Global 500 Evaluation

Jan 15th, 2024

This evaluation aims to assess deepfake detection methods using a curated set of 502 test cases. Detailed information about the curation is provided in the **Curation Protocol**. Summary on the evaluation attributes of each subsets is as follows:

- **Celeb-DF(v2)**: High-resolution clean and static faces
- **DFDC**: Diverse environmental and subject conditions
- **Google DFD**: Dynamic subject actions
- **FaceForensics++**: Diverse deepfake synthesis methods
- **AV-Deepfake1M**: Partially fake segments
- **Openforensics**: Multiple unique subject faces in a single frame
- **(*Additional*) Diffusion-based**: Video diffusion methods (only for fake samples)

1. Evaluation Metrics

The evaluation is conducted using 4 different measures. Since each subset in the curation reflects distinct evaluation attributes, we report both the *overall* metrics and *subset-wise* metrics. Please refer to the **Appendix A** for the detailed explanation of each metric.

- **Performance metrics**: (1) Accuracy, (2) F1-Score, and (3) AUC (AUC-ROC)
- **Calibration metrics**: (4) ECE (Expected Calibration Error) [1]

2. Evaluation Setup

- **____API [2]**: Offers a deepfake detection API for videos/images.
- **AltFreezing [3]**: Presented at CVPR 2023, this method is among the latest in state-of-the-art deepfake detection. The model is trained using FaceForensics++.

The ____ API provides predictions in 1-second intervals, while AltFreezing offers per-frame predictions. To ensure a fair comparison, we post-process these to achieve per-sample evaluation results. We employ two strategies for deriving overall predictions:

- **Averaged Fakeness Score**: Normalize predictions over all time frames, calculating an average fakeness score for each sample.
- **Peak Fakeness Score**: Assign the highest fakeness score observed at any point in time as the sample's overall prediction.

Note that the **Peak Fakeness Score** is more effective in identifying instances where only certain segments of the overall time frames are manipulated. Our main evaluation focuses on **Averaged Fakeness Score**. Please refer to the **Appendix B** for the implementation details.

3. Evaluation Result

____API is better: DFDC (Adversarial conditions) // GoogleDFD (Dynamic subject actions)
AltFreezing is better: FaceForensics++ (Diverse Synthesis Methods)

Both methods struggle to distinguish between real and fake samples on the subsets:

- Celeb-DF-v2 (High-resolution clean and static faces)
- AV-Deepfake1M (Partially fake segments)
- Openforensics (Multi-Face & Single Frame).

(1) Performance Analysis - Accuracy & F1-Score

Limited Unseen Generalization

- AltFreezing shows near-random performance across different subsets, except in the FaceForensics++ datasets on which it was trained.
- While ____API shows relatively reasonable performance in the DFDC and GoogleDFD subsets, this is likely because these datasets were included in its training phase.

Near-Random Performance on Subsets

- Both ____API and AltFreezing methods show significant limitations in distinguishing real and fake samples within subsets of AV-Deepfake1M and Openforensics.
- ____API consistently mislabels all samples as "real," whereas AltFreezing labels all as "fake," regardless of their actual labels.

(2) Performance Analysis - AUC

- Table 3 shows AUC of 0.9263 for Celeb-DF-v2 and 0.9094 for Google DFD. These results are in line with the AUCs reported in its official paper (0.895 and 0.985).
- The discrepancy between Accuracy/F1-Score and AUC shows that the model might perform well at certain thresholds, but the threshold used to calculate Accuracy and F1-Score might not be the optimal one (Please also see **Appendix C**.)

(3) Performance Analysis - ECE

- Table 4 shows that both ____API and AltFreezing exhibit quite high calibration errors (ranging from 0.3 to 0.5) in their confidence levels.
- With AltFreezing generally showing lower ECE values than ____API, indicating it consistently provides more reliable probability predictions.

