Audio-visual Correlation from Cross-modal Attention in Self-supervised Transformers on Videos of Musical Performances

Master Thesis - Computer Science
University of Twente

Hessel R. Bosma

Examination Committee:
dr. N. Strisciuglio (chair), E. Talavera Martínez PhD (supervision), and dr. D.V. Le Viet Duc

September 18, 2022

Abstract

Attention maps from transformer-based models using the self-attention mechanism are highly interpretable. Works in the vision-and-language domain train general models on large data sets using self-supervised methods, leveraging such attention values between modalities for multiple different learning tasks. Within the audio-visual domain, we see similar approaches, but these are often specialized towards one type of data set, like human speech, and require supervised data sets. This work introduces a general and more versatile audio-visual training framework, based on the approaches of the vision-and-language works. This framework can be applied to many different audio-visual learning scenarios. We apply this framework for the task of audio-source localization. Our implementation uses an audio-visual model based on two separate convolutional-based audio and visual embedding stages, and a single transformer-based encoder stage. This model is trained with the self-supervised proxy task of multi-modal alignment. Our new MUSIC-200k data set of 192 007 videos of musical performances (`https://github.com/HesselBosma/MUSIC200k`) was used for training and validation. Visual inspection of the source-localization results shows that the framework is valid for this particular learning task. These results suggest broader applicability of the framework, e.g. different learning tasks. The framework has some promising benefits like more general applicability, zero-shot learning capability, and requiring only non-supervised training data. However, the audio-source localization performance seems to be limited. Opportunities have been identified to increase performance on audio-visual learning tasks. But, these performance-increasing measures were not empirically tested in this work. Furthermore, due to time constraints, only a limited visual evaluation was performed instead of a more informative numerical evaluation. Thus, no direct accurate comparison can be made with the performance of other methods.

1 Introduction

The introduction of the self-attention mechanism and the transformer architecture mark a major shift in deep learning architecture. In the application of natural language processing, transformer-based networks [1–3] are outperforming the now almost obsolete recurrent neural network variants like LSTM [4] and GRU [5]. In computer vision tasks, transformers are starting to outperform the deep convolutional neural networks [6–8], which have been used and optimized extensively over the last decades [9–13]. Similar transformer-based architectures now compete with- or outperform the state-of-the-art in many tasks from almost all modes of deep learning. However, arguably one of the strongest properties of transformer-based architectures is their interpretability. Depending on the learning task, the attention maps in transformer layers can show how different parts of the input relate to each other and to the output of the model [3, 6].

This property of interpretability extends to multimodal transformer-based models as well. Such models use the self-attention mechanism to combine information from multiple data modalities, allowing for attention between both modalities, i.e. cross-modal attention. Analyzing these attention values can be extremely powerful. This is done extensively in the vision and language domain [14–21], where the cross-modal attentions are interpreted as correlations between the different parts of the image and text inputs. These models can often be trained in a self-supervised manner and work for all kinds of visual-textual data, and thus can be trained on massive general-purpose vision-and-text data sets. Similarly in the audio-visual domain [22–24], we see that cross-modal attention is starting to be used for audio-visual learning tasks like audio-source localization. However, within the audio-visual domain, we see mostly models specialized for different subsets of audio-visual data (like human speech [22]) and a heavy reliance on data supervision [23].

This works aims to generalize this audio-visual learning approach, drawing heavy inspiration from the vision-and-language domain. Such a general audio-visual learning approach would allow audio-visual models to be trained on much easier to obtain, and often larger, non-supervised audio-visual data sets. And these models could then be used for multiple and different audio-visual learning tasks without the need for fine-tuning. To determine the possibility of such a solution, we aim to answer the following research questions:

- *RQ1 - Can the cross-modal attention values from an audio-visual model trained using self-supervised methods be interpreted as audio-visual correlation?*
  - *Sub RQ1.1 - If so, can audio-visual correlations be used directly for audio-visual learning tasks like audio-source localization?*
- *RQ2 - How can such a self-supervised audio-visual learning framework that leverages cross-modal attention maps be implemented for a task like audio-source localization?*

To answer these questions a transformer-based audio-visual model is trained in a self-supervised way similar to the methods in the vision-and-language domain. From this model, the cross-modal attention maps are extracted and processed in different ways to perform audio-source localization. These audio-source localization results are visually inspected to determine the source localization performance and the existence of audio-visual correlation, though we do not strive to maximize this performance in this work. Furthermore, for the validation of this learning approach and the evaluation of such audio-visual models, a new large-scale audio-visual data set was created. Publicly available data sets are most useful for specific learning tasks like only containing human conversations or they are limited in size. Our new data set contains 192.007 videos of musical performances. Such videos are useful for the visual evaluation of more general audio-visual models on different learning tasks like source localization or audio separation, the first of which we do in this work. My work has the following contributions:

- A self-supervised audio-visual learning framework, that exploits the interpretability of attention maps in cross-modal transformer models.
- An implementation of this audio-visual learning framework on the task of audio-source localization as well as visual evaluation of this implementation.
- The MUSIC-200k data set. A large set of videos of musical performances for evaluating large-scale architectures on audio-visual learning tasks.

This thesis is structured as follows. In Section 2, the related works and the state-of-the-art are discussed. The scientific approach is described in Section 3. In this section, the learning framework is described in detail, as well as its components. Section 4 lays out the details of the experimental setup as well as the implementation details. Furthermore, this section introduces our new MUSIC-200k data set, as well as others used in this project. The results of the experiments are described in Section 5. These results and their implications are discussed in Section 6 as well as directions for future works. Finally, Section 7] will conclude this thesis.

2 Related Works

This work can be placed in between of two bodies of work. First, there are works exploring interpretability of attention maps in cross-modal applications of transformer based architectures. These are part of the more general research in transformers. And second, works that focus on audio-visual learning tasks with various

methods and architectures. This section will describe these bodies of work and how my work can be placed within those.

2.1 Audio-visual Learning

As the term would suggest, audio-visual learning encompasses learning tasks in which sound and visual information is used or required to solve a learning task. One such task is sound localization, where the goal is to find from which image region a certain sound is likely to originate. In this scenario, we can ask the question: What part of the image is producing sound? An example of this task is active speaker recognition, in which we want to identify the person speaking within a visual frame. Such problems can be solved with different levels of detail. Some works do localization based on regions of interest or object detection [22, 25], while others localize on the pixel level [26, 27]. The second common audio-visual learning task is audio separation, which is either visually aided [28] conditioned on a certain image region [27, 29]. In this scenario, we want to filter out the audio likely produced by an image region. The separation and localization scenarios are sometimes combined. In that case, we are looking for some general audio-visual correlation [26, 27], i.e. what parts of the sound are related to what parts of the image.

To learn this audio-visual correlation, cross-modal learning architectures can be trained in a self-supervised manner using the mix-and-separate [27] paradigm. The original audio signal is added to some randomly selected audio signal from the data set. The model is then tasked with predicting an audio mask for that signal so that it filters out the original signal based on the mixed signal and the original visual input. This way the model learns the relation between these two inputs of different modalities. This technique and others discussed later, can be used within the audio-visual framework proposed in this work.

2.2 Transformers

Transformer-based architectures have been used extensively in the field of Natural Language Processing. These models outperform recurrent and convolutional models in many language tasks while being parallelizable and faster to train [3]. A transformer processes a set of one-dimensional vectors of equal size. A transformer layer is a mapping of this set to a new one with the same size and dimension. For processing an item within this set, a transformer layer can draw information from every other item, weighed by some measure of relevance. This measure of relevance is the self-attention mechanism. A query, key, and value vector (Q,K,V) are computed from every item vector via a learned linear projection. The attention for an item is computed by measuring the similarity between its query and the keys of all other items. The output mapping of an item is an aggregate of the values of all other items within the set, weighted by this attention. This attention mechanism makes transformer-based architectures very interpretable [3, 6]. We can extract these attention values, which tell us how the model relates every part of the set to each other.

