DiVAS: Video and Audio Synchronization with Dynamic Frame Rates

Clara Fernandez-Labrador[1] Mertcan Akçay[1,2] Eitan Abecassis[3] Joan Massich[1] Christopher Schroers[1]
[1]DisneyResearch|Studios [2]ETH Zürich [3]Disney Entertainment and ESPN Technology
{clara.fernandez, mertcan.akcay, joan.massich, christopher.schroers}@disneyresearch.com
eitan.abecassis@disneystreaming.com

Abstract

Synchronization issues between audio and video are one of the most disturbing quality defects in film production and live broadcasting. Even a discrepancy as short as 45 milliseconds can degrade the viewer's experience enough to warrant manual quality checks over entire movies. In this paper, we study the automatic discovery of such issues. Specifically, we focus on the alignment of lip movements with spoken words, targeting realistic production scenarios which can include background noise and music, intricate head poses, excessive makeup, or scenes with multiple individuals where the speaker is unknown. Our model's robustness also extends to various media specifications, including different video frame rates and audio sample rates. To address these challenges, we present a model fully based on Transformers that encodes face crops or full video frames and raw audio using timestamp information, identifies the speaker and provides highly accurate synchronization predictions much faster than previous methods.

1. Introduction

Audio-Video (AV) synchronization is a basic expectation to anyone that is consuming video, whether through streaming, social media, cable television, theaters or any other form of media. From the lens of the camera to the eye of the consumer, there are many instances where errors can be introduced, such as during content mastering, third party modifications, content encoding, or client playback. Studies show that the viewer experience can be negatively affected by a mere 45 millisecond (ms) discrepancy [7]; this is equivalent to a delay of a single frame in a 90 minute film at 25 frames per second (fps). While commercial solutions [1] exist, their scale and capabilities are insufficient for production. Thereby, detecting and identifying the origin of synchronization issues remains a significant burden for quality control teams, as it is largely a manual process. Thus, there is a pressing need for an automated detection system that can accurately identify and resolve AV synchronization issues before they reach the viewer.

A first requirement for such an automated system is to identify relevant cues to reliably detect synchronization issues - scenes with clear relationships between visual and auditory stimuli [19]. However, many sound sources do not map to a clear visual signal, such as ambient sounds (traffic, rain, crowds), stationary sounds (car engine), and acousmatic sounds (narrator, background music). Similarly, some visual cues may not have a clear audio cue to detect synchronization issues (landscapes, still elements). The definitive cue for synchronization, and the most studied one in literature, is known as lip sync, where speakers' lip movements match the sound of spoken words. However, identifying such dialogue scenes is intricate and time-consuming, involving a combination of scene, speech, and face detectors, face tracking, and active speaker detection in most cases.

Another fundamental requirement for a synchronization model is to consume the content in its original form, since a simple operation like a frame rate conversion can introduce synchronization artifacts [4]. In video production, frame rates are used to invoke a certain feel to the content. For example, a feature film might have a lower frame rate to achieve a cinematic look, while a sports broadcast might have a higher frame rate to capture fast-moving action. Standard frame rates can range from 24 fps to 120 fps, thus challenging models to be robust enough to make reliable predictions on the original content. Previous lip sync methods [6, 8–10, 14, 15] have typically thrived on simpler datasets derived from BBC interviews [2] or TED talks [3], often containing a single speaker providing clean, continuous speech while facing the camera. These [2, 3] are the most common datasets used to benchmark AV synchronization models. Notably, both offer videos converted to 25 fps and cropped to the face region. Existing lip sync approaches [6, 8–10, 14, 15] leverage Convolutional Neural Networks (CNNs) achieving outstanding results on these simpler datasets, but their performance does not extend to more complex videos, as they exclusively operate on single-face clips and require an intermediate encoding of the in-

put, typically involving the conversion of videos to 25 fps. Moreover, we observe that previous methods are not tested against artificial offsets, nor consider predictions beyond individual clips.

In light of these challenges, we broaden the scope to include dialogue scene identification (not constrained to single-face clips) alongside synchronization assessment, yielding a more holistic solution for multimedia content creators and analysts. We make several contributions in this work: (i) We introduce our Dynamic Video and Audio Synchronization model, referred to as DiVAS, a transformer-based model that operates directly on raw audio and video. Different from previous lip sync methods, we do not rely on CNN feature extractors, which require fixed-size inputs, but directly feed a Transformer network with a variable number of video frames and audio samples depending on the original media specifications. To effectively handle the dynamic input, we propose a new positional encoding that leverages timestamp information, making our model robust to different frame and sample rates. (ii) DiVAS offers the flexibility to operate either on face crops, similar to previous approaches, or full frames, removing the significant cost of detecting and tracking faces and identifying the speaker. (iii) Our model outperforms state-of-the-art methods in several benchmark datasets, all while being significantly smaller and faster. (iv) We also evaluate DiVAS on a more challenging dataset which includes live action and animation movies, TV shows, musicals, documentaries and series' episodes setting a new baseline. (v) Finally, we consider synchronization not only at the clip but also at the title level. This allows us to provide more reliable predictions and categorize different types of synchronization issues, including constant offsets and drifts.

2. Related Work

The exploration of automated lip sync research dates back to 1991 [18]. The earliest works investigated generating talking avatars in-sync with a given speech. Some studies [17, 18] rely on *phonemes* to create a mapping from audio to lip movements, while others [27] classify Mel-frequency cepstral coefficients (MFCC) from audio into *visemes* using neural networks to create talking avatars. However, none of these works focus on evaluating the synchronization of audio and video.

