M2CF: Multi-Modal Contrastive Finetuning

Ruohua Li* **Felicia Luo*** **Rui Wang *** **Yitong Chen †**
{ruohual, zhixinlu, ruiwang3, yitongc}@andrew.cmu.edu

Abstract

In this project, we worked on the zero-shot audio-visual action recognition task. We conducted a detailed comparative analysis of multimodal approaches within the domain of human action recognition, focusing on the integration and synchronization of video and audio inputs, as well as their coordination with text labels. We introduce a specialized multimodal framework, Multi-Modal Contrastive Fusion and Finetuning (M2CF), designed to simultaneously improve unimodal encoders as well as inter-modal interactions and coordination. We utilized Low-Rank Adaptation finetuning on the image encoder to effectively leverage a large pretrained model while tailoring it to our specific task. A Cross-modal transformer selected meaningful auditory and visual information and supplemented one with another. Furthermore, a prompt learner dynamically adjusted the text representation to be better coordinated with the fused audio-visual representation. Together, our proposed framework demonstrated a substantial improvement upon unimodal and multimodal baselines in the benchmarking datasets UCF-101. It not only achieved outstanding performance in recognizing actions from seen data but also exhibited exceptional generalization abilities with unseen data. Our M2CFv2 outperformed the State-Of-The-Art ViFi-CLIP on action recognition by **7.1%** as measured in harmonic mean. Through this work, we demonstrated the potential of multimodal learning to transform the field of human action recognition, exploiting all information from existing data.

1 Introduction and Problem Definition

Human action recognition is a field focused on analyzing sequences of body movements captured through various media, defining each action within a spatio-temporal context(Pham et al., 2022). This process typically involves examining a series of static images to identify and label the actions of individuals depicted within. These images serve as complex visual signals that models must interpret to assign accurate action labels. Historically, unimodal models, which rely on a single type of data input, have been utilized to achieve notable success on several public datasets. Such models are adept at processing and learning from vast amounts of visual data to recognize and predict human actions effectively.

Despite the advancements in unimodal systems(Wang et al., 2023a; Qian et al., 2021; Wang et al., 2023b), there remains significant untapped potential in integrating additional modalities such as audio and text. These elements can provide context and nuances that purely visual data might miss, particularly in challenging scenarios like zero-shot or few-shot learning conditions. In these situations, models are required to correctly recognize actions or concepts that they have not been explicitly trained on, using very limited examples. Audio modality can enrich the feature representation of the actions, offering deeper insights that enhance the model's ability to distinguish features that are visually inseparable. And text modality allows image representation to be associated with natural language, enabling the models to have an alignment between visual and lexical, through which they can generalize from known actions to novel ones.

To address these limitations and improve the robustness and accuracy of action recognition systems, we propose the development of a multimodal framework. By integrating text, audio, and video inputs, this framework aims to leverage the strengths of each modality to provide a more comprehensive understanding of the actions. This holistic approach is expected to significantly enhance per-

*Equal Contribution – Alphabetical order
†Different Contribution

formance in human action recognition tasks, especially in more challenging scenarios where the capability to adapt to new, unseen actions is crucial. Our objective is to create a system that not only performs well on standard benchmarks but also excels in unseen data where diverse and complex actions are present.

In summary, in this project, we hypothesize that

- Without understanding language, it would be extremely difficult for a visual model to predict an unseen label. Therefore, we are using text labels as an additional modality. We assume that we could perform a zero-shot classification task on a dataset with classes that had not been seen during pretraining by utilizing a text encoder.
- Only a smaller network is necessary to coordinate unimodal representations outputted from large pretrained video and audio encoders. So that we can leverage the prior knowledge learned by the large models and distill mutual information or align them to a shared space with a small network requiring much fewer resources.
- Humans perceive and interact with their environment through multiple sensory modalities, primarily including sight and sound. This multimodal perception enables a more comprehensive understanding and recognition of the surroundings. Building on this natural human capability, we hypothesize that integrating audio input into systems designed to interpret or interact with the environment could enhance their predictive capabilities and overall accuracy. By incorporating sound, these systems can utilize additional context that may not be visually apparent.

Therefore, this project constructs a framework that effectively integrates multiple modalities—visual, audio, and textual—into a cohesive system for enhanced human action recognition. By leveraging the complementary strengths of these modalities, the framework is designed to improve the accuracy and robustness of action classification. Our framework gains improvement on both seen data and unseen (zero-shot learning) scenarios.

2 Related Work and Background

Human action recognition (Related Datasets) Human action recognition aims to identify and categorize a variety of human movements and analyze sequences of body movements captured through temporally sequenced static images(Pham et al., 2022). Among publicly available datasets, UCF-101(Soomro et al., 2012) stands as a significant dataset extensively utilized in the domain of action recognition research. Comprising 101 distinct action categories, it is divided into five main groups: Human-Object Interaction, Body-Motion Only, Human-Human Interaction, Playing Musical Instruments, and Sports. This dataset is an expanded version of the earlier UCF-50, adding 51 new action classes that feature audio, making it particularly advantageous for studies involving multimodal learning tasks. Our research leverages UCF-101 predominantly because it provides a comprehensive blend of audio-visual data that is essential for our experiments in multimodal frameworks, aiming to enhance the robustness and accuracy of action recognition systems. The dataset also enables us to perform zero-shot and few-shot transfers on unseen labels, which refer to the model's capability to quickly adapt to new action labels that it has never seen or only seen a few samples during training, respectively. Despite its relatively modest size when compared to larger datasets, UCF-101 offers a diverse range of activities that make it an invaluable resource for testing and refining action recognition algorithms. Therefore, we primarily used UCF-101 in our experiments.

In addition, Kinetics-400(Kay et al., 2017a) is so far the largest human activity dataset and can be seen as the successor to the standard benchmark UCF-101. It contains 400 human action classes with 400-1150 clips for each action. This dataset contains more variation since each clip is taken from a different video, whereas the UCF-101 dataset may have 7 clips from the same video. All of the clips are sourced from YouTube videos. Consequently, they are not professionally videoed and edited, meaning there is more noise in the dataset. It is an ideal dataset for pretraining if resources permit. However, considering the computing power available in this project, we only used a subset of Kinetics-400 for the ablation study.

Video and Audio representation learning (Unimodal Baselines) Video and audio representation learning has significantly advanced in recent years driven by the success of deep learning architectures. 2D CNNs have achieved state-of-the-art result for many down-steam tasks (He et al.,

2015; Szegedy et al., 2014). However, it lacks temporal understanding for video-specific tasks. Therefore, video techniques such as 3D-based CNN - I3D(Carreira and Zisserman, 2018) and S3D (Xie et al., 2018) have been pivotal. These models process video by capturing spatial and temporal features, enabling applications in action recognition, video classification, and more. Self-attention(Vaswani et al., 2023) based model has proven competitive if not better on many tasks such as Natural Language Processing (BERT(Devlin et al., 2019), GPT(Brown et al., 2020)), and Computer Vision (ViT(Dosovitskiy et al., 2021), BEiT(Bao et al., 2022), VideoMAE(Tong et al., 2022; Wang et al., 2023a)). Due to architectural consistency, transformers can convert to multi-model design with minimum changes

Similarly, in the realm of audio, deep learning techniques such as CNNs and RNNs have been extensively used to capture the temporal dynamics of audio signals. More recently, the introduction of transformers upon processed audios has furthered the field, allowing for a more nuanced understanding and generation of audio content. Models like the AudioMAE(Huang et al., 2023), leveraging the self-supervised training scheme, demonstrate improved performance in audio representation learning.

Modality Fusion (Prior Work) Modality fusion aims to leverage the complementary information available in different data types to improve the prediction outcomes. Previous studies have shown that a more effective fusion method translates to better performance. Early fusion involves the integration of features at the data or feature level, combining audio and visual signals before the learning process. By doing so, information from individual modalities is preserved as much as possible before the model learns to distill mutual information through fusion. Several works that use this method (Poria et al., 2016; Sun et al., 2018) allow for a more flexible fusion between modalities. On the other hand, late fusion typically involves merging features at or close to decision level (Nojavanasghari et al., 2016), accommodating diverse model architecture for independent unimodal processing.

There have been more intermediate approaches that combine early and late fusion methodologies to capitalize on the strengths of both techniques. Recent works such as Gated Fusion(Arevalo et al., 2017), Tensor Fusion(Zadeh et al., 2017), and low-rank Fusion(Liu et al., 2018) address the limitations of relying solely on one type of fusion and provide flexibility in handling diverse and complex activities.

Most recently, attention mechanisms have been applied to focus selectively on informative parts of the data streams. Many existing works such as LXMERT(Tan and Bansal, 2019a) and ViLBERT(Lu et al., 2019) have focused on utilizing the benefit of cross-modal attention to contextualize two modalities and dynamically weights the importance of different modalities. This approach ensures that the model pays more attention to the modality that is more informative and is selective on the less dominant modality.

Representation Coordination (Prior Work) Representation Coordination focuses on aligning features from different modalities to improve models' performance and generalizability. One of the foundation models, CLIP (Radford et al., 2021), addresses the challenge of representation coordination by learning to associate images with textual descriptions in a shared embedding space. Similar attempts further refine and extend the capabilities of multimodal learning and alignment to video and audio domains. Improved upon CLIP, Video Fine-tuned CLIP (Rasheed et al., 2023) finetunes the image-based model and incorporates temporal information through a simple pooling layer. CLAP (Elizalde et al., 2022) follows a similar structure and approach as CLIP, utilizes a transformer-based audio encoder and GPT-2 text encoder, and learns to maximize the cosine similarity between positive audio embedding and text embeddings and minimize that of negative pairs. To align video and audio modalities, XKD (Sarkar and Etemad, 2022) proposes a teacher-student setup for self-supervised cross-modal knowledge distillation, where one modality is masked and learns from the other unmasked modality through cross-modal attention.

Relevant techniques Prompt learning has emerged as a significant paradigm in natural language processing, allowing for more flexible and efficient use of pre-trained language models. Unlike traditional fine-tuning, which modifies a model's weights, prompt learning involves appending carefully designed prompts to the input, guiding the model to apply its pre-existing knowledge to new tasks. Existing approaches such as prompt tuning (Lester et al., 2021), prefix tuning (Li and Liang,

2021), a simple yet effective mechanism for learning "soft prompts" to condition frozen language models to perform specific downstream tasks, has shown effective in many domain.

Adapter tuning involves inserting small, trainable modules into a pre-trained model, while keeping the majority of the model's parameters frozen. This approach allows for efficient fine-tuning, as only the parameters of the adapters need updating, significantly reducing the computational cost and memory footprint compared to full model fine-tuning. Notable works such as (Houlsby et al., 2019; Hu et al., 2021) underline the potential of adapters not only to conserve resources but also to facilitate more rapid and flexible model adaptation to new domain of down-stream tasks.

3 Task Setup and Data

Our task is zero-shot audio-visual action recognition, namely, using both audio and video inputs to classify actions possibly on classes that are never seen during training. To experiment with baselines, we used the 51 classes of the UCF-101 dataset (Soomro et al., 2012) that contain both audio and video modalities. We followed (Mercea et al., 2022) and split them into a training set, a test-seen set, and a test-unseen set. The split details can be found in Table 1.

Inbalanced Data We analyzed the dataset statistics. Though the number of clips per class was relatively balanced (table 2), we noticed an imbalance distribution of clip lengths as shown in table 3 and 4. Such a phenomenon poses further challenges to the zero-shot classification task as the tuned model could be biased toward classes that had more data.

