Synthetic media detection

Dr. Symeon Papadopoulos
Principal Researcher @ CERTH/ITI
Head of MeVer group

AIDA AICET 2025 @ Thessaloniki, 16 July 2025

mever
Media Analysis, Verification and Retrieval Group

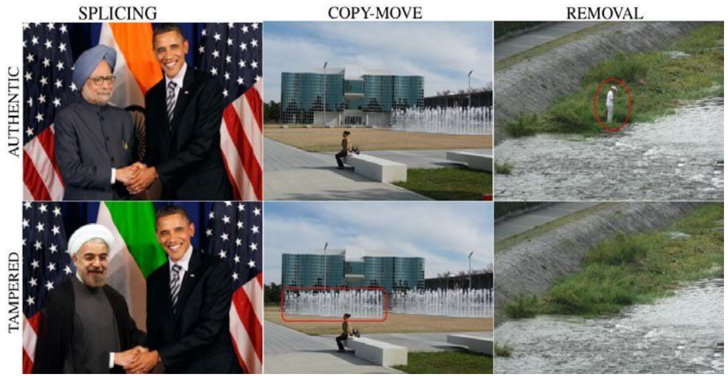
A few words about MeVer

mever
MEDIA ANALYSIS, VERIFICATION & RETRIEVAL GROUP

- A research group of Information Technologies Institute of CERTH, part of MKLab
- Emphasis on AI for multimedia analysis, retrieval, verification
- Key challenges: disinformation, bias/fairness, efficiency/scale, robustness
- Prominent role in numerous relevant EC funded projects (vera.ai, AI4Media, MedDMO, AI4Trust, disAI, AI-CODE, ELLIOT)
- Partnerships with end users and industry (AFP, DW, logically, UN-OHCHR, etc.)
- Tools & services

introduction

Manipulated & synthetic media



Manipulated Images: Digitally altered images

Synthetic Images: Entirely generated by AI

Deepfake Videos: Videos manipulated using AI (face swapping, reenactment, etc.)

Synthetic Videos: Temporally consistent visual synthesis

Common Generative Approaches

Encoders/Auto-encoders/GANs

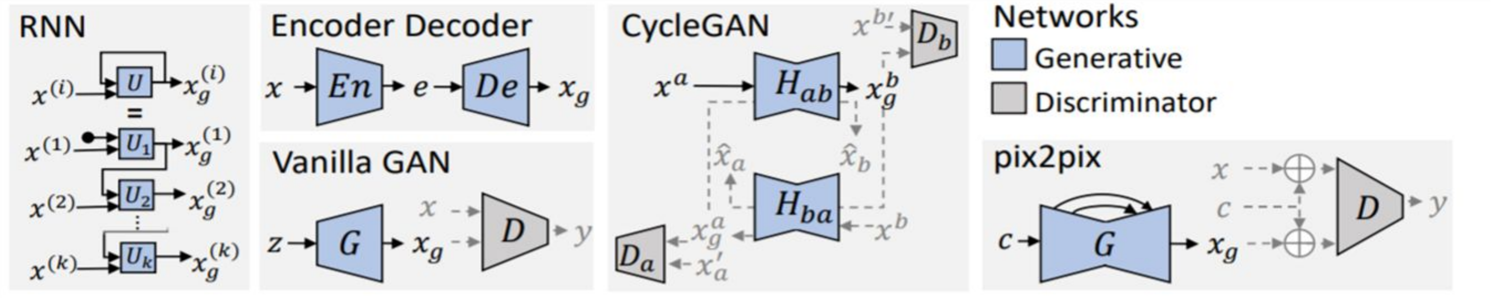
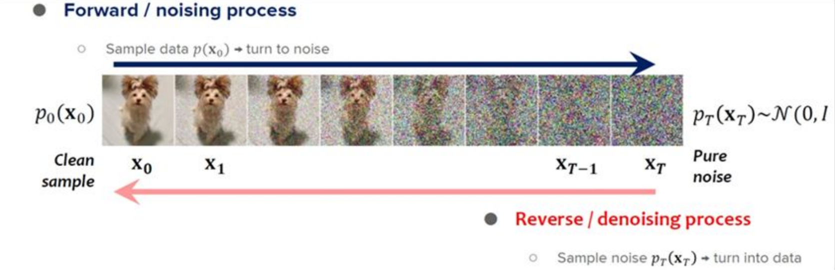
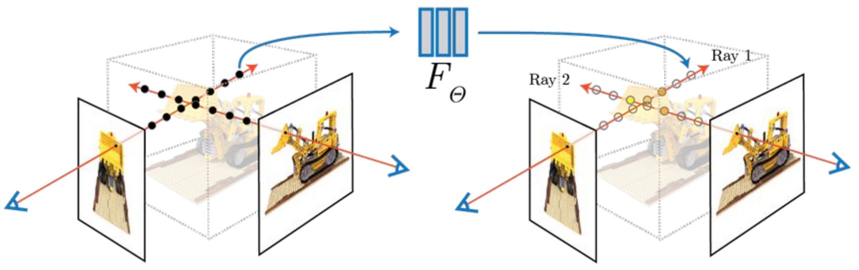


Fig. 4. Five basic neural network architectures used to create deepfakes. The lines indicate dataflows used during deployment (black) and training (grey).

Diffusion models



Neural radiance fields (NeRFs)/Gaussian splatting



Quality rapidly improving...

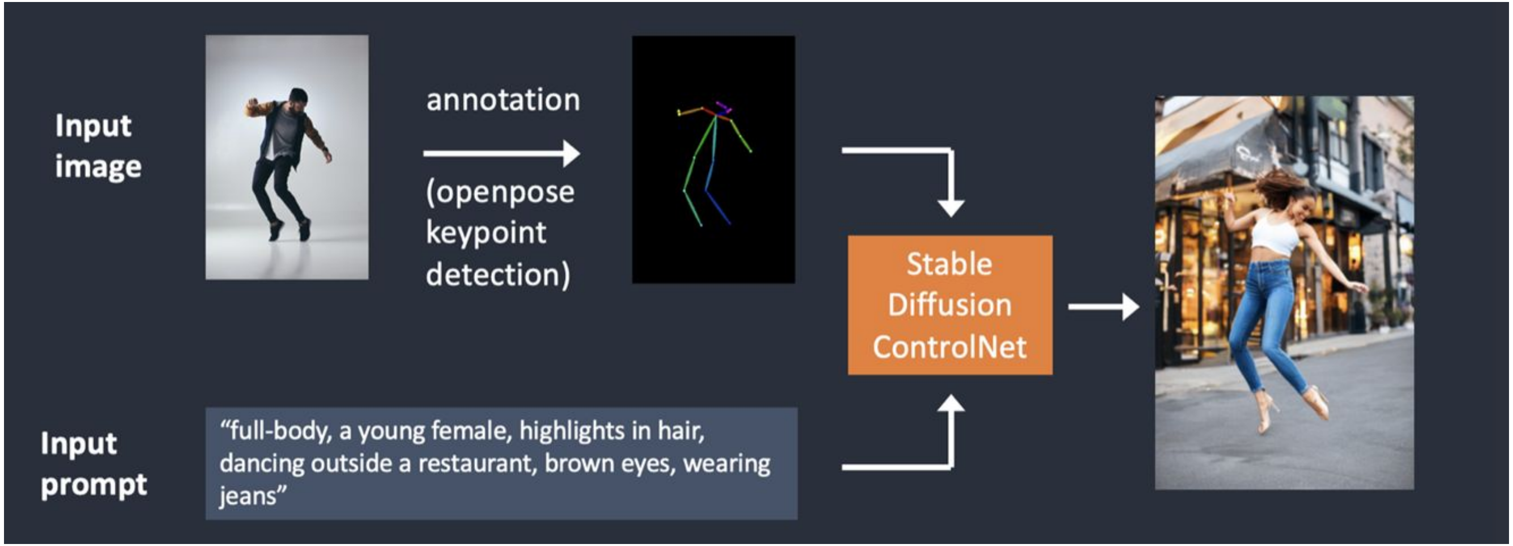
GANs

portrait, a beautiful young woman, glamour street medium format photography, feminine, shot on cinealta, night, pastel hues

Midjourney

Generation Control

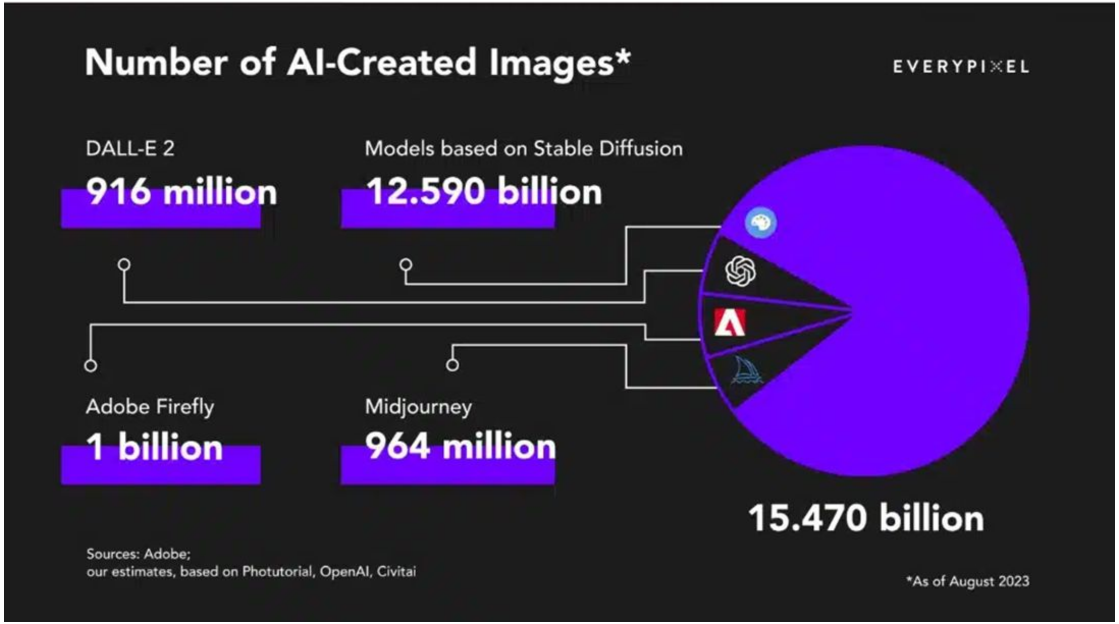
- Random latent variables (no control)
- Inversion of latent space + interpolation
- Text prompting
- Input image + text prompting