Subsequent approaches have predominantly focused on extracting audio and video features from dialog clips and aligning the two modalities based on maximum correlation. FaceSync [24] is the first to evaluate the synchronization of audio and video based on lip sync. Using MFCC and direct pixel values as audio and video features, respectively, FaceSync leverages statistical models to learn the correlation between the two signals. Marcheret et al. [20] are the first to utilize deep neural networks for the AV synchronization task and treats it as a classification problem with a single "in-sync" class and several "off-sync" classes. SyncNet [9] is still considered the reference model for AV synchronization. They propose a two-stream CNN that represents audio as MFCC features and video as a sequence of mouth regions converted to grayscale images. The model is trained with a contrastive loss. SyncNet also demonstrates that synchronization models can be leveraged to discriminate whether someone is speaking or not and is still used as a strong baseline in Active Speaker Detection (ASD) research [23, 25, 25]. Later, Perfect Match [10] proposes a new learning strategy replacing the contrastive loss with a multi-way matching objective. With the emergence of Transformers, new synchronization models appear and leverage this innovative architecture. However, these models do not entirely replace CNN-based feature encoders, but instead use Transformers to process features acquired by CNNs. VocaLiST [15] builds on the two-stream CNN encoders from [9] and incorporates a cross-modal Transformer optimized using binary classification. ModEFormer [13] further explores replacing the cross-modal Transformer architecture with distinct Transformers for audio and video streams. AVST [8] uses a similar architecture to VocaLiST [15] and trains the model on variable length video sequences. However, the flexibility of the Transformer architecture is hindered by CNN encoders, which work on fixed input lengths. Furthermore, AVST [8] for the first time explores training with full frames instead of face crops, but requires larger input length to get comparable performance. This is because the visual synchronization cue (mouth) can be very small compared to the whole frame, thus making the problem much harder. Similarly, [14] also explores working with full frames and proposes a new architecture that compresses the audio-visual tokens using sparse selectors to efficiently train the model. However, their prediction resolution of 200 ms steps is not suitable in practice [22] and they still depend on CNN encoders. We emphasize that recent synchronization models consistently integrate Transformers into CNNs. Nevertheless, Dosovitisky et al. [12] demonstrated that convolution-free Transformers achieve competitive results with CNNs while being more flexible. In this direction, a relevant work in multi-modal learning literature is VATT [5], which takes raw video, audio, and text as input and explores a modality-agnostic Transformer by sharing weights among the three modalities. The model is convolution-free and achieves state-of-the-art results in several downstream tasks.

In summary, although the aforementioned methods demonstrate impressive results on rather simple datasets like LRS2 [2] and LRS3 [3], they are not suitable for usage in live broadcasting and film production for several reasons. First, all models rely on CNNs, and are therefore unable to digest audio and video in its original form. This requires

video and audio conversions which are highly undesirable for professional use cases. Second, these models require audio processing to extract intermediate features, which adds significant computation time and might remove relevant information. Third, these methods do not generalize to complex scenarios. Finally, the large size of these models results in long inference times.

3. DiVAS

In this section, we introduce DiVAS, see Figure 1 for a complete overview. DiVAS encodes raw video and audio into latent representations using modality-specific Transformers and uses contrastive learning to discriminate between in-sync and out-of-sync samples. The model ingests short clips of fixed duration. Yet different from previous methods, the number of video frames and audio samples dynamically varies depending on the original frame and sample rates. DiVAS uses a novel positional encoding that leverages timestamp information, making the model robust to different media specifications and bringing audio and video to the same time scale.

3.1. Base Model

Video encoder. We design our video encoder as a 3D Vision Transformer $\mathcal{E}_{3D}$, which encodes the input video frames $x_v \in \mathbb{R}^{F\times H\times W\times C}$ into a learnable token $z_v = \mathcal{E}_{3D}(x_v)$, $z_v \in \mathbb{R}^D$. The number of frames F dynamically changes according to the video frame rate given a fixed input time t, $F = t * fps$. Note that we can either use the full video frames as input or face crops. The input frames are first divided into patches $\mathsf{x}_v^p \in \mathbb{R}^{f\times h\times w\times c}$, $p \in \{1, 2, ..., N\}$, with $N = HWF/hwf$. These patches are flattened and projected with a trainable affine layer into 1D vectors $z_v^p \in \mathbb{R}^D$, which serve as input sequence for the Transformer. The learnable token z_v is prepended to the input sequence and 3D sinusoidal positional encoding is added to inform the attention layer about the relative position of frames and image patches.

Audio encoder. Similarly, we design our audio encoder as a 1D Transformer $\mathcal{E}_{1D}$, which encodes the input audio signal $x_a \in \mathbb{R}^{1xS}$ into a learnable token $z_a = \mathcal{E}_{1D}(x_a)$, $z_a \in \mathbb{R}^D$. The number of audio samples S dynamically changes according to the audio sample rate given a fixed input time t, $S = t*sr$. The raw audio signal is first divided into patches $\mathsf{x}_a^p \in \mathbb{R}^{1\times s}$, $p \in \{1, 2, ..., M\}$, with $M = S/s$. These patches are then projected to a higher dimension by a learnable affine layer to create the tokens $z_a^p \in \mathbb{R}^D$. 1D sinusoidal positional encoding is added to the tokens and the extra learnable token z_a is prepended, which serve as input sequence to the Transformer. Unlike all previous methods which transform the raw audio signal into Mel-spectrograms or MFCC features, we directly operate on raw audio signals saving computation time and keeping all the signal information.

