Chaboom at ImageCLEF 2026 Deepfake Task: Multimodal Deepfake Detection and Generation

ImageCLEF-Deepfake at CLEF 2026

Sehwan Kim[1,*], Youngchan Lee[1] and Jinseo Kim[1]

[1]Department of AI and Big Data, Soonchunhyang University, Asan 31538, Republic of Korea

Abstract

This paper describes the system submitted to the ImageCLEF 2026 Deepfake Task [1], which comprises two sub-tasks: deepfake generation (image and audio modalities) and deepfake detection (image and audio modalities). For generation, we propose a landmark-constrained diffusion pipeline for images and an F5-TTS-based zero-shot voice cloning pipeline for audio. For detection, we employ a multi-model ensemble with hybrid scoring for images and a pretrained AASIST model for audio. Our system achieves a final score of 0.2486 for image generation, 0.3378 for audio generation, 0.5022 for image detection, and 0.7564 for audio detection on the official ImageCLEF 2026 evaluation. The landmark-constrained image generation achieves a scaled landmark distance score of 0.7645, while the audio detection pipeline demonstrates strong real-audio precision.

Keywords

deepfake detection, deepfake generation, diffusion model, voice cloning, ImageCLEF 2026

1. Introduction

The rapid advancement of generative AI has enabled the creation of highly realistic synthetic media, commonly referred to as deepfakes. Such content poses serious threats to information integrity and public trust, making both detection and controlled generation important research challenges [2].

The harms associated with deepfakes span a wide range of concerns, including identity fraud, non-consensual impersonation, reputational damage, and a broader erosion of trust in audio-visual evidence. As generative models continue to improve in realism and accessibility, the perceptual gap between synthetic and authentic media keeps narrowing, raising the bar for detection systems that must generalize to generation techniques they have not seen during development. At the same time, understanding state-of-the-art generation pipelines is itself valuable for detection research, since insight into how synthetic artifacts are produced can inform the design of more robust and generalizable detectors. This dual relationship motivates treating detection and generation as complementary problems rather than independent ones, and is the perspective adopted throughout this paper.

The ImageCLEF 2026 Deepfake Task [1] addresses this dual challenge by jointly benchmarking generation and detection across two modalities, image and audio. Unlike detection-only benchmarks, this joint formulation evaluates participants' generation systems not only on perceptual and identity-preservation quality but also on their ability to evade both participant-submitted and organizer-provided detectors, while detection systems are evaluated on their ability to generalize to diverse, participant-generated synthetic content. This setup mirrors the practical adversarial dynamic between deepfake creators and detectors more closely than either task considered in isolation. Concretely, the benchmark covers two complementary sub-tasks: (1) generating deepfake images and audio under specified constraints, and (2) detecting deepfakes across image and audio modalities. We participated in all four modality-task combinations.

Our main contributions are:

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

*Corresponding author.

kimsehwan22@sch.ac.kr (S. Kim); maxyounglee0000@gmail.com (Y. Lee); kjs20021224@gmail.com (J. Kim)

- A landmark-constrained image generation pipeline combining ControlNet, IP-Adapter, and a photorealistic diffusion backbone.
- An F5-TTS-based zero-shot voice cloning system with LUFS-normalized post-processing for audio generation.
- A multi-model ensemble with conditional score boosting for image deepfake detection.
- A chunk-based AASIST inference pipeline for audio deepfake detection.

The rest of this paper is organized as follows. Section 2 describes the task settings and evaluation metrics. Section 3 reviews related work on the models and datasets used in this study. Sections 4 and 5 detail our methodology for generation and detection, respectively. Section 6 presents experimental results and analysis. Section 7 concludes the paper.

2. Task Description

The ImageCLEF 2026 Deepfake Task consists of two sub-tasks, each covering image and audio modalities.

2.1. Sub-task 1: Deepfake Generation

2.1.1. Image Generation

Participants are given short reference videos (15–40 seconds) for each subject in a few-shot generation scenario, along with facial landmark constraints, and must synthesize photorealistic face images that satisfy the given geometric constraints while preserving the subject's identity. Only the face region is constrained; the rest of the body and the background may be generated using any method. Beyond geometric and identity fidelity, generated images are also expected to be classified as real by deepfake detectors, whether implemented by the organizers or by other participants in the Detection sub-task.

Dataset Specification: The development set contains 335 videos, each depicting a different subject and named `ID_{personID}_dev.mp4`. Generation constraints are provided in `generation_constraints.json`, whose keys follow the format `ID_{personID}_GT_frame{frameNumber}`; for each subject, between one and three frames must be generated. Each key specifies 478 normalized facial landmark coordinates extracted with MediaPipe FaceMesh (`refine_landmarks=True`), with x and y values in the $[0, 1]$ range relative to image size. Auxiliary MediaPipe Pose coordinates are also provided but are optional; they were not used in our system.

The submission set requires generation of 1,002 images at 256×256 resolution in PNG format, one for each key in `generation_constraints.json`, following the naming convention `ID_{personID}_GT_frame{frameNumber}.png` and placed in an `Images/` directory.

Evaluation Metrics: Image generation is evaluated using a composite scoring scheme that multiplies an Image Quality Score by a weighted combination of evasion rates against participant-submitted and organizer-provided deepfake detectors (participant weight: 0.7, organizer weight: 0.3). The Image Quality Score consists of the following components:

- **Identity Similarity Score**: Measures how well the generated image preserves the subject's identity, computed via the `buffalo_s` model from the InsightFace library between the generated image and the original development video.
- **Constraints (Landmark) Score**: Measures how well the generated image satisfies the facial landmark constraints, computed as the Euclidean distance between the 478 MediaPipe facial landmarks extracted from the generated image and the target landmarks in `generation_constraints.json`.

As with audio generation, the multiplicative scoring scheme emphasizes that high image quality alone is insufficient without evasion capability against detection systems.

2.1.2. Audio Generation

Participants receive reference audio recordings and target texts per subject, and must synthesize speech that mimics the vocal characteristics of the reference speaker while correctly producing the target content.

Dataset Specification: The reference set contains recordings from 16 speakers (IDs 2–17), each with multiple utterances categorized by duration: L (Long), M (Medium), and S (Short). All audio files are provided in WAV format at 16 kHz sample rate, 16-bit depth, and mono channel.

The submission set requires generation of 480 synthetic utterances following the naming convention `u_{id}_c_{L/M/S}_i_{idx}.wav`, comprising 16 speakers $\times$ 3 duration categories $\times$ 10 sentences. Target transcriptions are provided in `submission_set.xlsx`.