Weak Text Signal The text modality in our dataset is presented as action class labels. Compared to audio and video modalities, it contains much less information. However, many related works have spent some effort on generating customized prompts for zero-shot image classification (Pratt et al., 2023; Menon and Vondrick, 2022). In HMDB, UCF and Kinetics, Refining text knowledge with Optimal Spatio-Temporal Descriptor (Chen et al., 2023) could achieve exceptional performance on zero-shot/few-shot video recognition. Therefore, enriching the textual information, possibly utilizing LLMs, is necessary to construct a more text-rich environment for enhanced open-vocabulary human action recognition.

From our baseline experiments, we further noticed several issues regarding the audio data in the UCF-101 dataset in which the visual information is dominantly informative, whereas the audio data could be very noisy and possibly irrelevant:

Noise & Misalignment with Typical Audio The audio data for each class does not necessarily align

Dataset	#Classes	Size
Train	42	2.53 GB
Test-Seen	42	274 MB
Test-Unseen	9	853 MB

Table 1: UCF-101 audio-visual dataset split

Table 2: Numbers of clips per class of train (upper), test seen (middle), and test unseen (lower) datasets

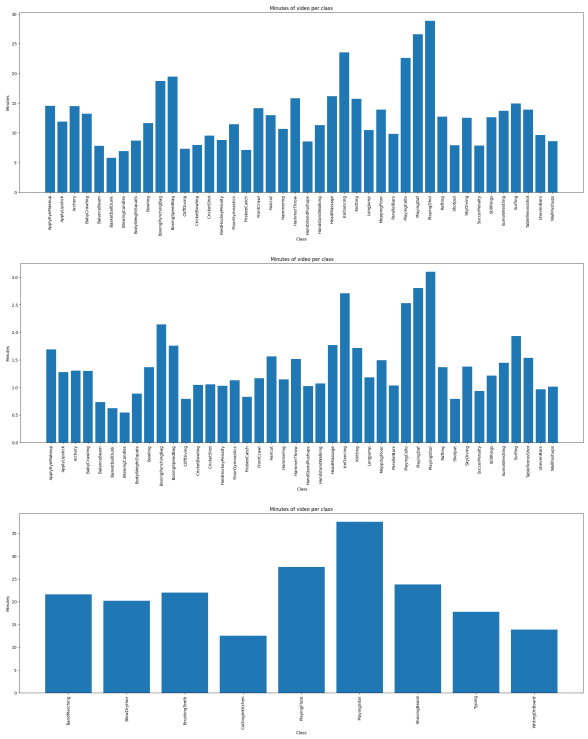


Table 3: Total length of clips per class of train (upper), test seen (middle), and test unseen (lower) datasets

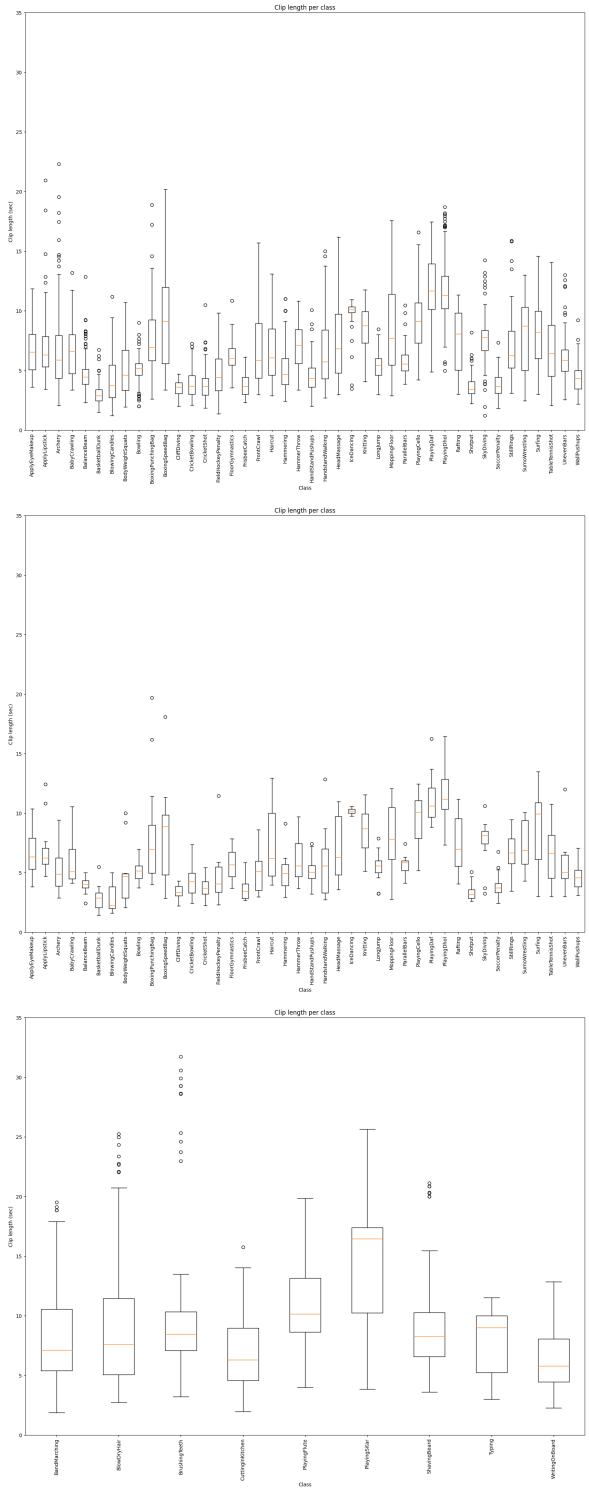


Table 4: Distribution of clip lengths per class of train (upper), test seen (middle), and test unseen (lower) datasets

with typical audio speech/sounds corresponding to such class. Examples include classes like Surfing and ApplyLipstick, in which some of the audios contain only background music without any meaningful information. In Surfing, a lot of audio in the test set contains mainly music, so it is rarely correctly classified. While, due to that Rafting and FrontCrawling classes contain audios that have the sound of "water", CLAP often classifies them into Surfing. These audios might carry information that is useful to classify the audio within the scope of this dataset, but this has down-effects on how we could leverage the prior knowledge from pretrained audio models in our prediction tasks.

Despite the fact that audio is able to provide meaningful information and assist vision to better classify some tasks. Some of the audios within UCF-101 also have a noticeable magnitude of noise in the background, mixed with useful audio information. This noise is a bit of a hindrance to the performance as it decreases the "distance" between audios from different classes. To better show such a problem, we visualized the audio feature representation learned by ResNet-18 shown in Figure 1 where we highlighted the part we suspect the model performed poorly. We can observe that the data from several categories has clustered together while some others fail to form a cluster.

Outstanding Non-Categorical Auditory Features Most audios in the dataset do not have outstanding features for specific classes. For example, many audios in the dataset are dominated by music (ApplyLipstick, Surfing, Haircut, etc.), background noise or silence (MoppingFloor), and so on. From our observations, these features tend to make pretrained models like CLAP to more likely predict all of them into one or two categories. In our experiments, they are usually classified as ApplyEyeMakeup, PlayingDaf or SumoWrestling. Many audios with instruction-style speeches are also classified under ApplyEyeMakeup. We could see that these features, though noticeable, are not highly categorical in this task. We also see this from figure 1. Opposite examples also exist in our task, especially for the "unseen" split which includes 9 classes like ShavingBeard, Typing, BandMarching. A comparison is in table 5 The audios of these classes are pretty useful. But the number of such classes is small.

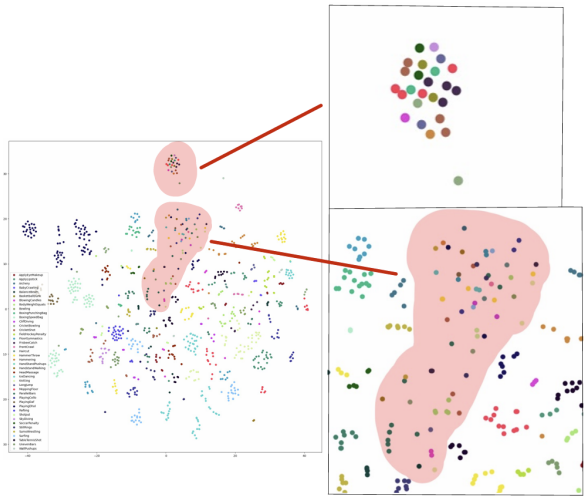


Figure 1: Audio feature representation analysis ResNet-18 is being used and we choose 96 frames per clip (about 4s of audio for each datapoint)

Video Lacks Diversity Although there are about 13,000 videos within UCF-101, it lacks sufficient variation for transformer-based deep learning methods compared with larger datasets such as Kinetics. Some categories simply do not have long enough clips to capture effective representation of temporal information. During our training and testing phase, if we increase our frames per clip to 96 or 120, we sometimes fail to capture any clips from some specific categories simply because that category doesn't have a long enough video clip. Additionally, even though UCF-101 has 13,000 clips, they are all taken from only 2.5k distinct videos (Kay et al., 2017b). For example, there are 7 clips from one video of the same person performing the same activity. This means that UCF-101 has very little variance, which prohibits modal to learning generalized feature representation, and it is difficult to transfer the model trained on UCF-101 to other similar tasks.

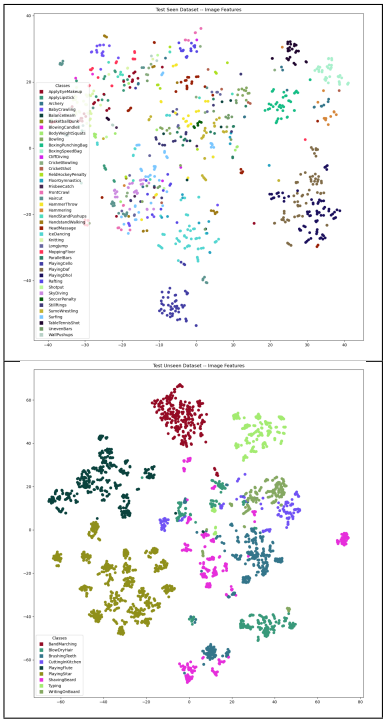


Table 5: t-SNE Comparison of CLAP on seen (upper) or unseen (lower) features. We could see that unseen features are apparently more categorical

4 Baselines

We examined and run the following unimodal and multimodal baselines, with analyses of baseline model results.

4.1 Unimodal Baselines

We include the following unimodal baselines. For abbreviation, we use V for Video, A for Audio, and T for Text.

1. **(V) ResNet-101 on Single-frame**
   As a very simple baseline, we finetune the pretrained ResNet-101 (He et al., 2015) on every single image frame and append a fully connected layer at the end for classification. We compare the prediction of every frame with the label to calculate loss and accuracy. This is meant to establish a baseline of how well an image encoder, solely by itself, could capture the categorical semantic information.

2. **(V) ResNet-18 + Self-attention**
   Working on 16-frame video samples, this model first uses a frozen ResNet-18 (He et al., 2015) image encoder trained on ImageNet](Deng et al., 2009) to obtain image embedding of each frame. Then, the sequence of image embeddings is added by the positional encoding and passed into a two-layer self-attention block to contextualize on other time-frames. The final layers include a pooling layer to average across time and a projection head for classification. With this baseline, we hope to build a temporally informed visual representation from a simple encoder.