Since it is a necessary component of my proposed audio-visual learning framework, only encoder transformers will be considered. An encoder transformer consists of one or multiple stacked transformer encoder layers. These encoder layers make use of the self-attention mechanism to learn to embed contextual information about the members of the input set of vectors. In the field of natural language processing, these vectors represent words that as a set form a sentence. Transformer-based language models like BERT [1], learn the meaning of sentences at increasing levels of abstraction for every transformer layer. BERT adds a learnable "class" vector to the input set, tasked with learning global meaning for the whole sentence. BERT demonstrates how transformer encoders can be pre-trained effectively using two self-supervised training methods. In next-sentence prediction, the model is asked to predict if two input sentences follow one another and hereby learning the meaning of entire sentences globally. While in masked language modeling the model learns the local structures within sentences by predicting masked out parts of the input sentence.

For computer vision applications, transformer encoders can be used in a similar way. Images have to be converted to a set of fixed-length tokens first. The vision transformer (ViT) [6] slices an image in non-overlapping patches. These patches are each linearly projected into one-dimensional vectors with a fixed length. This set of tokens is then encoded in the same way as BERT would encode a sentence, however, the class token is tasked with learning a representation of the image. From this representation, a prediction head can learn to perform computer vision tasks like classification, object detection, or image segmentation. Compared to models based on convolutions (CNN's) [9, 10, 12, 13], vision transformers perform well. Vision transformers struggle with dense prediction tasks like object detection and image segmentation because they are limited by the patch size of the patch embedding stage. Furthermore, transformer-based architectures require much larger data sets to train. To some extent, the Data-Efficient Image Transformer (DeiT) [7] and the Shifted Windows transformer (Swin) [8] have improved upon the vision transformer in these aspects. Transformer-based architectures now can compete, or even outperform state-of-the-art CNN-based architectures in almost all computer vision tasks. And just like in NLP, vision transformers can be pre-trained [30] well and the attention maps are highly interpretable [6, 30].

2.3 Multi-modal Transformers

The success of transformer-based architectures in NLP, as well as computer vision, shows that with only small adaptations, the same general architecture can be used across modes of data, with state-of-the-art performance in each. This is promising, as seems to be more cross-over between the research in different fields. Furthermore, this would suggest that transformer-based architectures would be well-suited to learning tasks that require multiple modes of data, i.e. multi-modal learning. The largest body of work researching these cross-modal applications, combine textual and visual information. Transformer-based architectures are used to co-process these types of data, to perform a wide range of vision-and-language tasks. In general, these architectures are based on two "stream" of data, one for each modality. Lu et. al. [18] describe how the language and visual streams of data are processed separately, allowing these streams to communicate through co-attention in a common transformer-like layer. The design of such co-attention layers can vary as well as how early in the network the two streams are allowed to communicate (early vs late fusion). The attention maps produced in these layers are highly interpretable, and can be used for various visual-linguistic reasoning/grounding tasks [17, 18, 31].

Some models [14, 15] are rather complex or specialized for a certain learning task. However, many architectures [16–21] are variations on a common design. They share a Bert-like transformer stage, where the streams of both modalities are embedded into tokens with a shared dimensionality. These tokens are simply concatenated and processed by a transformer layer as if they were a sentence as with BERT [1]. Extra embeddings or specialized tokens can be added to allow the transformer to distinguish between the modalities. With masked multi-modal learning, like Bert's masked token prediction, we can randomly mask input tokens and ask the model to predict what these should be. And in multi-modal alignment, similar to Bert's next sentence prediction, we can randomly switch the tokens from one of the modalities for tokens corresponding to some other sample. Then we ask the output class token to predict if these tokens are from the same data sample or not. These "proxy-tasks" [18,19] can be used to (pre-)train the multi-modal transformer stage of the architecture in a self-supervised way. However, depending on the length of the streams and learning tasks, both streams can be pre-trained on uni-modal tasks and the entire model can be trained end-to-end on the learning task at hand if needed.

This general architecture of Bert-like cross-modal transformers extends to the audio-visual domain as well. Instead of the textual stream we have an audio stream. Again we see state-of-the-art performance, as well as highly interpretable attention maps. These attention maps in some cases are directly used for some audio-visual tasks like audio-visual source localization [22, 23] or finding regions of interest [24]. However, these works are often limited to a specific learning task or require supervised data sets. My work explores how the more general approach from the vision-and-language domain can be translated to the audio-visual domain to be used with video data sets.

3 Approach

This section describes my proposed framework for self-supervised audio-visual learning, inspired by works from the vision-and-language domain [17–19]. This framework is general purpose. It applies to many different audio-visual data set types and it can be used for various learning tasks. The framework contains the following components

1. A transformer-based audio-visual model that allows for cross-modal attention at some point and can distinguish between the inputs of both modalities.
2. A self-supervised, proxy learning task that encourages, or ideally forces, the model to learn audio-visual correlation.
3. An extraction and interpretation method to extract this learned audio-visual correlation from the cross-modal attention maps, for solving learning tasks directly.

These components can be implemented in different ways to fit different audio-visual learning environments. This section will discuss how they can be implemented and how they work together. Within this project, this framework is implemented to perform audio-source localization. The implementation within this aims to demonstrate the purpose of the framework and its components and not how to best implement it for performance.

3.1 The Audio-visual Model

The purpose of the model within the framework is to model the audio-visual relations within videos. From a video, we separate the visual component v and the audio component a. Such a model f then learns to jointly represent the audio and visual components in some embedding $\hat{e}_{v,a}$. The model is parameterized by θ such that:

$$f_\theta(V, A) \rightarrow \hat{e}_{V,A}. \tag{1}$$

The audio-visual relations are internally represented within the model by means of attention values from the self-attention mechanism in transformer layers. Such a model can be implemented in different ways within the framework.

The General Audio Visual Transformer - The G-AVT (see Figure 1) is the implementation I propose within this work. It is an audio-visual encoder that is based on BERT [1] and ViT [6]. It draws inspiration from many works in the vision-and-language field [17–

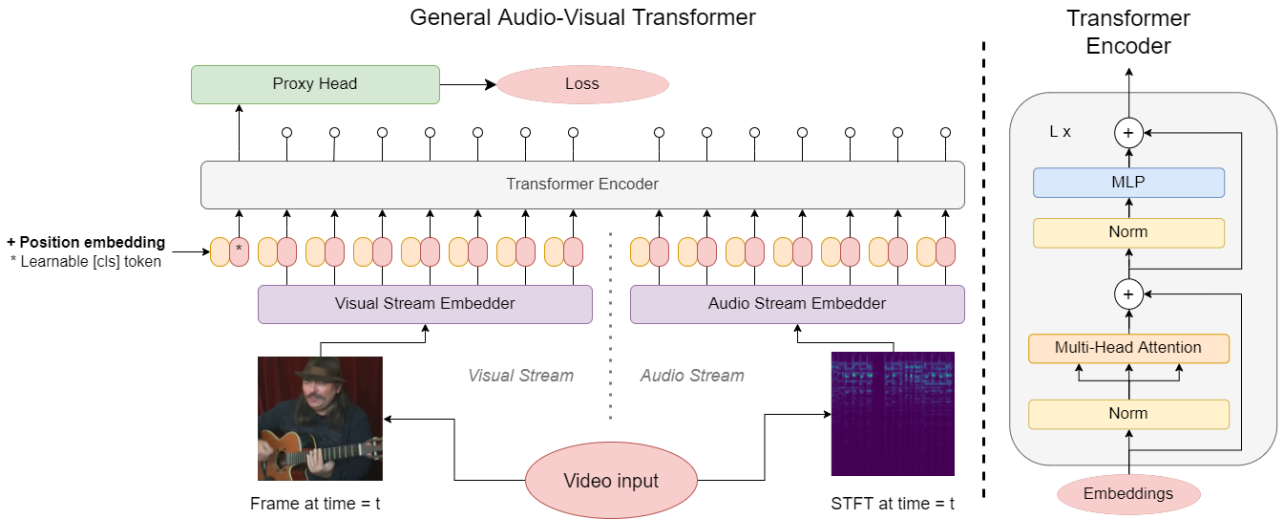


Figure 1: Diagram of the architecture of the General Audio Visual Transformer. Components like the proxy head, transformer encoder, image embedder, and audio embedder can all be implemented in different ways. This diagram shows how each component works together. See Figure 2 for the implementation of the embedders within the experiments of this project.