Synthetic images are proliferating

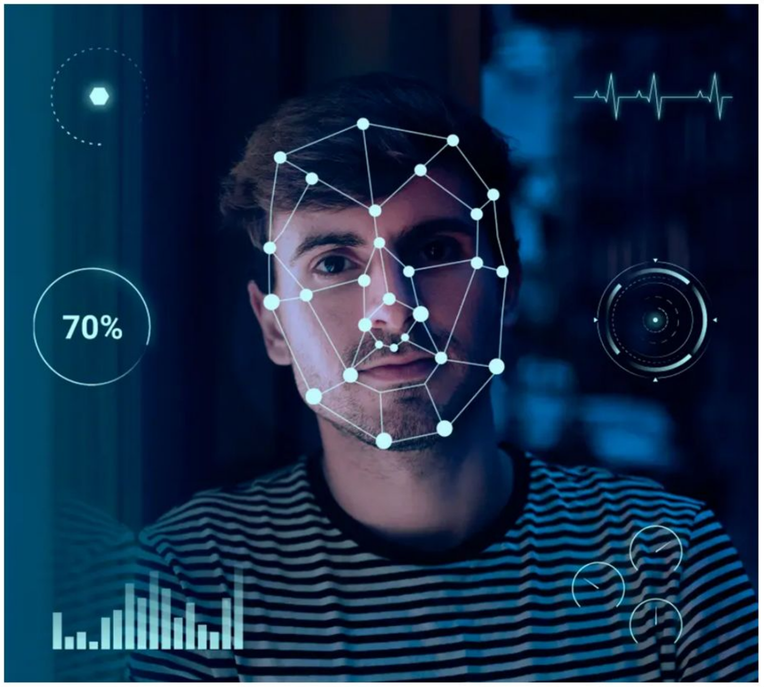
- Stability Diffusion
- MS Designer Image Creator
- Open AI DALL-E 3
- Imagen 3
- Midjourney
- FLUX. 1
- Grok
- Adobe Firefly

…



https://journal.everypixel.com/ai-image-statistics mever

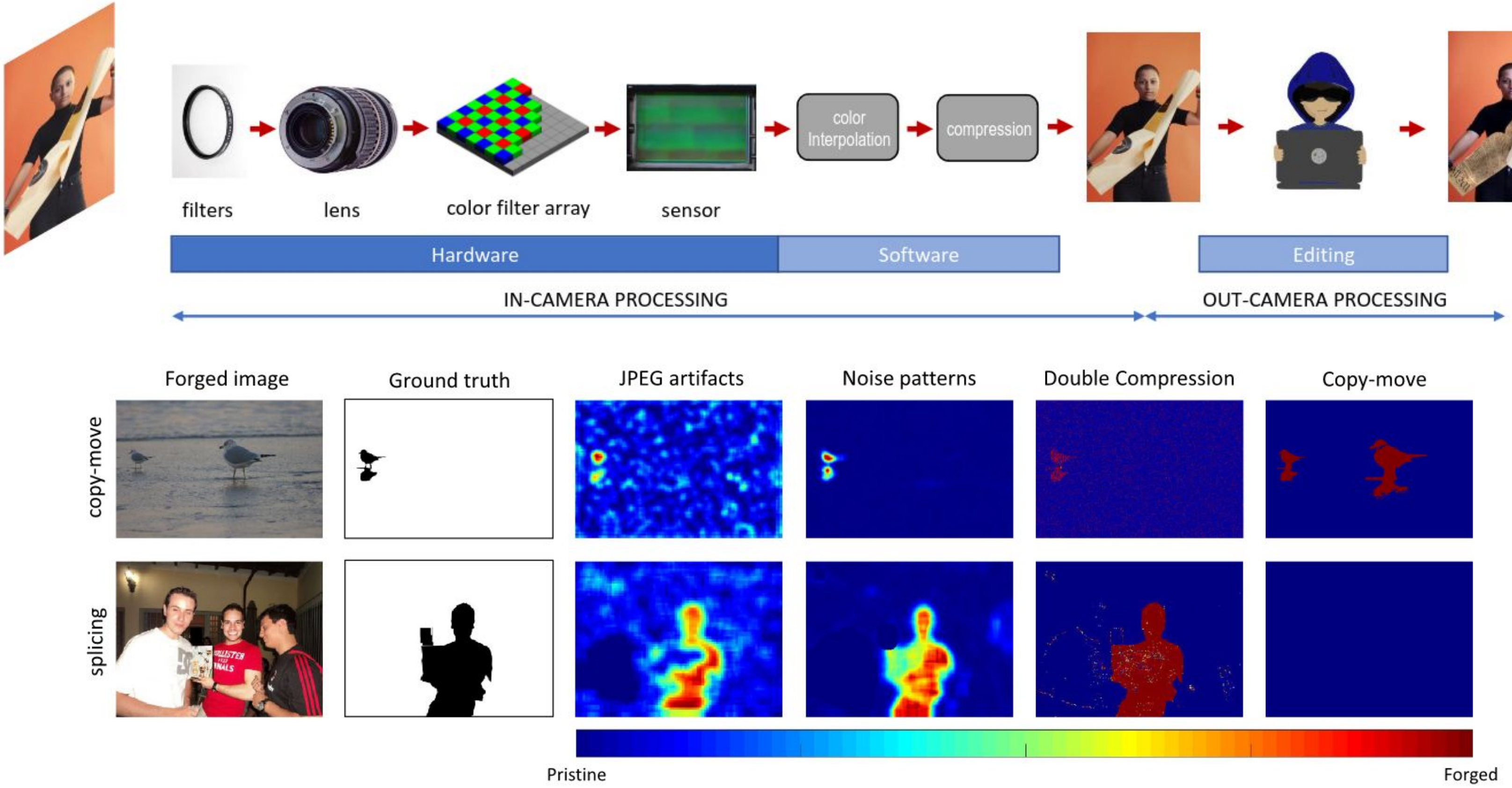
detection approaches



Manipulation vs Fully Synthetic Detection

- Manipulation detection focuses on identifying local modifications
- Tightly linked with the field of **image forensics**
- Common strategies include the identification of **local artifacts** that are different locally compared to rest of image (noise patterns, compression artifacts, frequency domain analysis, etc.) → **image forgery localisation**
- **Deep learning** approaches have become increasingly adopted for both manipulation detection and for **fully synthetic image detection** where they are a natural fit

Image Forensics: many traces - many methods

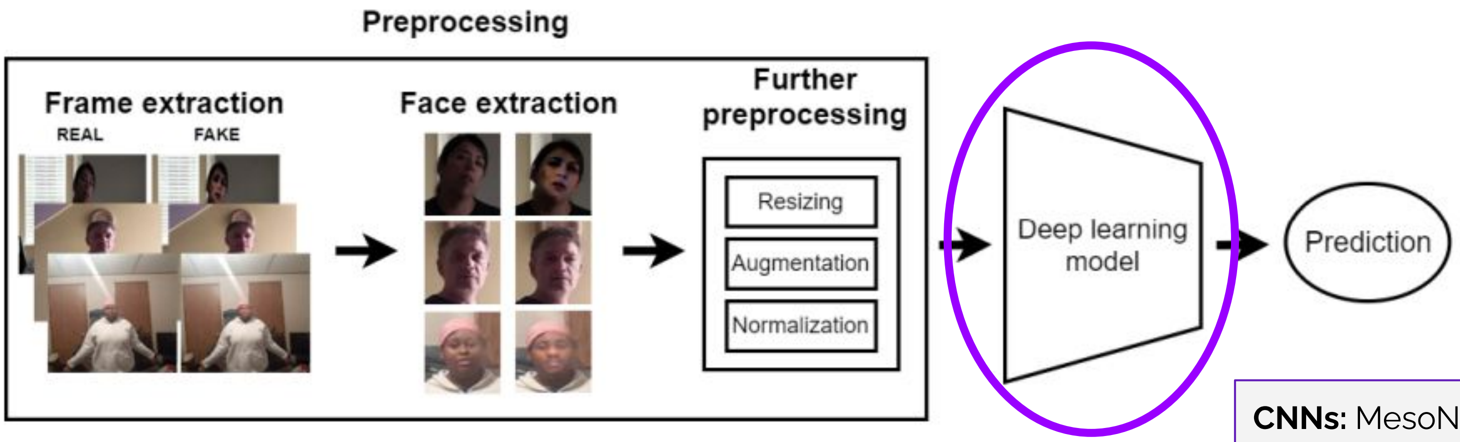


Verdoliva, L. (2020). Media forensics and deepfakes: an overview. *IEEE Journal of Selected Topics in Signal Processing*

synthetic media & deepfake detection

Deep learning	CNN · ViT · VLM
Spatial analysis	Local inconsistencies · Global inconsistencies
Biometric analysis	Eye blinking · Photoplethysmography · Other · Identity features
Temporal analysis	3D CNN · Temporal inconsistencies
Spectral analysis	Spectral artifacts · Frequency analysis networks
Multimodal analysis	Audio-visual inconsistencies

Deep learning-based DeepFake Detection



CNNs: MesoNet, ResNet, MobileNet, EfficientNet …

ViTs: DiT, DeiT, Swin, …

Baxevanakis, et al. (2022). The MeVer deepfake detection service: Lessons learnt from developing and deploying in the wild. In Proceedings of the 1st International Workshop on Multimedia AI against Disinformation (pp. 59-68).

Detection using physiological signals

Exploit inconsistencies in corneal specular highlights

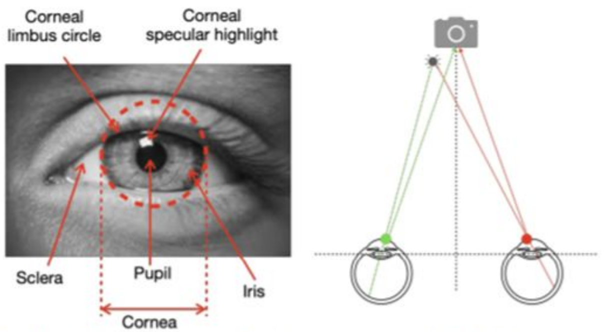


Fig. 3: *(left) Anatomy of a human eye. (right) The portrait setting with the corneal specular highlights.*

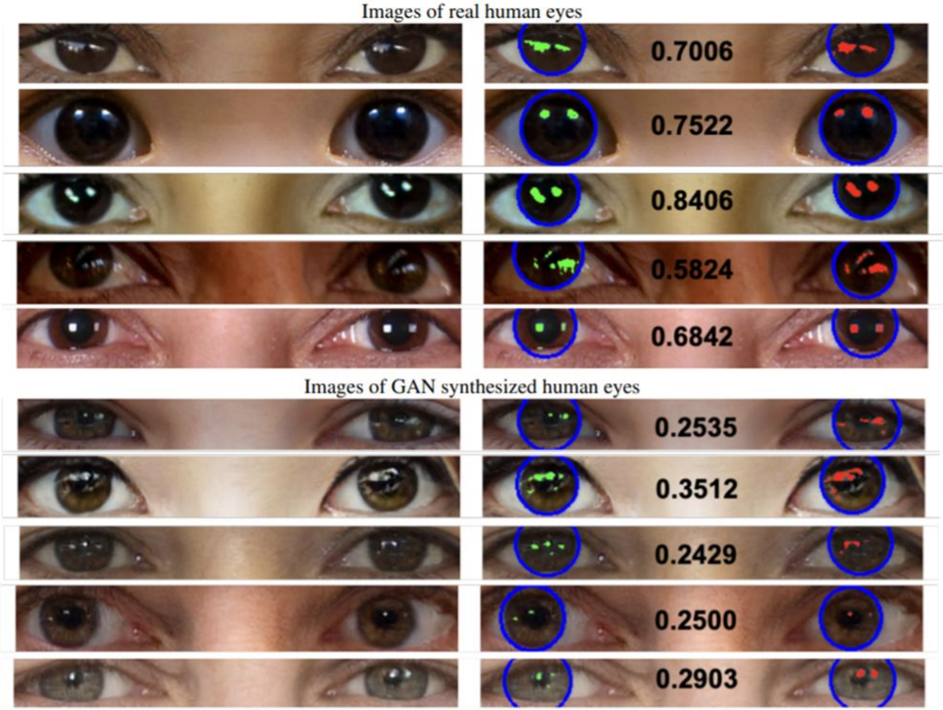


Fig. 5: *Corneal specular highlights from real human eyes (top) and GAN generated human faces (bottom). The right column corresponds to the detected corneal region (blue) and the specular highlights of two eyes (green and red). The IoU scores of the two corneal specular highlights are shown alongside the detections.*

Hu, S., Li, Y., & Lyu, S. (2021). Exposing GAN-generated Faces Using Inconsistent Corneal Specular Highlights. ICASSP 2021.