3. **(V) S3D Pretrained**
   To explore more advanced unimodal baselines, we adopt an S3D model (Xie et al., 2018) pretrained on Kinetics-400 (Kay et al., 2017a), which is a much larger dataset in the same domain as UCF-101. We only modify the final classification layer and then finetune S3D on our UCF-101 audio-visual training dataset. S3D has shown promising capability in video understanding. We expect this baseline to sufficiently capture the categorical information of the video.

4. **(V) VideoMAE Pretrained**
   Video transformers brought a series of improvements to computer vision tasks. To explore the possibility of state-of-the-art unimodal, we adopt VideoMAE (Tong et al., 2022) with ViT backbone. VideoMAE is pretrained on Kinetics-400 using tube masking and reconstruction objectives in a self-supervised manner. We take the pretrained encoder and place a linear layer after in order to fine-tune on UCF-101.

5. **(A) Self-attention on FBank**
   This baseline first projects the FBank features to an embedding space using two linear layers, and then uses a two-layer self-attention block to capture temporal information. The attended embeddings are lastly averaged across time and passed into a classification head. Similar to 1.1.2, We hope to establish a baseline of temporally informed auditory representation using minimal embedding design.

6. **(A) ResNet-18 on FBank**
   To capture temporal information across spectral features, we adopt ResNet as a naive starting point. This baseline first preprocesses the raw audio waves into FBank features. This practice is simply following (Huang et al., 2023), using torchaudio functions. We manipulate the parameters such that the output Fbank has an appropriate shape (224×224). Then, we put the features into a CNN layer with 1×1 kernels, input channels 1, and output channels 3 to increase the number of channels to 3. At last, we put the features into a ResNet-18 model pretrained on **ImageNet** dataset. The last fully-connected layer is replaced by a linear projection into **number of classes**.

7. **(A) AudioMAE Pretrained**
   AudioMAE (Huang et al., 2023) has been pretrained on AudioSet-2M (Gemmeke et al., 2017) dataset. We freeze the downloaded encoder and add a three-layer MLP after the encoder. In between the Linear layers, we add a BatchNorm layer followed by a ReLU activation. We hope to gauge how helpful is pretraining on the audio modality from this baseline.

4.2 Simple Multimodal Baselines

For abbreviation, we use EF for Early Fusion and LF for Late Fusion.

1. **(A+V)(EF) Self + Cross-modal Attention**
   Inspired by LXMERT (Tan and Bansal,

2019b), we include a baseline with a cross-modal transformer(Vaswani et al., 2023) to connect auditory and visual semantics. It builds upon unimodal baselines: after the unimodal self-attention layers, the input of each modality is fed into a two-layer cross-attention block that takes the current modality as query and the other as key and value. In all attention layers, residual connections ensure the improvement of the representation. We use the cross-modal representation for final temporal pooling and classification. We test two variants: one using the auditorily-contextualized representation of video for the final layers, and the other using visually-contextualized representation of audio. Together, we expect the two modalities to supplement each other and enhance mutual information.

2. **(A+V)(LF) S3D Pretrained + ResNet-18 on FBank**
   Since S3D with pretrained weight and ResNet-18 perform reasonably well on respective modalities, we include a baseline with different methods of late fusions trying to connect between video and audio modalities. Before fusion, we take Kinetics-400-pretrained S3D and UCF-101-finetuned Resnet-18, and we remove the last classification layers. We freeze S3D and Resnet-18 encoder during fusion training and project visual/audio embedding into a pre-defined dimensions. Therefore, assuming we have ideal encoders for audio and visual, we are only training fusion layers and projection layers. To experiment with different setups and possibilities, we apply three different types of fusions: **Summation**, **Concatenation**, **Gated Fusion**, **Tensor Fusion**(Zadeh et al., 2017), and **Low Rank Fusion**(Liu et al., 2018). Details of fusion strategies can be found in section 4.4.

4.3 Competitive Baselines

1. **(V+T) CLIP Pretrained**
   CLIP (Radford et al., 2021) has demonstrated its superiority in learning visual representation that's semantically aligned with text. Such is especially useful in zero-shot learning when the model has never seen some test classes during training, and CLIP chooses the classes whose text embedding is the most similar to the image embedding. Since CLIP has been pretrained on over 400 million image and text pairs, we expect it being able to generalize to action recognition domain. We test CLIP with ResNet-101 backbone and compare its prediction with the label of every single frame to evaluate the accuracy.

2. **(V+T) Video Finetuned CLIP**
   A SOTA model on zero-shot action recognition is ViFi-CLIP (Rasheed et al., 2023). Their strategy is simple yet effective: fine-tuning image-level CLIP with the video frames. Then a temporal pooling layer aggregate temporal information of the image-level embeddings. To work in low-data regimes, ViFi-CLIP also proposes to learn small-sized prompts on both the video and text branches and freeze the encoder to maintain generalizability while contextualize on the specific downstream task. We use a ViFi-CLIP model pretrained on Kinetics-400, and directly evaluate it on UCF-101.

3. **(A+T) CLAP**
   (Elizalde et al., 2022) follows a similar structure and approach as CLIP (Radford et al., 2021) but applied on Audio and Language modalities. The audio encoder is an audio transformer HT-SAT (Chen et al., 2022) trained on 22 tasks. The text encoder is an adapted GPT-2 (Radford et al., 2019) decoder, adapted by appending a special token at the end of each input text to produce sentence-level representations, then finetuned. The intuition of this task is that besides visual and language pairs of information, we also want to test the information of audio-text pair on a strong pretrained model.

4.4 Fusion Strategies

We have explore several fusion strategies using S3D pretrained on video and Resnet-18 on Audio FBank.

1. **Summation**: By assuming that the features from different modalities are aligned and of the same dimension, summation fusion is one of the simplest which utilizes element-wise addition of features. It is the most computationally efficient and works well if the assumption is met.

2. **Concatenation**: Concatenation fusion stacks the feature vectors from different modalities

end-to-end to create a longer feature vector. By using concatenation, we hope to retain all original features, allowing subsequent layers to learn how to best integrate the information from different modalities.

3. **Gated Shift Fusion**: This idea is a self-proposed fusion strategy inspired by Lecture 3.1 and the observation that audio generally gives weaker information than video/image. In the module, we learn a "gate" parameter on which we will apply tanh to make sure the values is ranged from -1 to 1. Then, we multiply the projected audio representation by the gate as a shift added to the projected video/image representations.

4. **Gated Fusion**: This idea is purposed by (Arevalo et al., 2017). Gated fusion is to use a gating mechanism that learns to assign weights to different modalities, indicating their importance or relevance. This "attention" on different modalities allows the model to dynamically adjust the influence and adaptively prioritize the most informative modality.

5. **Tensor Fusion**: Inspired by (Zadeh et al., 2017), we implement our own tensor fusion by creating a higher-order representation by computing the outer product of feature vectors from different modalities. In theory, tensor fusion could capture all possible interactions between elements across modalities.

6. **Low Rank Fusion**: Inspired and originated from (Liu et al., 2018), low rank fusion addresses the dimensionality issue of tensor fusion by approximating the high-dimensional tensor with a lower dimensional representation which utilizes matrix factorization. By balancing between capturing detailed intermodal interaction and maintaining computational efficiency, Low Rank Fusion would set a benchmark for multimodal fusion while being easier to train and optimize.

5 Proposed Model

Our proposed model takes in three modalities: Video, Audio, and Text. In our design, we have frozen Image Encoder (CLIP ViT-L/14), Audio Encoder(ViT-B), and Text Encoder(63M-Transformer) namely $E_v(\cdot)$, $E_a(\cdot)$ and $E_t(\cdot)$. Given dataset from UCF-101, we have video $V_i \in \mathbb{R}^{T\times H\times W\times C}$, audio $A_i \in \mathbb{R}^{C\times T}$ where $C \in \{1, 2\}$ is the number of sound tracks, and text label T_i. In summary, our model consists of two sections:

- Fusion between Video and Audio modality using cross-modality attention
- Coordination between Text Embedding and Fused Embedding.

Therefore, we call our model Multi-Modal Contrastive Fusion and Finetuning, or M2CF. Our framework has several variants as listed in table 6. One advantage of our framework is its versatility in applying audio modality depending on the data.

Fusion between Video and Audio Video embedding is generated from Vision Transformer (ViT-B/32, ViT-L/14). We decided to use CLIP's pre-trained checkpoint because its image embedding has been trained to coordinate with the text embedding. However, if we have sufficient data to train for the coordination, the VideoMAE (ViT-B/32) checkpoint is also available. Since CLIP was trained on diverse image-text pairs sourced from the internet, but UCF-101 contains only labeled human action videos, there is a domain gap between pre-training data and our training and testing datasets. Therefore, we decided to add Low-rank adaptation, which allows the model to adapt to our downstream tasks of human action recognition. We used the CLIP image encoder to encode each frame of the videos. So, the video embedding can be represented as:

$$e_v^{(i)}[t] = E_v\left(V_i[t]\right) + LoRA\left(V_i[t]\right) \quad \forall t \in [1, T]$$

Then, we could choose to either keep the temporal information or perform temporal pooling, which pools the embeddings from the temporal dimension into one embedding vector.

$$e_v^{(i)} = mean(e_v^{(i)}[1], e_v^{(i)}[2], \ldots, e_v^{(i)}[T])$$

For **audio embeddings**, we could generate from msCLAP (Elizalde et al., 2022) or AudioMAE (Huang et al., 2023). Both encoders have transformer architectures and patchify the preprocessed audio as inputs. For both encoders, we could choose to either keep all output embeddings separately or perform average pooling to gain a single vector (for msCLAP we changed their source code). We did not add LoRA to the audio encoder due to the low quality of our audio data and the computation resource limitation. As a result, our audio embedding is defined as:

$$e_a^{(i)}[n] = E_a\left(A_i[n]\right) \quad \forall n \in [1, N]$$
$$e_a^{(i)} = mean(e_a^{(i)}[1], e_a^{(i)}[2], \ldots, e_a^{(i)}[N])$$
if use average pooling

where N is the number of patches.

Finally, we perform a late fusion upon the Audio-Visual features. The fusion strategy is **Cross-Modality Attention** that takes one modality as input and use another modality as condition. To reasonably utilize this cross-attention mechanism, we choose to keep temporal information of video embeddings and all embedding vectors from audios. The cross-modality attention outputs two results:

$$e_{av} = cross\_attn\left(e_a, e_v\right)$$
$$e_{va} = cross\_attn\left(e_v, e_a\right)$$

where e_{av} represents the audio feature contextualized on video and e_{va} represents the video feature conditioned on audio. Both are usable and give reasonable results on seen datasets. We choose to use e_{va} because the pre-training data of the visual encoder is more similar to our visual training data. Whereas the pre-training audio dataset is clear environmental sound, our audio data consists of background noise, human speech, etc. There is a domain gap between our audio data and the audio encoder domain, whether it is msCLAP or AudioMAE.