[Table 1]: Main Results - Accuracy

	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.6853	0.5612	0.8608	0.8036	0.6100	0.5000	0.5333

AltFreezing	0.5956	0.5612	0.5253	0.5179	0.8700	0.5000	0.5000
LipForensics	0.6884	0.53	0.6939	0.7143	0.9800	0.5333	0.5000
GenConViT	0.7160	0.6939	0.7552	0.7321	0.8000	0.4828	0.6167

[Table 2]: Main Results - F1-Score

	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.5486	0.2182	0.8429	0.7556	0.3810	0.0000	0.1250
AltFreezing	0.7104	0.6950	0.6725	0.6747	0.8850	0.6667	0.6667
LipForensics	0.6165	0.1154	0.7134	0.6364	0.9796	0.2222	0.0000
GenConViT	0.7612	0.7581	0.8045	0.7887	0.8246	0.5161	0.6102

[Table 3]: Main Results - AUC

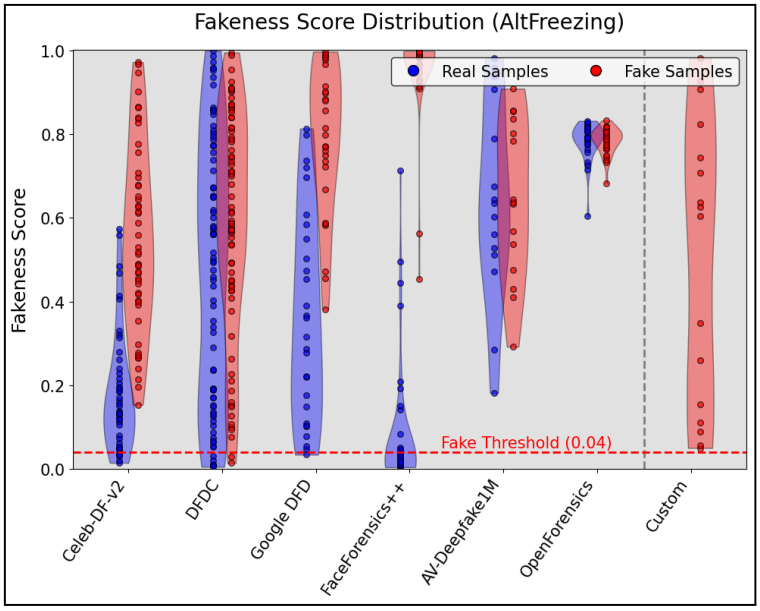
	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.7711	0.8176	0.9148	0.9745	0.8016	0.4267	0.4678
AltFreezing	0.7756	0.9263	0.6028	0.9094	0.9988	0.4844	0.4478
LipForensics	0.7886	0.8663	0.7474	0.8444	0.9992	0.6311	0.4478
GenConViT	0.8558	0.8142	0.9278	0.9911	0.9420	0.5524	0.5856

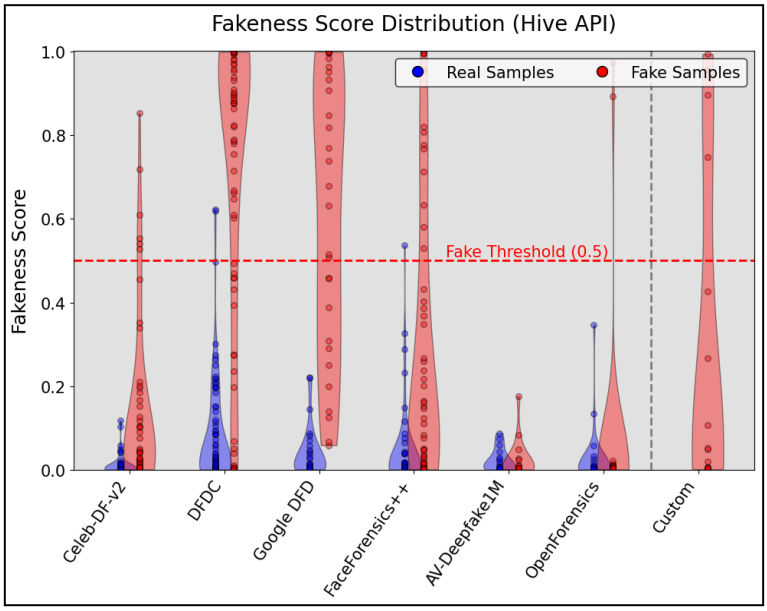
The model's AUC values being much higher than its Accuracy & F1-Score values indicates that if we could accurately determine the optimal threshold for each data subset, the model's performance could significantly improve. However, as the Figure 1 demonstrates, these optimal thresholds vary considerably between domains. In practical real-world scenarios, where the boundaries of domains are often indistinct, identifying both the specific domain of a sample and its ideal threshold is not feasible.