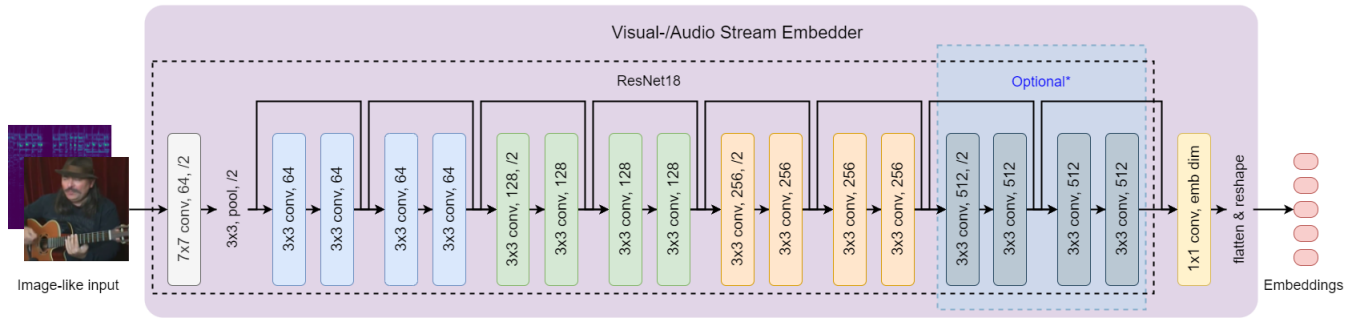


Figure 2: Diagram of ResNet18 [12] based implementation of the image/audio embedder used within the experiments of this work. The last residual block in blue* is skipped for the audio embedder. Note that the G-AVT in Figure 1 can be implemented with different embedder architectures than this, and this is by no means the best implementation.

21]. Similar to these vision-and-language architectures, the G-AVT can be split up into two streams of different modalities, and has three main components:

1. An embedding stage for the visual stream.
2. A separate embedding stage for the audio stream.
3. A transformer-based encoding stage that fuses the two streams.

From a video, we first extract a single frame and a fixed-length audio segment conditioned at the same moment in time. The video frame is reshaped to a predetermined size. The audio segment is converted using Short Time Fourier Transform [32], to an image-like representation with also a pre-determined fixed size. The embedding stages of each stream separately convert both inputs into an ordered set of fixed-length vectors. The vectors in each set share the same dimensionality and thus can be fused into a single set. Similar to BERT [1] we add learned positional embeddings. The encoder should also be able to distinguish between both inputs. Other works add an extra "mode" embedding to this positional embedding [17]. However, in this case, such an extra embedding is unnecessary because the inputs are of a fixed length. As a result, the tokens of each modality will always have the same positions within the fused ordered set, and thus the position contains information about the mode of the token. This means that we also don't need a special separation token like in BERT. However, a class token is added to the sentence, which is tasked with learning the context of the entire input.

3.2 Proxy Task Training

The purpose of the proxy task within the framework is to force the model to learn audio-visual correlation in a self-supervised way. In general, it takes the audio and visual inputs, as well as the output from the audio-visual model, and calculates the gradient to optimize the model:

$$L_{proxy}(V, A, f_\theta) \rightarrow \nabla\theta, \tag{2}$$

with $\nabla\theta$ being the gradient with which to optimize the parameters of the self-attention layers, and optionally any additional parameterized architectural features such as the embedding stages. To implement this, we can use one or multiple self-supervised audio-visual learning tasks. The G-AVT is extended with a proxy prediction head, see Figure 1. This proxy head is used during training and can be discarded at inference time. There can be many different such proxy tasks, each with its own proxy head.

Multi-modal alignment - This is the simplest proxy task and the easiest to implement. We randomly sample two videos from the data set $\{V_0, A_0\}$ and $\{V_1, A_1\}$. We randomly input either visual component V_0 or V_1 with $p = 0.5$ and the audio component a_0 as normal. We then ask the model to determine which visual component it sees, the one corresponding to the audio input or a different one:

$$\begin{gathered} proxy_{mma}(f_\theta(V_{match}, A_0) \rightarrow match, \\ with: \\ match \sim U(\{0, 1\}). \end{gathered} \tag{3}$$

We extend the class token output of the G-AVT with a proxy head with output dimension two and softmax activation, like in a two-class classification problem. During training, we randomly (p=0.5) switch one of the inputs with one randomly selected from the data set. We ask the model to predict if the two inputs match or not. In this way we force the G-AVT to compare the inputs globally and thus form cross-modal connections.

Mix-and-separate - This method is thorough and forces the model to look in more detail to the audio input. We mix audio input A_0 with N audio signals randomly selected from the data set A_n with $n = (1, .., N)$, taking advantage of the additive nature of audio signals, where:

$$A_{mix} = A_0 + \sum_{n=1}^{N} A_n. \tag{4}$$

We then ask the model to recover the original audio signal A_0 by looking at the mixed signal A_{mix} and the visual input V, as such:

$$proxy_{m\&s}(f_\theta(V, A_{mix})) \rightarrow A_0. \tag{5}$$

In practice we use $N = 1$, thus mixing the original audio signal with one randomly sampled one. The proxy head takes as input the class token or the audio tokens and outputs a two-dimensional audio mask. This audio mask m has the same dimensions as the input audio signal A_{mix}, and when multiplied together recovers the original audio signal A_0:

$$A_0 = m * A_{mix}. \tag{6}$$

This way the model is forced to pay attention to the audio signal with high resolution and compare it with the visual input globally. It is then expected that the model forms cross-modal connections with high acoustic resolution and precision.

Some other Proxy tasks that can potentially also be implemented successfully, include masked multi-modal learning [18] / masked language modeling [1] as well as self-distillation with no labels (DINO) [30]. These tasks don't force the model to form audio-visual connections but do encourage it to some extent.

3.3 Attention Maps Extraction

Given an unseen sample, we extract the attention maps from the audio-visual model. These attention maps can be processed in different ways. The choice of processing method depends on the audio-visual learning task at hand. Here I provide a general picture of different possible implementations. In this work, I experiment with different variations, which will be discussed in Section 4 and 5

First, we can take these attentions from a single attention head, or average over all heads. The latter should give a more complete picture. Similarly, we can look at a single transformer layer, or at the average attention over all layers. Here we expect to see a difference in abstraction level, depending on what layer we select. Furthermore, we can look at the attention in different directions or in both, i.e. the directions to or from a token.

The attentions from and to the class token tell us in general what the model is looking at, since this token is tasked with learning the global meaning of all inputs. This depends on the proxy task and how the proxy head is implemented. But to get an idea of correlations between the two inputs of different modalities, we can also look at the attentions in between the tokens from those streams, i.e. the audio-to-visual attentions or the visual-to-audio attentions.

Different from the attention direction, we can look at these attentions from different "perspectives". Grouping the incoming and outgoing attention values per visual token tell us something about how visual objects relate to all the different parts of the sound. This visual perspective can be used for vision tasks like source localization. Similarly, we can group the attention per audio token to view the audio-visual attentions from the perspective of the audio stream. This is useful to determine how different audio parts relate to all different visual parts. This audio perspective is useful for tasks like audio segmentation. Note that the perspective is about grouping and ultimately reducing the dimensionality of the attention matrix per group, which has nothing to do with the attention direction.

3.4 Audio-source Localization

We assume the cross-modal attentions to be analog for audio-visual correlation. These correlations can be leveraged for solving different audio-visual learning tasks. This project implements the framework for the task of

audio-source localization. We take the audio-visual attention maps from the visual perspective, i.e. grouped per visual token. For each visual token, we measure the average cross-modal attention to all audio tokens. This tells something about how related this visual token is to sound in general. We cannot be sure that these attention maps are correlating the audio and visual inputs. Rather we could interpret them as the prior audio-visual attention learned from the training data set. Normalization is needed if we are after the direct relation between the audio and visual inputs, i.e. the audio source location conditioned on the audio input. We can use Bayes' rule to calculate the conditional audio-source probability:

$$p_{source}(\theta, V, A) = \frac{attn(\theta, V, A)}{attn(\theta, V, 0)}. \tag{7}$$

Where the numerator is the evidence for the audio-source probability and the denominator is the prior audio-source probability. This normalization method draws inspiration from condition guidance [33] in generative learning.

4 Experimental Setup

This section will discuss the experimental setup used within this work. First, the new MUSIC-200k will be introduced. Then the evaluation method and experimentation methods are discussed, followed by the details of the implementations within those experiments.