M2CFv1	(V) ViT-L/14 (unfreeze last layer) (T) Text Encoder
M2CFv2	(V) ViT-L/14 + LoRA (T) Text Encoder + Prompt Learning
M2CFv3	(V) ViT-L/14 + LoRA (A) ViT-B (AudioMAE pretrained) (T) Text Encoder + Prompt Learning

Table 6: M2CF varients
*Vision and Text are using CLIP pretrained weights

Textual Representation and Coordination Admittedly, the lexical information available from the UCF-101 dataset is relatively limited compared to the auditory and visual information. Consequently, to enable the text encoder to effectively adapt to our specific downstream tasks without sacrificing its overall generalization capabilities, we have implemented a specialized strategy known as prompt learning. Essentially, this means that we prepend a carefully designed prefix to the input data, which helps guide the text encoder in generating context-appropriate responses. This prefix is designed to be both task-specific and informative, thereby allowing the encoder to maintain its performance across a variety of tasks while being particularly effective for the challenges presented by the limited lexical range of the UCF-101 dataset. We define our prompt learner θ_L which is essentially a learnable embedding prepend to our text label. Our text embedding can be defined as

$$e_t = E_t\left(\text{concat}\left(\theta_L, \text{tokenized}\left(T_i\right)\right)\right)$$

In contrast to the conventional CLIP approach in classification tasks that always put the text label into a template of "a video of <category>", tokenize, and then pass it into the text encoder, the prompt learner is initialized with the template "a video of <category>" or "X X X X X X X X X <category>" depending on the desired sequence length. Then, it learns how the initial tokenized embeddings change differently for different class labels to better fit the dataset. After encoding all the labels representation $e_t \in \mathbb{R}^{51\times768}$ and fused representation $e_{av} \in \mathbb{R}^{B\times768}$, we could coordinate (align) them via the cross-entropy objective.

5.1 Loss functions

Due to the fact that this is a classification task, and we only have 51 different texts, the conventional contrastive (InfoNCE) loss used in CLIP is not applicable in this case. Instead, we use Cross Entropy Loss as the primary objective function. For a batch of videos, audios, and labels, we have fused embedding e_{av} and text embedding e_t, respectively. Following a similar training scheme as ViFi-CLIP, the cosine similarities $sim(\cdot)$ between all the video-audio embeddings e_{av} and the corresponding text embeddings e_t is maximized to align two representations via cross-entropy objective.

$$\mathcal{L}_{CE} = -\frac{1}{B}\sum_{i=1}^{B}\log\left(\frac{\exp(\text{sim}(e_{av}^{(i)}, e_t^{(C_i)}))}{\sum_{j=1}^{C}\exp(\text{sim}(e_{av}^{(i)}, e_t^{(j)}))}\right)$$

- B: batch size.
- C: the number of classes.
- e_{av}: fused embedding
- e_t: text embedding
- C_i: index of class label of sample i

We have also experimented with several auxiliary loss functions. We use a Symmetric Cross Entropy in the same spirit as the contrastive loss used in CLIP to maximize the similarity between positive textual and audio-visual pairs and minimize that of negative pairs. However, as one textual representation may be the positive pair of several fused audio-visual representations in one batch, we compute the Binary Cross Entropy loss from the text dimension after calculating the cosine similarities:

$$\mathcal{L}_t = -\frac{1}{C}\frac{1}{B}\sum_{j=1}^{C}\sum_{i=1}^{B}[y_i\log(\hat{y_i}) + (1-y_i)\log(1-\hat{y_i})]$$

- $\hat{y_i} = \text{sim}(e_{av}^{(i)}, e_t^{(j)})$
- $y_i = 1$ if $(e_{av}^{(i)}, e_t^{(j)})$ is positive pair, and 0 otherwise

Therefore, the Symmetric Cross Entropy loss can be defined as:

$$\mathcal{L}_{SYM} = \frac{1}{2}\mathcal{L}_{CE} + \frac{1}{2}\mathcal{L}_t$$

In addition, we add Cosine Embedding loss to directly measure the cosine of the angle between the fused embedding vector and the text embedding, enhancing our objective to align positive pairs and separate dissimilar pairs. This loss can be defined as:

$$\mathcal{L}_{cos} = -\frac{1}{C}\frac{1}{B}\sum_{j=1}^{C}\sum_{i=1}^{B}\mathcal{L}_{cos}^{(ij)}$$

$$\mathcal{L}_{cos}^{(ij)} = \begin{cases} 1 - \text{sim}(e_{av}^{(i)}, e_t^{(j)}), & \text{if positive pair} \\ \max(0, \text{sim}(e_{av}^{(i)}, e_t^{(j)})), & \text{if negative pair} \end{cases}$$

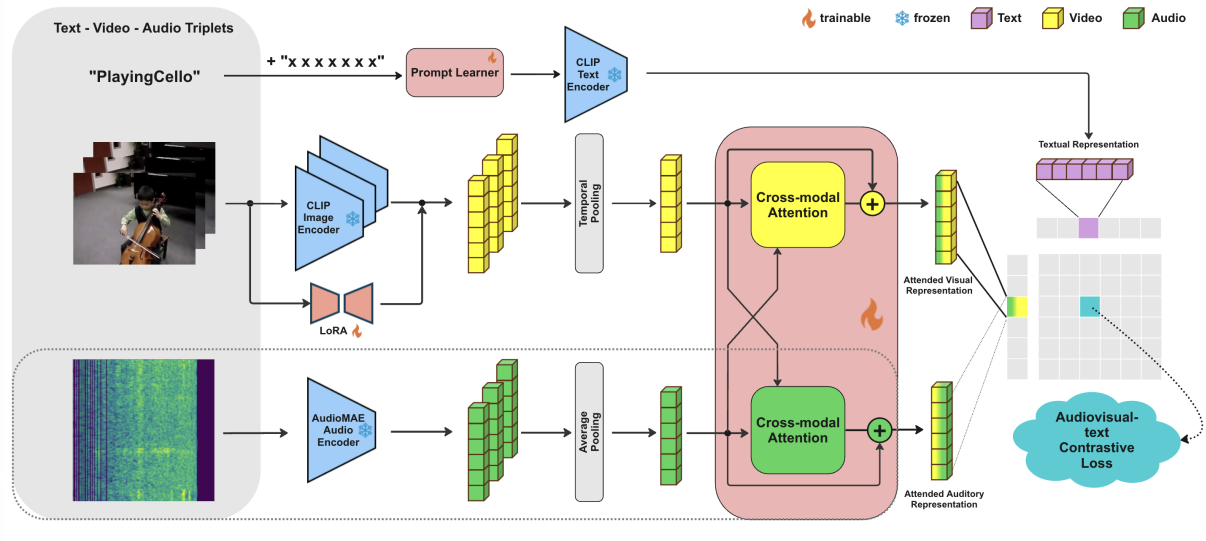


Figure 2: Proposed model architecture in training

We construct a composite loss from the Symmetric Cross Entropy and Cosine Embedding losses as:

$$\mathcal{L}_{COMP} = \frac{1}{2}\mathcal{L}_{SYM} + \frac{1}{2}\mathcal{L}_{cos}$$

We used a small network to ablate on the loss functions. We hypothesized that the composite loss would provide stronger supervision on the network and, therefore, could result in better accuracy and faster convergence. However, as shown in table 7, there was no significant difference between using CE and composite loss functions. We conjectured that this was because the Cosine Embedding loss and the Cross Entropy loss share the same goal to pull the positive pair closer and push the negative pairs further. However, using only the symmetric cross entropy only obtained half unseen accuracy, compared to the other two, suggesting that the model could not generalize well.

Besides, following the suggestions by Mark Chen (from team **CVPR2025_BestPaper**) we have also tried to use the Euclidean distance between audio-visual features and the textual features as loss. However, we need to note that if we are only using Euclidean distance as loss, then the similarity function should also be the Euclidean distance. Otherwise, the loss function is not reasonable. We tried two forms, the first is to use Euclidean distance as the similarity:

$$\mathcal{L}_{Euc1} = -\frac{1}{B}\sum_{i=1}^{B}\log\left(\frac{\exp(\|e_{av}^{(i)} - e_t^{(C_i)}\|)}{\sum_{j=1}^{C}\exp(\|e_{av}^{(i)}, e_t^{(j)}\|)}\right)$$

The second is to directly optimize over Euclidean distance:

$$\mathcal{L}_{Euc2} = \frac{1}{B}\sum_{i=1}^{B}\Bigg(\eta * \|e_{av}^{(i)} - e_t^{(C_i)}\| - \lambda\sum_{j\neq C_i}^{C}\|e_{av}^{(i)}, e_t^{(j)}\|\Bigg)$$

η and λ are hyperparameters that control the weights of positive and negative samples. Unfortunately we observe very poor performance using this objective. We believe the reason is that although cosine similarity relies mainly on the "angle" of embedding vector while ignoring Euclidean distance, it is reasonable when the embedding is high-dimensional as similar angles would be almost impossible. And Euclidean distance is too strong an assumption, so the model cannot learn well.

5.2 Changes to training data

For **visual data**, when loading the video dataset, we uniformly sub-sample the video clip into 16 frames per clip regardless of the number of input frames due to computational and implementation concerns.

Loss	Top-1 Accuracy% ↑ Seen	Unseen	HM
CE	70.9	71.28	71.09
Symmetric CE	65.91	37.98	48.19
Composite	71.33	70.36	70.84

Table 7: All using ViT-B/32 CLIP image encoder fine-tuned by LoRA, AudioMAE audio encoder, and one cross-modal attention layer, batch size 16, trained for 5 epochs.

For **audio data**, we used functions from *torchaudio.compliance.kaldi* to transform raw audio data into FBank. For different models and requirements, we use the parameters *num_mel_bins*, *frame_shit*, and *frame_length* to control the shape of the transformed audio.

For **text data**, we used templates filled with class name when no prompt learner is used. The templates are:

Modality	Template
Video	This is a video of ⟨class name⟩
Image	This is an image of ⟨class name⟩
Audio	We can hear ⟨class name⟩

Table 8: Modality and text templates

5.3 Hyperparameters and their effects

Frames per Clip When loading the video dataset, "frames per clip" decides the number of frames in a clip. It doesn't impact the visual encoder significantly because of the uniform subsampling. However, the audio clip is still determined by "frames per clip" before being fed into the FBank prepossessing. In this case, we can treat this hyperparameter as "audio length". During our analysis, we noticed a significant difference in model performance when we adjusted this hyperparameter. Therefore, we performed a small ablation study where we took a simple audio encoder, ResNet-18, apart from the rest of the model for analysis. In table 11, we tested the seen dataset on 50 frames (left), 96 frames (middle), and 120 frames (right) on their t-SNE visualization, respectively. When the audio length is 50 frames, the audio ResNet can hardly capture any meaningful representation, and the t-SNE can hardly form any clusters. This could also be reflected in test accuracy, which it can only achieve around 10%. However, if we increase the "frames per clip" to 96 frames, the embedding visualization where data points are in the same category starts to cluster together with around 60% testing accuracy. When we increase to 120 frames, each cluster is decently well separated apart from each other (75+% acc). This result shows that we need long enough audio information in order for the audio encoder to capture meaningful representation. However, some videos from UCF-101 do not have long enough audio, nor do they contain meaningful information, which we will discuss further in the qualitative analysis section.

Batch Size Batch size plays a critical role in contrastive learning, where it directly decides how many negative pairs to be deassimilated with the positive pair. A larger batch size provides a greater variety of negative samples, which is crucial for learning robust coordination between the audio-visual representation and the textual representation. The variety helps the model generalize better to unseen data by learning more about what distinguishes different classes from each other. In addition, batch size affects the stability of gradient estimates. Larger batch sizes generally lead to more stable and reliable gradient estimates because they average out the noise across more examples. In our ablation studies, we noticed significantly higher unseen accuracy with a larger batch size and, therefore, higher harmonic mean. It indicated that the model was better generalized and less overfitted to the training data. This is the key for an open-vocabulary model to succeed in the zero-shot task. However, a larger batch size also means more GPU memory is required. Therefore, we choose the greatest batch size possible in our final model training.