Common space projection. The learnable tokens (a.k.a. latent representations) from the video and audio encoders (z_v, z_a) are projected into a common space av by fully connected layers where they can be compared. Therefore, we project video and audio to a common space by $z_{v\to av} = f_{v\to av}(z_v)$ and $z_{a\to av} = f_{a\to av}(z_a)$ respectively.

Optimization. We use a contrastive learning approach as in [9]. During training, we produce negative samples with a 50% probability by artificially advancing or delaying the audio signal with respect to the video. The misalignment can be as small as 1 video frame and as large as possible, only limited by the clip length. We optimize the model by minimizing the distance between synchronized audio-video pairs and maximizing the distance between unsynchronized pairs, see Equation 1.

$$\mathcal{L} = \frac{1}{2N}\sum_{n=1}^{N} y_n d_n^2 + (1-y_n)\max(0, m-d_n)^2, \quad (1)$$

where $y_n \in \{0, 1\}$ is the binary target for in sync / out-of-sync audiovisual pairs, m is a margin value used as constraint, and $d_n = \|z_{a\to av} - z_{v\to av}\|_F$ is the Frobenius norm of the distance between the two latent representations.

3.2. Time Aware Positional Encoding

Convolutional models are restricted to fixed-size inputs. Therefore, a common practice we observe in all previous methods [6, 9, 10, 14, 15] is to use a fixed input of 5 video frames and 3200 audio samples, which is equivalent to 0.2 seconds at 25 fps and 16 kHz, respectively. DiVAS however, is purely based on Transformers, being able to handle inputs of varying sizes. Based on that, we fix the time input duration to $t = 0.2$ seconds following previous methods, but dynamically feed a different number of video frames and audio samples depending on the original frame and sample rates. We use 1D sinusoidal positional encoding to encode the audio samples [26] and 3D sinusoidal position encoding to encode the video frames, which can be formulated as a natural extension of the 1D case as in 2.

$$\begin{aligned}
e_{(\mathsf{x},2i)} &= \sin\frac{\mathsf{x}}{\tau^{\frac{6i}{D}}}, \quad e_{(\mathsf{x},2i+1)} = \cos\frac{\mathsf{x}}{\tau^{\frac{6i}{D}}} \\
e_{(\mathsf{y},2j+\frac{D}{3})} &= \sin\frac{\mathsf{y}}{\tau^{\frac{6j}{D}}}, \quad e_{(\mathsf{y},2j+1+\frac{D}{3})} = \cos\frac{\mathsf{y}}{\tau^{\frac{6j}{D}}} \\
e_{(\mathsf{z},2k+\frac{2D}{3})} &= \sin\frac{\mathsf{z}}{\tau^{\frac{6k}{D}}}, \quad e_{(\mathsf{z},2k+1+\frac{2D}{3})} = \cos\frac{\mathsf{z}}{\tau^{\frac{6k}{D}}}
\end{aligned} \quad (2)$$

where $(\mathsf{x}, \mathsf{y}, \mathsf{z})$ is the position of a patch in image plane and time with $\mathsf{x} \in \{0, 1, ..., H/h-1\}$, $\mathsf{y} \in \{0, 1, ..., W/w-1\}$, and $\mathsf{z} \in \{0, 1, ..., F/f-1\}$, $\tau = 10000$ and $i, j, k \in$

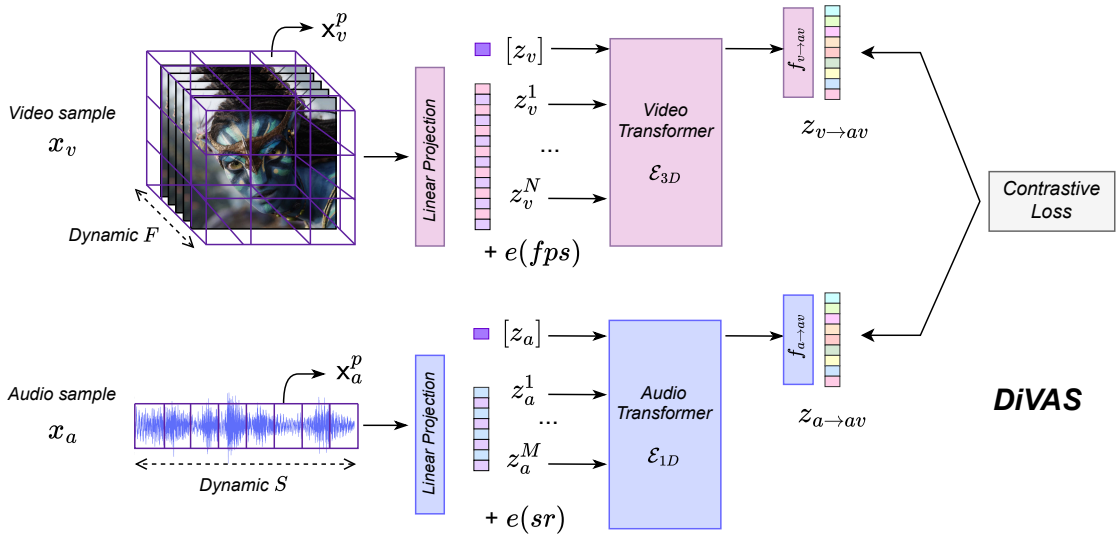

Figure 1. DiVAS architecture. The raw input signals are divided into patches and linearly projected into tokens. We concatenate the patch tokens together with the modality-token and add timestamp positional encoding to conform the input to the Transformer model. The modality-token output is sent to a linear layer that brings the modality specific representation to a common reference space where audio and video can be compared through a contrastive loss.