4.1 The MUSIC-200k Data Set

The MUSIC data set by Zhao et. al. [27] contains a total of 1313 videos of musical performances that have been hand-selected and labeled. However, because these videos have to be downloaded from YouTube, the usable size depends on their availability. As used in this project, the data set contained 1.121 total samples of which 20% was used for testing. For training the G-AVT, the size of MUSIC was found to be insufficient and no larger data set of musical performances existed.

For the purpose of training data-hungry audio-visual models like the G-AVT, I present the larger brother of the MUSIC data set: MUSIC-200k. As the name would suggest, this data set is much larger than the original, containing 192 007 individual videos of musical performances by artists. This data set is a subset of the Youtube8m [34] data set, which is a massive set of hand-labeled YouTube videos of various categories. Samples containing the label "musician" were extracted, while videos containing labels like "music video", "dance", or "trailer" were removed for a cleaner data set. The Youtube8m data set provides pre-extracted video and audio features and does not reference the Youtube video IDs directly. The MUSIC-200k does reference these video IDs so that the raw videos can be used for a broader range of audio-visual learning cases. The samples are labeled identically to the YouTube8m samples, using the same labeling dictionary. See Figures 3 and 4 for the distribution of labels and video lengths within the data set. The videos can be downloaded from YouTube directly, and stored in any desired format. This means that the usable size of MUSIC-200k also depends on the availability of these Youtube videos. At the time of this project, the available size of the train set was 149.067. 14.908 of these were used for validation. Random samples from the dataset set were taken to evaluate the quality of the videos. It was found that at least 95% of the videos contained music, as well as some of the corresponding musical instruments in the video frame for the majority of video duration. The quality of the samples in MUSIC-200k is a little less than the original MUSIC data set, since they are not each manually checked. There thus is a trade off between quality and quantity. MUSIC-200k has been made available on GitHub under the Apache 2.0 license (same as for Youtube8m): `https://github.com/HesselBosma/MUSIC200k`).

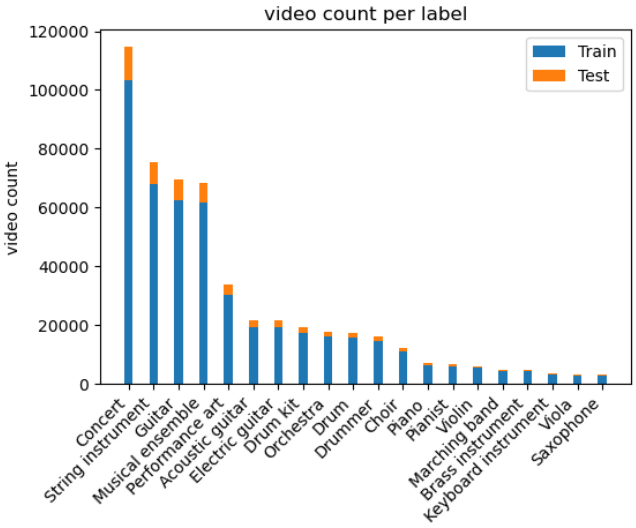


Figure 3: Distribution of the most common labels in the MUSIC-200K data set.

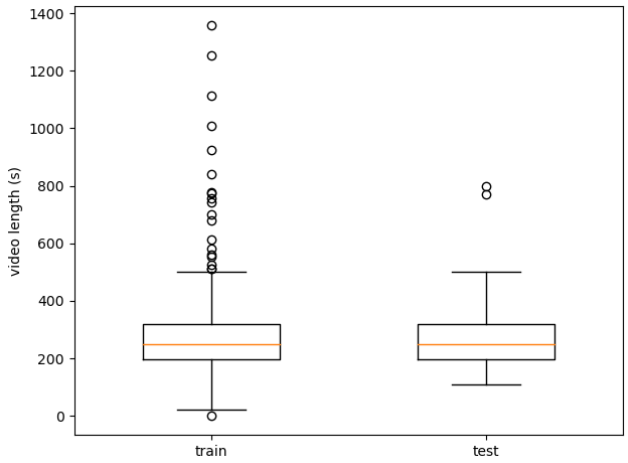


Figure 4: Box plot of the lengths of the videos in the MUSIC-200k data set.

4.2 Visual Evaluation

In the experiments, we aim to validate the proposed audio-visual learning framework for audio source localization. This validation is based on visual inspection of audio-source maps. For a video frame and audio segment, we use the framework to construct an audio source map. This audio source map is a heat map of how likely the sound heard in the audio segment is to originate from different image regions. We then view this map side by side with the visual frame, allowing us to determine what objects within the visual frame the model thinks are the audio source. Because we are using videos of performances with musical instruments, we would like to see regions with instruments to be highlighted. Similarly, in videos where people are singing, we want to see their faces or mouths being highlighted. The results in Section 5 will contain such side-by-side maps for the reader to inspect.

We also want to learn how to implement the new framework. The proposed framework has many variables to adjust, such as architectures, training methods, and attention map processing methods. We are not looking to maximize performance here, we looking for some configuration that produces satisfactory results based on the visual evaluation of the localization heat maps. In this project, different experiments were done to build up to this configuration. First, different audio-visual models were trained on the proxy tasks of mix-and-separate and mainly multi-modal alignment. This allows us to determine an architecture for the audio-visual model component of the framework together with a proxy task. We then use this trained model to experiment with different processing and normalization methods for extracting the source-localization maps.

4.3 Implementation Details

Data storage - The data sets used in this work include large amounts of videos. Some of these videos have a long duration and contain high-quality audio and video. Storing all this data in maximum quality was unfeasible given the available resources, so some compromises were made to reduce the storage size. First, the videos are limited in duration between 15 and 90 seconds. These video sections are extracted from the middle of the video, to reduce the chances of sampling video introductions, ending scenes, etc. The audio part of these sections was stored as a full-length, multi-channel raw time series in a .wav file, with a reduced sampling rate of 11025Hz and a bit rate of 176kbps. From the video part, the individual frames are stored with a reduced frame rate of 0.2 frames per second and a margin of half the frame rate. Meaning, that for a 90-second video, 17 images are stored. The image resolution was reduced to 480 pixels vertically, conserving the aspect ratio. The images were stored were compressed and stored in a .jpg format. In total, the training set of MUSIC-200k takes up around 497Gb on disk.

Pre-processing and augmentation - From a sample, a random image frame is selected as well as the corresponding audio segment. From the stored image we extract a 128x128 image tensor. At training time we apply image augmentation components:

- Rotation with a randomly selected angle of maximum of 20 degrees in each direction and a probability of 0.4. The rotated image is cropped to fit a rectangular frame.
- Random image resize of between 0.7 and 1 times the original image size.
- Randomly stretching the aspect ratio with a maximum of 25%.
- Cropping the image randomly to a 128x128 format.
- Random adjustment to the brightness of 30% maximum.
- Random adjustment to the contrast of 30% maximum.
- Random adjustment to the saturation of 20% maximum.
- Random adjustment to the hue of 10% maximum.

At inference time, only a center crop is used. This image augmentation was originally implemented to reduce overfitting on the small MUSIC data set, but might be unnecessary for the much larger MUSIC-200k data set.

For the audio, we select the corresponding section to the image frame with an approximate length of 6 seconds (65535 audio samples). We randomly adjust the volume to a maximum of 50% during training time. On this time series signal, we apply Short Time Fourier Transform (STFT). With a frame length of 1022 and a hop length of 256, obtaining a 512 x 256 time-frequency representation. This image is re-sampled in log-frequency scale resulting in a 256 x256 image-like representation of the audio. Re-sampling the image in log-frequency scale is done to place more emphasis on the lower frequencies, which contain most of the fundamental frequencies and overtones of musical instruments. This is similar to the common practice of applying Mel-Frequency scale [35], to make the representation more similar to the frequency composition of human hearing.

Architectures - To implement the G-AVT from Figure 1, the four modular components have to be implemented. We need to define:

1. An audio embedding stage
2. A visual embedding stage
3. An encoder stage that merges the two streams
4. A proxy head during training

Experiment	1	2	3	4
G-AVT version	G-AVTa	G-AVTa	G-AVTb	G-AVTc
Data set	MUSIC	MUSIC-200k	MUSIC-200k	MUSIC-200k
Proxy task	MMA/M&S	MMA/M&S	MMA	MMA
Loss function	Cross entropy/MSE	Cross entropy/MSE	Cross entropy	Cross entropy
Batch size	6	6	32	64
Optimiser	ADAM	ADAM	ADAMW	ADAMW
Learning rate	10^-6	10^-6	10^-5	10^-5
Weight decay	0.01	0.01	0.01	0.01
Dropout	0.1	0.1	0.1	0.1

Table 1: Optimisation configurations of the main experiments. Each experiment marks a key development within this work, and the results of each experiment are discussed in Section 5. See Table 2 for the details of the G-AVT implementations for every particular version.

