GIM-VOC which contains data from VOC manipulated by Stable Diffusion.

Metrics: We evaluate the performance of the IMDL method on both the image manipulation detection task and localization task. For the detection task, we use accuracy (Cls.acc) as our evaluation metric. For the localization task, the pixel-level AUC and F1 score are adopted.

Dataset	Image Content	Image Size	Num. Images		Manipulations Category		
			# Authentic Image	# Forged Image	Traditional	Gen.Reprint	Gen.Removal
FaceForensics++ (Rossler et al. 2019)	Face	480p-1080p	1,000	4,000	✗	✓	✗
DFDC (Dang et al. 2020)	Face	240p-2160p	19,154	100,000	✗	✓	✗
DeeperForensics (Jiang et al. 2020)	Face	1080p	50,000	10,000	✗	✓	✗
Columbia Gray (Ng, Chang, and Sun 2004)	General	128 × 128	933	912	✓	✗	✗
CASIA V2.0 (Dong, Wang, and Tan 2013)	General	320 × 240-800 × 600	7,200	5,123	✓	✗	✗
IMD2020 (Novozamsky, Mahdian, and Saic 2020)	General	1062 × 866	35,000	35,000	✓	✗	✗
Coverage (Wen et al. 2016)	General	400 × 486	100	100	✓	✗	✗
AutoSplice (Jia et al. 2023)	General	256 × 256 - 4232 × 4232	3,621	2,273	✗	✓	✗
HiFi-IFDL (Guo et al. 2023)	General	-	-	1,000,000 (200,000)	✓	✓	✗
CocoGlide (Guillaro et al. 2023)	General	256×256	-	512	✗	✓	✗
GIM	General	64×64-6000×3904	1,140,000	1,140,000	✗	✓	✓

Table 1: Summary of image manipulation datasets. Image content, image sizes, and manipulation techniques are reported. HIFI-IFDL includes mainly entirely synthesized images with a small portion of traditional manipulation or face image manipulation.

Total Num. of Image	Image Classes	Image per Class	Metrics(%)		
			Cls.Acc	F1	AUC
2,800	10	280	63.1	25.5	79.5
28,000	100	280	74.8	32.5	79.9
180,000	100	1800	91.3	52.9	86.9
360,000	200	1800	91.3	52.9	87.0
500,000	500	1000	91.3	52.9	87.0

Table 2: Dataset Scale Experiment: Effect of Dataset Scale on Base Models (SegFormer-b0, ResNet-50).

Degradation	Metrics(%)	
	Cls.Acc	F1
-	91.3	52.9
JPEG compression	83.1	45.7
Gaussian blur	90.9	40.1
Downsample	88.3	37.1

Table 3: Degradation Experiment: Impact of Degradation on Base Models (SegFormer-b0, ResNet-50).

Settings: Two settings are proposed to evaluate the performance and generalization. In the mix-generator setting, the models are jointly trained on the GIM-SD, GIM-GLIDE and GIM-DDNM training set and tested on the correspondence test dataset respectively to evaluate the performance. In the cross-generator setting, the models are trained on the GIM-SD training set and tested on the GIM-GLIDE, GIM-DDNM and GIM-VOC test sets to explore the generalization.

Method

To address the challenges of generative manipulation, we propose GIMFormer that adopts a dual encoder and decoder architecture. Our framework includes several components: the ShadowTracer, the Frequency-Spatial Block (FSB), and the Multi Windowed Anomalous Modeling (MWAM) module. Figure 3 gives an overview of the framework. For the RGB image input x, we first extract its learned trace map t. Then, both x and t are fed into a two-branch network, where the four-stage structure is used to extract pyramid features F_i ($i \in [1, 4]$). The RGB branch is composed of FSB, Transformer Block (Xie et al. 2021) and WMAM. The tracer branch consists of a Transformer Block and WMAM. In the fusion step, the feature rectification module (FRM) and feature fusion module (FFM) (Zhang et al. 2023) are used for feature fusion. The four-stage fused features are forwarded to the decoder for final detection $\hat{y}$ and location $\hat{M}$.