Batch Size	Top-1 Accuracy% ↑ Seen	Unseen	HM
8	73.18	7.06	12.88
16	70.61	32.69	44.69
32	62.62	58.62	60.55

Table 9: All using ViT-B/32 CLIP image encoder fine-tuned by LoRA, AudioMAE audio encoder, and one cross-modal attention layer, symmetric loss objective, and trained for 3125 steps.

Number of Cross-modal Attention Layers The number of cross-modal attention layers indicates the complexity of the fusion network. More layers are more capable of learning and fusing informa-

tion from both modalities and result in higher accuracy. However, too complex a fusion network may be prone to overfitting as it can also learn from the noise and features specific to the training data. This could have a negative effect in a zero-shot setting, where the goal is not only to achieve good performance in the seen classes but also to generalize well to unseen classes. As we found in the ablation studies shown in table 10, the more cross-modal attention layers, the higher the seen accuracy but the lower the unseen accuracy. Such indicated that the model had learned biasing the seen classes and was less generalizable to the unseen classes.

Cross-modal Attention Layers	Top-1 Accuracy% ↑ Seen	Unseen	HM
1	70.9	71.28	71.09
2	71.61	63.83	67.50
3	73.18	58.93	65.29

Table 10: All using ViT-B/32 CLIP image encoder fine-tuned by LoRA, AudioMAE audio encoder, and batch size 16, trained for 5 epochs.

Learning Rate During the training process, we have to use learning rate that is very small, as small as $2 * 10^{-6}$ to gain reasonable improvement. On the other hand, when using learning rate as large as 10^{-4}, the performance degrades a lot during training process and we completely lose generalizability to unseen test datasets. We think there are two reasons:

1. We are using Adam optimizer. And Adam itself requires relatively small learning rates.

2. This is the most important reason. We are using pretrained large models as encoders. They have been pre-aligned and converged on large datasets. As a result, when fine-tuning we should find more fine-grained features to improve and adjust the output features in very small scale. So, a very small learning rate is better, while standard learning rate leads to the model to deviate from the pretrained embedding space.

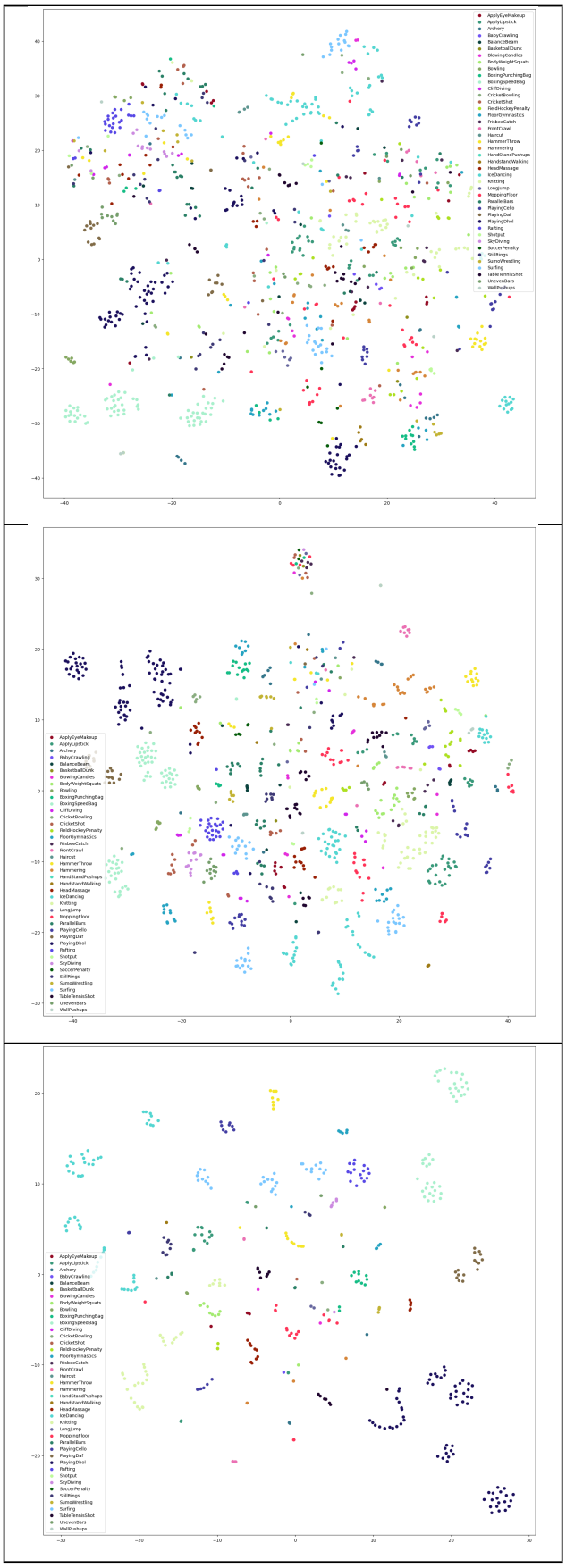


Table 11: t-SNE Comparison of audio ResNet-18 on between different "frames per clip" setup.
Top: 50 frames; Middle: 96 frames; Bottom: 120 frames

6 Results

All the result including baseline and proposed method has been included in the table 12. We report the seen and unseen accuracy, the harmonic mean on all the baselines and proposed models.

6.1 Extrinsic Metrics Used

Accuracy is a common metric used in any classification task. Defined as the percentage of the number of correct predictions of the total number of predictions, it provides the most simple and straightforward evaluation of how well a model predicts the class label. In our experiments, we reported the accuracy of the seen(S) and unseen(U) classes separately.

Harmonic Mean (HM) defined as $\frac{2US}{U+S}$, has been used in many studies on zero-shot and few-shot classification (Mercea et al., 2022; Qian et al., 2022; Rasheed et al., 2023). It provides a robust measure of a model's generalized performance across different datasets. The harmonic mean is particularly significant in the zero-shot setting because it penalizes the model heavily if there is a large discrepancy between the seen and unseen accuracy. This means that a model must perform reasonably well on both seen and unseen classes to achieve a high harmonic mean score.

6.2 Results Analysis

Unimodal Baselines Generally speaking, unimodal baselines using audio inputs performed noticeably worse than those using video inputs, even if the audio baselines had been pretrained on large datasets. On the vision side, we have tested 2D CNN, 3D CNN, and transformer-based approaches. Although we cannot reason unseen performance, the seen accuracy improves as the vision encoder progressively becomes more complicated. We observed that S3D and ViT-based VideoMAE already achieved very competitive results, indicating that video modality is a relatively strong modality. On the audio side, we noticed that CNN-based and transformer-based models achieved similar results. Though neither result from audio is competitive, this indicates the inherent limitation of the data: the audio was noisy, and some of them had very weak information. For example, videos like "Mopping Floor" included human voice, which was not representative nor predictive of the human action that lay behind this audio clip. This could account for the worse performance of unimodal approaches on audio. However, some of the classes do have audio with predictive power. Like the "Apply Eye Makeup" videos containing people illustrating how to do eye makeup.

Fusion Strategy In general, from table 22, we notice that fusion on Vision and Audio achieves better accuracy than Vision or Audio modality alone. Between different fusion strategies, we find that Summation and Concatenation showed promising results among other strategies. Concatenation was better than Summation fusion, showing that visual and audio do not necessarily reside in the same dimension. However, Tensor Fusion and Gated Fusion, which were supposed to capture more fusion pattern, perform worst by slight margins. This could be attribute to the complexity of these two model where both showed signs of overfitting during training. Low Rank Fusion was able to achieve best result because it balanced between efficiency and capturing all possible interaction between elements.

Early Fusion vs. Late Fusion We observed that early fusion did not get competitive accuracy. We speculated that early fusion mechanism fuses data input together and increased the complexity of feature space. The model requires a large amount of training data to better generalize. Late fusion models, on the other hand, achieved impressive performance. We speculate that late fusion methods leveraged pretrained models that had established a good knowledge base on modality-specific understanding.

Zero-shot Performance We tested open-vocaluary methods, CLIP and its descendants, that built text embeddings from the label names and used the label with highest similarity to the visual/auditory embeddings as the prediction. We found a general trend that the test accuracies on unseen classes were greater than the accuracies on seen classes. This is expected as such conclusion has been mentioned in Table 5. However, such result is also encouraging proving that CLIP, as a text-image foundation model, as a strong generalization performance on human action recognition.

Proposed Models In our study, the proposed methods, specifically M2CF, demonstrated superior performance on the Seen dataset. Although M2CF did not outperform ViFi-CLIP on the Un-

Methods	Top-1 Accuracy% ↑ Seen	Unseen	Harmonic Mean
Unimodal Baselines			
(V) ResNet-101 on Single-frame	80	0	0
(V) ResNet-18 + Self-attention	76	2.2	4.28
(V) S3D Pretrained	89	0	0
(V) VideoMAE Pretrained	96.8	0	0
(A) Self-attention on FBank	19.6	2.8	4.9
(A) ResNet-18 on FBank	72	0	0
(A) Frozen AudioMAE + MLP	68	0	0
Multimodal Baselines			
(A+V)(EF) Self + Cross-modal Attention	77.3	1.6	3.14
(A+V)(LF) S3D Pretrained + ResNet-18 on FBank	94	0	0
Competitive Baselines			
(V+T) CLIP Pretrained	56	77	64.84
(V+T) Video Finetuned CLIP Pretrained	79.8	**99.7**	88.64
(A+T) CLAP Pretrained	23	64	33.84
Proposed Models			
(V+T) M2CFv1	98	59.68	74.18
(V+T) M2CFv2	**99.86**	90.47	**94.93**
(V+A+T) M2CFv3	97.57	88	92.54

Table 12: Results from baselines and proposed models.

seen dataset, it remained highly competitive among other approaches. A key strength of M2CF is its ability to effectively fit the Seen dataset while maintaining robust performance on the Unseen dataset. Consequently, our enhanced model, M2CFv2, achieved the best results when assessed using the Harmonic Mean. It is important to acknowledge that there was a slight performance decline after integrating the audio module into our model (M2CFv3). Despite this, the Harmonic Mean results were still exceptional, showing significant improvement over the baseline CLIP model.

7 Analysis

***All data are being recorded and plotted in the Appendix section due to formatting.**

7.1 Intrinsic Metrics Used

t-distributed stochastic neighbor embedding (t-SNE) t-SNE algorithm (van der Maaten and Hinton, 2008) is designed to visualize high-dimensional data. The goal of t-SNE is to take a high-dimensional dataset and reduce it to a lower-dimensional space to facilitate visualization while preserving the relative distances between points as much as possible. For our baselines, we use t-SNE visualization to understand how well the video and audio data of the seen and unseen classes were encoded. Plotting the video and audio representations using t-SNE, we can tell if the learned representations are well-clustered by the true labels or if any class is mixed with what other classes. In addition, by comparing visualizations between seen and unseen datasets using the same model, we can extrapolate the reason behind poor unseen performance. Comparing unimodal and multi-modal baselines that share the same components shows how additional modality improves or deteriorates the representation.

Confusion Matrix A Confusion Matrix is a tabular representation that allows us to understand the accuracy of prediction and, more importantly, to see the types of errors the model is making. By using the confusion matrix, we hope to identify which particular two classes our model is confusing and which classes can be improved by adding another modality. It will give us a general insight into where the model is performing well and where it is not.