$\{0, 1, ..., D/6 - 1\}$ so that each third of the positional encoding encodes the position in the respective dimension.

Regular sinusoidal encoding is agnostic to the relative time between video frames or audio samples, as it just encodes the sequence order. Therefore, we propose a time aware positional encoding, which encodes not only the natural order of the frames and samples but also the relative time distance between them, providing exact timestamp information. We do so by applying a temporal factor that depends on the video frame rate and audio sample rate, which also brings audio and video to the same time scale. For video we modify z such that $\mathsf{z} = \{0, 1, ..., F/f - 1\}\frac{100f}{fps}$, whereas for audio we use $\mathsf{p} = \{0, 1, ..., S/s-1\}\frac{100s}{sr}$, where 100 is used as scaling factor.

4. Evaluation Protocol

4.1. Offset Prediction

We follow the same approach as SyncNet [9] to get the alignment between audio and video. Given a clip of length L measured in video frames, we take a sliding-window approach and divide it into W overlapping video and corresponding audio samples with a step size of 1, getting (x_v^w, x_a^w) where $W = L-F+1$. Then, DiVAS is used to get video and audio features for every window $(z_{v\to av}^w, z_{a\to av}^w)$. We compute the distance between every video feature and all audio features in the range $[-v_{\text{shift}}, .., 0, .., +v_{\text{shift}}]$, obtaining a distance matrix $D \in \mathbb{R}^{W\times O}$, with $O = v_{\text{shift}} * 2+1$ potential offsets. Note that a significant drawback of classification methods [8, 10, 15] is that they need to run the model for every audio-video combination i.e. $W * O$ times, making evaluation very computationally expensive, whereas DiVAS only needs to be run W times. Given D, we average the distance values over all windows $\overline{D} \in \mathbb{R}^O$ to mitigate the influence of non-relevant samples in a clip (e.g. silence in speech) and look for the minimum distance to find the actual offset $o = v_{\text{shift}} - \text{argmin}(\overline{D})$. The confidence of the prediction can be computed as $c = \text{median}(\overline{D}) - \text{min}(\overline{D})$. A high confidence value means that, for a particular offset, there is a predominant peak which gives the alignment between the two signals. A low confidence value means that video and audio are uncorrelated, e.g. off-screen dialogue or non-speakers, see Figure 2. This means that the model's ability to find alignment between audio and video can be leveraged to assess whose lips are correlated with the speech.

4.2. Robustness

Previous methods [8–10, 15] are evaluated only on the original videos, assuming 0 offset. None of them perform a robustness evaluation to actual offsets. Only [14] applies synthetic offsets at inference time, but, as a classification model, is limited to 3 classes (audio leads, in sync, audio lags) or 21 classes (-2 to +2 seconds with 200 ms step size). We propose to test models against random artificial offsets in the range $\pm v_{\text{shift}}$, simulating real scenarios where syn-

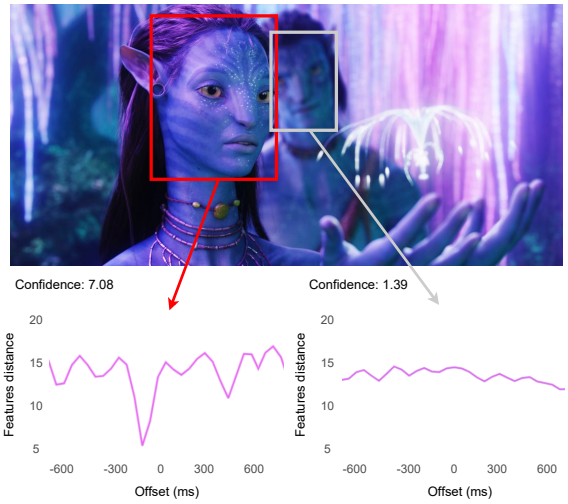

Figure 2. Non-speakers' mouth movements show no correlation with speech. We represent non-speakers with a gray bounding box. Speakers' mouth movements instead, exhibit a clear peak of correlation with speech for a specific alignment in time, indicating the offset between audio and video. In the example, dialog is behind with respect to the video by 150 ms and hence, the speaker is highlighted in red.

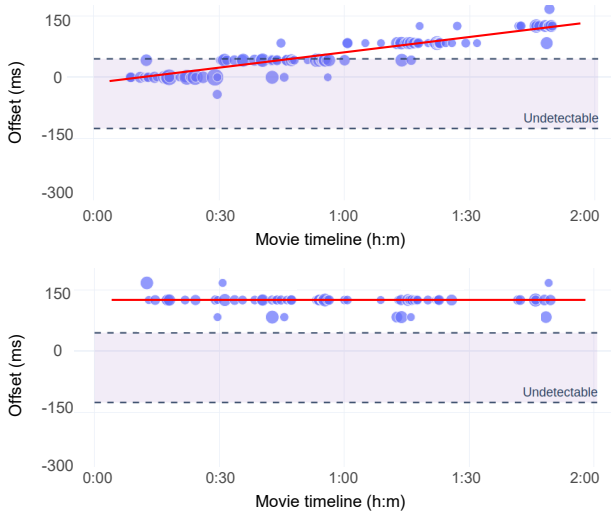

Figure 3. Synchronization timeline for a given title showing a drift where audio and video start in sync and by the end of the movie they are out of sync (top image) and constant offset where dialog is ahead with respect to the video by 125 ms (bottom image).

chronization issues occur. Note that DiVAS is not limited to a specific number of classes.