ShadowTracer

Prior manipulation detection methods mainly focus on "cheapfake" and rely on visible traces. These artifacts include distortions and sudden changes caused by manipulation of the image structure. However, generative tampering makes significant alterations to the content with no apparent frequency or structural inconsistency. As shown in Figure 4, these subtle traces are displayed with inherent patterns that are not visible traces with inconsistent edges.

ShadowTracer aims to capture the inherent characteristics and subtle traces of the generative models. For a manipulated image, our objective is to learn a mapping g_ϕ to map the tampered image to its latent disturbed pixel values, where g_ϕ represents a neural network with trainable parameters ϕ. Our key observation is that the differences introduced by generative models in data distribution exhibit inherent patterns, and deep neural networks can attempt to reconstruct these variations. At the training stage, we generate pairs of the image x_i and the tampered image $G(x_i)$, the manipulation trace can be calculated by $t_i = G(x_i) - x_i$. The objective function for training g_ϕ can be formulated as :

$$\min_{\boldsymbol{\phi}} \left\{ \mathcal{L}_r(g_{\boldsymbol{\phi}}\left(G\left(\mathbf{x}_i\right)\right), t_i) \right\} \tag{1}$$

where $\mathcal{L}_r(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$. Furthermore, the mapping network needs to detect subtle tampering traces and be robust to various real-world image degradations. For this reason, image pairs are generated by mixing original and manipulated images and incorporating diverse degradation operations at the training stage. Specifically, given an input image I, we segment the region of interest and perform generative manipulation to obtain I_m. The mix-up (Zhang et al. 2017) strategy is utilized to the original and manipulated images to obscure obvious tampering traces. Following this, we subject the images to the degradations mentioned above to obtain the final manipulated image. The network is trained on 64×64 pixel patches randomly sampled from the dataset and the loss in Eq. 1 is adopted.

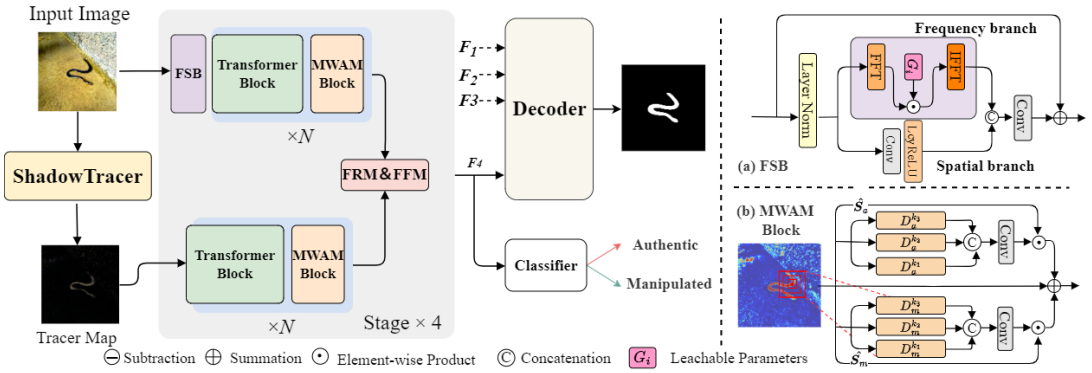

Figure 3: GIMFormer architecture. ShadowTracer extracts trace map t from the input image x. The encoder combines x and t to generate pyramid features F_i across four stages, which are sent to the decoder for manipulation detection and localization.

Figure 4: Generative manipulation leaves subtle traces, ShadowTracer identifies intrinsic patterns and reconstructs underlying tampering perturbations.

Frequency-Spatial Block

When degradation operations are applied, artifacts in manipulated images are tricky to perceive. To improve the local expressive and harvest discriminative cues in manipulated images, we design a Frequency-Spatial Block (FSB) to extract forgery features in the frequency and spatial domains.