Artifact-oriented detection

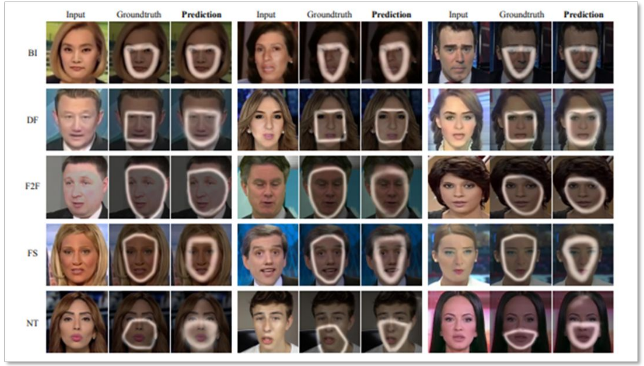
Visual artifacts

Matern, et al. (2019). Exploiting visual artifacts to expose deepfakes and face manipulations. In 2019 IEEE Winter Appl. of Computer Vision Workshops (WACVW) (pp. 83-92)

Limited resolution

Li, Y., & Lyu, S. (2019). Exposing DeepFake Videos By Detecting Face Warping Artifacts. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 46-52).

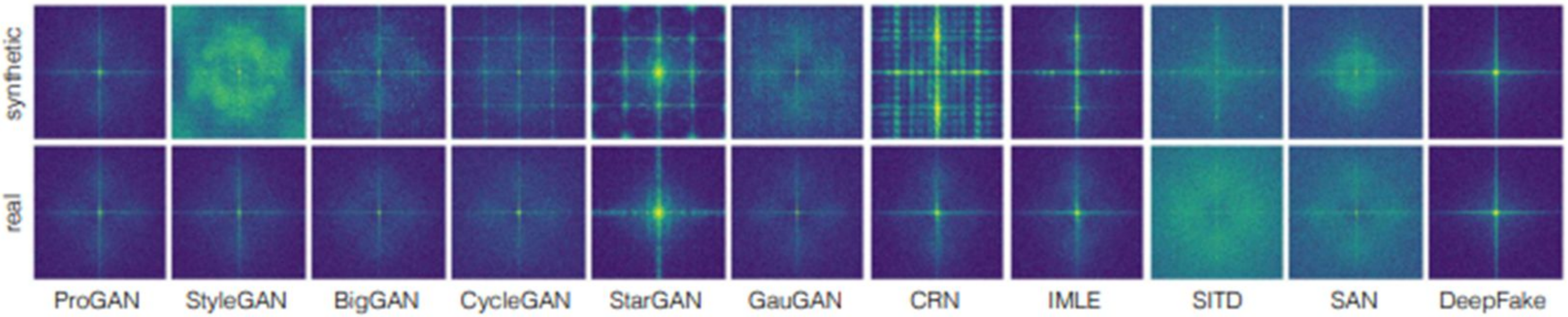
Blending regions



Li, et al. (2020). Face x-ray for more general face forgery detection. In Proc. of IEEE/CVF Conference on CVPR (pp. 5001-5010).

Spectral analysis

Premise: Common up-sampling methods, i.e. known as up-convolution or transposed convolution, are causing the inability of such models (GANs) to reproduce spectral distributions of natural training data correctly.

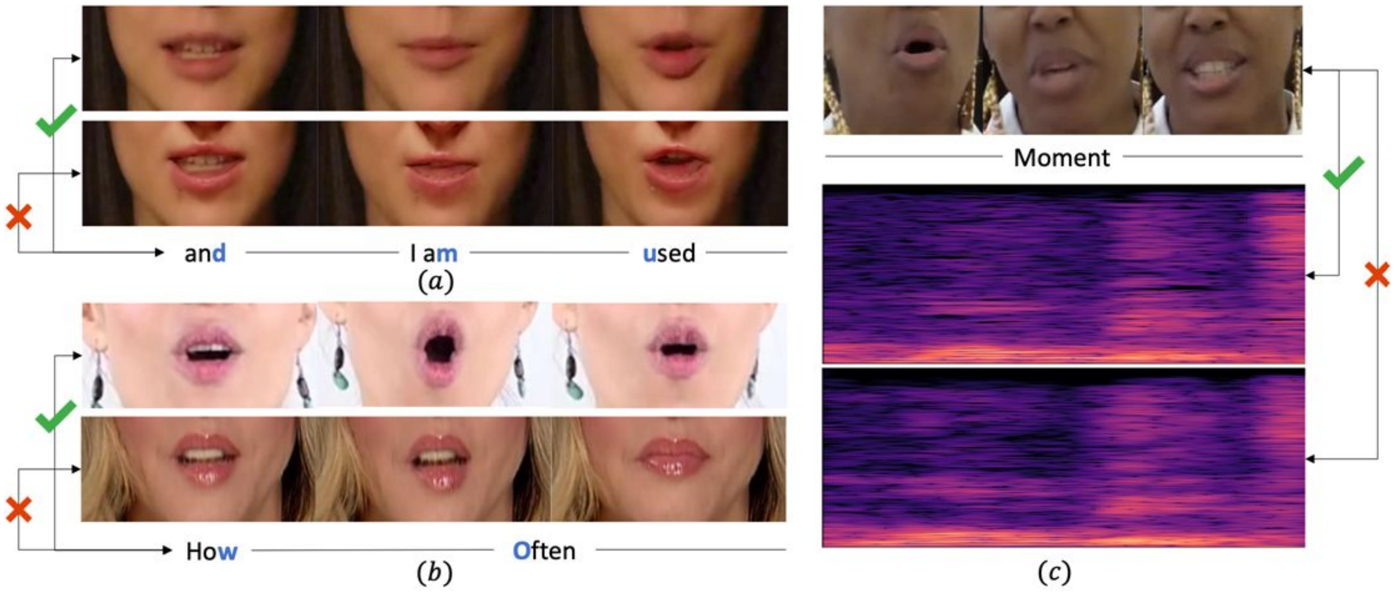


Wang, et al. (2020). CNN-generated images are surprisingly easy to spot... for now. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (Vol. 7).

Durall et al. (2020). Watch your Up-Convolution: CNN Based Generative Deep Neural Networks are Failing to Reproduce Spectral Distributions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 7890-7899).

Multimodal deepfake detection

Spatio-temporal convolutional residual blocks & 1D CNN fusion with attention components



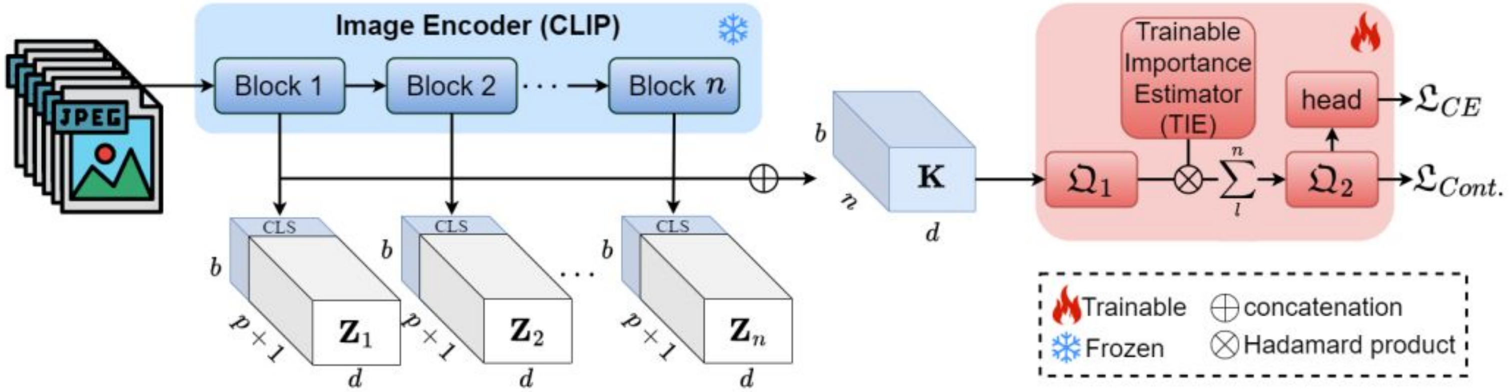
Zhou, Y., & Lim, S. N. (2021). [Joint audio-visual deepfake detection](). In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 14800-14809).

recent contributions

overview of recent contributions

- **RINE**: Representations from INtermediate Encoder blocks (ECCV ’24)
- **SPAI**: Any-Resolution AI-Gen. Image Detection by Spectral Learning (CVPR ‘25)
- **TextureCrop**: Enhancing SID through Texture-based Cropping (WACV WS’25)
- **SIDBench**: A Python Framework for Reliably Assessing Synthetic Image Detection Methods (ICMR WS’ 24)

RINE: Representations from Intermediate Encoder-blocks



- Leveraging the low-level visual information from the intermediate Transformer blocks
- Learning a forgery-aware vector space on top of CLIP's image representations
- Training on ProGAN data and evaluating on GAN, Diffusion outperforms SotA
- Indicative results (AP) on Synthbuster (Bammey, 2023): **96.2**% (DALL-E 2), **100%** (Stable Diffusion 1.4), **97.4%** (Midjourney)
- Decent performance on high-resolution in-the-wild AI-generated content

Koutlis, C., & Papadopoulos, S. (2024). [Leveraging representations from intermediate encoder-blocks for synthetic image detection](). In European Conference on Computer Vision (pp. 394-411). Springer, Cham.

RINE: Implementation details

- Backbone: CLIP ViT-L/14
- Data augmentations include: Gaussian blurring, JPEG compression, random cropping, random horizontal flip
- Resizing is omitted as it eliminates synthetic traces
- We train RINE for only 1 epoch (~8 min)!
- Batch size 128, learning rate 1e-3, Adam optimizer
- Light hyperparameter tuning on the fusion mechanism
- One NVIDIA GeForce RTX 3090 Ti GPU

Koutlis, C., & Papadopoulos, S. (2024). Leveraging representations from intermediate encoder-blocks for synthetic image detection. In European Conference on Computer Vision (pp. 394-411). Springer, Cham.