4.3. Tolerances

In literature, a prediction is considered correct if it is within a certain tolerance with respect to the expected value. More specifically, it is considered correct if it lies in the range ±1 frames, corresponding to ±40 ms at 25 fps. In [14], they instead use a tolerance of ±5 frames, corresponding to ±200 ms at 25 fps, which exceeds the accepted limits [22]. Instead, we consider the standard tolerance in the literature, ±40 ms, and propose to include the undetectable [-125, 45] ms and acceptable [-185, 95] ms tolerances according to ITU recommendations [22]. We consider tolerances in the time domain to generalize to all video frame rates.

4.4. Title Predictions

Previous methods [8–10, 14, 15] only consider synchronization at the clip level, thereby being limited to only predict constant offsets. However, there are four types of sync issues that happen in reality: constant offset, drift early, drift late and intermittent offset. As far as we know, we devise for the first time how to get a synchronization assessment for an entire title, contemplating the aforementioned diversity of issues. Given a title, it is split into dialog scenes, which are not constrained to single-face clips. We make a prediction for every face and keep only high confidence predictions, hence removing non-speakers and scenes with off-screen dialog. Then, we use a RANSAC-based algorithm to exclude outlier predictions and find a linear model that describes the title synchronization. We look at the slope and magnitude of the regression line to asses whether the audio is in sync with the video, the audio leads / lags by a constant offset or has a drift early / late. We measure the confidence of the general prediction as the agreement among clip predictions. For visualization, we propose a synchronization movie timeline in which we display the predicted offset in ms for every dialog scene. Such visualization would help quality control teams to quickly and intuitively analyze synchronization issues without manually checking the entire movie, see Figure 3. The size of each sample represents the confidence of the prediction. Generally, we observe that dialog scenes cover $\sim 30\%$ of the title and are well spread from beginning to end, providing enough samples to provide a synchronization assessment. In the top example, the audio starts in sync with the video and drifts progressively earlier. In the bottom example, dialog is ahead with respect to the video by a constant offset of 125 ms.

5. Experiments

5.1. Implementation Details

Both audio and video Transformer encoders have 3 layers, 3 heads and an MLP dimension of 1024. We use patch size $(h, w, f) = (16, 16, 1)$ for video and $s = 128$ for audio. Face crops are resized to $(H, W, C) = (96, 96, 3)$ whereas full frames are resized to $(H, W, C) = (240, 240, 3)$ so that the mouth is big enough. To avoid changing the aspect ratio, we keep the height but remove the sides to make it squared, assuming that the speaker is not in the extremes

Model	Accuracy (%)	#Par	FPS
SyncNet [9]	99.4 | 99.4 | 99.4	13.6M	265
VocaLiST [15]	98.7 | 98.7 | 98.8	80.1M	13
DiVAS	99.4 | 99.4 | 99.5	4.8M	4500

Table 1. Results on LRS2 dataset [2]. The models are tested against random offsets. Accuracies are given for different tolerances, i.e. [-40, 40] ms | [-125, 45] ms | [-185, 90] ms. Note that DiVAS has ∼3 and ∼16 times less parameters than [9] and [15] respectively, and is ∼17 and ∼350 times faster.

of the frame. We observed no decrease in performance and it reduces the number of patches. F and S vary according to the frame and sample rates, respectively, and correspond to 0.2 seconds duration. For example, we use 6 frames for 30 fps and 3200 samples for $16kHz$. We convert audio to mono. In the contrastive loss, we use a margin value of $m = 20$. For evaluation, we consider $v_{\text{shift}} = 15$ as previous approaches. We implement the network in Pytorch [21] and train it end-to-end from scratch using the Adam optimizer [16]. We use a learning rate of 10^{-5} and a batch size of 128 and train the model until convergence. We use a single Nvidia GeForce RTX 3090 in all our experiments.

5.2. SOTA Comparison

Note that DiVAS is not directly comparable to previous approaches, as it is the first method that works with original media specifications. In this experiment, we test our model on the benchmark dataset LRS2 [2], which consists of face-cropped videos transformed to 25 fps, hence not exploiting the full potential of DiVAS. We show a comparison with SyncNet [9] and VocaLiST [15] in Table 1. As stated in Section 4, every model is tested against random artificial offsets and accuracies are given for different temporal tolerances. This is different from previous works where they only consider in-sync videos for evaluation. We use the public code and trained models from [9, 15] and follow the same evaluation protocol as [9] for fair comparison, which is explained in Section 4. DiVAS achieves close to perfect performance, similar to [9], and has ∼3 and ∼16 times less parameters than [9] and [15], respectively, and is ∼17 and ∼350 times faster. Note that FPS values exclude the common preprocessing and are only with respect to the synchronization models. Our increase of speed is not only due to the model size but also to the fact that DiVAS ingests raw audio rather than spectral features, which are computationally expensive. Also, as explained in Section 4.1, VocaLiST [15] as a classification model needs to be run more times to predict the offset. We noticed that, when testing the models against artificial offsets, SyncNet [9] outperformed more recent approaches like VocaLiST [15].