Inspired by the recent work (Rao et al. 2021; Lee-Thorp et al. 2021; Zhang et al. 2022), FSB consists of two branches: a frequency branch and a spatial branch as shown in Figure 3. In the frequency branch, the input X is converted into the frequency domain $\mathcal{F}_T(X)$ using the 2D FFT. A learnable filter G_i is multiplied to modulate the spectrum and capture the frequency information. Subsequently, the inverse FFT is applied to convert the feature back to the spatial domain, resulting in the extraction of frequency-aware features X_{f}. In the spatial branch, the input X is processed through convolution layers and LeakyReLU function to enhance the expressiveness of the features and obtain refined spatial features X_{s}. Then X_{f} and X_{s} are concatenated and passed through convolution layers and the LeakyReLU function to obtain enhanced information, which is then combined with the original input X through element-wise summation. The total process can be formulated by:

$$\begin{aligned} X_{\mathrm{f}} &= \hat{\mathcal{F}}_T(\mathcal{F}_T(X) \odot G_i) \\ X_{\mathrm{s}} &= \mathrm{Conv_L}\left(\mathrm{Conv}\left(X\right)\right) \\ X_{out} &= \mathrm{Conv_L}([X_{\mathrm{f}}, X_{\mathrm{s}}]) + X, \end{aligned} \tag{2}$$

where $\odot$ denotes the Hadamard product , $\mathrm{Conv_L}$ denotes convolution with LeakyReLU and $[\cdot]$ denotes concatenation.

Multi Windowed Anomalous Modelling Module

Image manipulation causes discrepancies at the pixel level. Genuine pixels are expected to exhibit consistency with neighboring pixels, while manipulated pixels may deviate and display anomalies. Motivated by previous works (Wu, AbdAlmageed, and Natarajan 2019; Kong et al. 2023) that explore local inconsistencies. To effectively capture the pixel-level inconsistency between the manipulated and real region, we introduce the Multi Windowed Anomalous Modelling (MWAM) module to model these differences at multiple scales for fine-grained features.

As shown in Figure 3, for input feature $F \in H \times W \times C$, we calculate the difference between pixel and its surroundings within a local window in two branches via Eq.3.

$$\begin{aligned} D_u^k[i,j] &= (F[i,j] - F_u^k[i,j])/\sigma^*, \\ \sigma^* &= \mathrm{maximum}(\sigma(F), 1e^{-5} + w_\sigma) \end{aligned} \tag{3}$$

where $u \in \{a, m\}$ denotes average or maximum branches, $\sigma(F)$ is the standard deviation of F, and w_σ is a learnable non-negative weight vector of the same length as σ, F_a^k and F_m^k are calculated from the average and maximum values of the $k \times k$ windows in each pixel. Different sizes k are selected to model the inconsistency at different scales. Then, the obtained $N = 3$ different-scale D_a^k and D_m^k are concatenated and fed into a convolutional network to obtain an anomaly map M_a and M_m of the same size as the original input. Additionally, the anomaly score mask $\hat{\boldsymbol{S}}_{\boldsymbol{u}} \in H \times W$ of the feature is computed using.

$$\begin{aligned} \hat{\boldsymbol{f}}_u &= \mathrm{DConv}\left(\boldsymbol{f}\right), \\ \hat{\boldsymbol{S}}_u &= \mathrm{Sigmoid}\left(\mathrm{Conv}(C,1)\left(\hat{\boldsymbol{f}}_u\right)\right), \end{aligned} \tag{4}$$

where the DConv means a 3×3 Depth-Wise convolution layer. The element-wise multiplication between anomaly score $\hat{\boldsymbol{S}}_u$ and the anomaly map M_u capture the anomaly information. Next, we calculate the element-wise summation between the resulting anomaly-aware map and the input feature map X to obtain an anomaly-sensitive feature map. The whole process can be described as:

$$\hat{X} = X + \hat{\boldsymbol{S}}_a \times M_a + \hat{\boldsymbol{S}}_m \times M_m \tag{5}$$