Evaluation Metrics: Audio generation is evaluated using a composite scoring scheme that multiplies an Audio Quality Score by a weighted combination of evasion rates against participant-submitted and organizer-provided deepfake detectors (participant weight: 0.7, organizer weight: 0.3). The Audio Quality Score consists of the following components:

- **NISQA MOS**: Non-Intrusive Speech Quality Assessment, predicting Mean Opinion Score (1–5 scale) for perceptual quality.
- **ECAPA / WavLM Score**: Speaker similarity measured via cosine similarity of embeddings from ECAPA-TDNN and WavLM models.
- **WER / CER**: Word Error Rate and Character Error Rate, measuring transcription accuracy against target text.
- **Evasion Score**: Success rate against participant-submitted and organizer-provided deepfake audio detectors.

The multiplicative scoring scheme emphasizes that high audio quality alone is insufficient without evasion capability against detection systems.

2.2. Sub-task 2: Deepfake Detection

Sub-task 2 focuses on binary deepfake detection for both image and audio modalities. For each input sample, participants must submit a prediction indicating whether the sample is real or deepfake. Throughout this task, label 0 denotes real media and label 1 denotes deepfake media.

2.2.1. Image Detection

In the image detection track, participants are given 24,404 PNG images and must classify each image as real or deepfake. The task requires detecting not only obvious facial manipulation artifacts but also subtle synthetic patterns such as unnatural texture, inconsistent illumination, boundary artifacts around the face, and global image-level generation traces. Since both real and generated images may contain compression noise, blur, or other low-level artifacts, the task also requires careful calibration to avoid false positives on real images. The submission file contains the image identifier and the predicted binary label for each image.

2.2.2. Audio Detection

In the audio detection track, participants are given 11,520 WAV files and must classify each file as real or deepfake. The audio files vary in duration and sampling rate, so detection systems must handle variable-length waveforms before producing a single file-level prediction. The submission file contains the audio file identifier and the predicted binary label for each file.

2.3. Evaluation Metrics

Table 1 summarizes the evaluation metrics for each modality and sub-task.

Table 1
Evaluation metrics per sub-task and modality

Sub-task	Modality	Metrics
Generation	Image	Landmark Error, Identity Score
Generation	Audio	WER, CER, Speaker Similarity, Audio Quality, Evasion Score
Detection	Image	Final Score (Acc.-based)
Detection	Audio	Final Score (Acc.-based)

3. Related Work

3.1. Image Deepfake Generation

Controllable face image generation typically combines a pretrained diffusion backbone with conditioning modules that constrain geometry and identity. In this work, facial landmarks were extracted using MediaPipe FaceMesh [3], and high-level facial attributes were classified in a zero-shot manner using CLIP [4]. The generation backbone was Stable Diffusion 1.5 [5], guided by ControlNet [6] to enforce the target landmark geometry and by IP-Adapter [7] to preserve the subject's identity from a reference image. Sampling was performed with the UniPC multistep scheduler [8] to reduce the number of denoising steps required for high-fidelity synthesis.

3.2. Audio Deepfake Generation

Zero-shot voice cloning systems synthesize speech in an unseen speaker's voice from a short reference recording, without speaker-specific fine-tuning. We considered three pretrained text-to-speech models for this task: SpeechT5 [9], a unified-modal encoder-decoder model for joint speech and text processing; XTTS-v2 [10], a GPT-based multilingual zero-shot TTS model; and F5-TTS [11], a flow-matching-based TTS model, which was selected for the final system. Speaker similarity between generated and reference audio was measured using Resemblyzer [12], and output loudness was normalized following the EBU R128 [13] broadcast loudness standard.

3.3. Image Deepfake Detection

Image deepfake detection has been widely studied using pretrained visual classifiers that capture facial artifacts, texture inconsistencies, and synthetic image patterns. In this work, we used three publicly available pretrained detectors for image-based deepfake classification. The first two models, `deepfake-detector-model-v1` [14] and `Deepfake-Detect-Siglip2` [15], were selected as deepfake-oriented detectors. In particular, the SigLIP2-based detector provides a vision-language representation [16] that differs from conventional image-only classifiers. The third model, `ai-image-detection` [17], was included as a general AI-generated image detector to capture global synthetic artifacts such as unnatural texture, color, and background patterns.

3.4. Audio Deepfake Detection

For audio deepfake detection, we adopted AASIST [18], an audio anti-spoofing model that integrates spectral and temporal graph attention mechanisms. AASIST operates on raw waveform inputs and has been widely used for detecting spoofed or synthetic speech. In our system, AASIST was used as a pretrained detector in a chunk-based inference pipeline to handle variable-length WAV files.

3.5. Validation Datasets for Detection

We also used public datasets only for supplementary internal validation. Celeb-DF v2 [19] was used to analyze the behavior of the image detection pipeline on real and fake face images, while ASVspoof

2021 DF [20] was used to validate the audio detection pipeline on bonafide and spoofed speech samples. These datasets were not used to train the models or to optimize ensemble weights.

4. Methodology: Deepfake Generation

4.1. Image Generation

We propose a four-stage pipeline for landmark-constrained deepfake face image generation. Given a set of reference videos and facial landmark constraints, our system synthesizes photorealistic 256×256 face images that preserve the target person's identity while conforming to the specified facial geometry. The pipeline consists of: (1) frontal source frame extraction, (2) CLIP-based visual attribute detection, (3) landmark-to-conditioning image translation, and (4) constrained image synthesis via a tri-component diffusion model.

4.1.1. Source Frame Extraction and Attribute Detection

Frontal Source Frame Extraction.

To obtain a high-quality identity reference for each subject, we extract the most frontally-aligned frame from each development video. For each video, up to 150 frames are uniformly sampled to reduce computational overhead. MediaPipe FaceMesh [3] is applied to each sampled frame to obtain 478 normalized facial landmark coordinates. Frontal alignment is quantified by the horizontal displacement of the nose-tip landmark (index 1) from the image center:

$$\text{frontality score} = |x_{\text{nose}} - 0.5| \tag{1}$$

The frame minimizing this score is selected as the identity reference image, saved as `person_{id}_source.png`.

CLIP-Based Visual Attribute Detection. To construct person-specific generation prompts, we perform zero-shot visual attribute classification using CLIP (ViT-B/32) [4] on each extracted source frame. Four attributes are estimated:

- **Glasses / Beard** (binary): Classified via softmax over positive and negative text prompt sets. A positive probability exceeding a threshold of 0.55 is treated as presence.
- **Gender** (male / female): Two-way softmax over `"a photo of a man"` vs. `"a photo of a woman"`.
- **Age group** (young / middle-aged / elderly): Three-way softmax over age-descriptive text prompts.

The resulting attribute dictionary (`source_features.json`) is passed to the generation stage to personalize the positive prompt for each subject.