[Table 4]: Main Results - ECE

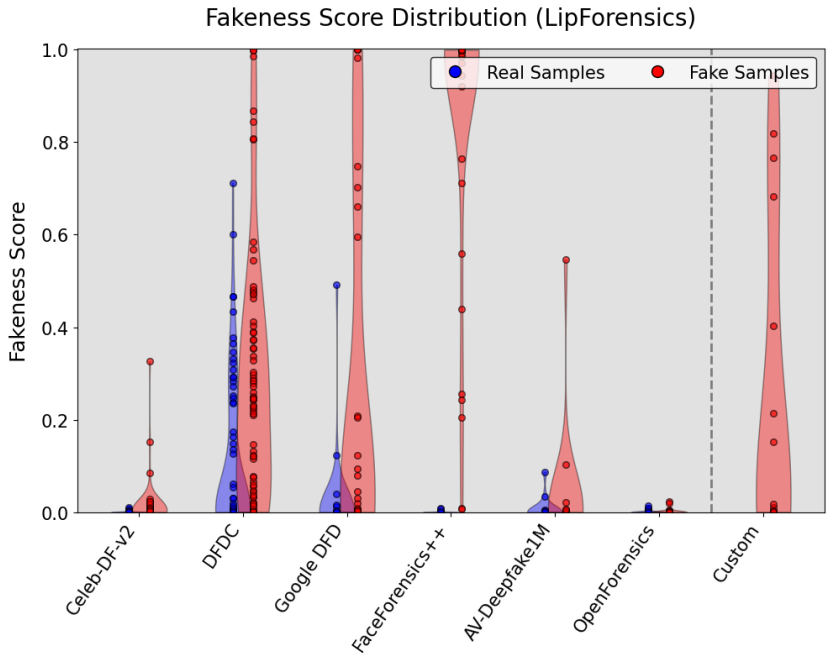
	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.4707	0.5134	0.4749	0.5374	0.4725	0.4863	0.4889
AltFreezing	0.3020	0.4345	0.2783	0.2830	0.4734	0.3044	0.2853
LipForensics	0.4628	0.5013	0.3977	0.5110	0.5226	0.5128	0.5128