Method	Params	GFLOPS	GIM-SD			GIM-GLIDE			GIM-DDNM		
			Cls.Acc	F1	AUC	Cls.Acc	F1	AUC	Cls.Acc	F1	AUC
ManTranet(Wu, AbdAlmageed, and Natarajan 2019)	**4.0**	1009.7	61.1	37.5	80.8	71.0	49.1	83.3	54.0	33.1	74.9
MVSS-Net(Dong et al. 2022)	146.9	160.0	56.1	23.2	72.0	61.3	33.1	74.9	49.2	14.1	70.1
SPAN(Hu et al. 2020)	15.4	**30.9**	53.2	35.6	79.3	60.0	39.5	81.2	59.2	32.1	73.6
PSCC-Net(Liu et al. 2022)	4.1	107.3	52.3	31.5	83.8	66.5	53.7	86.4	56.3	41.8	85.8
ObjectFormer‡[1](Wang et al. 2022)	14.6	249.6	59.1	26.8	85.2	70.1	40.1	85.2	54.3	33.1	86.8
Trufor‡[1](Guillaro et al. 2023)	67.8	90.1	67.1	44.1	84.5	80.2	59.3	93.0	63.3	44.5	87.6
IML-VIT(Ma et al. 2024)	91.0	136.0	65.2	53.9	84.3	71.3	69.3	92.1	71.2	53.9	85.7
SegFormer(Xie et al. 2021)	27.5	41.3	64.3	46.2	83.3	78.1	56.8	88.7	69.3	40.2	84.6
GIMFormer (Ours)	95.9	96.2	**70.9**	**58.6**	**88.2**	**83.9**	**77.3**	**95.42**	**76.7**	**56.3**	**88.3**

Table 4: Benchmarking IMDL models to evaluate performance. Cls.Acc(%), AUC(%) and F1(%) Params(M) are reported.

Method	GIM-SD			GIM-VOC			GIM-GLIDE			GIM-DDNM		
	Cls.Acc	F1	AUC	Cls.Acc	F1	AUC	Cls.Acc	F1	AUC	Cls.Acc	F1	AUC
ManTranet(Wu, AbdAlmageed, and Natarajan 2019)	73.1	43.2	80.2	63.2	27.4	72.7	58.2	24.5	74.6	39.5	16.8	58.3
MVSS-Net(Dong et al. 2022)	56.1	25.1	82.2	56.3	21.2	73.2	53.2	23.17	73.1	48.1	10.1	50.2
SPAN(Hu et al. 2020)	52.2	47.3	56.9	52.2	29.9	55.2	50.0	30.1	56.1	44.2	14.3	56.7
PSCC-Net(Liu et al. 2022)	58.2	48.4	87.3	54.8	39.4	83.8	51.1	29.6	79.8	40.5	4.4	48.5
objectFormer‡[1](Wang et al. 2022)	67.1	39.5	87.9	58.2	29.1	81.2	54.2	17.7	81.0	49.2	7.2	59.0
Trufor‡[1](Guillaro et al. 2023)	74.0	49.9	86.2	65.2	31.9	80.1	50.1	22.4	76.3	49.2	5.7	53.1
IML-VIT(Ma et al. 2024)	77.1	58.1	88.9	54.3	47.2	83.2	52.1	23.1	71.1	50.3	7.2	52.3
SegFormer(Xie et al. 2021)	71.9	53.1	87.1	60.0	28.2	81.0	54.1	21.0	75.1	50.2	4.7	51.3
GIMFormer (Ours)	**78.9**	**61.7**	**90.6**	**67.2**	**50.0**	**84.0**	**67.0**	**39.3**	**81.0**	**52.3**	**19.4**	**60.1**

Table 5: Benchmarking IMDL models to evaluate generalization. Cls.Acc(%), AUC(%) and F1(%) are reported.

Loss Function

For detection, we adopt a lightweight backbone in (Wang et al. 2020) on the fourth stage feature for binary prediction $\hat{y}$. For localization, we utilize the MLP decoder (Xie et al. 2021) as the segmentation head to obtain a predicted mask $\hat{M}$. Given the ground-truth label y and mask M, we train GIMFormer with the following objective function:

$$\mathcal{L} = \mathcal{L}_{cls}(y, \hat{y}) + \mathcal{L}_{seg}(M, \hat{M}), \tag{6}$$

where both $\mathcal{L}_{cls}$ and $\mathcal{L}_{seg}$ are binary cross-entropy loss.