RINE: Results

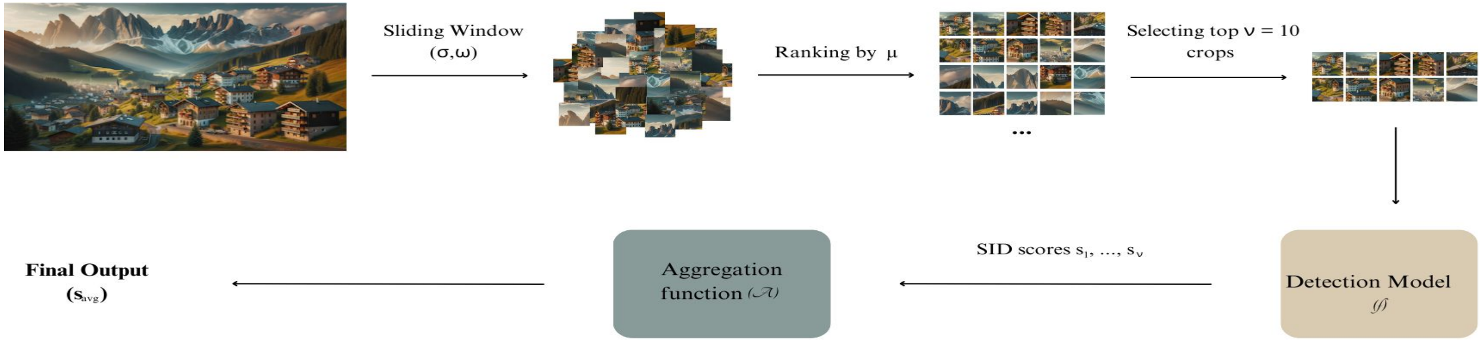
Table 2: Accuracy (ACC) scores of baselines and our model across 20 test datasets. The second column (# cl.) presents the number of used training classes. Best performance is denoted with **bold** and second to best with underline. Our method yields +10.6% average accuracy compared to the state-of-the-art.

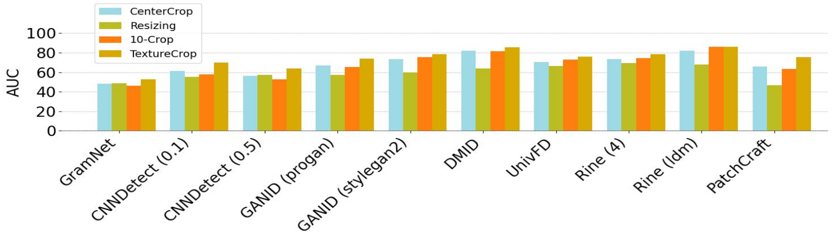
		Generative Adversarial Networks								Low level vision		Perceptual loss			Latent Diffusion			Glide				
method	# cl.	Pro-GAN	Style-GAN	Style-GAN2	Big-GAN	Cycle-GAN	Star-GAN	Gau-GAN	Deep-fake	SITD	SAN	CRN	IMLE	Guided	200 steps	200 CFG	100 steps	100 27	50 27	100 10	DALL-E	AVG
Wang [9] (prob. 0.5)	20	**100.0**	66.8	64.4	59.0	80.7	80.9	79.2	51.3	55.8	50.0	85.6	92.3	52.1	51.1	51.4	51.3	53.3	55.6	54.2	52.5	64.4
Wang [9] (prob. 0.1)	20	**100.0**	84.3	82.8	70.2	85.2	91.7	78.9	53.0	63.1	50.0	90.4	90.3	60.4	53.8	55.2	55.1	60.3	62.7	61.0	56.0	70.2
Patch-Forensics [10]	†	66.2	58.8	52.7	52.1	50.2	96.9	50.1	58.0	54.4	50.0	52.9	52.3	50.5	51.9	53.8	52.0	51.8	52.1	51.4	57.2	55.8
FrePGAN [27]	1	95.5	80.6	77.4	63.5	59.4	99.6	53.0	70.4	-*	-	-	-	-	-	-	-	-	-	-	-	-
FrePGAN [27]	2	99.0	80.8	72.2	66.0	69.1	98.5	53.1	62.2	-	-	-	-	-	-	-	-	-	-	-	-	-
FrePGAN [27]	4	99.0	80.7	84.1	69.2	71.1	**99.9**	60.3	70.9	-	-	-	-	-	-	-	-	-	-	-	-	-
LGrad [28]	1	99.4	96.1	94.0	79.6	84.6	99.5	71.1	63.4	50.0	44.5	52.0	52.0	67.4	90.5	93.2	90.6	80.2	85.2	83.5	89.5	78.3
LGrad [28]	2	99.8	94.5	92.1	82.5	85.5	99.8	73.7	61.5	46.9	45.7	52.0	52.1	72.1	91.1	93.0	91.2	87.1	90.5	89.4	88.7	79.4
LGrad [28]	4	99.9	94.8	**96.1**	83.0	85.1	99.6	72.5	56.4	47.8	41.1	50.6	50.7	74.2	94.2	**95.9**	95.0	87.2	90.8	89.8	88.4	79.7
DMID [15]	20	**100.0**	**99.4**	92.9	96.9	92.0	99.5	94.8	54.1	90.6**	55.5	**100.0**	**100.0**	53.9	58.0	61.1	57.5	56.9	59.6	58.8	71.7	77.6
UFD [16]	20	99.8	79.9	70.9	95.1	98.3	95.7	99.5	71.7	71.4	51.4	57.5	70.0	70.2	94.4	74.0	95.0	78.5	79.0	77.9	87.3	80.9
	1	99.8	88.7	86.9	99.1	**99.4**	98.8	99.7	**82.7**	84.7	**72.4**	93.4	96.9	**77.9**	96.9	83.5	97.0	83.8	87.4	85.4	91.9	90.3
RINE (Ours)	2	99.8	84.9	76.7	98.3	**99.4**	99.6	**99.9**	66.7	**91.9**	67.8	83.5	96.8	69.6	96.8	80.0	97.3	83.6	86.0	84.1	92.3	87.7
	4	**100.0**	88.9	94.5	**99.6**	99.3	99.5	99.8	80.6	90.6	68.3	89.2	90.6	76.1	**98.3**	88.2	**98.6**	**88.9**	**92.6**	**90.7**	**95.0**	**91.5**

* Hyphens denote scores that are neither reported in the corresponding paper nor the code and models are available in order to compute them.
** We applied cropping at 2000x1000 on SITD [46] for DMID [15] due to GPU memory limitations.
† Patch-Forensics has been trained on ProGAN data but not on the same dataset as the rest models. For more details please refer to [10].

Koutlis, C., & Papadopoulos, S. (2024). Leveraging representations from intermediate encoder-blocks for synthetic image detection. In European Conference on Computer Vision (pp. 394-411). Springer, Cham. / GitHub: https://github.com/mever-team/rine

TextureCrop: Enhancing SID through Texture-based Cropping

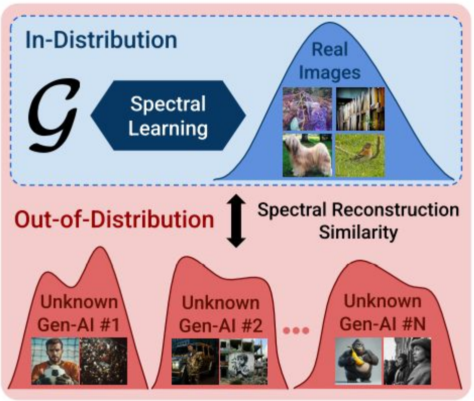




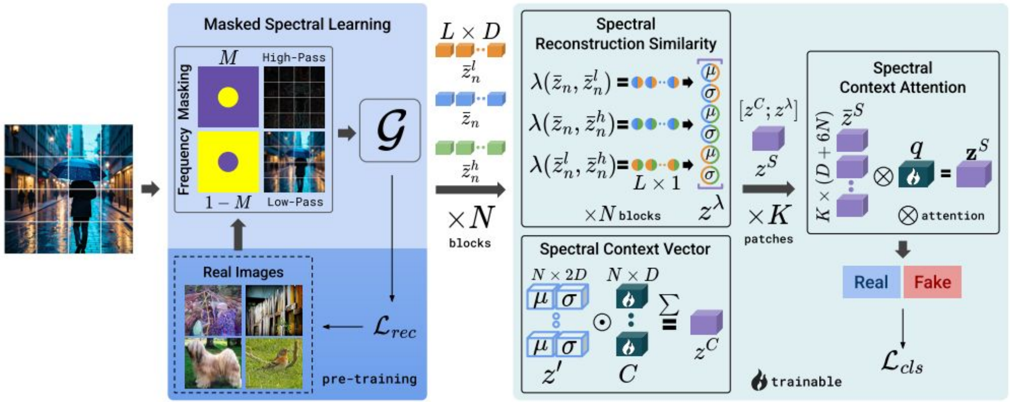
TextureCrop is a pre-processing technique designed to enhance synthetic image detection on high resolution images by selectively analyzing texture-rich regions within the image.

Konstantinidou et al. (2024). TextureCrop: Enhancing Synthetic Image Detection through Texture-based Cropping. WACV Workshop 2025

SPAI: Any-Resolution AI-Generated Image Detection by Spectral Learning



Architecture Overview



Key Idea: The spectral distribution of real images constitutes an invariant and highly-discriminative pattern for the task of AI-Generated Image Detection.

→ **Corollary**: Given a model of the spectral distribution of real images, AI-Generated images can be detected as Out-Of-Distribution (OOD) samples of this model.

SPAI uses:	**Frequency Reconstruction pre-text task** to model the spectral distribution of real images. **Spectral Reconstruction Similarity** to detect Gen-AI images as OOD samples of this model. **Spectral Context Attention** to capture subtle spectral inconsistencies in any-resolution images.