GenConViT	0.4194	0.4041	0.4178	0.4503	0.4710	0.4302	0.5075

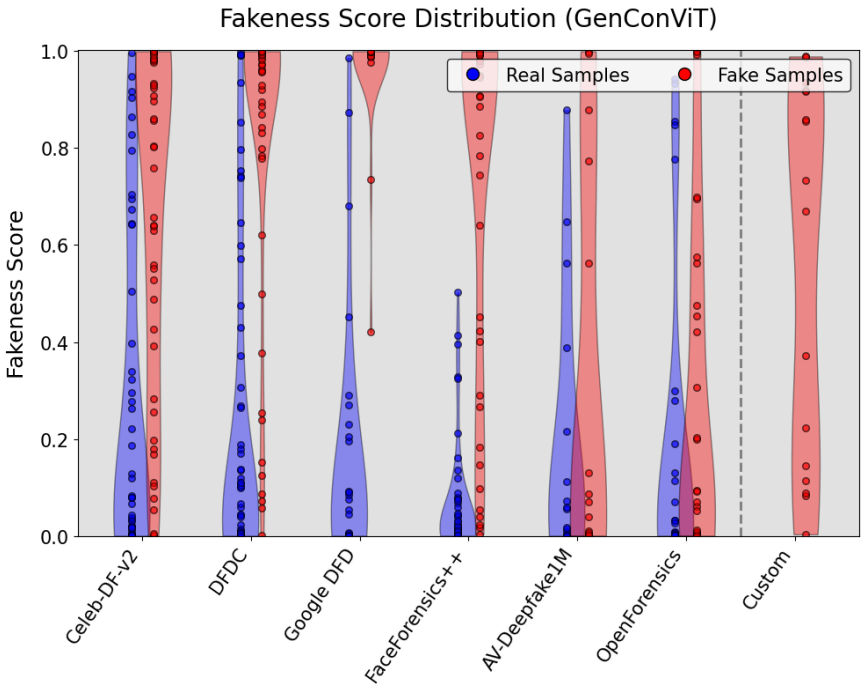




[Figure 1]. Visualization of Fakeness Score of ____API(Left) and AltFreezing(Right)



[Figure 2]. Visualization of Fakeness Score of LipForensics



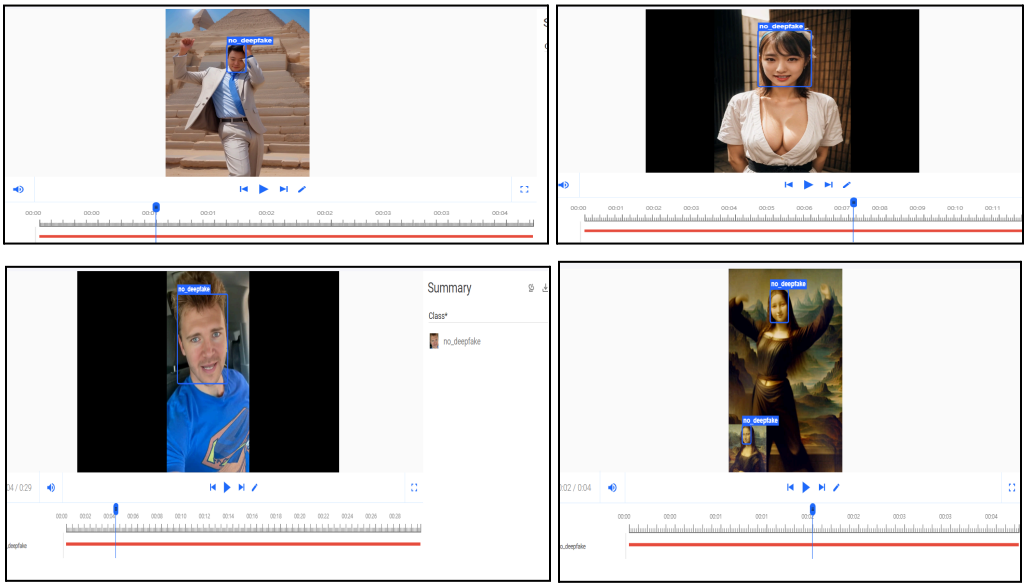
[Figure 3]. Visualization of Fakeness Score of GenConViT

(3) Custom Diffusion-Based Fake Samples

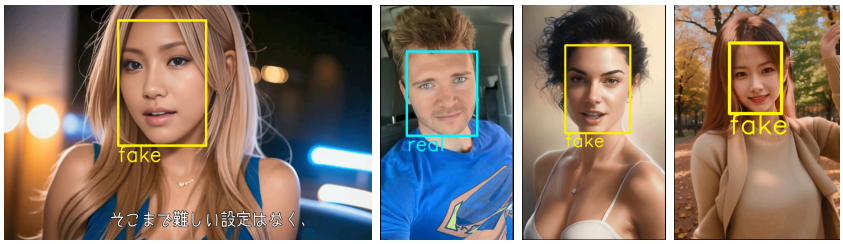
In Table 5, both methods are evaluated on 16 diffusion-based custom fake samples. Note that those samples have no corresponding original samples. While ____API struggles to detect Diffusion-based generation samples, AltFreezing predicts surprisingly well on those samples.