Implementation Details

Our approach includes two separate training steps. First, we train the ShadowTracer using the dataset generated from ImageNet. This training process follows a similar data generation method as mentioned in the former chapter. Then, we train the encoder and decoder of the model according to the two settings in GIM, as described in the previous section. We train our models on 8 V100 GPUs with an initial learning rate of $6e^{-5}$ which is scheduled by the poly strategy with power 0.9 over 20 epochs. The optimizer is AdamW (Loshchilov and Hutter 2017) with epsilon $1e^{-8}$ weight decay $1e^{-2}$, and the batch size is 4 on each GPU.

Comparison with Existing Methods

We compare our methods with various state-of-the-art IMDL methods [1] and the vanilla SegFormer (MiT-B2) (Xie et al. 2021). As shown in Table 4 and Table 5, these methods are tested in two settings to evaluate the performance. Note that some methods are not explicitly designed for image-level detection, in which case we use the maximum of the prediction map as the detection statistic. All methods are immersed in the same implementation details. More experiments are included in the Appendix.

Mix-generator Comparison. Table 4 reports the performance of all methods. GIMFormer outperforms other methods, demonstrating its superior ability to identify generative manipulation. Existing methods exhibit low pixel-level F1 scores on this benchmark, indicating that tampering areas are not accurately identified. The qualitative results for visual comparisons are illustrated in Figure 5.

Cross-generator Comparison. Table 5 reports the generalization of all the methods. The results show that GIMFormer outperforms all other methods in both in-domain and cross-domain. For in-domain experiments, our method catches the subtle artifacts inherent in manipulation and accurately localizes them. Other methods may encounter confusion as they attempt to learn specific content, potentially leading to challenges in accurately detecting and localizing generative tampering patterns. For cross-domain experiments, GIMFormer demonstrates well generalization in detecting manipulation using different generative models, as shown in the results of the GIM-GLIDE and GIM-DDNM testsets. Besides GIMFormer works well on data generated by the same generator from different distributions, while existing methods have an obvious performance drop, as shown in the results of the GIM-VOC testset. The qualitative results for visual comparisons are illustrated in Figure 6.

[1] ‡ indicates that the original paper does not provide the code, we reproduce the code and evaluate it under the same settings.

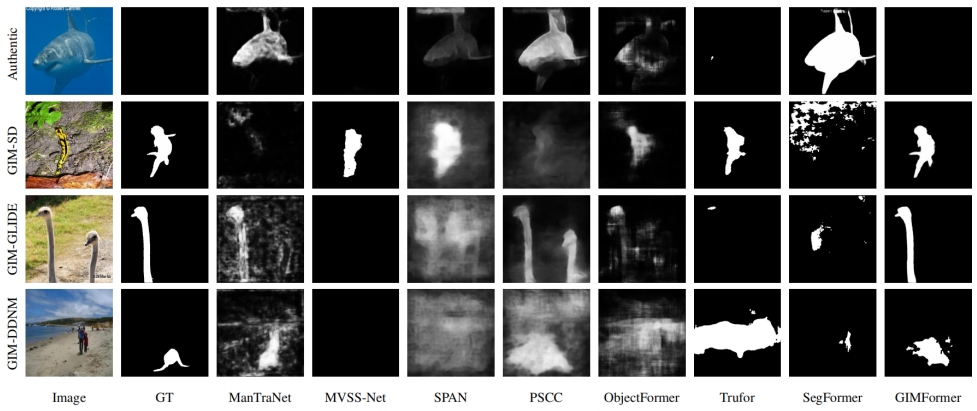

Figure 5: Qualitative results on GIM of comparing GIMFormer with state-of-the-art methods performance.