Karageorgiou, et al. (2025). Any-Resolution AI-Generated Image Detection by Spectral Learning. CVPR 2025

SPAI: Any-Resolution AI-Generated Image Detection by Spectral Learning

Image Size	< 0.5 MPixels			0.5 - 1.0 MPixels						> 1.0 MPixels				AVG
Approach	**Glide**	**SD1.3**	**SD1.4**	**Flux**	**DALLE2**	**SD2**	**SDXL**	**SD3**	**GigaGAN**	**MJv5**	**MJv6.1**	**DALLE3**	**Firefly**	
NPR [66]	72.2	89.6	60.5	19.8	3.9	12.5	18.1	60.6	83.2	15.3	19.8	97.1	38.0	45.4
Dire [72]	33.3	59.9	61.3	45.7	52.2	68.5	46.9	49.2	36.3	41.9	50.3	65.2	49.9	50.8
CNNDet. [71]	59.2	59.0	61.2	39.8	71.5	57.5	67.4	30.2	73.4	48.8	56.7	23.5	73.4	55.5
FreqDet. [23]	43.6	92.3	92.7	36.5	47.4	42.5	66.5	69.8	63.2	36.9	27.5	42.2	80.9	57.1
Fusing [34]	63.0	62.8	62.2	57.5	76.7	66.9	62.1	38.8	80.4	64.0	74.0	25.2	76.3	62.3
LGrad [65]	76.5	82.4	83.4	74.9	85.7	60.7	70.2	12.7	89.9	69.2	79.6	30.0	42.0	65.9
UnivFD [52]	63.3	80.8	81.2	36.3	91.4	84.3	78.3	28.6	86.2	57.1	60.5	31.0	95.5	67.3
GramNet [48]	78.2	83.9	84.3	78.6	85.2	66.7	77.8	19.2	85.0	63.8	84.9	42.9	38.0	68.4
DeFake [63]	86.1	64.2	63.6	90.5	41.4	66.2	52.3	87.7	71.7	67.0	87.5	93.3	39.4	70.1
PatchCr. [77]	78.4	95.7	96.2	86.9	81.8	95.7	96.7	33.8	98.0	79.0	96.1	28.1	79.1	80.4
DMID [7]	73.1	100.0	100.0	97.2	54.3	99.7	99.6	67.9	67.9	99.9	94.4	41.3	90.2	83.5
RINE [39]	95.6	99.9	99.9	93.0	93.0	96.6	99.3	39.1	92.9	96.4	81.2	41.8	82.9	85.5
SPAI (Ours)	90.2	99.6	99.6	83.0	91.1	96.5	97.4	75.9	85.4	94.5	84.0	90.2	96.0	**91.0**

Table 1. Comparison against state-of-the-art. Average AUC over 5 sources of real images is reported. Lower values are highlighted in red, while higher values are highlighted in green. Best overall average value is highlighted in bold, while second best is underlined. Our approach generalizes across all the considered generative approaches, even on ones producing imagery of extreme fidelity, such as SD3, where the single method [63] that scores better was required to explicitly train on relevant data.

Karageorgiou, et al. (2025). Any-Resolution AI-Generated Image Detection by Spectral Learning. CVPR 2025

SIDBench - A Python Framework for Reliably Assessing Synthetic Image Detection Methods

Why SIDBench?
Systematic evaluation that is easily extensible to new generative models and detectors

Evaluation
1. Overall Performance Accuracy
2. Threshold Calibration
3. Influence of Training Data
4. Influence of Image Resolution
5. Influence of Image Transformations (Gaussian blurring, Cropping, Resizing, JPEG recompression)

Modular architecture → incorporating new models

pyTorch Datasets & DataLoaders for incorporating new datasets

Schinas, M., & Papadopoulos, S. (2024). SIDBench: A Python framework for reliably assessing synthetic image detection methods. Proceedings of 3rd ICMR2024 Workshop on Multimedia against Disinformation (MAD'24) / Github: https://github.com/mever-team/sidbench

SIDBench - A Python Framework for Reliably Assessing Synthetic Image Detection Methods

Integrated SID Models

Categorization on two dimensions

- **Backbone architecture**
  - *ResNet* pretrained on ImageNet
  - *ViT* pretrained on CLIP
- **Input Features**
  - Raw images
  - Fingerprints: *frequencies, texture patterns, noise patterns*

12 state-of-the-art SID models (since 2020) trained on proGAN, StyleGAN or LDM images (Wang2020 & Ojha2023)

		Input features	
		Raw Images	Fingerprints
Backbone Architecture	ResNet + ImageNet	CNNDetect, DIMD (ResNet18)	LGrad, GramNet Fusing, NPR, FreqDetect
	ViT + CLIP	UnivFD, RINE, DeFake	-
	Other	-	PatchCraft

Evaluation Data

Family	Method	Source	#Images
Unconditional GAN	ProGAN	LSUN	8.0k
	StyleGAN	LSUN	12.0k
	StyleGAN2	LSUN	16.0k
	BigGAN	ImageNet	4.0k
Conditional GAN	CycleGAN	-	2.6k
	StarGAN	CelebA	4.0k
	GauGAN	COCO	10.0k
Perceptual loss	CRN	GTA	12.8k
	IMLE	GTA	12.8k
Low-level vision	SITD	Raw camera	0.36k
	IMLE	Standard SR	0.44k
Deepfake	FaceForensics++	Videos of faces	5.4k
Text-to-image Diffusion	Latent Diffusion	LAION	3.0k
	GLide	LAION	3.0k
Guided Diffusion	Guided [9]	ImageNet	1.0k
Auto-regressive	DALLE	LAION	1.0k

Wang2020

Ojha2023

- High-quality, high-resolution images from 9 generative models (1000 per model)
- *DALL·E2 & DALL·E3, Adobe Firefly, Midjourney v5, Stable Diffusion (1.3, 1.4, 2, XL), Glide*
- Original images from *Raise-1k*

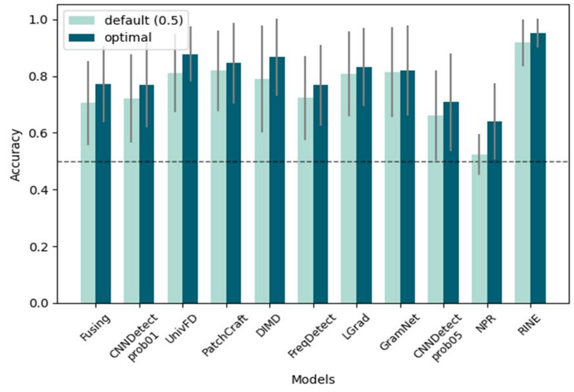
Synthbuster

SIDBench - Evaluation

method	Generative Adversarial Networks								Low level vision		Perceptual loss		Guided	Latent Diffusion			Glide			Dall·E	AVG
	Pro GAN	Style GAN	Style GAN2	Big GAN	Cycle GAN	Star GAN	Gau GAN	Deep fake	SITD	SAN	CRN	IMLE		200 steps	200 CFG	100 steps	100 27	50 27	100 10		
CNNDetect (prob 0.5) [38]	**100.0**	73.6	68.0	59.3	80.8	80.9	79.6	50.9	78.06	50.0	87.95	94.35	52.3	51.1	51.4	51.3	53.3	55.6	54.2	52.5	66.26
CNNDetect (prob 0.1) [38]	78.70	86.90	84.60	52.30	85.65	92.15	78.70	53.55	90.28	50.46	85.95	85.80	62.00	53.85	55.20	55.10	60.30	62.70	61.00	56.05	71.54
LGrad [37]	99.85	89.0	85.1	84.25	87.3	99.4	83.4	52.35	78.89	79.68	53.55	53.55	76.1	88.7	90.6	89.7	87.3	89.65	89.35	89.0	82.34
DIMD [6]	**100.0**	**99.1**	90.8	96.8	91.35	99.35	94.0	67.15	**96.11**	56.62	**99.35**	**99.35**	53.85	57.55	59.4	57.6	56.65	58.7	58.85	69.65	78.11
FreqDetect [10]	99.5	90.8	72.3	82.2	79.05	94.4	81.65	63.85	66.39	51.14	59.95	60.05	57.65	78.95	76.65	79.25	52.1	53.3	49.65	81.5	71.52
Fusing [14]	99.9	82.35	80.8	75.7	83.4	91.65	73.95	54.5	82.78	52.51	87.65	89.5	62.7	53.15	54.25	53.6	60.0	63.1	60.8	53.4	70.78
GramNet [21]	**100.0**	82.9	85.65	67.45	74.05	100.0	57.55	62.55	72.22	81.51	50.05	50.05	79.5	**98.45**	**98.45**	**98.7**	**91.75**	**93.4**	**95.55**	87.75	81.38
NPR [36]	50.0	50.0	49.95	50.0	49.7	50.0	50.0	54.75	83.06	50.0	50.1	50.05	50.25	49.95	49.95	49.95	51.55	52.35	51.45	50.0	52.15
UnivFD [25]	99.85	83.85	75.65	95.05	98.2	96.05	99.45	68.05	62.22	56.62	56.6	68.1	69.65	94.4	74.0	95.0	78.5	79.05	77.9	87.3	80.77
RINE [18]	**100.0**	88.0	**94.05**	**99.5**	**99.3**	**99.75**	**99.6**	**80.3**	90.56	68.26	90.45	91.45	76.1	98.25	88.2	98.6	88.75	92.55	90.7	**95.0**	**91.47**
PatchCraft [43]	**100.0**	91.85	89.3	95.25	69.05	100.0	71.2	56.15	88.06	**88.58**	50.05	50.05	**80.5**	91.0	90.05	90.9	78.9	82.4	85.2	85.5	81.7

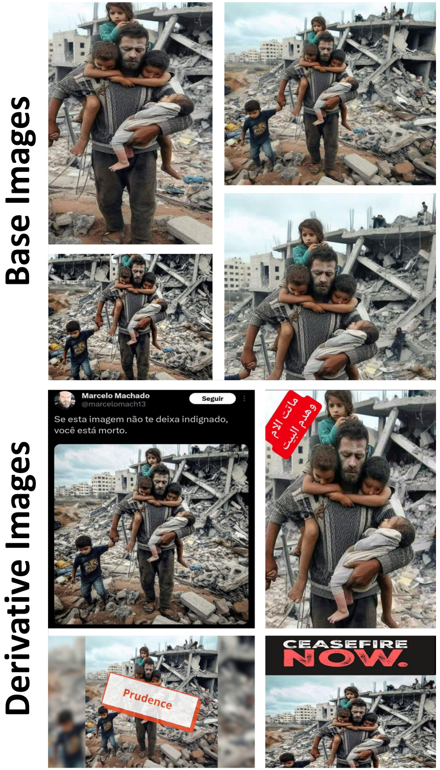
Detectors trained on GAN images

- Exhibit decent to good generalization to other ***GANs***
- Generally, perform poorly on ***Diffusion Models***, with some notable exceptions (GramNet, RINE, PatchCraft, and LGrad)
- *Threshold calibration:* improvements in terms of accuracy but challenging in real-world applications



Detection challenges - Derivative Images

- Detection methods are developed to detect "base" images, i.e. images that look like actual photos.
- Image content often circulates online in the form of "derivative" images (inclusion in memes and screenshots, addition of synthetic text, image in a photo, etc.)
- In many cases SID algorithms detect the later post-processing operations, instead of the actual signal of the image, plausibly increasing performance and causing several false positives.
- In our recent study, **considering only a subset of "base" synthetic images collected in the wild, decreases the average performance of 12 popular SID approaches by 6% in terms of AUC.**



CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS   école normale supérieure paris—saclay   université PARIS-SACLAY   AFP

Karageorgiou, D., Bammey, Q., Porcellini, V., Goupil, B., Teyssou, D., & Papadopoulos, S. (2024). Evolution of Detection Performance throughout the Online Lifespan of Synthetic Images. Presented at ECCV24 TWYN.