4.1.2. Landmark-to-Conditioning Image Translation

Each generation constraint in `generation_constraints.json` contains 478 normalized (x, y) facial landmark coordinates produced by MediaPipe FaceMesh with `refine_landmarks=True`. To convert these into a ControlNet-compatible conditioning image, we render the landmarks onto a blank 512×512 canvas using the exact color and stroke specification of the CrucibleAI/ControlNetMediaPipeFace training data [6]. The region-to-color mapping is summarized in Table 2.

Reproducing the precise training-time rendering protocol is critical: any deviation in color or stroke width introduces a domain shift between the conditioning image and the ControlNet's learned feature space, degrading landmark adherence at inference.

Table 2
Landmark region color specification (BGR) for ControlNet conditioning image rendering

Region	Color (BGR)
Face oval	(10, 200, 10)
Mouth / Lips	(10, 180, 10)
Right eye / brow	(10, 200, 180) / (10, 220, 180)
Left eye / brow	(180, 200, 10) / (180, 220, 10)
Right iris (idx 468)	(10, 200, 250)
Left iris (idx 473)	(250, 200, 10)

Table 3
Inference configuration for constrained image synthesis

Parameter	Value
Base model	Realistic Vision V6.0
Scheduler	UniPC multistep
Denoising steps	30
Guidance scale	7.5
ControlNet scale	1.0
IP-Adapter scale	0.6
Generation resolution	512×512
Submission resolution	256×256 (LANCZOS)

4.1.3. Constrained Image Synthesis

The image generation stage integrates three complementary components into a single Stable Diffusion 1.5 [5] pipeline.

ControlNet [6] (CrucibleAI/ControlNetMediaPipeFace) enforces facial geometry by conditioning the denoising process on the rendered landmark image. The conditioning scale is set to 1.0 to maximize landmark fidelity.

IP-Adapter [7] (`ip-adapter_sd15`) injects identity information extracted from the source frame via cross-attention on CLIP image embeddings. The adapter scale is set to 0.6, balancing identity preservation against layout flexibility.

Realistic Vision V6.0 serves as the base UNet checkpoint, fine-tuned for photorealistic human portraiture. Combined with the UniPC multistep scheduler [8] over 30 denoising steps and a guidance scale of 7.5, this backbone produces natural skin texture and lighting.

The generation prompt is dynamically constructed from the detected attributes following the template:

> *"RAW photo, a photo of a [gender] person [wearing glasses], 8k uhd, dslr, soft lighting, high quality, film grain, Fujifilm XT3"*

All images are generated at 512×512 resolution and downsampled to the required 256×256 submission size using LANCZOS resampling. Table 3 summarizes the inference configuration.

Design Rationale: IP-Adapter vs. ControlNet Scale Trade-off.

The two conditioning signals — identity (IP-Adapter) and geometry (ControlNet) — represent inherently competing objectives. A high `ip_scale` encourages the model to replicate fine facial features of the source person, which may override the target landmark layout. Conversely, a high `controlnet_scale` enforces strict geometric adherence at the cost of identity drift.

We empirically set `ip_scale` = 0.6 and `controlnet_scale` = 1.0 as a working balance, prioritizing landmark constraint satisfaction — which is directly evaluated via Euclidean distance to ground-truth landmarks — while retaining sufficient identity signal for recognition-based scoring. A quantitative ablation of this trade-off is presented in Section 6.1.

Table 4
Speaker similarity (SECS) for different reference selection strategies. Higher is better.

Method	Description	SECS
Method A	Single L reference	**0.9498**
Method B	L+M+S ensemble	0.7052 (-0.245)
Method C	Three L ensemble	0.8228 (-0.127)

4.2. Audio Generation

We employ F5-TTS [11], a flow-matching-based text-to-speech model with zero-shot voice cloning capabilities, for audio generation. Our complete pipeline consists of four components: (1) reference audio selection from provided recordings, (2) F5-TTS inference for speech synthesis, (3) a four-stage post-processing pipeline including high-frequency noise removal, silence trimming, LUFS-based loudness normalization, and fade in/out application, and (4) adversarial evasion experiments to improve detector evasion scores.

4.2.1. Model Selection and Reference Strategy

Model Selection.
We evaluated three candidate text-to-speech models for this task:

- **SpeechT5** [9]: A unified-modal speech/text model with encoder-decoder architecture. While computationally efficient, its zero-shot voice cloning capability was limited for our requirements.
- **XTTS-v2** [10]: A GPT-based multilingual TTS with strong zero-shot performance. However, it requires longer reference audio (typically 6–10 seconds) and higher computational resources, making it less suitable for our single-reference constraint.
- **F5-TTS** [11]: Selected for final implementation due to superior zero-shot voice cloning with single reference samples, open-source availability (MIT license for code, CC-BY-NC-4.0 for model), and efficient inference.

F5-TTS (Fairytaler that Fakes Fluent and Faithful Speech) combines flow matching with Diffusion Transformers (DiT). The architecture consists of:

- **ConvNeXt V2 blocks** for text feature extraction from phoneme sequences
- **DiT blocks** for diffusion-based mel-spectrogram transformation
- **Sway Sampling** for optimized inference speed without quality degradation

The model contains approximately 300M parameters and was pre-trained on the Emilia dataset comprising 100,000 hours of multilingual speech across multiple languages and acoustic conditions.

Reference Selection Strategy. A key design decision in zero-shot voice cloning is determining the optimal reference audio selection strategy. We hypothesized that ensemble averaging of multiple reference samples across duration categories (L/M/S) would capture richer speaker characteristics than a single reference. To test this hypothesis, we conducted a controlled experiment with three approaches:

- **Method A (Baseline):** Single L-category (long duration) reference audio per speaker
- **Method B:** Ensemble of three references (one from each L, M, and S category), averaged in speaker embedding space
- **Method C:** Ensemble of three L-category references, averaged in speaker embedding space

Table 4 presents speaker similarity (SECS) measured using Resemblyzer [12] on a validation subset of generated samples.

Contrary to initial expectations, single reference audio significantly outperformed ensemble approaches by 0.13–0.24 points in speaker similarity. Ensemble averaging dilutes speaker-specific characteristics by mixing different prosodic patterns, speaking rates, and acoustic contexts. Based on these results, we selected Method A (single L-category reference) for final submission.

4.2.2. Inference and Post-processing

F5-TTS Inference.

For each target text in the submission set, we perform inference using F5-TTS's standard API:

```
wav, sr, _ = tts.infer(
    ref_file = ref_path,      # Reference audio (L-category)
    ref_text = ref_text,      # Reference transcription
    gen_text = target_text,   # Target text to synthesize
    remove_silence = False    # Handled in post-processing
)
```

where `ref_path` points to a single L-category reference recording for the target speaker, `ref_text` is its manual transcription, and `target_text` is the sentence to be synthesized.