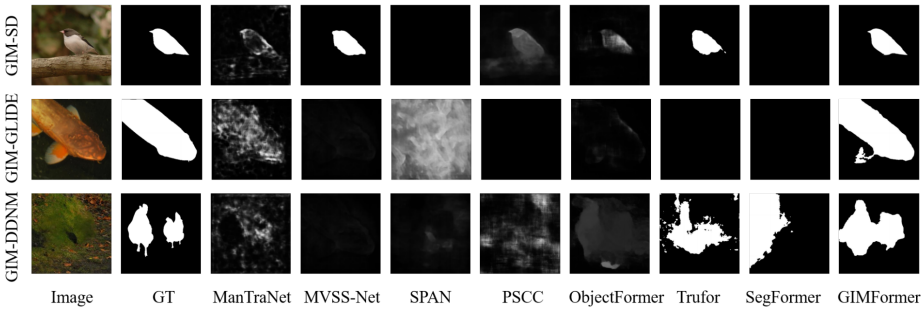

Figure 6: Qualitative results on GIM of comparing GIMFormer with state-of-the-art methods generalization capability.

Variants	Cls.Acc	F1	AUC
Baseline	78.1	56.8	88.7
+ShadowTracer	79.9	67.1	92.0
+ShadowTracer+WMAM	80.2	73.4	94.0
+ShadowTracer+WMAM+FSB	**83.9**	**77.3**	**95.4**

Table 6: Ablation results of GIMFormer variants. Cls.Acc(%), AUC(%) and F1(%) scores are reported.

Variants	GIM-GLIDE			GIM-DDNM		
	Cls.Acc	F1	AUC	Cls.Acc	F1	AUC
GIMFormer w ShadowTracer	83.1	78.1	95.4	77.4	58.8	89.4
GIMFormer w/o ShadowTracer	80.0	68.9	91.3	73.1	51.3	86.1

Table 7: Generalization Experiments of ShadowTracer. Cls.Acc(%), AUC(%) and F1(%) scores are reported.

Ablation Analysis

Effectiveness of the Proposed Module. We consider a simple baseline proposed in (Xie et al. 2021) and gradually integrate new key components. Experiments are carried out on the GIM-GLIDE testset. The quantitative results are listed in Table 6. The result shows that ShadowTracer brings significant improvements to the vanilla baseline. With the MWAM, there is an increase of 6.29% in F1 and 2% in AUC, which indicates that the differential information at multiple scales is crucial for accurate tampering localization. The use of FSB to dynamically harvest complementary frequency and spatial cues improves performance, particularly in detection. The results verify that ShadowTracer, FSB and MWAM effectively improve the performance of the baseline model.

Generalization of ShadowTracer. To evaluate the generalization of ShadowTracer, we train ShadowTracer using data generated by Stable Diffusion and subsequently hold its weights fixed. We proceed to train the backbone on the GIM-GLIDE and GIM-DDNM trainsets, with and without the incorporation of ShadowTracer. The result in Table 7 reveals that leveraging the pretrained ShadowTracer increases performance in cross-generator IMDL tasks. Across GIM-GLIDE and GIM-DDNM, ShadowTracer achieves up to 9% F1, 4% AUC and 4% accuracy improvements.

Conclusion

We address the challenge of detecting and locating generative manipulation and provide a reliable database GIM for AIGC security. This dataset leverages multiple generators to provide diverse generative manipulation data. Based on this, we design a benchmark for IMDL methods with two settings. We also introduce GIMFormer, a novel transformer-based IMDL framework. Extensive experiments demonstrate that GIMFormer achieves SOTA performance.

Acknowledgments

This work is partially supported by NSFC (No. 62402318, 62376153, 24Z990200676)

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Chai, L.; Bau, D.; Lim, S.-N.; and Isola, P. 2020. 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*, 103–120. Springer.

Chen, Y.; Wei, P.; Liu, Z.; Wang, B.; Yang, J.; and Liu, W. 2023. FASTC: A Fast Attentional Framework for Semantic Traversability Classification Using Point Cloud. In *ECAI 2023*, 429–436. IOS Press.

Cozzolino, D.; Poggi, G.; and Verdoliva, L. 2015. Splicebuster: A new blind image splicing detector. In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*, 1–6. IEEE.