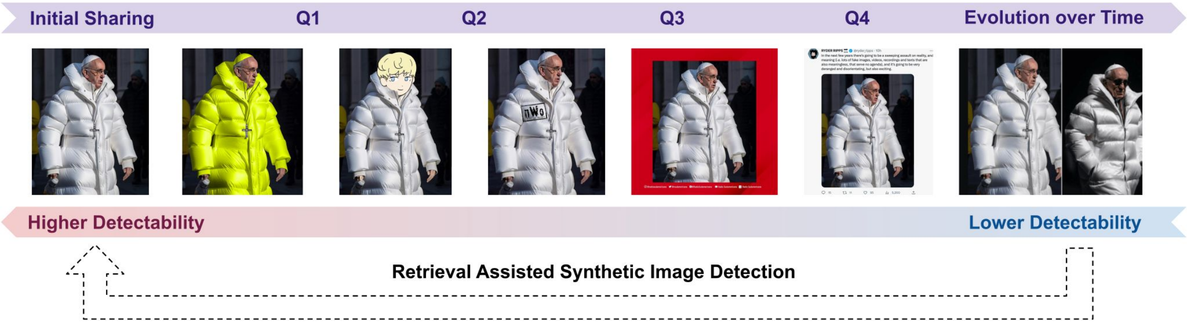
Evolution of Detection Performance throughout the Online Lifespan of Synthetic Images

Performance of popular SID algorithms on synthetic images collected in the wild.

Some perform even worse than random guessing!

Algorithm	Accuracy
GramNet	43.2
UnivFD	45.8
PatchCraft	47.9
Fusing	49.3
CNNDetect	49.4
FreqDetect	49.5
Dire	51.5
DIMD	53.0
NPR	59.2
Rine	61.8
LGrad	68.1
DeFake	72.8

- While state-of-the-art methods exhibit strong performance on lab-generated data, **they fail to discriminate between synthetic and real image cases collected in the wild, without further preconditions.**
- An image is continuously post-processed and reshared after its initial online appearance, leading to an average **3.2% drop in AUC between Q1 and Q4, even when considering only base images.**
- **Retrieval Assisted Synthetic Image Detection** exploits the early copies of an image submitted to a detection system, to facilitate the detection of the heavy post-processed late ones. **Increases AUC performance by 7.8% on average**.

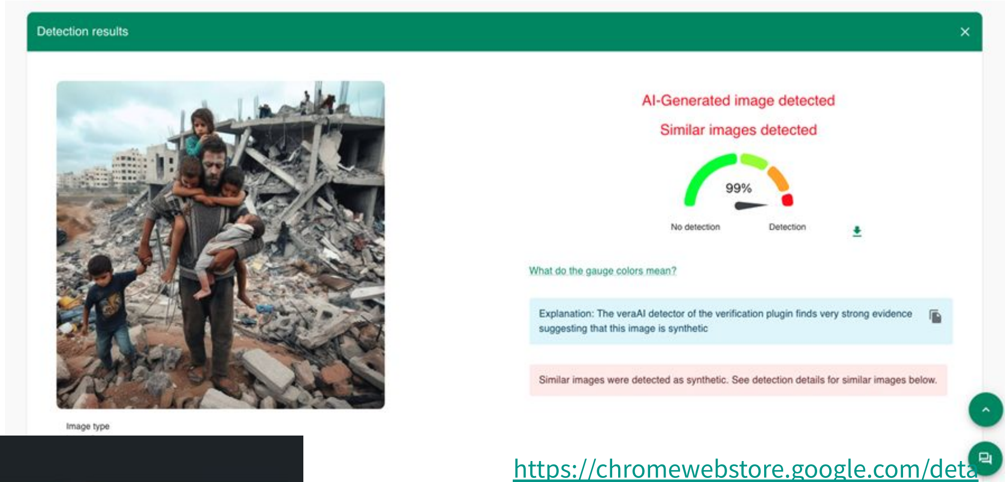


Karageorgiou, D., Bammey, Q., Porcellini, V., Goupil, B., Teyssou, D., & Papadopoulos, S. (2024). Evolution of Detection Performance throughout the Online Lifespan of Synthetic Images. Presented at ECCV24 TWYN.

parting thoughts

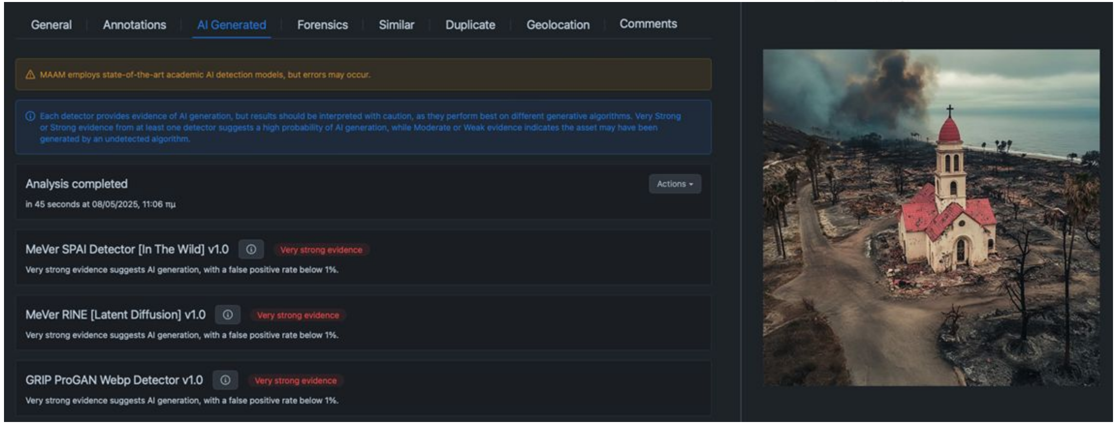
Putting results in practice

Verification Plugin



https://chromewebstore.google.com/detail/fake-news-debunker-by-inv/mhccpoafgdgbhnjfhkcmgknndkeenfhe?hl=en

Media Asset Annotation Management (MAAM)



https://maam.mever.gr/

Challenges when putting results in practice

Technical robustness and efficiency

- Some of the methods are compute-intensive / require GPU
- Managing multiple requests, esp. on large video files
- Managing multiple media formats and fetching from multiple sources

Reliability in the wild

- Over time new types of deepfakes emerge that older detectors fail to detect
- Several inputs are not appropriate for analysis, e.g. low resolution, high compression
- Out-of-distribution samples in terms of domain (e.g. X-Ray images, cartoons) lead to unpredictable results

Explaining results to end users

- Output of detectors is often a score in [0,1] (or 0-100), but that's not necessarily calibrated, nor should it be interpreted as a confidence value
- Limitations should be properly communicated

The Liar’s Dividend

“not all lies involve affirmative claims that something occurred (that never did): some of the most dangerous lies take the form of denials”

Paradox: The more widespread the public is educated about the hyper-realistic capabilities of deepfake generation, the more likely it is that an authentic piece of media can be (misleadingly) rejected as a deepfake!

Chesney, B., & Citron, D. (2019). Deep fakes: A looming challenge for privacy, democracy, and national security. Calif. L. Rev., 107, 1753.

Detector model cards

A standard way to inform and guide new users.
It contains:

- model architecture details
- datasets used
- evaluation results
- versioning scheme
- caveats and recommendations
- factors that affect performance

Model Card - DeepFake Detection Service

Model Details

- Developed by: CERTH-ITI Media Verification Team
- Model date: 03/02/2022
- Model version: 1.0. In this version, an ensemble of five models is deployed. Compared to previous versions, one model has been added, and several functionalities have been refactored to improve robustness.
- Processing pipeline:
  - Download the image/video from the input URL.
  - In the case of images:
    1. Use a Face Detector to detect all faces in the image.
    2. Feed each face to the model ensemble to get a DeepFake probability score in range $(0, 1)$.
  - In the case of video:
    1. Segment the input video into shots.
    2. For each shot, use a Face Detector to detect faces in the shot's frames.
    3. Perform a Face Clustering scheme to discard wrongly detected faces from the detector and organise the remaining faces into groups.
    4. Feed each face to the model ensemble to get a DeepFake probability score in range $(0, 1)$.
    5. Use an Aggregation Strategy to derive a video-level DeepFake probability for the entire video.
       (a) The face predictions of each face cluster are averaged to generate a cluster prediction.
       (b) Segment predictions derive based on the maximum prediction of their clusters.
       (c) The final video-level prediction is the maximum segment prediction.
- Model input: Video or image url.
- Model output: The video-level DeepFake probability, and the probability for each detected person in each video shot. Probabilities closer to 0 means **real** and closer to 1 means **fake**.
- Model type:
  - DeepFake prediction: a five model ensemble is used:
    1. a vanilla EfficientNet-b4,
    2. a Transformer head based on DETR with *fixed* positional embeddings on top of an EfficientNet-b4,
    3. a Transformer head based on DETR with *learned* positional embeddings on top of an EfficientNet-b4,
    4. a Multi-head Transformer based on DETR on top of an EfficientNet-b4,
    5. a vanilla EfficientNet-V2-m
  - Face Detection: we use the facenet-pytorch library.
  - Face Clustering: we employ the method described in this paper, where we extract face features using the pretrained InceptionResnetV1 provided in facenet-pytorch library, and used DBSCAN for clustering.
  - Shot segmentation: the feature extraction and similarity calculation described in this paper are used to extract peaks in the graph of distances of the consecutive frames.
- Citation details: (CERTH-ITI Media Verification Team, 2022) MeVer DeepFake Detection service.
- Feedback & Contact: {spirosbax, georgekordopatis, papadop}@iti.gr

Intended Use

- Primary intended use: Detect whether the faces present in the image or video from the provided URL have been manipulated using Deep Learning methods (DeepFake).
- Primary intended users: Journalists and media verification companies/organizations/groups.
- Out-of-scope uses:
  - The service cannot detect audio manipulations.
  - The service cannot detect if the images/videos have been tampered with using non-facial manipulations or other forgeries (e.g. splicing, copy-move, inpainting).
  - The service does not provide localized predictions on the extracted faces.
  - The service does not process videos longer than 12 minutes containing more than 50 shots due to reliability issues. Refer to the *Caveats and Recommendations* section.

Relevant Factors

- Factors for which service performance may vary are:
  - Manipulations: whether the networks have been trained with the presented DeepFake manipulation method or not. Refer to the *Training Data* section for more information.
  - Background faces: if there are many background low-resolution faces in the input image/video, it may affect the service's final prediction because it treats all detected faces equally.
  - Image/Video quality: blurry or low quality faces can affect the predictions.
  - Adversarial Attacks: alterations in the images/videos to evade detection can affect service performance.

Metrics

- Model performance measures:
  - Balanced Accuracy: defined as the mean of the recall computed on each class.
  - AUC: Area Under the Receiver Operating Characteristic
- Metrics decision: since the evaluation datasets are unbalanced, we want to avoid skewed metrics that might favor one class or alter the datasets (e.g. use sampling to balance the dataset).
- Decision threshold: a face prediction greater than 0.5 is considered **Fake** whereas a prediction lower or equal than 0.5 is considered **Real**.