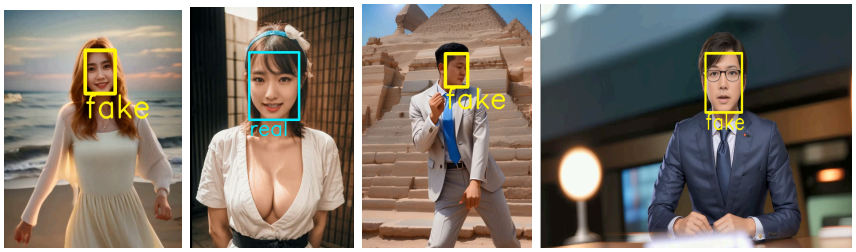
[Table 5]: Fakeness prediction on custom fake samples. Success: (O) and Failure: (X)

Method	Fake Samples (File Index 251 - 266)															
____API	x	x	x	x	O	O	x	O	x	x	x	x	x	x	x	x
AltFreezing	O	O	O	O	O	O	O	X	O	X	O	O	O	O	O	O
LipForensics	x	O	x	O	x	x	O	x	O	x	x	x	x	x	x	x



[Figure 2]: ____API prediction results on custom fake samples.





[Figure 3]. AltFreezing prediction results on custom fake samples.

[1]. Naeini, Mahdi Pakdaman, Gregory Cooper, and Milos Hauskrecht. "Obtaining well calibrated probabilities using bayesian binning." *Proceedings of the AAAI conference on artificial intelligence*. Vol. 29. No. 1. 2015.

[2]. “Deepfake Detection.” *Documentation* | ____, docs.the____.ai/docs/deepfake-detection.

[3]. Wang, Zhendong, et al. "AltFreezing for More General Video Face Forgery Detection." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023.

Appendix

A. Evaluation Metrics

Accuracy: $\frac{|Correct\ Samples|}{|Total\ Samples|}$: This metric calculates the proportion of correct samples out of the total (500 samples). Since the curation set is balanced in terms of labels (real:fake = 1:1), this accuracy also reflects the balanced accuracy.

F1-Score: $\frac{2 \times (Precision \times Recall)}{(Precision + Recall)}$: As a blend of *precision* (true positives out of all positive predictions) and *recall* (true positives out of all actual positives), F1-Score offers a balanced evaluation of model performance.

AUC (AUC-ROC): Area under the receiver operating curve: As a measure that plots true positive rate (sensitivity) against false positive rate (1-specificity), AUC-ROC provides an overall evaluation of a model's ability to distinguish between classes. The closer the AUC is to 1, the better the model is at correctly classifying true positives and true negatives.

ECE (Expected Calibration Error): $\sum_{i=1}^{n} \frac{|B_i|}{|Total\ Samples|} \times |Acc(B_i) - Conf(B_i)|$:

Imagine you have a weather app on your phone that predicts the chance of rain each day. Let's say on Monday, the app tells you there's a 70% chance of rain, so you take your umbrella to school. But it doesn't rain. On Tuesday, it says 70% again, and again it doesn't rain. If this keeps happening, you'd start to think that the app isn't very good at predicting rain, even though it seems pretty sure about it. The ECE is a way to measure how much we can trust the probabilities given by a prediction system, in our case, a model that predicts whether a video is a deepfake or not.

- Perfect Calibration: If the app is perfectly calibrated and says there's a 70% chance of rain, then out of 10 days when it says 70%, it should actually rain on about 7 of those days.
- Overconfident Predictions: If the app says there's a 70% chance of rain on 10 different days, but it only rains on 3 of those days, the app is overconfident. It's telling you it's pretty sure it will rain, but it's wrong most of the time.
- Underconfident Predictions: If the app says there's a 30% chance of rain, but it ends up raining 7 out of 10 times, then the app is underconfident. It's not sure it will rain, but it actually rains a lot.

In the case of detecting deepfakes, if a model has a high ECE, it means that when it says it's 90% sure, it might be right a lot less often than that, which would make it unreliable.

So the ECE gives us a way to say, "Hey, this model says it's really sure about its predictions, but should we believe it?" The lower the ECE, the more we can trust it.