Model	Accuracy (%)	Accuracy-60fps (%)
Base	94.0 | 95.8 | 97.2	82.2 | 82.8 | 98.6
DiVAS	94.4 | 96.2 | 97.4	96.8 | 97.7 | 99.2

Table 2. Results on LRS3-multfps dataset [3]. The models are tested against random offsets. Accuracies are given for different tolerances, i.e. [-40, 40] ms | [-125, 45] ms | [-185, 90] ms. Both models are trained with videos up to 30.0 fps. We show results on videos up to 30.0 fps (left) and generalization to 60.0 fps (right).

5.3. Timestamp Positional Encoding

Table 2 shows the effect of using timestamp information in positional encoding. To do so, we use the LRS3 dataset [3] keeping the original frame rates of the videos, referred to as LRS3-multfps. Following the observations of [14], the resulting videos were encoded using the MPEG-4 Part 10 (H.264) codec and audios using the AAC codec. These media specifications avoid temporal artifact leakage that may lead to a trivial solution when training for audio-visual synchronisation. We use the face crop version of the dataset in this experiment. LRS3-multfps has 7 different frames rates including 23.976, 24.0, 25.0, 29.97, 30.0, 59.94 and 60.0 fps. We curate new splits to keep a balance of frame rates getting 44.5k train, 1.7k validation and 4.6k test clips. All videos with 59.94 and 60.0 fps were kept as a separate test split to evaluate generalization to unseen and higher frame rates. We compare DiVAS to our base model which uses regular positional encoding. Both models were trained on LRS3-multfps videos up to 30.0 fps and tested on videos with similar frame rates (left) and videos with 59.94 and 60.0 fps (right). In both cases, we apply random artificial offsets to the videos and accuracies are again given for different temporal tolerances. We observe that DiVAS achieves better results by leveraging timestamp information and, more importantly, is able to generalize to unseen frame rates, whereas the base model suffers more than 10% drop of performance for stricter tolerances. Note that working with original frame rates is crucial to providing reliable predictions, especially for professional use cases such as live broadcasting and film production. As far as we know, DiVAS is the first AV synchronization method which is frame rate independent.

5.4. Challenging Data

We test DiVAS on more challenging videos which can include background noise and music, intricate head poses, faces covered with makeup, and animated characters. To do so, we build a large dataset that includes 218k dialog scenes, with a total of ∼140 hours, from 170 titles including live action and animation movies, TV shows, musicals, documentaries and series' episodes. The process of identifying dialog scenes involves a careful combination of scene

and speech detectors. Additionally, it often requires face detectors and trackers to isolate the synchronization cue in the video if face crops are used as input. The framework should not be constrained to single-face clips, as that would reduce the coverage of the title being analyzed. However, this poses another challenge of identifying who is the speaker or even if there is a speaker at all (i.e off-screen speech). While previous datasets [2, 3] contain a single person talking in the video, this is not the case when we look at the film industry. After manually labeling the active speakers in our dataset, we observed that among all the scenes containing speech and faces, only 65% contain actual speakers. The other 35% of the scenes include faces that are not talking, such as in documentaries and when panning to the reaction of a listener. Of the valid dialog scenes, 60% contain a single person talking, 30% show a single speaker among other individuals, and 10% include dialogue between multiple speakers in the scene. The frame rate of videos in this complex dataset range from 23.97 to 59.94 fps, whereas the sample rate of audio is constant at 48 kHz.

Table 3 shows how VocaLiST and DiVAS trained on previous benchmark datasets [2, 3] generalize to the new dataset. While both models struggle to generalize to more challenging scenarios, DiVAS outperforms VocaLiST for most relevant tolerances. Note that VocaLiST requires 25 fps videos, so both models trained on LRS2 [2] are tested on converted videos. We observe that DiVAS trained on LRS3-multfps [3] generalizes better than when trained on LRS2 [2]. This might be due to the fact that original media specifications are kept, however the model still struggles due to the complexity of the data, especially for stricter tolerances. Finally, we train and test DiVAS on the new dataset. We test accuracy on each of the different scenarios, namely single person talking, one person talking among other individuals, and dialogues with multiple speakers, but observed very similar behaviour and thus we report average values only. As expected, we observe a significant increase of accuracy when the model is exposed to a similar level of difficulty during training.

However, production-type use cases demand higher accuracy and should be considered at the title level instead of the clip level. The last row of Table 3 collects our results obtained at the title level as explained in Section 4.4, which allow to exclude outliers and provide more reliable predictions, as shown in Figure 3. We evaluate every title against random artificial constant offsets and drifts applied accordingly to all clips in the title.