Post-processing Pipeline. Raw F5-TTS output undergoes a four-stage post-processing pipeline:

Stage 1: High-Frequency Noise Removal. We apply a 4th-order Butterworth low-pass filter with 7500 Hz cutoff frequency using `scipy.signal.butter` and `filtfilt` for zero-phase filtering.

Stage 2: Silence Trimming. Using `librosa.effects.trim` with `top_db=25`, we remove leading and trailing silence, ensuring consistent utterance boundaries.

Stage 3: LUFS Loudness Normalization (Key Innovation). Initial generations exhibited an 18.1 dB LUFS discrepancy (reference: -41.1 dBFS, generated: -23.0 dBFS). We apply EBU R128 [13] loudness normalization using `pyloudnorm`:

```
import pyloudnorm as pyln

meter = pyln.Meter(sample_rate)
lufs_ref = meter.integrated_loudness(y_reference)
lufs_gen = meter.integrated_loudness(y_generated)
y_normalized = pyln.normalize.loudness(
    y_generated, lufs_gen, lufs_ref
)
```

This achieves perfect loudness matching (0.0 dB difference), significantly improving perceptual naturalness.

Stage 4: Fade In/Out. We apply 10 ms linear fades at utterance boundaries to prevent audible clicks or discontinuities.

4.2.3. Adversarial Evasion Experiments

To improve evasion scores, we conducted comprehensive experiments with adversarial perturbation strategies. We trained a CNN-based surrogate detector on MFCC features achieving 99.8% accuracy on F5-TTS samples.

Strategy A: Waveform-Level PGD Attack. We applied Projected Gradient Descent attacks directly to audio waveforms with epsilon ranging from 0.001 to 0.1 over 40–100 iteration steps. *Result:* Evasion rate remained at 0%. The fundamental issue is domain mismatch: PGD crafts perturbations in waveform space, but detectors operate on MFCC features extracted via cascaded transformations that attenuate adversarial noise.

Strategy B: Feature-Space MFCC Manipulation. We directly manipulated MFCC features by shifting generated MFCCs toward real audio patterns, then reconstructed audio using Griffin-Lim algorithm. *Result:* Evasion rate remained 0%, with speaker similarity dropping to 0.0 (complete voice collapse). MFCC-to-waveform reconstruction is inherently lossy and phase-ambiguous.

Conclusion. Unlike computer vision, audio perturbations face cascaded transformations that fundamentally alter signal representation. For our final submission, we prioritized high-quality generation with proper LUFS normalization over ineffective adversarial perturbations.

Table 5
Multi-view image preprocessing used for image detection

View	Purpose
Original	Preserves global background, illumination, color, and texture.
Flip	Test-time augmentation to reduce directional bias.
Face Crop	Focuses on internal facial artifacts such as eyes, mouth, and skin.
Wide Face Crop	Includes face boundary, hair, neck, and nearby background.

Table 6
Pretrained models used for image deepfake detection

Model	Role	Weight
`deepfake-detector-model-v1`	Primary deepfake detector	0.45
`Deepfake-Detect-Siglip2`	SigLIP2-based auxiliary detector	0.30
`ai-image-detection`	AI-generated image detector	0.25

5. Methodology: Deepfake Detection

5.1. Image Detection

Given 24,404 PNG images, the objective of the image detection system is to assign a binary real-or-deepfake label to each sample. Since no ground-truth labels were available for the official detection set during development, our system was designed as a pretrained-model-based scoring pipeline followed by rule-based score correction.

For each input image, we first generate four complementary views: Original, Flip, Face Crop, and Wide Face Crop. These views are passed to three pretrained image detectors, producing a total of 12 fake scores per image. For each model, the four view-level scores are combined into a hybrid score. The three model-level hybrid scores are then aggregated by a weighted ensemble to obtain an ensemble fake score. In addition, we extract noise- and frequency-domain auxiliary features to identify cases where the model ensemble assigns a low fake score despite exhibiting synthetic artifacts. A conditional score boosting step is finally applied before thresholding the final fake score into the submission label.

5.1.1. Multi-view Input Preprocessing

Instead of applying each detector to the original image only, we expand each image into four views. This design was intended to capture both global scene-level artifacts and local facial inconsistencies. Table 5 summarizes the four input views.

The face crop is obtained using OpenCV-based face detection [21]. When multiple faces are detected, the largest detected face is selected under the assumption that the target identity occupies the main region of the image. The wide face crop is generated by expanding the detected bounding box to include surrounding context such as hair, neck, and nearby background regions.

5.1.2. Model Ensemble

We used three publicly available pretrained image classification models. The ensemble was designed to combine two deepfake-specific detectors with one general AI-generated image detector. Table 6 lists the models and their ensemble weights. The ensemble weights were not learned or optimized from labeled validation performance, since no ground-truth labels were available for the official detection set during system development. Instead, we manually assigned role-based heuristic weights according to the intended contribution of each detector. Therefore, the weights should be interpreted as design choices rather than performance-derived coefficients.

Model A, `prithivMLmods/deepfake-detector-model-v1` [14], was used as the primary detector because it directly targets real-versus-deepfake image classification; accordingly, it received the largest weight of 0.45. Model B, `prithivMLmods/Deepfake-Detect-Siglip2` [15], was used as a SigLIP2-based auxiliary detector to complement Model A with a structurally different vision-language representation [16], and was assigned a weight of 0.30. Model C, `capcheck/ai-image-detection` [17], was included to capture more general AI-generated image artifacts beyond facial manipulation, such as background texture, color, and global synthetic patterns. Since this model served as a supplementary signal rather than a task-specific face deepfake detector, it received the lowest weight of 0.25.

Unlike a standard single-detector baseline that applies one model only to the original image and directly thresholds its output, our image detection pipeline applies each detector to four complementary views of the same image. This produces 12 view-level fake scores per image. These scores are first summarized into model-level hybrid scores and are then combined using the role-based ensemble weights. This design allows the system to incorporate both global image-level artifacts and localized facial inconsistencies.

5.1.3. Score Computation and Boosting

Hybrid Score.