Relevant Datasets

- FaceForensics++ (FF++): The dataset is organized in two manipulation categories, *Identity Swap* and *Expression Swapping*, each of which have two manipulation methods, *FaceSwap* and *DeepFakes*, and *NeuralTextures* and *Face2Face*, respectively. It contains 1k videos for each manipulation derived from the combination of 1k real videos. Evaluation on FF++ provides a performance indicator on different manipulation categories and methods. It should also be mentioned that compared to more recent datasets (e.g. CelebDF, DFDC.), the DeepFake quality in FF++ is visibly worse.
- CelebDF-V2 (CelebDF): Comprised of videos from celebrity interviews that have been manipulated using an improved version of the DeepFake manipulation method used in the FF++. It consists of 590 real and 5639 fake videos.

1

- DeepFake Detection Challenge (DFDC): Published by Facebook in the context of a DeepFake Detection Challenge, it contains 20K videos from hundreds of paid actors that have been used to generate 100K manipulated videos using improved DeepFake, FaceSwap methods, and three GAN-based manipulations. Due to its size and quality, it is often used both in research and production.
- WildDeepFake (WDF): This is one of the most recent datasets (2021) and in contrast to the previously mentioned datasets where the manipulations were applied automatically. It contains real-world DeepFakes scraped from various video-sharing websites as well as their corresponding real versions. It consists of 3.8k real and 3.5k fake videos. Due to its real-world nature, it is considered a challenging dataset.

Evaluation Data

- Datasets: FaceForensics++, CelebDF-V2, WildDeepFake
- Preprocessing: The WildDeepFake dataset is already preprocessed via the procedure described on the original paper. For each video in the FF++ and CelebDF datasets, we follow the same processing scheme used in the service. All face images are resized to 300×300 and normalized by the ImageNet mean and standard deviation.
- Postprocessing: we use the *Aggregations Strategy* described in the *Model Details* for all of the evaluation datasets.

Training Data

- Models $1 - 4$ were trained on the DFDC dataset while model number 5 was trained on the WildDeepFake dataset.
- We expect that the models will demonstrate good performance on facial manipulations included in the *DFDC* and *WildDeepFake* datasets, i.e. Identity Swap manipulations based on DeepFake, FaceSwap, and GAN-based algorithms, and various real-world DeepFake manipulations. Thus, we expect the service to be accurate when detecting DFDC manipulations and more sensitive to real-world manipulations.

Ethical Considerations

- Ricks and harms: The service presented should be used only as an **auxiliary** decision-making tool for anyone assessing the validity of an image/video; thus, the results should not be seen as the absolute truth.

Caveats and Recommendations

- Manipulation methods: the performance of DeepFake detectors highly depends on the manipulations they have seen during training. For example, if a detector is trained using only one kind of DeepFake manipulation, it would perform very poorly in real-world DeepFakes since there are numerous manipulations. The generalization to novel manipulations is an open issue in the research community that almost all approaches suffer from, including our service. Our training data contain various manipulations, yet we cannot guarantee good performance on unseen manipulations due to this generalization issue.
- Multiple faces: it is recommended that the multimedia inputs (videos or images) to the service contain only the face(s) in question and not any background faces that may distract the detection process and affect the final result.
- Video quality: it is also recommended that the input media be of the best quality possible since factors like quality and compression significantly affect the service performance.
- Video length: to ensure high-quality predictions and avoid computational overload, it is not recommended to submit very long videos and with many shots (c.f. *Out-of-scope uses*).
- Adversarial attacks: an adversarial attacker might affect the service performance by using methods such as a Projected Gradient Descent (PGP) attack. Even though these attacks might not be visible to the naked eye, they can fool a DeepFake detector into assessing that a DeepFake video is real.
- Facebook videos: The service does not guarantee successful processing of Facebook videos due to the strict Facebook policies that restrict video downloading.

Quantitative Analyses

Manipulation	Balanced Accuracy	AUC
FaceSwap	78.40%	86.74%
DeepFakes	86.20%	94.68%
NeuralTextures	57.65%	62.76%
Face2Face	59.02%	64.02%

Table 1: Balanced Accuracy and AUC for each manipulation in the FF++ dataset.

Dataset	Balanced Accuracy	AUC
FaceForensics++	70.31%	77.05%
CelebDF	82.75%	92.59%
WildDeepFake	84.94%	93.73%

Table 2: Balanced Accuracy and AUC for the service evaluated on three datasets.

Dataset	norm-1	norm-2	norm-inf
FaceForensics++	70.31%	64.04%	50.53%
CelebDF	82.75%	76.01%	50.00%
WildDeepFake	84.94%	63.04%	50.00%

Table 3: Balanced Accuracy scores on three datasets adversarially manipulated with the PGP attack (hyperparameters: $eps = 0.2$).

Performance Intuition

- *Balanced Accuracy* is the average of the accuracy in each class. Since our datasets are unbalanced, it would be misleading to just report the overall accuracy of the system. For example, in a dataset where 90% of the data are DeepFakes, a naive classifier that outputs only ones regardless of the input would get 90& accuracy, which is misleading. Owning to its intuitive nature, we consider Balanced Accuracy to be our primary metric to gauge our ensemble's performance.
- *Area Under the Curve (AUC)* takes into account the Miss Rate or, in other words, how often the model wrongly thinks a DeepFake is Real, as well as the True Positive Rate, meaning how often the model correctly classifies DeepFakes. Thus the AUC is an overall metric describing these two rates, and in a classification system, such as ours, higher is better. However, it does not consider the 0.5 probability threshold, which is an essential parameter in our setting; therefore, we consider it as an auxiliary metric.
- From the tables 1,2 it is evident that our system performs much better on the CelebDF and WildDeepFake datasets rather than the FF++. It can be argued that this is due to our training data lacking *Expression Swap* examples which is one of the two manipulation categories of that dataset (c.f. *Relevant Datasets* and *Training Data* sections for more information).
- In table 1 the *FaceSwap* and *DeepFakes* manipulations belong in the *Identity Swap* category while the rest in the *Expression Swapping* category. Since we observe worse performance

2

Takeaways

- Synthetic media detection is a complex challenge as there are numerous types of generative models and ways to blend and manipulate generative and authentic content
- A lot of the essence of building AI solutions for synthetic media detection is selecting/adapting an appropriate deep learning architecture with the goal of separating between synthetic/manipulated and authentic content
- A lot of challenges ahead:
    - Arm's race nature of synthetic media detection
    - Access to platform data
    - Data annotation
    - Computational requirements

Acknowledgements

Despoina Konstantinidou
Research Assistant

Dimitris Karageorgiou
PhD Candidate @ UvA

Manos Schinas
Senior Research Engineer

Dr. Christos Koutlis
Post-doc Researcher

Dr. Hannes Mareen
Post-doc Researcher @ UGent

Dr. Luisa Verdoliva
Professor @ Univ. Naples

Olga Papadopoulou
Senior Manager

Dr. Nikos Sarris
Senior Manager

Dr. Yiannis Kompatsiaris
ITI & MKLab Director

Dr. Symeon Papadopoulos
Principal Researcher

Dr. Panagiotis Petrantonakis Assis.
Professor @ AUth

Dr. Efstratios Gavves
Assoc. Professor @ UvA

vera.ai
AI-CODE
Mediterranean Digital Media Observatory
ELLIOT
AI4TRUST

mever

Thank you

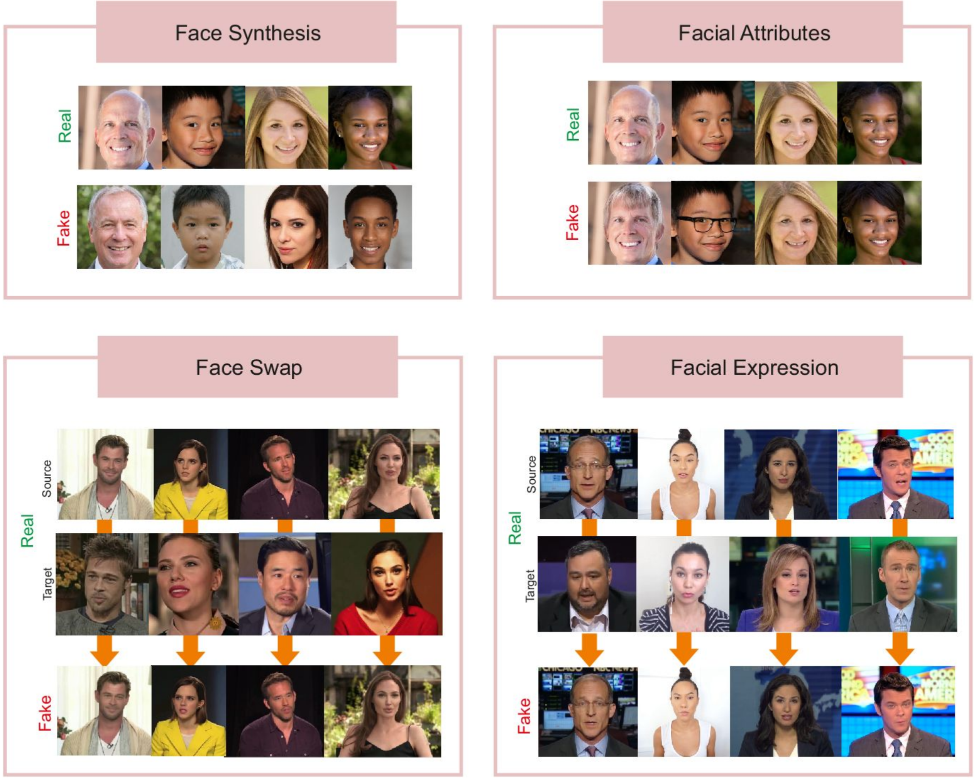
Reach me at papadop @ iti.gr
Follow @sympap &
@meverteam

Deepfake detection

Content, generated (at least partly) by deep neural networks, that seems authentic to human eye.

Four main types of face DeepFakes: a) Entire face synthesis, b) Attribute manipulation, c) Identity swap, d) Expression swap. Lip syncing and voice generation are also common types in video and audio content.

Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. Information Fusion, 64, 131-148.