The audio and visual embedding stages can be completely different architectures. This allows for selecting an architecture that fits each modality best. However, in this work, we use similar architectures for both of these embedding stages, for convenience. In theory, because we represent the audio as an STFT, we can treat it just like an image. The first experiments used VIT-like [6] patch embedding stages. These divide the image into non-overlapping patches of equal size and use a single learned linear projection matrix to project the patches to the one-dimensional vectors. However, the best performing model used a patch-embedded architecture based on ResNet18 [12] instead. The features extracted by the ResNet18 are converted to the embedding dimension of the encoder stage using a 1x1 convolution. We then reshape and flatten this output to retrieve the embeddings. A diagram of the visual patch embedder can be found in Figure 2.

The encoder stage is based on BERT's transformer encoder [1]. The encoder takes as input a set of one-dimensional, fixed-length embeddings, to which a class embedding is concatenated with the same dimension. To these, we add learned positional embeddings to encode positional and modality information. The encoder consists of an adjustable number of multi-head self-attention layers with an adjustable number of heads. The G-AVT uses an MLP with two linear layers with adjustable hidden dimensions to calculate the query, key, and value. And dot product similarity is used to calculate the attention values. See Table 2 for the full model configurations.

Depending on the proxy task we extend the class token output of the encoder stage with different proxy heads, see Figure 1. In the experiments in this project, we focus on multi-modal alignment and use a simple linear projection as proxy head. This is a linear layer with an input size equal to the embedding dimension of the embedding stage and output size of 2. After a softmax layer, we have a two-class classification head with classes "match" and "no match" to represent the classes generated by the multi-modal alignment learning task.

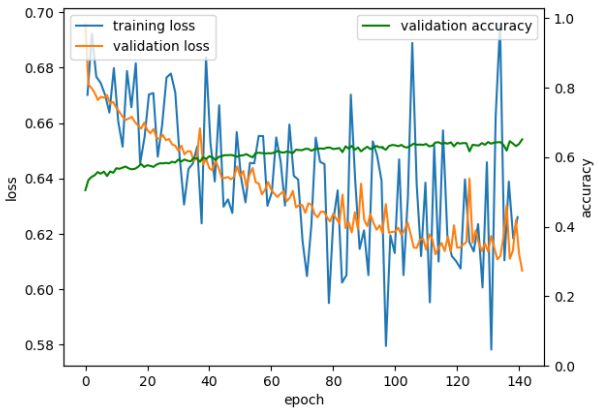
	G-AVTa	G-AVTb	G-AVTc
Embed type	Patch	ResNet	ResNet
Emded dim	128	256	512
Hidden dim	256	512	1024
Layers	8	8	2
Heads	8	8	16
Activation	GeLu	GeLu	GeLu
Norm type	Layer	Layer	Layer

Table 2: G-AVT architecture implementation hyper-parameters used per version.

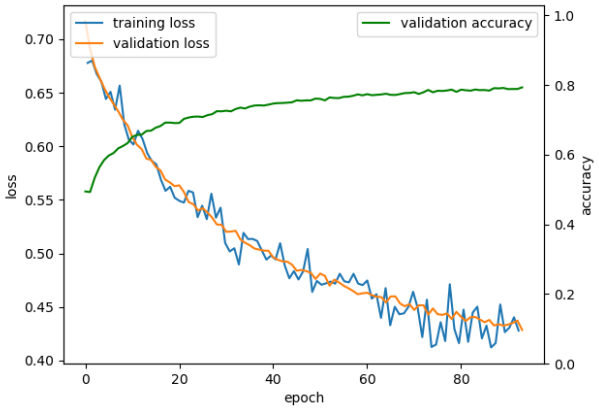
Optimisation - The optimization process depends on the version of the G-AVT that is trained, as well as the proxy task. All variations were trained on a Linux machine using various GPU's depending on their availability. Most training was done with a single GPU. However, when available, data parallelism was used to split the workload over two GPU's and increase the batch size. Gradient clipping was used to prevent exploding gradients. All training was done in 32-bit precision so no mixed-precision training and no gradient check-pointing was used. These techniques could have sped up the training process or allowed the use of bigger models. The optimization of the models required much time training on the computing cluster, which contributed to the time constraints experienced in this project. For further details on the optimization process, see Table 1. These values are by no means the optimal hyper-parameters, as there was no time to do parameter sweeps. Furthermore, optimal performance on the proxy tasks was never the goal.

5 Results

This section describes the results of the experiments. First, we look at the proxy task performance of different architectures for the audio-visual model. Then we take the best performing model and try different techniques for processing the attention maps.



(a) Experiment 2 (see Table 1) - G-AVTa with patch embedder



(b) Experiment 3 (see Table 1) - G-AVTb with ResNet feature extractor

(c) Experiment 4 (see Table 1) - G-AVT with ResNet feature extractor and increased embedding dimension (best result)

Figure 5: The performance on the multi-modal alignment proxy task from different experiments, see Table 1. NOTE: I artificially applied smoothing to the training loss curve, since I mistakenly only saved this loss on an interval, instead of averaging over the entire epoch. Making the original loss curve unreadable. This explains some of the noise within this curve, compared to the validation loss.

5.1 Audio-visual model

One of the biggest challenges in this project was finding a well-performing audio-visual model, constrained by computing resources, and limited time. Many experiments were done to find a model that could be used for audio-source localization.

Experiment 1 - First, version a of the G-AVT (see Table 2) was trained with multi-modal alignment as well as mix-and-separate on the smaller MUSIC [27] data set, as described in Table 1. The model was found to overfit immediately. This result showed that a larger data set is required and led to the creation of the MUSIC-200k data set.

Experiment 2 - The same experiment was repeated, but this time on the much larger MUSIC-200k data set. This solved the overfitting issue, although the performance of the G-AVTa on the proxy task is still poor, as can be seen in Figure 5a. We can see that validation accuracy on the multi-modal seems not to be able to reach even 70%. This means that the model is only barely able to predict if a sound matches its visual input. This poor performance is also reflected in the quality of the attention maps, which to the human eye look close to random noise. Furthermore, the mix-and-separate proxy task was found to be much more difficult to train on. Finding the right architecture for the M&S head was also difficult. So it was decided that the following experiments would use multi-modal alignment only.

Experiment 3 - Instead of the patch embedding stage from ViT, a convolutional feature extractor was implemented, creating the G-AVTb (see Table 2). This was found to significantly increase the performance of the G-AVT on the multi-modal alignment task, see Figure 5b. We now see that the validation accuracy of the multi-modal alignment task approach 80%. This increase in performance was also reflected in the attention maps. We find that the model attends more to sound-producing objects in general. However, the results are very inconsistent and not yet satisfactory, as the final results will be.

Experiment 4 - The G-AVTb uses a rather small embedding dimension of only 256, compared to VIT. The smallest version of ViT has an embedding dimension of 768 [6]. Hypothetically, a larger embedding dimension is needed to capture the audio-visual information at the abstraction level required for performing the multi-modal alignment. Therefore for the last experiment, the depth of the encoder was decreased to allow for this increased embedding dimension under GPU memory constraints. This resulted in version c of the G-AVT, see Table 2. As can be seen in Figure 5c, this model performed the best yet. We now see that the validation accuracy on the multi-modal alignment task has well surpassed 80%. Although this is likely due to the increased training time, since the accuracy at epoch 100 is similar to that in experiment 3 at epoch 100 (see Figure 5b). The attention maps resulting from this model are also improved. We thus use this model to exper-