Each model receives the four preprocessed views and produces four fake scores: s_{orig}, s_{flip}, s_{face}, and s_{wide}. We combine these view-level scores in two ways. First, we compute a maximum score to preserve strong fake evidence that may appear only in a specific crop:

$$S_{\text{max}} = \max(s_{\text{orig}}, s_{\text{flip}}, s_{\text{face}}, s_{\text{wide}}). \tag{2}$$

Second, we compute a weighted average score to incorporate all views more smoothly:

$$S_{\text{weighted}} = 0.30 \cdot s_{\text{orig}} + 0.15 \cdot s_{\text{flip}} + 0.30 \cdot s_{\text{face}} + 0.25 \cdot s_{\text{wide}}. \tag{3}$$

The view weights in S_{weighted} were also manually assigned according to the expected contribution of each view. The original image received a weight of 0.30 because it preserves global information such as background, illumination, color, and texture. The face crop also received a weight of 0.30 because it focuses on the most discriminative facial regions, including the eyes, mouth, and skin texture. The wide face crop was assigned a slightly smaller weight of 0.25 because it captures boundary regions such as hair, neck, and nearby background, which may reveal inconsistencies between the face and its surroundings. The flipped view received the lowest weight of 0.15 because it mainly served as a test-time augmentation view for reducing directional bias rather than providing an independent semantic view.

The final model-level hybrid score is computed as:

$$S_{\text{hybrid}} = 0.65 \cdot S_{\text{max}} + 0.35 \cdot S_{\text{weighted}}. \tag{4}$$

The max component reduces missed detections for cases where only one view contains a strong artifact, while the weighted component suppresses unstable predictions caused by a single noisy view. The three model-level hybrid scores are then combined as:

$$S_{\text{fake}} = 0.45 \cdot S^{A}_{\text{hybrid}} + 0.30 \cdot S^{B}_{\text{hybrid}} + 0.25 \cdot S^{C}_{\text{hybrid}}. \tag{5}$$

These model-level weights follow the same role-based heuristic described above. Model A was emphasized as the primary deepfake detector, while Models B and C were used as complementary detectors with smaller weights. No labeled development set was used to estimate these coefficients.

Noise/Frequency Auxiliary Features. Preliminary inspection of ensemble outputs showed that some images with low fake scores still exhibited visually suspicious low-level artifacts. To compensate for such cases, we extracted auxiliary features from the spatial noise domain and frequency domain.

Table 7
Conditional boosting criteria for image detection

Condition	Boost	Type
Fake-score bottom 10% + auxiliary top 10%	+0.20	Strong
Fake-score bottom 20% + auxiliary top 20%	+0.10	Weak
Otherwise	+0.00	None

These features were not used as a standalone classifier; instead, they provided an auxiliary signal for conditional score correction.

The noise features include blur, edge density, residual noise statistics, and blockiness. Blur is measured using the variance of the Laplacian, where low sharpness may indicate unnatural smoothing. Residual noise is computed by subtracting a Gaussian-blurred image from the grayscale input, which highlights fine local noise patterns. Blockiness estimates JPEG-like or synthesis-related block boundary artifacts. Edge density is used to describe the amount of local structural detail.

The frequency features are extracted from the two-dimensional Fourier transform of the grayscale image. We compute low-, mid-, and high-frequency energy ratios, as well as spectral entropy. The high-frequency and entropy terms are used to measure fine texture and the complexity of the frequency distribution.

Unlike the pretrained visual detectors, which mainly capture semantic and face-level cues, these auxiliary features were intended to capture low-level artifacts such as compression traces, residual noise patterns, and abnormal frequency distributions.

All feature values are normalized with robust min–max scaling. The normalized features are aggregated into a noise score S_{noise} and a frequency score S_{freq}. The auxiliary score is computed as:

$$S_{\text{aux}} = 0.5 \cdot S_{\text{noise}} + 0.5 \cdot S_{\text{freq}}. \tag{6}$$

Conditional Score Boosting. A direct weighted average between S_{fake} and S_{aux} was not used in the final submission. In preliminary analysis, the auxiliary score distribution was generally lower than the ensemble fake score distribution, and simple averaging often reduced the final fake score instead of recovering suspicious low-score samples. Therefore, we applied the auxiliary score only under specific conditions.

The conditional boosting rule is defined in Table 7. If an image belongs to the bottom 10% of ensemble fake scores and the top 10% of auxiliary scores, a strong boost of 0.20 is added. If it belongs to the bottom 20% of ensemble fake scores and the top 20% of auxiliary scores, but not the strong-boost group, a weaker boost of 0.10 is added. The final fake score is clipped to the range $[0, 1]$:

$$S_{\text{final}} = \text{clip}(S_{\text{fake}} + b, 0, 1), \tag{7}$$

where b is the conditional boost value.

Two final threshold settings were retained. The threshold 0.75 candidate predicted 17,380 images as deepfake, corresponding to 71.22% of the official image test set. The threshold 0.70 candidate predicted 19,007 images as deepfake, corresponding to 77.88%.

To analyze the behavior of the noise/frequency-based correction before interpreting the official results, we additionally evaluated the two settings on a balanced public validation subset derived from Celeb-DF v2 [19], consisting of 500 real frames and 500 fake frames. Table 8 reports the validation results. This validation was used only to characterize the pipeline behavior and not as an official task score.

The validation results show that the noise/frequency-based correction helped preserve high fake-class recall, but the real-class accuracy remained low. This indicates that some real frames also contained noise/frequency patterns similar to the auxiliary cues used for boosting, which limited the separation between real and fake samples.

Table 8
Internal validation results on a balanced Celeb-DF v2 subset

Candidate	Acc.	Prec.	Recall	F1	Real Acc.
th075	0.533	0.519	0.888	0.655	0.178
th070	0.517	0.510	0.910	0.653	0.124

Table 9
Statistics of the target audio files used in the detection pipeline

Property	Value
Total files	11,520
Channels	Mono for all files
Sample rates	16 kHz (10,080), 22.05 kHz (480), 24 kHz (960)
Minimum duration	0.320 s
Mean duration	5.305 s
Maximum duration	28.073 s

5.2. Audio Detection

For audio deepfake detection, we used a single-model pipeline based on AASIST, an audio anti-spoofing model that integrates spectral and temporal graph attention mechanisms [18]. The target detection set contained 11,520 WAV files, and each file was classified as either real or deepfake. The pipeline consisted of four main stages: audio file inspection, waveform preprocessing, chunk-based AASIST inference, and file-level fake probability aggregation. Compared with a simple file-level baseline that processes a single fixed segment, our pipeline performs chunk-based inference so that longer utterances can be represented by multiple temporal regions before file-level aggregation.

The overall audio detection flow was as follows. First, each WAV file was loaded and checked for basic signal properties such as sample rate, duration, channel count, clipping ratio, and silence ratio. The waveform was then converted to a mono signal and resampled to 16 kHz to match the AASIST input setting. Since the audio files had variable durations, short audio files were padded to a fixed input length, while long audio files were split into multiple chunks. AASIST was applied to each chunk, and the chunk-level scores were aggregated into a single file-level fake probability. Finally, a threshold of 0.5 was used to generate the binary prediction.