5.5. Video Frames versus Face Crops

DiVAS works either with face crops or full frames as video input. We explore the trade-offs of each setting using the LRS3-multfps dataset [3]. The first setting is the most studied in literature [6, 9, 10, 15] and yields outstanding performance. However, this comes at the expense of requiring additional face detection and tracking, which is extremely time consuming specially if high accurate detectors are used such as RetinaFace [11]. Additionally, this setting requires comparing the audio with every face to find whose lips are correlated to the audio, which sometimes can go up to 20-30 faces in crowded scenes. The second setting is much harder, as the network needs to first localize the sound source which usually represents a very small fraction of pixels (mouth) in the entire frame. We visualized the self-attention of the $[z_v]$ token and realized that until half of the training, the model focuses on hand movements rather than on the mouth, which are generally more noticeable, especially for medium and long shots. As a consequence, the model needs more time to converge. However, we only observe a slight decrease in performance and this setting reduces a lot of overhead at inference time. Specifically, it does not require the use of face detectors and trackers and only needs to be run once, independently of the number of faces present in the scene. This framework is also more

Model	Trained on	Accuracy (%)
VocaLiST [15]	LRS2	59.2 | 60.1 | 80.5
DiVAS	LRS2	54.5 | 73.9 | 82.1
DiVAS	LRS3-multfps	61.9 | 81.9 | 90.7
DiVAS	New data	89.4 | 93.7 | 97.0
DiVAS	New data	93.5 | 96.4 | 99.4

Table 3. Results on our new dataset. Artificial offsets are applied to all videos. While DiVAS trained on LRS2 shows a better generalization than VocaLiST [15], models trained with current benchmark datasets do not handle complex scenarios well. Training with more complex data improves the results at clip level (top rows) and taking the problem to the title level (last row) allows us to guarantee reliable and accurate solutions for production standards.

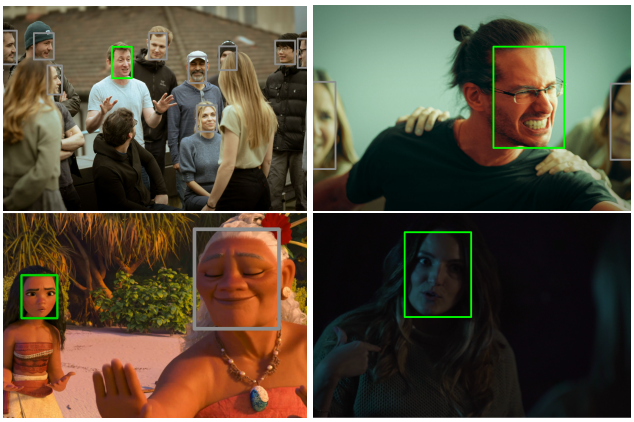

Figure 4. Active Speaker Detection in scenes with multiple individuals, makeups, extreme lighting conditions and in animation content.

Model	Accuracy (%)	Prep (FPS)
DiVAS	94.4 | 96.2 | 97.4	9.4
DiVAS-frame	93.6 | 95.6 | 96.9	317

Table 4. Results on LRS3-multfps dataset [3]. Accuracies are given for different sensitivity thresholds, i.e. [-40, 40] ms | [-125, 45] ms | [-185, 90] ms. DiVAS working with face crops is highly accurate, at the expense of time-consuming preprocessing. DiVAS operating on the entire frame shows slightly worse performance, but the end-to-end solution is much quicker and efficient. The preprocessing runtime is calculated for Full HD video resolution.

Model	Acc_{21}	Acc_{21}^{tol}	#Par
Iashin et al. [14]	80.7	96.9	55.3M
DiVAS-frame	92.2	94.0	17M

Table 5. Results on LRS3-multfps dataset [3]. For comparison, here we follow the evaluation protocol of [14] and measure accuracy on a 21-class offset grid ranging from -2.0 to +2.0 sec with a step size of 200 ms. In [14], they allow 200 ms tolerance (tol) which is too high for real applications. When no tolerance is considered, DiVAS achieves 10% higher accuracy.

generic and shows the potential of sound source localization, making it promising to capture audio-visual relationships beyond speech. In order to quantitatively show the trade-offs of these two settings, we compare DiVAS trained on face crops and full frames in Table 4. DiVAS-frame version is almost as accurate as DiVAS using face crops and requires $\sim$ 30 times less preprocessing time. Even if the model only saw single-speaker scenes during training (TED talk videos) we observed reasonable sound source localization in our new dataset where self-attention is over the speaker's face even in scenes with multiple individuals.

In Table 5 we compare DiVAS-frame to Iashin et al. [14] which is also a full frame approach. Note that [14] requires video conversion to 25 fps as previous approaches. DiVAS without tolerance achieves 10% higher accuracy and is not limited to 21 classes. Note that the 200 ms tolerance proposed in [14] is too high for real applications.

5.6. Active Speaker Detection

We evaluate our model for the task of Active Speaker Detection (ASD), which aims to detect who is speaking in one or more speakers scenarios. We use the confidence c of our model to determine whether or not someone is speaking, see Figure 2. Table 6 compares how DiVAS trained on public benchmark datasets generalizes to more complex data with the same model trained on videos of similar difficulty. We report F_1 score, a widely used metric for ASD. Unlike in Section 5.4, we observe a difference when evaluating ASD on the three different scenarios described, namely scenes

Model	Trained on	S	SI	MS	Avg
DiVAS	LRS2	69.8	78.6	48.8	65.7
DiVAS	LRS3-multfps	68.5	77.7	53.6	66.6
DiVAS	New data	91.8	92.0	70.6	84.8

Table 6. F_1-score for Active Speaker Detection on our new dataset. Three conditions are evaluated: one speaker (S), on speaker among other individuals (SI) and multiple speakers (MS).

with one speaker (S), one speaker among other individuals (SI), and dialogues with multiple speakers (MS), therefore we report separate and average values. This distinction is not made in previous papers and we believe it is essential to fully understand the behaviour of AV synchronization and ASD models. We observe that scenes with multiple speakers are consistently more challenging. This is because we average predictions over the entire clip, which is unfavourable if the person speaks for a very short period. DiVAS is able to localize the speaker and hence, provide reliable synchronization assessments, even in complex scenes with multiple people, makeups, extreme lighting conditions and in animation content. See some examples in Figure 4.