Dang, H.; Liu, F.; Stehouwer, J.; Liu, X.; and Jain, A. K. 2020. On the detection of digital face manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern recognition*, 5781–5790.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. IEEE.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Dong, C.; Chen, X.; Hu, R.; Cao, J.; and Li, X. 2022. Mvss-net: Multi-view multi-scale supervised networks for image manipulation detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3539–3553.

Dong, J.; Wang, W.; and Tan, T. 2013. Casia image tampering detection evaluation database. In *2013 IEEE China summit and international conference on signal and information processing*, 422–426. IEEE.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88: 303–338.

Guan, H.; Kozak, M.; Robertson, E.; Lee, Y.; Yates, A. N.; Delgado, A.; Zhou, D.; Kheyrkhah, T.; Smith, J.; and Fiscus, J. 2019. MFC datasets: Large-scale benchmark datasets for media forensic challenge evaluation. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, 63–72. IEEE.

Guarnera, L.; Giudice, O.; Guarnera, F.; Ortis, A.; Puglisi, G.; Paratore, A.; Bui, L. M.; Fontani, M.; Coccomini, D. A.; Caldelli, R.; et al. 2022. The face deepfake detection challenge. *Journal of Imaging*, 8(10): 263.

Guillaro, F.; Cozzolino, D.; Sud, A.; Dufour, N.; and Verdoliva, L. 2023. TruFor: Leveraging all-round clues for trustworthy image forgery detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 20606–20615.

Guo, X.; Liu, X.; Ren, Z.; Grosz, S.; Masi, I.; and Liu, X. 2023. Hierarchical fine-grained image forgery detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3155–3165.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Hu, X.; Zhang, Z.; Jiang, Z.; Chaudhuri, S.; Yang, Z.; and Nevatia, R. 2020. SPAN: Spatial pyramid attention network for image manipulation localization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, 312–328. Springer.

Huang, Y.; Juefei-Xu, F.; Guo, Q.; Liu, Y.; and Pu, G. 2022. FakeLocator: Robust localization of GAN-based face manipulations. *IEEE Transactions on Information Forensics and Security*, 17: 2657–2672.

Jia, S.; Huang, M.; Zhou, Z.; Ju, Y.; Cai, J.; and Lyu, S. 2023. AutoSplice: A Text-prompt Manipulated Image Dataset for Media Forensics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 893–903.

Jiang, L.; Li, R.; Wu, W.; Qian, C.; and Loy, C. C. 2020. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; Dollar, P.; and Girshick, R. 2023a. Segment Anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 4015–4026.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023b. Segment anything. *arXiv preprint arXiv:2304.02643*.

Kong, C.; Luo, A.; Wang, S.; Li, H.; Rocha, A.; and Kot, A. C. 2023. Pixel-inconsistency modeling for image manipulation localization. *arXiv preprint arXiv:2310.00234*.

Lee-Thorp, J.; Ainslie, J.; Eckstein, I.; and Ontanon, S. 2021. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*.

Liu, X.; Liu, Y.; Chen, J.; and Liu, X. 2022. PSCC-Net: Progressive spatio-channel correlation network for image manipulation detection and localization. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(11): 7505–7517.

Liu, Z.; Liu, S.; Ning, Z.; Yang, J.; and Liu, W. 2024. CD-NGP: A Fast Scalable Continual Representation for Dynamic Scenes. arXiv:2409.05166.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Lu, F.; Dong, S.; Zhang, L.; Liu, B.; Lan, X.; Jiang, D.; and Yuan, C. 2024a. Deep Homography Estimation for Visual Place Recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 10341–10349.

Lu, F.; Lan, X.; Zhang, L.; Jiang, D.; Wang, Y.; and Yuan, C. 2024b. CricaVPR: Cross-image Correlation-aware Representation Learning for Visual Place Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16772–16782.

Ma, X.; Du, B.; Jiang, Z.; Du, X.; Hammadi, A. Y. A.; and Zhou, J. 2024. IML-ViT: Benchmarking Image Manipulation Localization by Vision Transformer. arXiv:2307.14863.