5.2.1. Preprocessing and Chunk-based Inference

Before model inference, we performed a file-level inspection step to verify that all audio files could be read correctly and to summarize the basic properties of the dataset. Table 9 summarizes the statistics obtained from the 11,520 target audio files. All files were mono, but the sample rates differed across files. Therefore, all waveforms were resampled to 16 kHz before being passed to the model.

The AASIST input length was set to 64,600 samples, which corresponds to approximately 4.04 seconds at 16 kHz. Audio files shorter than this length were repeat-padded to form one fixed-size chunk. For longer files, we selected up to five chunks from uniformly spaced temporal positions across the waveform rather than using only the first segment. This chunk selection strategy was used to cover multiple temporal regions of long utterances while keeping the inference cost bounded, and it allowed the detector to capture localized spoofing cues that may appear only in part of an utterance.

Chunk-based AASIST Inference. We used AASIST as the primary audio deepfake detector. AASIST receives raw waveform chunks and outputs logits for spoof/fake and bonafide/real classes. For each preprocessed audio file, the waveform was divided into one or more chunks of 64,600 samples. Each chunk was forwarded through the AASIST model, and the fake probability for each chunk was extracted from the model output.

For most files in the official target set, one chunk was sufficient because the duration was shorter

Table 10
ASVspoof 2021 DF validation results for audio detection

Acc.	Prec.	Recall	F1	Real Acc.	Fake Acc.
0.900	0.848	0.975	0.907	0.825	0.975

than or close to the model input length. However, longer files were represented by multiple chunks, enabling the model to capture localized fake cues across different temporal regions. In the final target set, 9,748 files were represented by one chunk, while the remaining files required two or more chunks.

5.2.2. Fake Probability Aggregation

For files with multiple chunks, we aggregated chunk-level fake probabilities into a single file-level score. Let $\bar{s}$ denote the mean fake probability over all chunks of a file, and let $s_{\max}$ denote the maximum chunk-level fake probability. The final file-level fake probability was computed as

$$p_{\text{fake}} = 0.80 \cdot \bar{s} + 0.20 \cdot s_{\max}. \tag{8}$$

The mean score was used as the main decision signal because it provides a stable estimate of the overall file-level prediction across all chunks. The maximum score was included with a smaller weight to capture localized spoofing cues that may appear strongly in only a short temporal segment. This aggregation was therefore designed to balance overall file-level stability with sensitivity to short fake regions. We then applied a fixed threshold of 0.5: files with $p_{\text{fake}} \geq 0.5$ were predicted as Deepfake, while the remaining files were predicted as Real.

On the official target set, this pipeline predicted 7,514 files as Deepfake and 4,006 files as Real, corresponding to 65.23% and 34.77% of all audio files, respectively. To further validate the behavior of the pipeline, we constructed a balanced validation subset from ASVspoof 2021 DF, which is a speech deepfake benchmark containing bonafide and spoofed speech samples [20]. The subset contained 200 real and 200 fake samples. Table 10 reports the validation results.

The validation result shows that the AASIST-based pipeline detected fake speech with high recall while maintaining a reasonable real-class accuracy. This indicates that the audio detector was less biased toward predicting all samples as fake than the image detector, although the validation subset was smaller and should be interpreted as a supplementary analysis rather than the official evaluation.

6. Experiments and Results

6.1. Image Generation Results

Table 11 presents the internal evaluation results measured on the 1,002 valid generated images, and Table 12 presents the official ImageCLEF 2026 evaluation results.

Landmark Distance Analysis. The internal evaluation yielded a mean landmark distance of 0.00758. Since MediaPipe FaceMesh coordinates are normalized to $[0, 1]$, this corresponds to sub-pixel displacement at the 256×256 submission resolution, confirming that the ControlNet conditioning pipeline effectively enforces the provided geometric constraints. The official evaluation confirms this result, with a landmark distance of 0.007600 translating to a scaled Landmark Distance Score of 0.7645.

Identity Score Analysis. The internal identity score (mean cosine similarity 0.194, measured with InsightFace buffalo_s) indicates limited identity preservation. The official evaluation assigns an Identity Distance Score of 0.0000, reflecting a high identity distance (0.1749) between generated and reference faces. This outcome is consistent with the trade-off described in Section 4.1: the ControlNet conditioning scale of 1.0 strongly prioritizes geometric fidelity, which constrains the IP-Adapter scale of 0.6 from recovering subject-specific features such as skin tone and facial structure.

Table 11
Internal evaluation results for image generation (1,002 valid samples)

Metric	Tool	Score
Landmark Distance (mean)	MediaPipe FaceMesh	0.00758
Landmark Distance (median)	MediaPipe FaceMesh	0.00705
Landmark Distance (max)	MediaPipe FaceMesh	0.03665
Identity Score (mean)	InsightFace buffalo_s	0.194
Identity Score (median)	InsightFace buffalo_s	0.193
Identity Score (min)	InsightFace buffalo_s	−0.101
Evasion Rate — ViT-based [22]	deepfake_vs_real_image_detection	98.1%
Evasion Rate — CNN-based [23]	Deep-Fake-Detector-Model	6.9%
Mean Evasion Rate	—	52.5%

Table 12
Official evaluation results for image generation

Metric	Score
Raw Metrics	
Landmark Distance	0.007600
Identity Distance	0.174900
Scaled Scores	
Landmark Distance Score	0.7645
Identity Distance Score	0.0000
Image Quality Score	0.3823
Final Scores	
Image Quality Score	0.3823
Deepfake Evasion Score (Participants)	0.5657
Deepfake Evasion Score (Organizers)	0.8478
Final Score	**0.2486**

Evasion Rate Analysis. The internal evasion evaluation revealed a large disparity between detector architectures. The ViT-based detector [22] classified 98.1% of generated images as real, while the CNN-based detector [23] detected 93.1% as deepfake, yielding a mean evasion rate of 52.5%. The official Deepfake Evasion Score (Organizers) of 0.8478 indicates that organizer detectors also classified the majority of generated images as real, suggesting that the diffusion pipeline produces globally coherent images that confound semantics-based detectors, while local texture artifacts remain detectable by CNN-based models.

Overall Discussion. The final official score of 0.2486 is primarily constrained by the Identity Distance Score of 0.0000, which carries substantial weight in the composite Image Quality Score (0.3823). Future work should focus on increasing the IP-Adapter scale or adopting a higher-capacity identity encoder to improve subject fidelity while maintaining landmark adherence.

6.2. Audio Generation Results

Table 13 presents the official ImageCLEF 2026 evaluation results for our audio generation system.