Per-class accuracy, Confidence, and Cosine Similarity We use per-class accuracy to evaluate a model's performance on each individual class within a dataset. It is particularly useful in a scenario where the dataset is imbalanced or when the cost of misclassification varies significantly between classes. It helps us to identify if a model is biased towards certain classes and informs us of how the model makes its decisions.

We also plot per-class confidence to show models' confidence in the true label, which is defined below. We hope to see models highly confident in predicting the correct label.

$$\text{Confidence}_C = \frac{1}{N}\sum_{i=1}^{N}(p_{ic}) \tag{1}$$

- N is the number of samples with true label C
- p_{ic} refers to the probability that the model predicted class C for sample i

For models using the contrastive learning paradigm, we visualize the per-class average cosine similarity between embeddings from the video or audio input and the true text label. This can be understood as the confidence these models have in the true label. Similar to confidence, we also hope to see models have higher cosine similarity between embeddings from different modalities of the true pair.

$$\text{Cosine\_Similarity}_C = \frac{1}{N}\sum_{i=1}^{N}\frac{u \cdot v}{\|u\|\|v\|} \tag{2}$$

- u represents audio/video input embedding
- v represents true text label embedding

7.2 Insights from Intrinsic Metrics

Poor Zero-shot Performance using Classification Head From the t-SNE visualizations, we noticed unseen representations learned by simple baselines that do not involve text modality were actually clustered. Though not well separated, the learned representations should be clustered enough for a low-accuracy prediction. However, the evaluated unseen accuracies on these baselines were as low as less than 3% as in table 12, far below what we would expect. We speculated the issue happened at the classification head, which was a fully connected layer to project learned representation to the final predicted probabilities. During training, the models were trained on 42 seen classes, and the classification head learned the probability distribution over these 42 classes. However, during testing on unseen classes, the models were asked to predict the probability of the 9 seen classes using weights learned from the 42 classes. This inconsistency made it nearly impossible for these baselines to predict the correct label.

To address the generalization issue, we chose to avoid using a fully connected layer after each encoder. This approach helps prevent catastrophic forgetting in our foundation model. Hence, we have tried to apply two scenarios:

1. M2CFv1: Freeze the entire encoder layer except the last transformer layer.

2. M2CFv2: Freeze the entire encoder layer and add the LoRA adapter.

The result and difference can be seen in Table 13. Admittedly, either approach will preserve some generalization performance compared with 3% when using FC layers. However, we still noticed some forgetting phenomena while fine-tuning the last layer. Therefore, we decided to use LoRA for finetuning, which not only allows for targeted adjustments to the model while keeping the majority of parameters frozen, reducing computational costs, but also preserves its ability to generalize to unseen data.

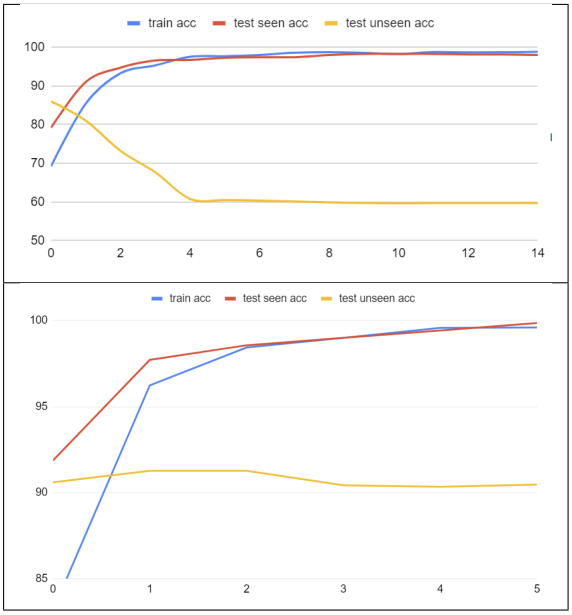


Table 13: Result from last layer tuning (Top), and LoRA tuning (Bottom)

Audio as an Auxiliary Modality for Classification Task Since we hypothesized that audio would help to classify some tasks that are difficult using visuals alone, we performed a comparative analysis between unimodal - S3D (Video only) (Xie et al., 2018), and simple multi-modal - low-rank fusion (Liu et al., 2018) on S3D (Video) with Resnet-18 (Audio) (He et al., 2015). We visualize their feature embedding respectively shown in Table 14 where the top one is feature embedding of S3D, and the bottom part is fusion modal on both vision and audio. The blue region represents the cluster of class "MoppingFloor". We can observe that this class is able to form a more obvious cluster with the aid of audio modality. However, it is not necessarily true that audio is always able to improve feature representation, especially on UCF-101, noted as the red region on the bottom plot of Table 14. There are still some regions where the multi-modal is struggle learning. Adding audio might cause the fusion modal to deteriorate.

We notice similar behavior in the different variants of our proposed model. As shown in table 15, our M2CFv2, which only takes video as input and coordinates with the text label, has been able to improve upon competitive baselines and push apart embeddings from similar categories "StillRings", "BalanceBeam", and "FloorGymnastics". However, since these videos have very similar audio, that is, all crowd noise, cheers, and music on sports competition venues, the model (M2CFv3) once again becomes confused after taking the audio input into account.

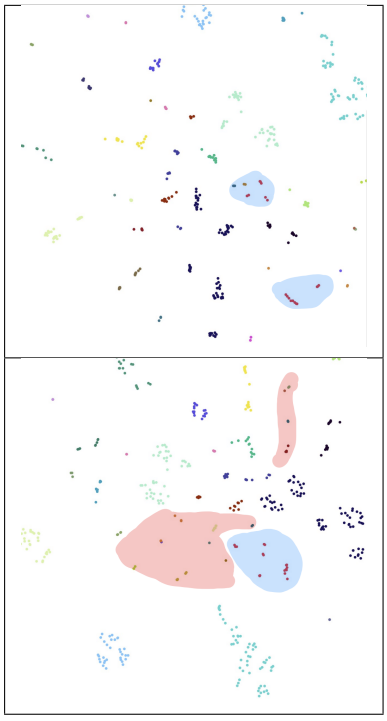


Table 14: t-SNE Comparison between S3D (Top) and low-rank fusion (Bottom)

Contrastive Learning to coordinate different modalities: CLIP, ViFi-CLIP, and our M2CF

1. **CLIP and ViFi-CLIP**: CLIP combines single-frame image and language representation in order to perform arbitrary class prediction. However, many mistakes can still be found. For example, it is very typical for CLIP to confuse "CricketBowling" and "CricketShot", as shown in their confusion matrix comparison in table 16. One reason we suspect is that the CLIP visual encoder fails to capture temporal information. In contrast, ViFi-CLIP is less confused about these two classes. ViFi-CLIP used temporal pooling to cooperate with temporal information while coordinating with language information. Besides, CLIP cannot capture the semantic or hierarchical structure of an image. In other words, CLIP's vision encoder can only understand what object appeared in an image without understanding the dynamic relationship between them. An example of such phenomenon is a group of classes including: "StillRings", "BalanceBeam", and "FloorGymnastics". Their image representation might be more align with texts like "A person is doing gymnastics in an indoor gym." The language label, which is a description of a series of actions from humans, might not be aligned with what the image encoder understands. However, ViFi-CLIP can understand the difference between these classes through the movement changing across time. Table 15 shows the t-SNE visualization on the gymnastics category using both models. We can see that with ViFi-CLIP, these representations are better separated.

2. **Proposed M2CF model**:

   Table 17 presents a t-SNE visualization comparing the original CLIP model, ViFi-CLIP, and our proposed model, M2CFv2. All of these models coordinate between the visual representation and the textual representation. Our method demonstrates a significant advantage over the traditional CLIP model, as observed in the t-SNE plots. The CLIP model, lacking temporal comprehension, results in scattered embeddings that fail to form dense clusters. In contrast, our model's embeddings exhibit well-defined clustering and clear separation. Even compared to the previous SOTA unimodal vision model, VideoMAE, whose t-SNE visualization can be found in the Appendix and shows clusters with tails, our M2CFv2 still outperforms it in terms of seen accuracy and embedding clustering. This distinction in t-SNE visualizations translates into the reduced confusion observed in Table 16, where our approach markedly outperforms the CLIP model. Additionally, categories such as "Still Rings," "Balance Beam," and "Floor Gymnastics" are particularly challenging to differentiate due to their similarity and the complexity of their backgrounds. As illustrated in Table 15, our model shows a distinct improvement in semantic understanding even when compared to CLIP and ViFi-CLIP. While our method does not achieve perfect separation, the finetuned visual encoder significantly reduces error rates in these three categories.

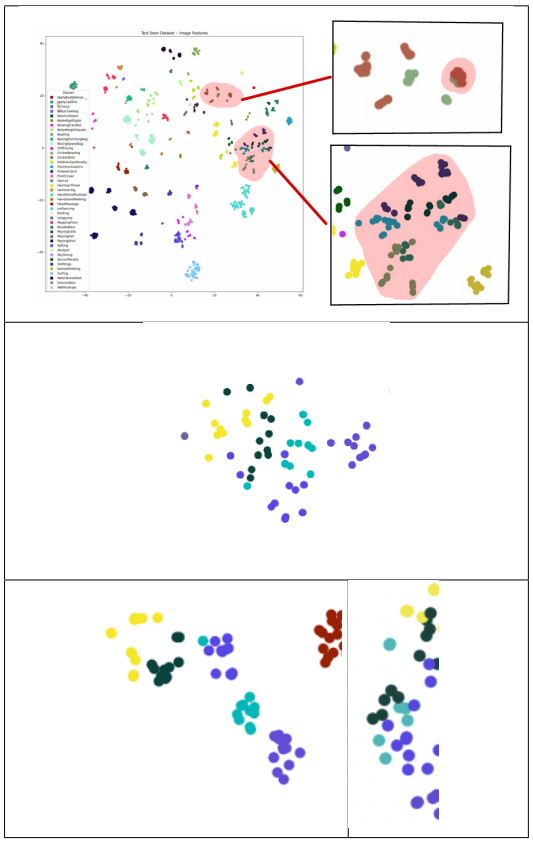


Table 15: t-SNE comparison between CLIP (Top), ViFi-CLIP (Middle), Our M2CFv2 (Bottom Left), and M2CFv3 (Bottom Right) on the gymnastics category