B. Implementation Details

B.1 Method Setups

____API Although ____'s dashboard displays predictions for every frame, its API response offers results in 1-second intervals. Without method details from ____, a score threshold of 0.5 is applied for evaluation. This choice is supported by the alignment of these results with ____'s visualized outcomes, suggesting that ____ likely uses a 0.5 threshold.

AltFreezing For compatibility, image samples are converted into 1-second videos at 30 frames per second for evaluation with AltFreezing. The score threshold of 0.04, suggested as the optimal threshold in the paper, is used for calculating Accuracy and F1-Score.

B.2 Method Efficiency

A notable finding is that AltFreezing's processing time on an NVIDIA RTX A5000 GPU for a 30-second 1080p video ranges between 30-40 seconds, whereas ____API (async) completes the task in under 10 seconds. Although the resources utilized by ____API are not specified, this efficiency suggests that ____API either employs a lighter model or implements steps for efficiency, such as downsampling resolution during detection process.
In the case of GenConViT, a single GPU (Quadro RTX 5000) is used with 3GB of RAM allocated. GenConViT uses Variational AutoEncoder (VAE) and ConvNet (from Meta) as a backbone. The face recognition and localization are processed by dlib library under the hood. This helps GenConViT manage to inference 500 samples in less than an hour.
For LipForensics, a single GPU (RTX 2080Ti) is utilized. The algorithm begins by extracting frames in batch, a process that takes approximately 1.5 hours. Following this, landmark data is extracted using FAN, which requires 2 GB of RAM and takes about 5 hours. This landmark data is then used to crop images to get the mouth only frame and taking around 1 minutes with 1.5 GB of RAM only. As a result, LipForensics is able to process 500 samples in approximately 8 hours

[Table 6]: Required cost on curation dataset inference

Method	Used Resource	Time spent until completion
____API (Async)	Unknown	< 5 mins*
AltFreezing	1 RTX A5000 GPU	7 hrs**
GenConViT	1 Quadro RTX 5000	< 1h**
LipForensics	1 RTX 2080 Ti	8 hrs

* ____API appears to internally distribute multiple requests across various resources.
** With parallel processing, this estimated time could be faster.

[lip forensics as-is]
8.mp4 (fake), size: 2.4 MB
+ extract frame --> 11.61 s --> 456 frames
+ extract landmark --> 45.44 s --> 456 frames
+ crop mouth --> 17.41 s --> 456 frames
+ inference --> 9.4 s
+ total = 83.86 s

[LipForensics downsizing]
+ Only process 25 frames (sampling) (the minimum default number of Lipforensics model)
- Extracting Frames: Approximately 0.64 seconds
- Extracting Landmarks: Approximately 2.49 seconds.
- Cropping Mouth: Approximately 0.95 seconds
- Inference (Lipforensics detection) with 25 frames: Approximately 9.4 seconds
+ Total time: 13.48 Seconds

[Lip forensic next step]
- Incorporate dlib

[genconvit as-is]
- the evaluation script used 10 frames extracted from each video
- “hog” an option from dlib to extract face was used

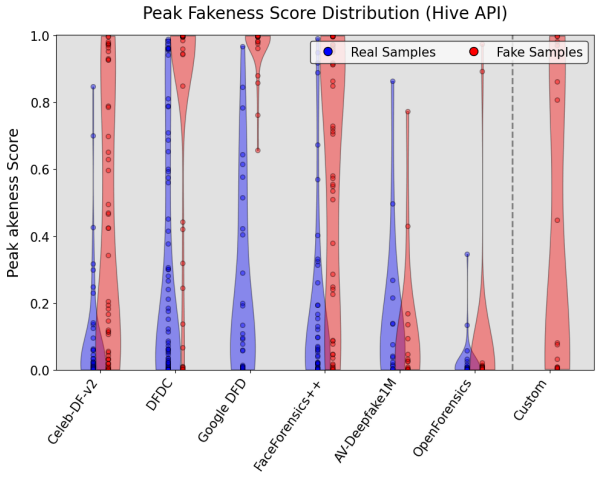
[genconvit improvement]
- dlib - library for face recognition, actually failed to detect face from around 40 videos and disregarded them. So we improved the script.
-