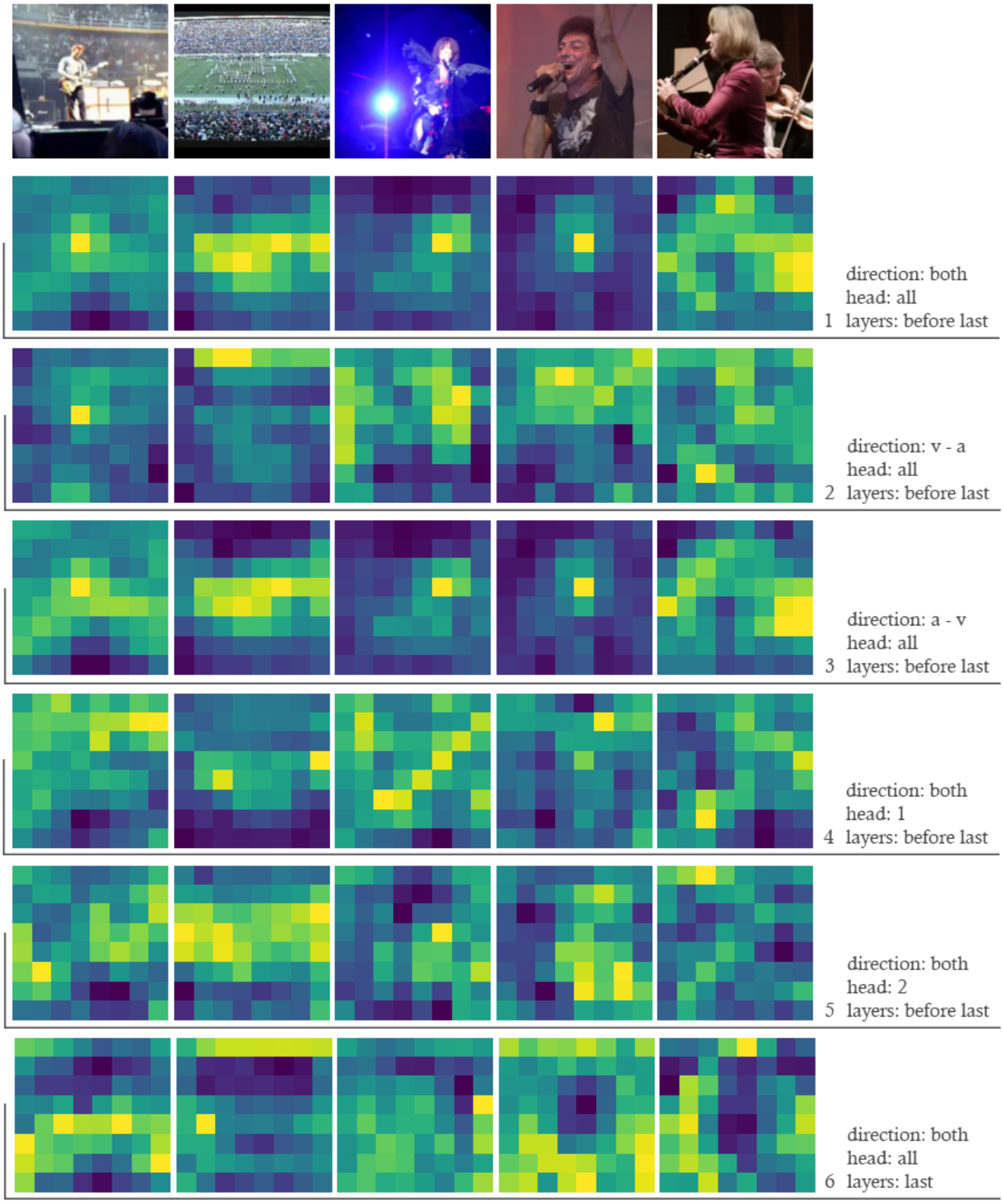


Figure 6: Audio-source localization maps produced by the G-AVTc (see Table 2) in the final experiment (see Table 1 and Figure 5c). Every row contains represents a different set of processing parameters, to reduce the values to a single 8x8 map. These parameters are: the direction of the cross-modal attention values, the attention heads over which we average the attention values, and similarly the encoder layer over which we average the attention values. The columns represent a single unseen data sample. The localization maps were created using the normalization using the normalization method from Equation 7.

iment with the different processing methods for these attention maps, see Figure 6.

5.2 Attention maps

From G-AVTc (Table 2) trained on multi-modal alignment as can be seen in Table 1 and Figure 5c, we take the raw attention maps for some unseen samples. We use the normalization technique from Equation 7 to convert the attention maps to sound localization maps. If we look at the first row of audio-source maps in Figure 6, we can see that the maps highlight the musical instruments and the mouths of singing people similar to what we would expect. Furthermore, we see that also the marching band is highlighted in the second sample. However, the maps do still contain some noise and are still inconsistent. See Appendix B for a larger set of randomly selected samples.

Within Figure 6 we can also inspect the results of some variations in the attention map processing method. We can see how the attention maps look in different directions. We see that taking the audio-to-visual attention in row 3 seems to produce more consistent results than taking the visual-to-audio attention values as can be seen in row 2. Taking an average of both values, as in row 1, results in localization maps that are similar to the audio-to-visual attention in row 3, which suggests that the audio-to-visual attentions are relatively larger in magnitude than visual-to-audio attentions. Thus the audio-visual model seems to have a preference for determining audio-visual relations in this direction.

We can also look at how different attention heads learn to pay attention to different image regions. If we take the localization maps resulting from just a single attention head, as in rows 4 and 5 of Figure 6, we can see that the results are much more random than combining the attention values from all heads as in row 1. This suggests that the attention maps from the individual attention heads are not very interpretable, and looking at all attention heads simultaneously is the better analog for audio-visual correlation here.

Finally, we could also look at the attention values from different layers. If we take a look at row 6 from Figure 6, we see that the localization maps from the last layer are bad. They do not highlight musical instruments or singing people as we would expect. This is because, for the last layer, a gradient is only determined for the class token with how the proxy head was implemented. Thus the attention values from the audio and visual tokens are ignored for the optimization with the proxy task. This means that the before the last layer is the last layer with meaningful cross-modal attention values. If a model contains more than two transformer layers we expect to see attention maps at a lower level of abstraction if we take earlier layers. However, this could not be tested since the G-AVTc only contains two layers (see Table 2).

6 Discussion

This section discusses the validity of the proposed learning framework. This validity depends on the answers to the main research questions within this work. These questions are discussed in the subsections, as well as how my new data set could be useful in future projects.

6.1 RQ1 - Solving learning tasks with correlation from attention maps

We have seen promising results for our application of the framework on audio-source localization. Visual inspection of the source-localization results on performances with musical instruments shows that in many cases, the audible instruments are highlighted. This means that we can indeed use the cross-modal attention maps from an audio-visual model, trained in a self-supervised way, for at least one audio-visual learning task directly. Although this is just one example of one such learning task, this suggests that we can indeed process such attention maps in ways to interpret them as audio-visual correlation. This opens the door for different implementations of these attention maps for different audio-visual learning tasks like audio separation. However, this was not empirically tested within this work. If this is the case, it would make this audio-visual learning framework very versatile. A trained audio-visual model could then be used for multiple different audio-visual learning tasks, and training the model does not require supervised data sets. Furthermore, the framework is not restricted to tasks related to only musical instruments, but should also be able to be used for tasks regarding different types of audio-visual data such as human speech.

There are some major limitations of my work I should discuss. First, the evaluation method is extremely limited. Visual inspection of the audio source localization results is prone to human errors and biases. Due to a time shortage, this option was chosen. Ideally, this work would have included a numerical evaluation on a secondary supervised data set allowing for comparison with other works. Or at least human inspection using focus groups. These methods would have more accurately quantified the performance on the task of audio-source localization. This performance is the second limitation of this work. Although we cannot accurately quantify the performance, visual inspection is enough to determine that the quality of the source localization maps is poor compared to that achieved by some other works [24, 26–29]. However, these works have different limitations such as a reliance on training supervision or being limited to a certain type of audio-visual data such as human speech. Furthermore, the performance of our framework heavily depends on the implementation of its components and the chosen audio-visual learning task. This brings us to the second main research question of this work.

6.2 RQ2 - Framework implementation

We have seen that the implementation of the components within the framework greatly influences the proxy task performance and the performance of the audio-source localization task. Because of limited compute availability and time shortage, the implementation of these components within this work is far from optimal. Thus there should be much room for improvement by more carefully considering the choice of these components.

The selection of audio and visual embedding stages has been found to be important in determining the performance of the model on the proxy tasks, and thus the quality of the attention maps. ViT-like patch embedding stages make the model too difficult to optimize and likely need much larger data sets. Convolutional embedding stage in contrary work well. A features extractor based on ResNet18 [12] was used in my works, so there is an opportunity for the use of deeper versions of ResNet, or newer architectures like ResNext [36]. However, using deeper feature extractors means that the audio and visual streams are fused later. We thus miss the opportunity to analyze the cross-modal attention with a lower lever of abstraction. But this could be a necessary trade-off for achieving good proxy-task performance and attention maps of satisfactory quality.