Ng, T.-T.; Chang, S.-F.; and Sun, Q. 2004. A data set of authentic and spliced image blocks. *Columbia University, ADVENT Technical Report*, 4.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

Novozamsky, A.; Mahdian, B.; and Saic, S. 2020. IMD2020: A large-scale annotated dataset tailored for detecting manipulated images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops*.

Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, 8821–8831. PMLR.

Rao, Y.; Zhao, W.; Zhu, Z.; Lu, J.; and Zhou, J. 2021. Global filter networks for image classification. *Advances in neural information processing systems*, 34: 980–993.

Ren, T.; Liu, S.; Zeng, A.; Lin, J.; Li, K.; Cao, H.; Chen, J.; Huang, X.; Chen, Y.; Yan, F.; Zeng, Z.; Zhang, H.; Li, F.; Yang, J.; Li, H.; Jiang, Q.; and Zhang, L. 2024. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. arXiv:2401.14159.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Rossler, A.; Cozzolino, D.; Verdoliva, L.; Riess, C.; Thies, J.; and Nießner, M. 2019. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*.

Verdoliva, L.; and Cozzolino, D. 2022. IEEE Image and Video Processing Cup Synthetic Image Detection.

Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. 2020. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10): 3349–3364.

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

Wang, Y.; Yu, J.; and Zhang, J. 2022. Zero-shot image restoration using denoising diffusion null-space model. *arXiv preprint arXiv:2212.00490*.

Wen, B.; Zhu, Y.; Subramanian, R.; Ng, T.-T.; Shen, X.; and Winkler, S. 2016. COVERAGE—A novel database for copy-move forgery detection. In *2016 IEEE international conference on image processing (ICIP)*, 161–165. IEEE.

Wu, Y.; AbdAlmageed, W.; and Natarajan, P. 2019. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9543–9552.

Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34: 12077–12090.

Yan, S.; Li, O.; Cai, J.; Hao, Y.; Jiang, X.; Hu, Y.; and Xie, W. 2024. A Sanity Check for AI-generated Image Detection. *arXiv preprint arXiv:2406.19435*.

Yin, H.; Bai, J.; Huang, S.; and Chen, J. 2023. How information on soft labels and hard labels mutually benefits sound event detection tasks. *Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge*.

Yin, H.; Chen, J.; Bai, J.; Wang, M.; Rahardja, S.; Shi, D.; and Gan, W.-s. 2025. Multi-granularity acoustic information fusion for sound event detection. *Signal Processing*, 227.

Ying, Q.; Zhou, H.; Qian, Z.; Li, S.; and Zhang, X. 2023. Learning to immunize images for tamper localization and self-recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zampoglou, M.; Papadopoulos, S.; and Kompatsiaris, Y. 2015. Detecting image splicing in the wild (web). In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 1–6. IEEE.

Zhang, D.; Huang, F.; Liu, S.; Wang, X.; and Jin, Z. 2022. Swinfir: Revisiting the swinir with fast fourier convolution and improved training for image super-resolution. *arXiv preprint arXiv:2208.11247*.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhang, J.; Liu, H.; Yang, K.; Hu, X.; Liu, R.; and Stiefelhagen, R. 2023. CMX: Cross-modal fusion for RGB-X semantic segmentation with transformers. *IEEE Transactions on Intelligent Transportation Systems*.

Zhang, Q.; Liu, X.; Li, W.; Chen, H.; Liu, J.; Hu, J.; Xiong, Z.; Yuan, C.; and Wang, Y. 2024. Distilling Semantic Priors from SAM to Efficient Image Restoration Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 25409–25419.

Zhang, Q.; and Qi, Y. 2024. Can MLLMs Guide Weakly-Supervised Temporal Action Localization Tasks? *arXiv preprint arXiv:2411.08466*.

Zhu, M.; Chen, H.; Yan, Q.; Huang, X.; Lin, G.; Li, W.; Tu, Z.; Hu, H.; Hu, J.; and Wang, Y. 2023. GenImage: A Million-Scale Benchmark for Detecting AI-Generated Image. *arXiv preprint arXiv:2306.08571*.