Audio Quality Analysis. Our system achieved strong audio quality with a composite Quality Score of 0.7150. NISQA MOS of 3.14 indicates good perceptual quality, and transcription accuracy was excellent (WER: 7.93%, CER: 3.93%).

A notable finding is substantial variation in speaker similarity across evaluation models: WavLM (0.8622) vs. ECAPA-TDNN (0.5959), with our internal Resemblyzer validation (0.9498) showing even higher scores. This 0.27-point gap highlights that speaker similarity depends heavily on the embedding

Table 13
Official evaluation results for audio generation

Raw Metrics	
NISQA MOS	3.1405
Speaker Similarity (WavLM)	0.8622
Speaker Similarity (ECAPA)	0.5959
Speaker Similarity (Mean)	0.7291
Word Error Rate (WER)	0.0793
Character Error Rate (CER)	0.0393
Scaled Scores	
NISQA MOS Score	0.5351
ECAPA Score	0.5959
WavLM Score	0.8622
WER Score	0.9207
CER Score	0.9607
Composite Scores	
Audio Quality Score	0.7150
Evasion Score (Participants)	0.5521
Evasion Score (Organizers)	0.2868
Final Score	**0.3378**

model architecture.

Evasion Score Analysis. Limited evasion capability severely constrained our final score despite strong audio quality. Against participant detectors, we achieved 55.21% evasion, but only 28.68% against organizer detectors. The final score calculation demonstrates the multiplicative bottleneck:

$$\begin{aligned} \text{Final} &= 0.7150 \times (0.5521 \times 0.7 + 0.2868 \times 0.3) \\ &= 0.7150 \times 0.4725 = \mathbf{0.3378} \end{aligned} \tag{9}$$

Key Findings.

- **Single reference superiority:** Single L-category reference outperformed ensemble approaches by 0.13–0.24 points.
- **LUFS normalization impact:** Eliminated 18.1 dB perceptual discrepancy, improving naturalness.
- **Adversarial audio limitations:** Both waveform and feature-space perturbations failed (0% evasion).
- **Metric variability:** Speaker similarity varied by 0.27 points across models, emphasizing evaluation protocol importance.

6.3. Image Detection Results

Table 14 presents the official ImageCLEF 2026 evaluation results for our image detection system.

The proposed image detector achieved relatively high accuracy on deepfake subsets, with 0.8351 on organizer baseline deepfakes and 0.7136 on the weighted participant deepfake subset. This suggests that the multi-view ensemble and noise/frequency-based correction were effective for identifying many fake images. However, the system performed poorly on real-image subsets, achieving only 0.1826 on organizer real data and 0.2874 on the hidden organizer real ground-truth subset. Since the real GT subset has a high weight in the final score, this false-positive behavior substantially reduced the final image score to 0.5022.

Table 14
Official evaluation results for image deepfake detection

Metric	Score
Baseline Deepfakes (Images)	0.8351
Organizers Real Data (Images)	0.1826
Organizers Real GT Data (Images)	0.2874
Participant Deepfake Data (Images)	0.7098
Participant Deepfake Data Weighted (Images)	0.7136
Final Score (Images)	**0.5022**

Table 15
Official evaluation results for audio deepfake detection

Metric	Score
Baseline Deepfakes (Audio)	0.5411
Organizers Real Data (Audio)	0.8750
Organizers Real GT Data (Audio)	0.8438
Participant Deepfake Data (Audio)	0.6950
Participant Deepfake Data Weighted (Audio)	0.6932
Final Score (Audio)	**0.7564**

These results indicate that the auxiliary artifacts used for correction were not sufficiently discriminative for real-image calibration. Although noise and frequency cues helped recover many fake samples, some real images also exhibited similar blur, residual noise, blockiness, or frequency patterns, leading to many real images being classified as Deepfake. In future work, we would introduce a real-image validation set for calibration and refine the noise/frequency auxiliary features to reduce false positives while maintaining deepfake recall.

6.4. Audio Detection Results

Table 15 presents the official ImageCLEF 2026 evaluation results for the audio deepfake detection task.

The audio detector achieved a final official score of 0.7564. In contrast to the image detector, the audio system performed strongly on real-audio subsets, obtaining 0.8750 on organizer real data and 0.8438 on the hidden organizer real ground-truth subset. This suggests that the AASIST-based pipeline was relatively effective at avoiding false positives on authentic audio samples.

For deepfake audio, the system achieved 0.6950 on participant-generated fake data and 0.6932 on the realism-weighted participant deepfake subset, showing moderate robustness to synthetic speech produced by other teams. However, the accuracy on organizer baseline deepfakes was lower at 0.5411, indicating that the single pretrained AASIST model was less effective for some organizer-generated audio attacks. Overall, the audio detector showed a more balanced behavior than the image detector, especially with respect to real-data accuracy, but the results also suggest that a single-model audio pipeline may be insufficient for covering all deepfake generation methods.

6.5. Discussion

Across all four sub-tasks, our systems exhibited consistent strengths in modality-specific technical execution but revealed common limitations in cross-domain robustness and identity fidelity.

For generation, both pipelines achieved competitive quality in their primary objectives — landmark adherence for images and speech intelligibility for audio — but were constrained by secondary metrics. The image pipeline achieved a landmark distance score of 0.7645 but scored 0.0000 on identity preservation, indicating that the ControlNet geometry constraint overwhelmed the IP-Adapter identity

signal. The audio pipeline achieved a WER score of 0.9207 and a WavLM speaker similarity of 0.862, but the organizer evasion score (0.2868) limited the final score to 0.3378 due to the multiplicative scoring formula.

For detection, the audio pipeline (final score 0.7564) significantly outperformed the image pipeline (0.5022). The key differentiator was real-data precision: the AASIST-based audio detector maintained high accuracy on genuine audio (0.8750 on organizer real data), whereas the image ensemble aggressively classified many real images as deepfake (0.1826), resulting in a high false-positive rate. Both pipelines shared the same root cause for this failure: the absence of a labeled validation set during development made it impossible to calibrate thresholds or auxiliary scoring rules against real-data distributions.

A cross-modal pattern is that inference-only systems without task-specific fine-tuning or labeled calibration data face a precision–recall trade-off that is difficult to optimize blindly. In both detection modalities, recall for the target class (fake) was high but precision on the complementary class (real) was not controlled. Future work across all four sub-tasks should prioritize labeled validation data for threshold calibration and fine-tuning to close the gap between internal and official evaluation performance.