Similarly, we see that we do not sacrifice much proxy-task performance if we reduce the depth of the final encoder stage, suggesting that much of the heavy lifting is done by these deep feature extractors. Analyzing the attention maps from this shorter encoder stage is simpler since we have fewer layers to process. Tough similar, the attention maps from this shorter encoder stage seem to be of slightly higher quality. This could mean that in a deeper encoder stage the correlations are much more hidden within cascades of attention instead of direct first-degree attention values. However, this is purely speculation, since the independent variables were not controlled between the two runs with different encoder depths. Furthermore, the evaluation method used is not reliable.

Within my work, I was able to show the validity of my framework for one simple implementation using just one of the proxy tasks. Multi-modal alignment training was found to be an effective training method. It would however also be interesting to look at the mix-and-separate learning task since hypothetically it would force the audio-visual model to look at the audio signal with more detail. This could increase the quality of the attention maps from the audio perspective such that they could potentially be used for visually based audio segmentation as well. This would be an interesting topic for future work. Similarly, for future research, it would be interesting if using deeper and more advanced architectures can increase the performance of the framework such that it is competitive within the state of the art. Using high-performance deep convolutional feature extractors for both embedding stages seems like a clear direction for improvement, given the availability of computing resources.

6.3 MUSIC-200k in context

In this project, the new MUSIC-200k data set has been key for the development of our newly presented audio-visual learning framework. This data set can be placed in between other large audio-visual data sets like AVSPeech [37] and YouTube8m [34] in terms of data variance. MUSIC-200k is suitable for a broader set of audio-visual learning tasks than AVSPeech which is only useful in the context of human speech. At the same time, it is more restricted in such audio-visual learning tasks than YouTube8m. This well-defined variation within the data makes the data set well suited for developing and testing general-purpose audio-visual learning algorithms, that require large amounts of video data, such as the framework presented within this work.

There are some limitations to consider. First, the availability of the samples is tied to the availability of the YouTube links, and thus it can change. Similarly, the quality of the audio and video signals is limited by those YouTube uploads. Furthermore, not every sample within the MUSIC-200k data set has been manually checked for quality, unlike the original MUSIC [27] data set. Inspecting a random sample showed that the data set contains some bad samples. These bad samples could be videos where the acoustic and visual parts don't correlate much, like videos with background music, image overlays, or any video where the sound source is not directly filmed. However, the random sample contained fewer than 5% of such bad samples. Although there also exists some variation in quality within those samples, throughout their duration, which has not been checked manually. There appears to be a slight trade-off between quality and quantity here, compared with the MUSIC [27] data set. Despite these limitations, MUSIC-200k will be a good choice for testing and developing many general-purpose audio-visual learning tasks.

7 Conclusion

This work presents a new audio-visual learning framework. We have trained a transformer-based audio-visual model on self-supervised proxy learning tasks. The attention maps from the self-attention layers of this model are processed and normalized to solve downstream audio-visual learning tasks. The primary goal of this work has been to validate this learning approach. In particular for the downstream task of audio-source localization. Through visual inspection, we have seen that the audio-source maps produced by the implementation of our new framework can be used for source localization. This thus validates our learning framework for the task of source localization in particular and suggests that the cross-modal attention maps can be interpreted as audio-visual correlation. This in turn suggests that these attention

maps can also be used for multiple different audio-visual learning tasks. Although only visual inspection and no numerical evaluation were done within this work, the audio-source localization results appear to be of poorer quality than many other approaches.

It was found that the architecture used for the audio-visual model, greatly influences the performance of the framework. And because our implementation of the framework was constrained by computing resources and a shortage of time, there is likely much room for improving the performance on the task of audio-source localization. If the performance of the framework can improve such that it is competitive within the state-of-the-art, then our method would have notable benefits over the competition. First, it is a general purpose framework, which is not constrained to any type of audio-visual data and can thus be implemented in many different scenarios without much tailoring. Second, after training, we can use the same model and weights for solving multiple different learning tasks without the need for fine-tuning. And finally, we do not require data supervision and can thus use more easily obtainable general-purpose large-scale audio-visual data sets, like our new MUSIC-200k data set.

Directions for future work - The new framework has some promising benefits, but its performance needs to improve. The implementation of the framework has much room for improvement. The audio-visual model in particular. Assuming more available computing resources, future works can try larger models. I would suggest using more advanced deep convolutional feature extractors for both the embedding stages and a larger embedding dimension. These works could explore the effect of the depth of the transformer encoder stage. Besides increasing performance, accurate quantification of this performance would be required to place the framework amongst other methods within the state-of-the-art. Finally, since we have seen that the cross-modal attention maps can be interpreted as audio-visual correlation, it would be interesting to see how well these could be used for solving audio-visual learning tasks other than source localization. Audio separation based on the image region is among the possibilities. For these tasks, we would need accurate quantification of the performance as well. If good performance can be reached in one or preferably more audio-visual learning tasks, then the framework would be an excellent option among the audio-visual state-of-the-art. Then the framework can be useful in real-world applications, which would be another possible topic for future works.

References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[5] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[7] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 347–10 357.

[8] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[9] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, contour and grouping in computer vision*. Springer, 1999, pp. 319–345.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[14] X. Liu, L. Li, S. Wang, Z.-J. Zha, L. Su, and Q. Huang, "Knowledge-guided pairwise reconstruction network for weakly supervised referring expression grounding," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 539–547.

[15] R. Zellers, Y. Bisk, A. Farhadi, and Y. Choi, "From recognition to cognition: Visual commonsense reasoning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6720–6731.

[16] Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu, "Uniter: Universal image-text representation learning," in *European conference on computer vision*. Springer, 2020, pp. 104–120.

[17] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "Visualbert: A simple and performant baseline for vision and language," *arXiv preprint arXiv:1908.03557*, 2019.

[18] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," *Advances in neural information processing systems*, vol. 32, 2019.

[19] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7464–7473.

[20] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, and N. Carion, "Mdetr-modulated detection for end-to-end multi-modal understanding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1780–1790.

[21] J. Xu, S. De Mello, S. Liu, W. Byeon, T. Breuel, J. Kautz, and X. Wang, "Groupvit: Semantic segmentation emerges from text supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 134–18 144.

[22] T.-D. Truong, C. N. Duong, H. A. Pham, B. Raj, N. Le, K. Luu *et al.*, "The right to talk: An audio-visual transformer approach," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1105–1114.

[23] L. Zhu and E. Rahtu, "Visually guided sound source separation and localization using self-supervised motion representations," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 1289–1299.

[24] Y.-B. Lin and Y.-C. F. Wang, "Audiovisual transformer with instance attention for audio-visual event localization," in *Proceedings of the Asian Conference on Computer Vision*, 2020.

[25] R. Gao and K. Grauman, "Co-separating sounds of visual objects," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3879–3888.

[26] R. Arandjelovic and A. Zisserman, "Objects that sound," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 435–451.

[27] H. Zhao, C. Gan, A. Rouditchenko, C. Vondrick, J. McDermott, and A. Torralba, "The sound of pixels," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 570–586.

[28] R. Gao and K. Grauman, "Visualvoice: Audio-visual speech separation with cross-modal consistency," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021, pp. 15 490–15 500.

[29] H. Zhao, C. Gan, W.-C. Ma, and A. Torralba, "The sound of motions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1735–1744.

[30] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9650–9660.

[31] L. Zhou, H. Palangi, L. Zhang, H. Hu, J. Corso, and J. Gao, "Unified vision-language pre-training for image captioning and vqa," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 13 041–13 049.

[32] J. Allen, "Short term spectral analysis, synthesis, and modification by discrete fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 235–238, 1977.

[33] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[34] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: A large-scale video classification benchmark," 2016.

[35] S. S. Stevens, J. Volkmann, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *The journal of the acoustical society of america*, vol. 8, no. 3, pp. 185–190, 1937.

[36] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[37] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K. Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein, "Looking to listen at the cocktail party," *ACM Transactions on Graphics*, vol. 37, no. 4, pp. 1–11, aug 2018.