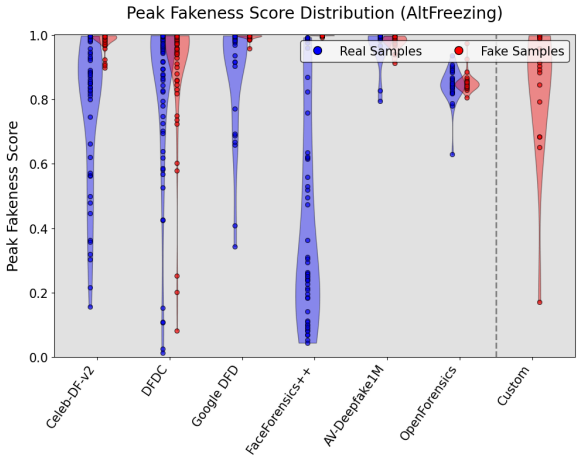
C. Peak Fakeness Score

The results in Table 7 display the AUC performance based on Peak Fakeness Score. When compared to Table 3, the scores show mixed results.

[Table 7]: Main Results - AUC

	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.7569	0.7934	0.9105	0.9886	0.7972	0.4133	0.4678
AltFreezing	0.7746	0.9075	0.6058	0.8788	0.9996	0.4000	0.5467





D. The evaluation results of the API tested on smaller datasets due to the constraint of a 200 API call limit.

Main Results - Accuracy (516)

original	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.6853	0.5612	0.8608	0.8036	0.6100	0.5000	0.5333
AltFreezing	0.5956	0.5612	0.5253	0.5179	0.8700	0.5000	0.5000

MORE FALSE 4:1 (214 : 57)

new_version	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M	Open Forensics
____API	0.7380	0.6066	0.8333	1.0	0.0.6818	0.3889	0.5333
AltFreezing	0.8450	0.8197	0.8333	0.7778	0.9394	0.7778	0.5000
Reality Def	0.8514	0.9016	0.8555	0.9444	0.8333	0.7222	

Only False (187)

new_version	Overall	Celeb DF v2	DFDC	Google DFD	FaceForensics++	AV-Deepfake1M
____API	0.8340	0.7333	0.9649	1.0	0.8684	0.2727
AltFreezing	0.9079	0.8222	0.8772	1.0	1.0	0.9091
Reality Def	0.8634	1.0	0.8596	0.9524	0.8684	0.6364
Sensity	0.7841	0.9778	0.7368	1.0	0.8421	0.3636

* Each vendor brings something valuable to the table. Reality Defender, for example, may be pricey, but they provide decent justifications for the detection. All APIs show better performance when tested on a higher number of deepfake videos and fewer real ones. Reality defender did much better than ____ in this scenario.

E. The list of repos we looked at but didn't decide to work on.

Index	**Project**	**Things that worked**	**Things that did not work**	**Remedy**
1	**RealForensics**	Pretrained model can be loaded	Missing pretrained model for audio branch.	Fixed the dependencies of getting face Landmarks (rely on `face-alignment` project) Implemented landmark extraction code
2	FACTOR (face forgery)	Frame (image) extraction worked Pretrained model can be loaded	Requires “real” counterparts video as reference for the detection, which is non-sense. We do not have them for the collected video	The project depends on `FaceX-Zoo` project. Made the code for extracting images from mp4 file work.
3	FACTOR (audio visual)	Pretrained weights is provided (huBERT-based)		Fixed the dependencies. Obtained pretrained model.

4	**UniversalFakeDetect**	All code worked	Image only	modified DataLoader and inference code to produce fake scores tried different backend models (RN50, CLIP:ViT) RN50 only improved CLIP:ViT a bit to ~50% accuracy. 0.66 F1.
5	**HiFi_IFDL**	All code worked	Image only	Both pretrained models loaded properly
6	CMUA-Watermark	NA	Pretrained models cannot be downloaded (Authors turned off the hosting service)	Repo check
7	FedForgery	NA	Cuda 9.0 (too old)	
8	https://arxiv.org/abs/2307.06272v1 Image (Diffusion)		No code	