7. Conclusion

In this paper, we presented our system for the ImageCLEF 2026 Deepfake Task [1], covering both detection and generation across image and audio modalities. For detection, the audio pipeline achieved a final score of 0.7564, demonstrating strong real-audio precision with the AASIST model, while the image ensemble reached 0.5022, limited by false positives on real images due to the absence of labeled calibration data. For generation, the landmark-constrained diffusion pipeline produced geometrically faithful face images (landmark distance score 0.7645) but showed limited identity preservation (identity distance score 0.0000), reflecting the inherent tension between ControlNet geometry enforcement and IP-Adapter identity injection. The F5-TTS-based audio generation achieved strong speech intelligibility (WER score 0.9207) and speaker similarity (WavLM 0.862), but the final score of 0.3378 was constrained by the organizer evasion score (0.2868).

Future work should focus on labeled validation sets for threshold calibration in detection, higher-capacity identity encoders and dynamic conditioning schedules for image generation, and audio-domain adversarial evasion strategies that operate beyond waveform-level perturbations.

Declaration on Generative AI

During the preparation of this work, the author(s) used Claude (Anthropic) in order to: assist with paper writing, text editing, grammar and spelling check, and LaTeX formatting. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] D. Stanciu, A. Radu, R. Bolborici, M. Negru, A. Ene, V. Vasilescu, A.-A. Nicolae, B. Ionescu, L. Ştefan, M. Constantin, M. Dogariu, A. Andrei, Overview of ImageCLEF 2026 deepfake task: Multimodal detection and generation of deepfakes, in: CLEF 2026 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, Jena, Germany, 2026.

[2] B. Ionescu, H. Müller, D. Stanciu, A. Radu, R. Bolborici, M. Negru, A. Ene, V. Vasilescu, A.-A. Nicolae, L. Ştefan, M. Constantin, M. Dogariu, A. Andrei, H. Damm, T. M. G. Pakull, A. Ben Abacha, A. García Seco de Herrera, C. M. Friedrich, R. Brüngel, L. Reinartz, H. Schäfer, C. S. Schmidt, B. Bracke, P. Nath, B. Eryılmaz, M. Hjuler, D. Fabre, C. Lemaire, B. Lecouteux, D. Schwab, D. Dimitrov, M. S. Hee, M. Ahsan, S. Ahmad, D. Zlatkova, G. Pachov, Z. Xie, P. Nakov, I. Koychev, J. E. Heras Rivera, D. K. Low, W. Yim, J. Ruzevick, D. Child, M. Kurt, Z. Sun, F. Xia, M. Yetisgen, A. Radzhabov,

Y. Prokopchuk, V. Kovalev, D. Karpenka, S. A. Hicks, S. Gautam, M. A. Riegler, V. Thambawita, P. Halvorsen, M. El Sakka, J. Mothe, A. Băicoianu, C. Florea, M. Ivanovici, Overview of ImageCLEF 2026: Multimodal challenges in medicine, science, agritech, and security, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), Springer Lecture Notes in Computer Science LNCS, Jena, Germany, 2026.

[3] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, M. Grundmann, MediaPipe: A framework for building perception pipelines, in: Workshop on ML Systems, NeurIPS, 2019. URL: https://arxiv.org/abs/1906.08172.

[4] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning (ICML), 2021.

[5] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution image synthesis with latent diffusion models, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

[6] L. Zhang, A. Rao, M. Agrawala, Adding conditional control to text-to-image diffusion models, in: IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

[7] H. Ye, J. Zhang, S. Liu, X. Han, W. Yang, IP-Adapter: Text compatible image prompt adapter for text-to-image diffusion models, arXiv preprint arXiv:2308.06721 (2023).

[8] W. Zhao, L. Bai, Y. Rao, J. Zhou, J. Lu, UniPC: A unified predictor-corrector framework for fast sampling of diffusion models, arXiv preprint arXiv:2302.04867 (2023).

[9] J. Ao, R. Wang, L. Zhou, C. Wang, S. Ren, Y. Wu, S. Liu, T. Ko, Q. Li, Y. Zhang, Z. Wei, Y. Qian, J. Li, F. Wei, SpeechT5: Unified-modal encoder-decoder pre-training for spoken language processing, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2022, pp. 5723–5738. URL: https://aclanthology.org/2022.acl-long.393. doi:10.18653/v1/2022.acl-long.393.

[10] Coqui AI, XTTS-v2, https://huggingface.co/coqui/XTTS-v2, 2023. Hugging Face model repository.

[11] Y. Chen, Z. Niu, Z. Ma, K. Deng, C. Wang, J. Zhao, K. Yu, X. Chen, F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching, arXiv preprint arXiv:2410.06885 (2024). URL: https://arxiv.org/abs/2410.06885. arXiv:2410.06885.

[12] Resemble AI, Resemblyzer: Voice analysis in python, https://github.com/resemble-ai/Resemblyzer, 2019. GitHub repository.

[13] European Broadcasting Union, EBU R 128: Loudness Normalisation and Permitted Maximum Level of Audio Signals, Technical Report R 128, European Broadcasting Union, 2023. URL: https://tech.ebu.ch/publications/r128.

[14] prithivMLmods, deepfake-detector-model-v1, https://huggingface.co/prithivMLmods/deepfake-detector-model-v1, 2025. Hugging Face model repository.

[15] prithivMLmods, Deepfake-Detect-Siglip2, https://huggingface.co/prithivMLmods/Deepfake-Detect-Siglip2, 2025. Hugging Face model repository.

[16] X. Zhai, B. Mustafa, A. Kolesnikov, L. Beyer, Sigmoid loss for language image pre-training, arXiv preprint arXiv:2303.15343 (2023). URL: https://arxiv.org/abs/2303.15343. arXiv:2303.15343.

[17] capcheck, ai-image-detection, https://huggingface.co/capcheck/ai-image-detection, 2025. Hugging Face model repository.

[18] J. weon Jung, H.-S. Heo, H. Tak, H. jin Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, N. Evans, AASIST: Audio anti-spoofing using integrated spectro-temporal graph attention networks, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 6367–6371.

[19] Y. Li, X. Yang, P. Sun, H. Qi, S. Lyu, Celeb-DF: A large-scale challenging dataset for deepfake forensics, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 3207–3216.

[20] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, H. Delgado, ASVspoof 2021: Accelerating progress in spoofed and

deepfake speech detection, in: Proceedings of the ASVspoof Challenge Workshop, 2021, pp. 47–54. doi:10.21437/ASVSPOOF.2021-8.
[21] G. Bradski, The OpenCV library, Dr. Dobb's Journal of Software Tools (2000).
[22] dima806, deepfake_vs_real_image_detection, https://huggingface.co/dima806/deepfake_vs_real_image_detection, 2023. Hugging Face model repository, ViT-based deepfake detector.
[23] prithivMLmods, Deep-Fake-Detector-Model, https://huggingface.co/prithivMLmods/Deep-Fake-Detector-Model, 2024. Hugging Face model repository, CNN-based deepfake detector.