6. Conclusions

This paper investigates the automatic assessment of audio and video synchronization, specifically targeting professional use cases such as film production and live broadcasting. It is important that such a system consumes the content in its original form, without modifying the media specifications, such as frame rate. Additionally, the model should perform on complex content seen in the wild, including dialogue scenes with background noise and music, intricate head poses, excessive makeup, and multiple individuals wherein the speaker is unknown. We find that audio-video synchronization can be addressed by leveraging multi-modal architectures fully based on transformers that can ingest raw audio and video. Transformers are able to differentiate essential parts of the input and can do so for inputs of variable length. This unique property was exploited to introduce a novel positional encoding that leverages timestamp information to make our model frame and sample rate independent. Our model can work either with face crops or full frames, avoiding expensive preprocessing. DiVAS is small and fast, and can predict different synchronization issues such as constant offsets or drifts. Additionally, our proposed model can be used for the task of Active Speaker Detection. Future work could focus on leveraging the full frame version of DiVAS for more general sounds such as transient sounds.

References

[1] Baton lipsync — automatic audio-video sync detection. www.interrasystems.com/BATON-LipSync.php. 1

[2] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 44(12):8717–8727, 2018. 1, 2, 6, 7

[3] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. LRS3-TED: a large-scale dataset for visual speech recognition. *CoRR*, abs/1809.00496, 2018. 1, 2, 6, 7, 8

[4] Sucharu Aggarwal and Alka Jindal. Comprehensive overview of various lip synchronization techniques. In *2008 International Symposium on Biometrics and Security Technologies*, pages 1–6. IEEE, 2008. 1

[5] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34:24206–24221, 2021. 2

[6] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proceedings of the European conference on computer vision (ECCV)*, pages 435–451, 2018. 1, 3, 7

[7] Elizabeth J Carter, Lavanya Sharan, Laura Trutoiu, Iain Matthews, and Jessica K Hodgins. Perceptually motivated guidelines for voice synchronization in film. *ACM Transactions on Applied Perception (TAP)*, 7(4):1–12, 2010. 1

[8] Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Audio-visual synchronisation in the wild. In *British Machine Vision Conference (BMVC)*, 2021. 1, 2, 4, 5

[9] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13*, pages 251–263. Springer, 2017. 2, 3, 4, 6, 7

[10] Soo-Whan Chung, Joon Son Chung, and Hong-Goo Kang. Perfect match: Improved cross-modal embeddings for audio-visual synchronisation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3965–3969. IEEE, 2019. 1, 2, 3, 4, 5, 7

[11] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641*, 2019. 7

[12] A Dosovitskiy, L Beyer, A Kolesnikov, D Weissenborn, X Zhai, and T Unterthiner. Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[13] Akash Gupta, Rohun Tripathi, and Wondong Jang. Modeformer: Modality-preserving embedding for audio-video synchronization using transformers. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. 2

[14] Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Sparse in space and time: Audio-visual synchronisation with trainable selectors. In *British Machine Vision Conference (BMVC)*, 2022. 1, 2, 3, 4, 5, 6, 8

[15] Venkatesh S Kadandale, Juan F Montesinos, and Gloria Haro. Vocalist: An audio-visual synchronisation model for lips and voices. In *Interspeech*, pages 3128–3132, 2022. 1, 2, 3, 4, 5, 6, 7

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[17] B.E. Koster, R.D. Rodman, and D. Bitzer. Automated lip-sync: direct translation of speech-sound to mouth-shape. In *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, pages 583–586 vol.1, 1994. 2

[18] John Lewis. Automated lip-sync: Background and techniques. *The Journal of Visualization and Computer Animation*, 2(4):118–122, 1991. 2

[19] Shao Li, Qi Ding, Yichen Yuan, and Zhenzhu Yue. Audio-visual causality and stimulus reliability affect audio-visual synchrony perception. *Frontiers in Psychology*, 12:629996, 2021. 1

[20] Etienne Marcheret, Gerasimos Potamianos, Josef Vopicka, and Vaibhava Goel. Detecting audio-visual synchrony using deep neural networks. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015. 2

[21] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NIPS*. 2019. 6

[22] ITU Radiocommunication. Relative timing of sound and vision for broadcasting. 2, 5

[23] Muhammad Shahid, Cigdem Beyan, and Vittorio Murino. S-vvad: visual voice activity detection by motion segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2332–2341, 2021. 2

[24] Malcolm Slaney and Michele Covell. Facesync: A linear operator for measuring synchronization of video facial images and audio tracks. In *Advances in Neural Information Processing Systems*. MIT Press, 2000. 2

[25] Ruijie et al. Tao. Is someone speaking? exploring long-term temporal features for audio-visual active speaker detection. In *29th ACM Multimedia*, pages 3927–3935, 2021. 2

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[27] G. Zoric and I.S. Pandzic. A real-time lip sync system using a genetic algorithm for automatic neural network configuration. In *2005 IEEE International Conference on Multimedia and Expo*, pages 1366–1369, 2005. 2