A Random Samples from MUSIC-200k

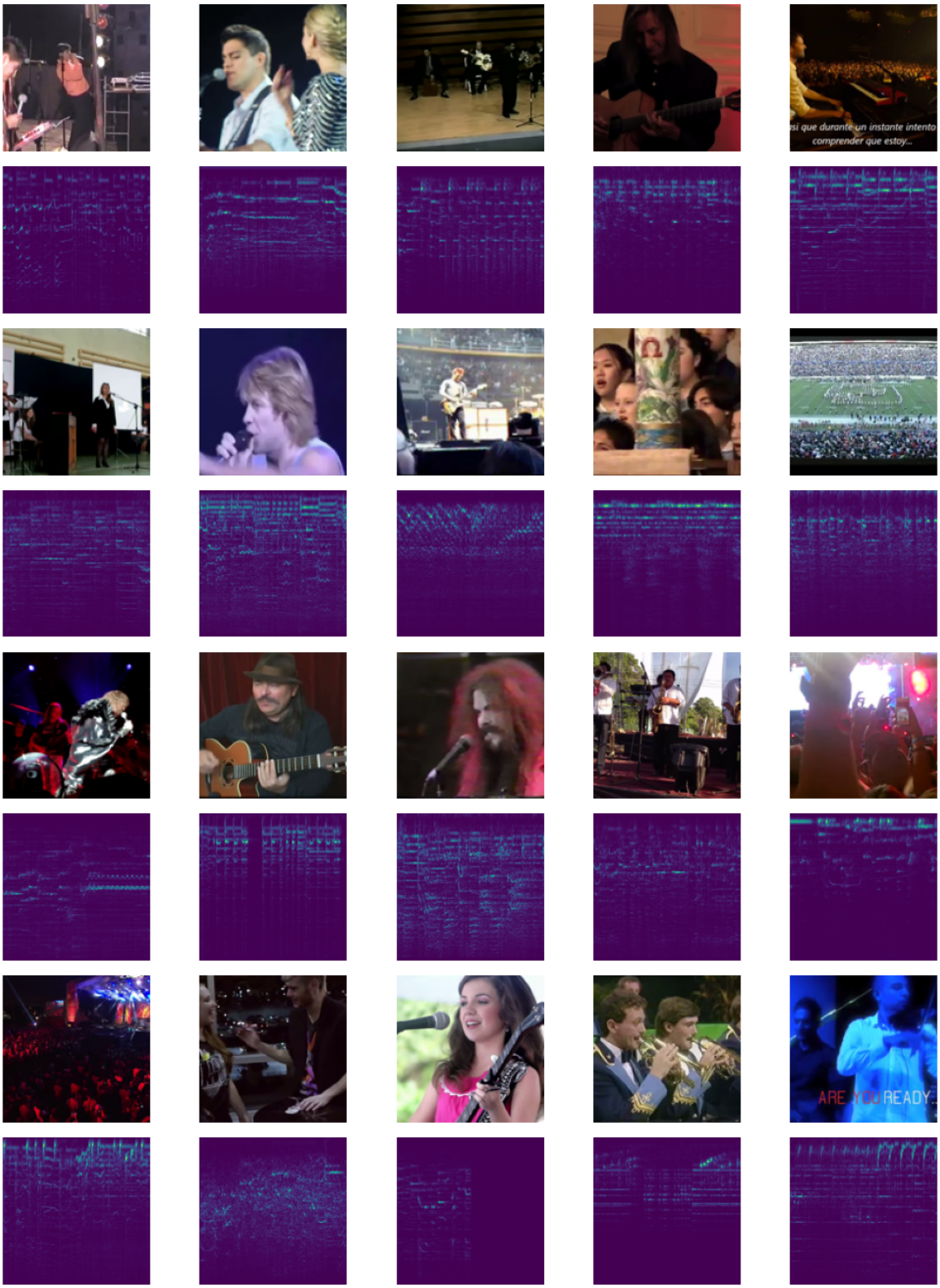


Figure 7: Samples from the MUSIC-200k data set. On top, a randomly selected video frame, as well as the short-time Fourier transform representation of the corresponding audio section.

B Audio-source Localisation Results

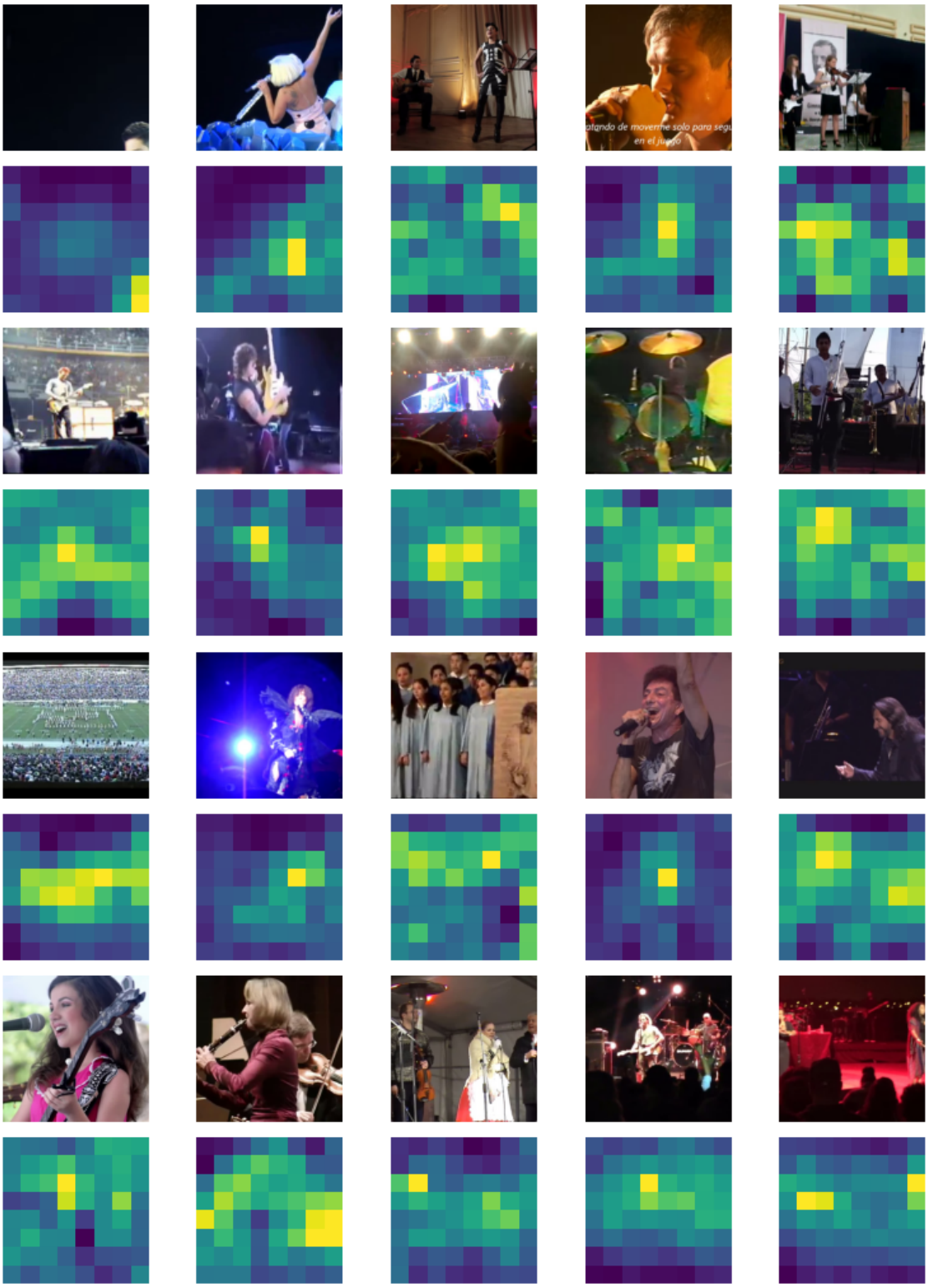


Figure 8: Random selection of audio-source localization maps from G-AVTc with the normalization method from Equation 7. The same processing parameters were used as in row 1 of Figure 6. We take the attentions in the direction from the audio to the visual tokens. We take the attentions of the last encoder layer. We average the attentions over all attention heads. And finally, we group the attentions per visual token to retrieve the source localization maps above.