Text-to-Image diffusion models

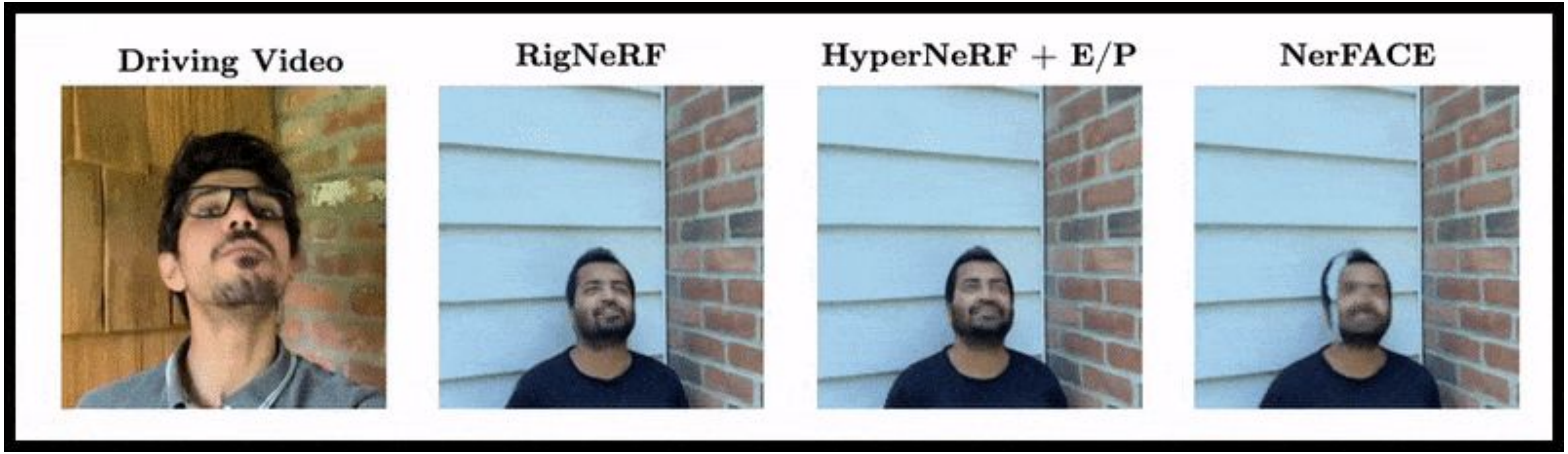
"An astronaut riding a horse in photorealistic style"

https://openai.com/dall-e-2/

"Teddy bears swimming at the Olympics 400m Butterfly event."

https://imagen.research.google/

RigNeRF: Fully Controllable Neural 3D Portraits



3D morphable face models + NeRF enable the full control of head pose and facial expressions learned from a single portrait video!

Athar, S., Xu, Z., Sunkavalli, K., Shechtman, E., & Shu, Z. (2022). Rignerf: Fully controllable neural 3d portraits. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 20364-20373).

A new breed of Generative AI tools & services

Pika Labs

Synthesia

Midjourney

runway

Leonardo.Ai

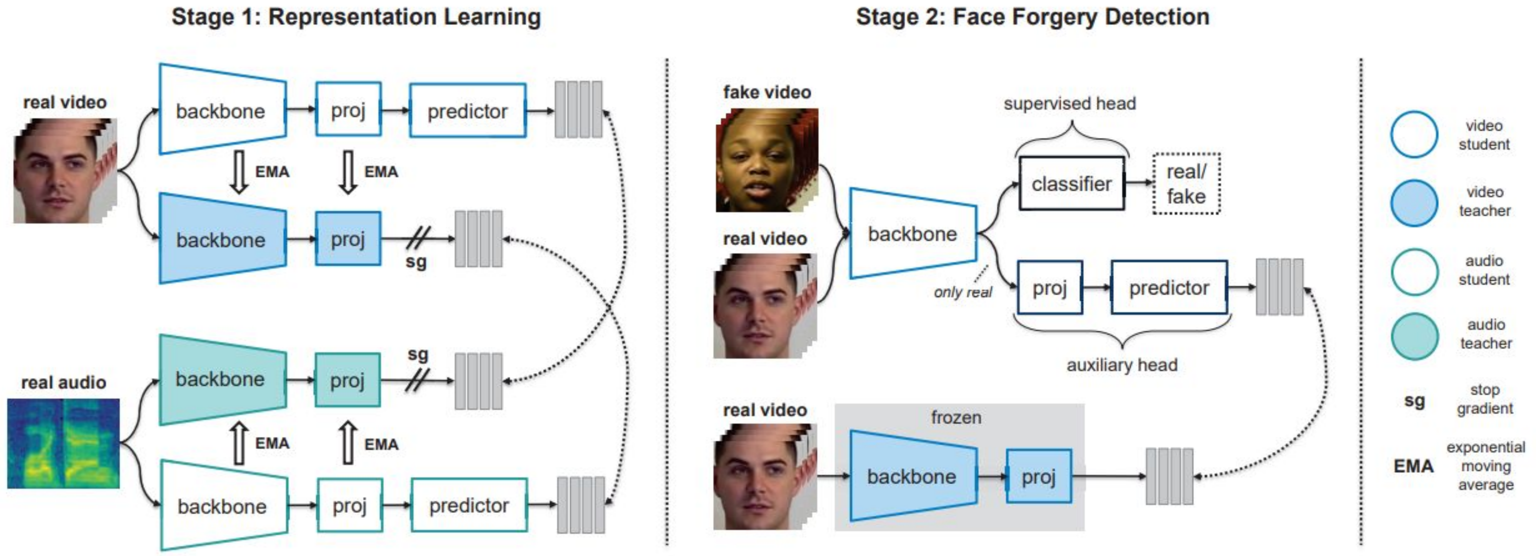
DALL·E 3

Craiyon

stability.ai

Multimodal deepfake detection

First: learn temporally dense video representations in a self-supervised way. Second: perform multi-task learning: forgery prediction & embedding reconstruction



Haliassos, A., Mira, R., Petridis, S., & Pantic, M. (2022). Leveraging real talking faces via self-supervision for robust forgery detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 14950-14962).

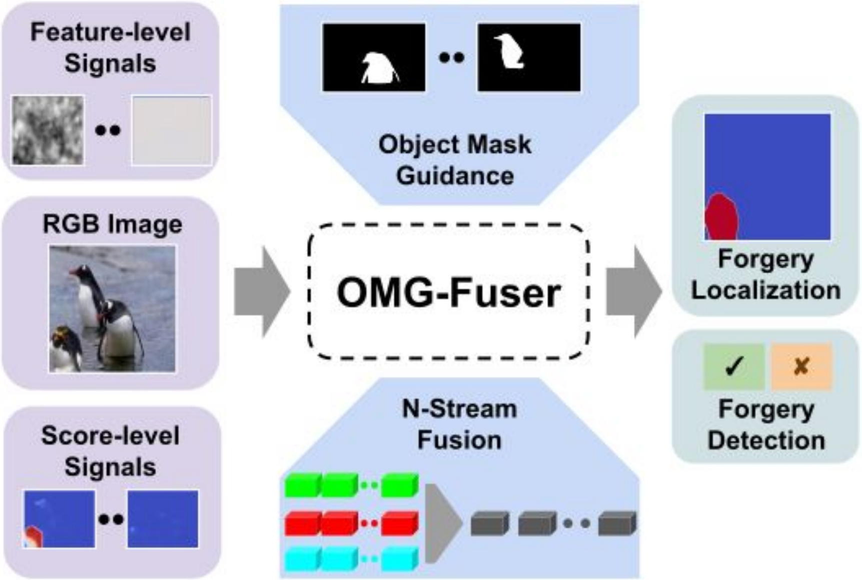
Image forensics: OMG-Fuser fusion transformer

Detecting **if** and **where** an image has been semantically altered:

Image Forgery Detection & Localization

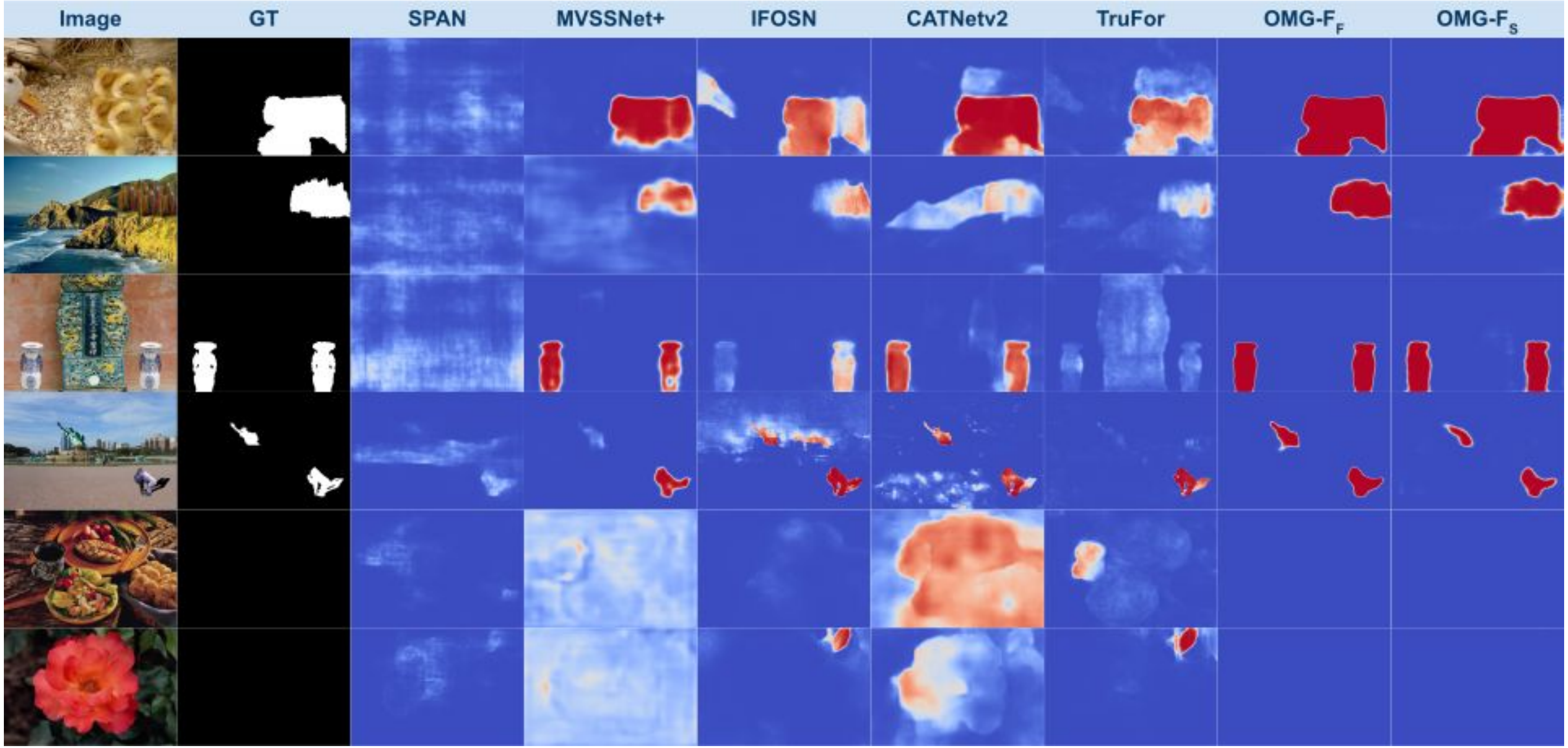
New modular architecture - **OMG-Fuser**:

- Exploits the knowledge encoded in large pretrained image segmentation models.
- Fuses an arbitrary number of clues.
- More than 20% performance increase across several established benchmarks.



Karageorgiou, D., Kordopatis-Zilos, G., & Papadopoulos, S. (2024). Fusion Transformer with Object Mask Guidance for Image Forgery Analysis. CVPR Workshop on Media Forensics (WMF) 2024

OMG-Fuser: Example outputs



Karageorgiou, D., Kordopatis-Zilos, G., & Papadopoulos, S. (2024). Fusion Transformer with Object Mask Guidance for Image Forgery Analysis. CVPR Workshop on Media Forensics (WMF) 2024