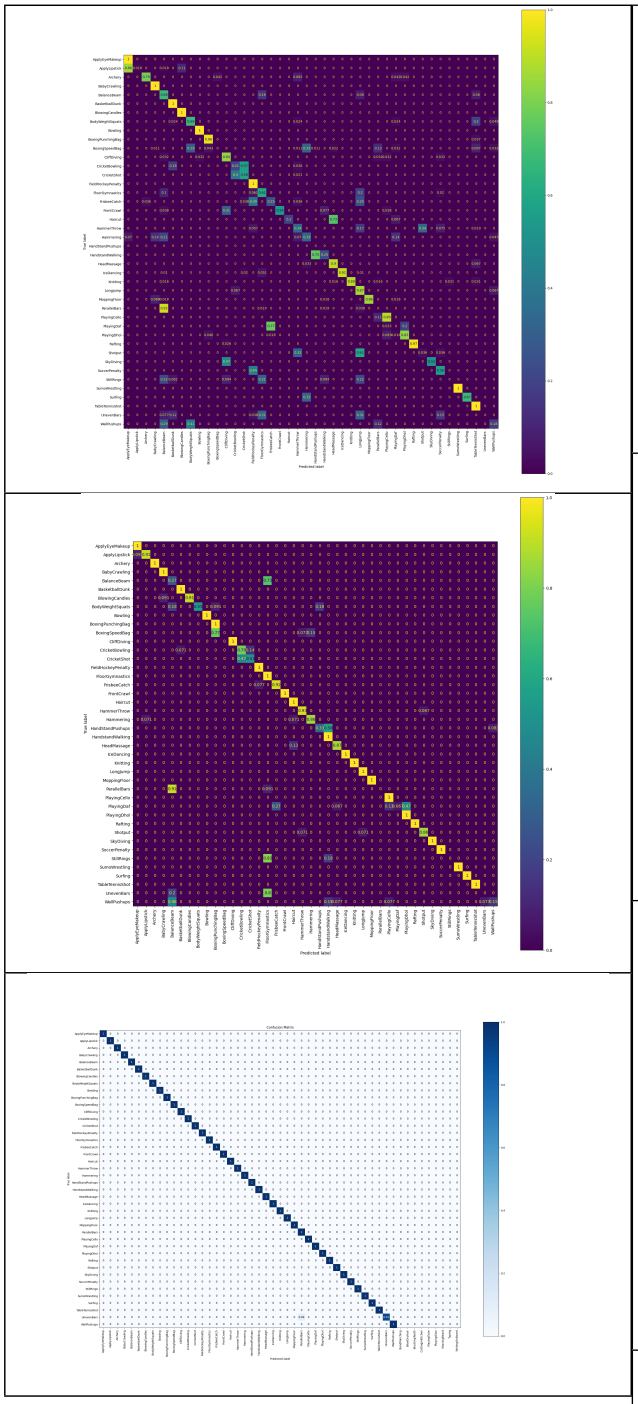


Table 16: Confusion matrix comparison between CLIP (Top) and ViFi-CLIP (Middle) and Our M2CFv2 (Bottom) on seen dataset

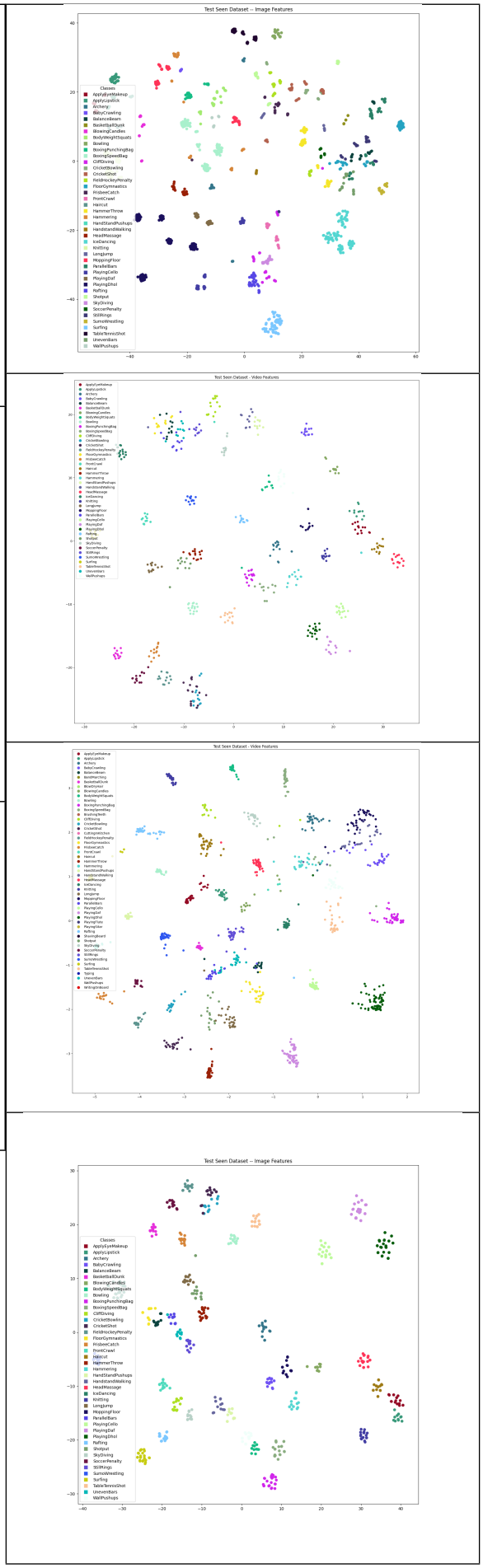


Table 17: t-SNE comparison between CLIP (Top), ViFi-CLIP (Second), VideoMAE (Third), and Our M2CFv2 (Bottom)

7.3 Qualitative Analysis and Examples

Audio as a complimentary modality: Comparison between (V) S3D and (A+V) S3D+Resnet Before analysis of our purposed model, we analyzed how audio behaved as an additional modality together with visual in contrast with vision alone. t-SNE visualization and confusion show that we do need high-quality audio feature representation in order for modality fusion to have a significant improvement. We acquired some examples that we hope better strengthen our perspectives. The results are shown in Table 19, where we sample some error cases that are representative.

1. **Ground Truth: Boxing Punching Bag, Playing Cello**: These two examples show that a good audio representation can improve existing visual understanding. In UCF-101, "Boxing Speed Bag" and "Boxing Punching Bag" are two classes with high visual similarity but distinct audio signature. From a visual perspective, both categories are illustrated as a person with a boxing glove punching some form of a bag. It is extremely difficult to identify which bag that person is punching, and the shape of the bag is difficult to discern due to the low resolution and multiple irrelevant objects. However, these two classes can be effortlessly distinguished by listening to the pitch and rhythm of punching. In our case, by adding audio understanding to S3D, the modal is able to predict the correct category. Similarly, the same theory holds true between "Playing Dhol" and "Playing Cello" where audio is much more informative than visual.

2. **Ground Truth: Archery, Cricket Bowling, Front Crawl**: These three cases are some examples where audio modality is less dominant regarding decision-making. In the case of "Archery" and "Front Crawl" (true label), our visual learner (S3D) is paying more attention to the background information instead of semantics. In our example, both "Archery" and "Floor Gymnastics" have the indoor gym as a background. "Sky Diving" and "Front Crawl" both have blue (sky and water) as the background. It is easier for a model to learn this information as its distinctive categorical feature and ignore the actual human action (semantics). In other words, S3D is learning more background than semantics and learning more spatially than temporally. When adding audio modality, these two examples result in the other model's predictive decision. Fortunately, since the "Archery" class has a relatively silent background and the arrow shooting sound is in contrast with "Floor Gymnastics," where audience noise and music are dominant, the model (S3D + ResNet) is able to predict the correct label. In the "Front Crawl" case, since the audio is more like a rhythmic base, the model incorrectly classifies it as "Boxing Punching Bag". "Cricket Bowling" is another similar example where the audio is mostly noise, causing the fusion model's performance to deteriorate.

Contrastive Learning to coordinate different modalities: Example analysis on CLIP, ViFi-CLIP, and the proposed M2CF CLIP and ViFi-CLIP are three examples from the CLIP family, which utilizes contrastive loss to coordinate different modalities. Our proposed model, M2CF, also used a similar strategy to coordinate the audiovisual representation with the textual representation.

1. **CLIP and ViFi-CLIP**: Given examples shown in Table 19, we can observe that the CLIP model is able to find a decent relationship between language and image without finetuning on UCF-101. However,since CLIP operates in single-frame images, the visual encoder (using either ResNet-101 or ViT-base) fails to capture temporal information, but ViFi-CLIP used temporal pooling to cooperate with temporal information while coordinating with language information. With this simple strategy, we notice that ViFi-CLIP is able to correctly classify classes such as "Archery", "Front Crawl", and "Writing on Board", which requires continuous understanding along the time axis, as in table 19.

   However, ViFi-CLIP could still be wrong sometimes, as shown in table 19, it misclassified "Cricket Bowling" as "basketball Dunk". Unfortunately, this is a challenging video showing people running and bowling cricket on a basketball court, and it is extremely unclear to see a cricket ball moving.

2. **Proposed M2CF model**: Example results are shown in Table 19 20. Generally speaking, our proposed method doesn't make the same

mistakes compared with our baseline models. This really shows that our method has a reasonably well understanding of both semantic and temporal information. It is able to mutually improve single-frame understanding exhibited in CLIP and sequential temporal understanding demostrated in ViFi-CLIP without sacrificing either. However, there are still a few mistakes made during inference shown in Table 18. Within seen data, our model mistakenly classifies "UnevenBars" with "Parallel Bars". Admittedly, these two are hard classes which really has a its resemblances. However, one category in our model that made the most mistakes is "Writing on Board" from unseen data. This similar mistake has also been common in our baseline models. We speculate the reason might be due to the fact that our visual encoder and text encoder have not seen similar data during pretraining. And these two are the unseen classes that our model is not trained on.

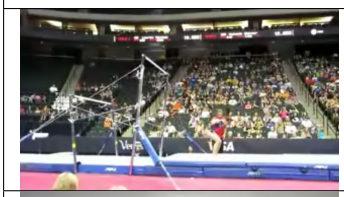
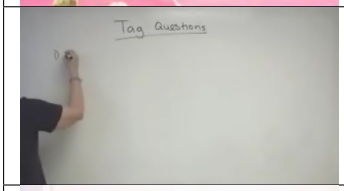
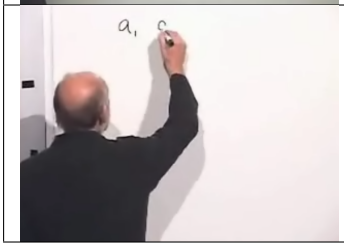
Sample Data	*M2CF*	True Label
	Parallel-Bars	Uneven-Bars
	Typing	Writin-gOnBoard
	Typing	Writin-gOnBoard

Table 18: Our purposed method's mistakes
Correct prediction and ground-truth are being labeled as blue, incorrect ones are red

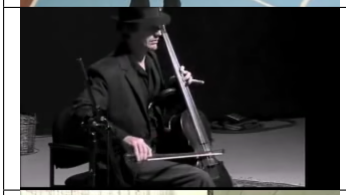
Sample Data	*1.1.3 (V) S3D*	*1.2.2 (A+V) S3D +Resnet*	*1.3.3 (A+T) CLAP*	*1.3.1 (V+T) CLIP*	*1.3.2 (V+T) ViFi-CLIP*	M2CF	True Label
	Boxing-Speed-Bag	Boxing-Punching-Bag	Playing-Daf	Cliff-Diving	Boxing-Punching-Bag	Boxing-Punching-Bag	Boxing-Punching-Bag
	Floor-Gym-nastics	Archery	Shotput	Ham-merThrow	Archery	Archery	Archery
	Play-ingDhol	Playing-Cello	Playing-Dhol	Playing-Cello	Playing-Cello	Playing-Cello	Playing-Cello
	Cricket-Bowling	Long-Jump	Shotput	Cricket-Bowling	Bas-ketball-Dunk	Cricket-Bowling	Cricket-Bowling
	Sky-Diving	Boxing-Punching-Bag	Body-Weight-Squats	Boxing-Punching-Bag	Front-Crawl	Front-Crawl	Front-Crawl

Table 19: Examples of models' prediction
Correct prediction and ground-truth are being labeled as blue, incorrect ones are red

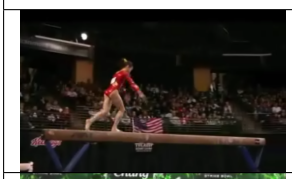
Sample Data	1.1.2 (V) ResNet-18 + Self-attention	1.1.5 (A) Self-attention on FBank	1.2.1 (A+V)(EF) Self + Cross-modal Attention	M2CF	True Label
	Balance-Beam	LongJump	ParallelBars	Balance-Beam	Balance-Beam
	Bowling	PlayingDaf	Bowling	Bowling	Bowling
	Hammering	ApplyEye-Makeup	Knitting	Knitting	Knitting
	Mop-pingFloor	Knitting	Mop-pingFloor	TableTennis-Shot	TableTennis-Shot
	TableTennis-Shot	TableTennis-Shot	TableTennis-Shot	TableTennis-Shot	TableTennis-Shot

Table 20: Examples of models' prediction
Correct prediction and ground-truth are being labeled as blue, incorrect ones are red

8 Future work and Limitations

Cross-Attention Fusion layer may damage model to generalize well The integration of a cross-attention fusion layer aims to enhance the interaction between different modalities, yet this approach may inadvertently impair the model's ability to generalize. As shown in table 10, although an increasingly number of cross-modal attention layers result a better fusion result shown as seen accuracy, we encounter a degradation of unseen accuracy. Despite the fact that the the visual, text, and audio encoder hasn't loss their generalization performance, the fusion layer might become overly specialized to the training data's distribution and fail to perform well on unseen labels. Future work could explore adaptive fusion techniques that allows models more effective fuse vision and audio while preserving its generalization ability.

Limited lexical diversity makes it hard to train from scratch The model's dependency on extensive and varied lexical input to train effectively from scratch is a significant hurdle especially on human action recognition where text input is just labels. This not only makes text representation difficult to align with other modality resulting poor generalization performance, but also hinder us to train any model from scratch. As a result, our model is highly dependent on the assumption that visual and text representation is pre-aligned. Future work could explore options such as text augmentation or generating text ground truth using retrieval techniques.

Noisy audio makes fusion with video challenging In some of the videos, the outstanding audio feature does not actually align with the domain of encoders pretrained on good-quality or simply typical sounds. For example, in the class "ApplyLipstick", we should expect typical audio to have speech of a person instructing on how to apply lipstick. However, in some of the corresponding audios, the BGM takes dominant position. This is what we call "noise" in our dataset. One possible techniques is using auto-encoder to denoise irrelevant audio information. Another interesting idea is to use audio retrieval to extract text description and convert audio modality into a more discrete, well explored text modality.

Additional auxiliary loss function to better align and fuse modalities At the current stage we are mainly utilizing Cross Entropy Loss for optimization. But given our condition of limited data resources and poverty in textual information, one possible improvement would be to use auxiliary loss function to better align and fuse the modalities, and map the generated embeddings into a more regular and meaningful target domain. We already have several attempts, such as symmetric ce loss that try to mimic InfoNCE under the condition that we do not have 1-to-1 mapping between text and audio-visual features, or Euclidean distance losses, etc. It might also be a good possibility to explore approaches that try to keep the structures of embedding space to enable better generalizability.

Model modality agnostic performance When multiple modalities of data present, we need fusion modules to fuse the generated embeddings (or early fuse the data). However, this triggers problems when at inference time some of the modalities are absent. One possible future direction is to research on how to make the model work even if we do not have data for some of the modalities. In our work, this could be the audio modality. In UCF-101, 50 of the classes do not have audio data at all.

Large-scale pretraining on high-quality datasets During our experiments, we've noticed the audio data of the UCF-101 dataset may contain limited meaningful information or be irrelevant to the class label. Since it was sourced from YouTube videos, some of the audio is music edited after the video was filmed and has no correspondence to either the video or the text class label. More limitations of the UCF-101 dataset have been discussed earlier, including noisy audio, scarce video clip diversity, etc. Due to computing resource limitations, we have not been able to train on other datasets of larger size and greater diversity, such as Kinetics-400 (Kay et al., 2017a), and ActivityNet-200 (Heilbron et al., 2015). We think our proposed model is capable of effectively fusing auditory and visual information and coordinating with textual information. The meaningful correspondence of audio, video, and text is crucial. Other audio-visual-textual models (Akbari et al., 2021) have utilized datasets with such correspondence on other domains, such as HowTo100M (Miech et al., 2019), EpicKitchen (Damen et al., 2020), and AudioSet (Gemmeke et al., 2017). We expect our proposed model performance would be greatly improved if pretrained on large, high-quality datasets. As detailed in table 21,

a preliminary ablation study shows that pretraining on Kinetitcs and then finetuning on UCF-101 improved the harmonic mean by 59.86% compared to purely finetuned on UCF-101.

		Top-1 Accuracy% ↑		
Dataset	Ep	Seen	Unseen	HM
All UCF	7	43.65	48.52	45.96
K400 Pretrain	1	51.08	50.68	50.88
UCF Finetune	6	**65.62**	**83.46**	**73.47**

Table 21: All using ViT-B/32 CLIP image encoder finetuned by LoRA, AudioMAE audio encoder, and one cross-modal attention layer, and Composite loss objective.

9 Ethical Concerns and Considerations

Even though our model has the capability of using both or either video and audio input, we have not robustly tested the performance of such scenarios. Therefore, it would be too premature to use our model with only audio input and could cause significantly unreasonable predictions. In addition, the training dataset is only a small set of human actions of typical daily activities, sports, and playing musical instruments. It can not comprehensively represent all kinds of human actions. Though our intention was to build an open-vocabulary model that can generalize to unseen actions, it would still be irresponsible to use it in a completely different domain without careful finetuning and testing. This is especially crucially if it were to be applied in high-stakes tasks, for example, in security surveillance and medical monitoring. Even if the model prediction is correct, any decisions should still be made through human inspection instead of fully automated, as any errors could lead to inappropriate and unfair actions based on incorrect data interpretation.

Besides, it is critical that this tool is used by the correct hands. Machine-automated human action recognition could help in the surveillance of dangerous events and getting medical assistance in time. It may also help building management to better plan spaces and energy consumption based on usage patterns. However, this is also a double edged sword, where continuous monitoring could be of privacy concern. The audio and video data is generally considered to contain personal-identifiable information that people are less preferred to share. Monitoring human behavior could also be used to manipulate public behavior and enforce conformity to what is deemed "acceptable" or "normal". Corporations could exploit the technology to analyze consumer behavior in invasive ways, leading to hyper-targeted and potentially manipulative advertising strategies. Another type of malicious use would be if the input (video or audio) is adversarially perturbed. Malicious users add patch or sound snippets to affect the prediction result of the model. These kind of adversarial perturbations might lead to failure to correctly predict actions of object humans and might lead to severe situations in safety-critic scenarios.

In addition, we also acknowledge the biases inherent to the training dataset and, therefore, exhibited in the model prediction. Our model uses pretrained weights of CLIP, which has been confirmed to contain societal stereotypes (Alabdulmohsin et al., 2024). Though finetuning could mitigate the pre-existing bias in CLIP, our training dataset, UCF-101, is also prone to a similar issue. It is by no means a comprehensive dataset that reflects all demography and all human actions in the real world. Some of the classes are of high quality with clear correspondence between video, audio, and text, while some are much less, for example, audio containing indistinguishable noise and music edited afterward. These would make our model biased toward classes it understands better.

References

Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. 2021. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text.

Ibrahim Alabdulmohsin, Xiao Wang, Andreas Steiner, Priya Goyal, Alexander D'Amour, and Xiaohua Zhai. 2024. Clip the bias: How useful is balancing data in multimodal learning?

John Arevalo, Thamar Solorio, Manuel Montes y Gómez, and Fabio A. González. 2017. Gated multimodal units for information fusion.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2022. Beit: Bert pre-training of image transformers.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Joao Carreira and Andrew Zisserman. 2018. Quo vadis, action recognition? a new model and the kinetics dataset.

Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2022. Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection.

Tongjia Chen, Hongshan Yu, Zhengeng Yang, Zechuan Li, Wei Sun, and Chen Chen. 2023. Ost: Refining text knowledge with optimal spatio-temporal descriptor for general video recognition.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. 2020. The epic-kitchens dataset: Collection, challenges and baselines.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2022. Clap: Learning audio concepts from natural language supervision.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–970.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. 2023. Masked autoencoders that listen.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. 2017a. The kinetics human action video dataset.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. 2017b. The kinetics human action video dataset.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Zhun Liu, Ying Shen, Varun Bharadhwaj Lakshminarasimhan, Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. 2018. Efficient low-rank multimodal fusion with modality-specific factors.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Sachit Menon and Carl Vondrick. 2022. Visual classification via description from large language models.

Otniel-Bogdan Mercea, Lukas Riesch, A. Sophia Koepke, and Zeynep Akata. 2022. Audio-visual generalised zero-shot learning with cross-modal attention and language.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips.

Behnaz Nojavanasghari, Deepak Gopinath, Jayanth Koushik, Tadas Baltrušaitis, and Louis-Philippe Morency. 2016. Deep multimodal fusion for persuasiveness prediction. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI '16, page 284–288, New York, NY, USA. Association for Computing Machinery.

Hieu H. Pham, Louahdi Khoudour, Alain Crouzil, Pablo Zegers, and Sergio A. Velastin. 2022. Video-based human action recognition using deep learning: A review.

Soujanya Poria, Iti Chaturvedi, Erik Cambria, and Amir Hussain. 2016. Convolutional mkl based multimodal emotion recognition and sentiment analysis. pages 439–448.

Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. 2023. What does a platypus look like? generating customized prompts for zero-shot image classification.

Rui Qian, Yeqing Li, Zheng Xu, Ming-Hsuan Yang, Serge Belongie, and Yin Cui. 2022. Multimodal open-vocabulary video classification via pre-trained vision and language models.

Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. 2021. Spatiotemporal contrastive video representation learning.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Hanoona Rasheed, Muhammad Uzair Khattak, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Fine-tuned clip models are efficient video learners.

Pritam Sarkar and Ali Etemad. 2022. Xkd: Cross-modal knowledge distillation with domain alignment for video representation learning.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William W. Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2014. Going deeper with convolutions.

Hao Tan and Mohit Bansal. 2019a. Lxmert: Learning cross-modality encoder representations from transformers.

Hao Tan and Mohit Bansal. 2019b. Lxmert: Learning cross-modality encoder representations from transformers.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. 2022. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. 2023a. Videomae v2: Scaling video masked autoencoders with dual masking.

Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Lu Yuan, and Yu-Gang Jiang. 2023b. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning.

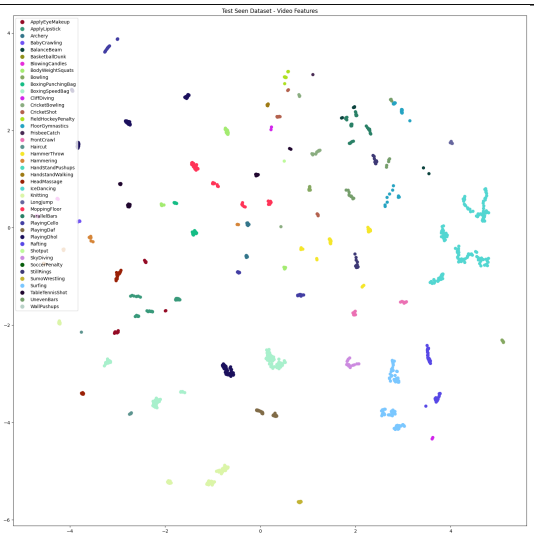
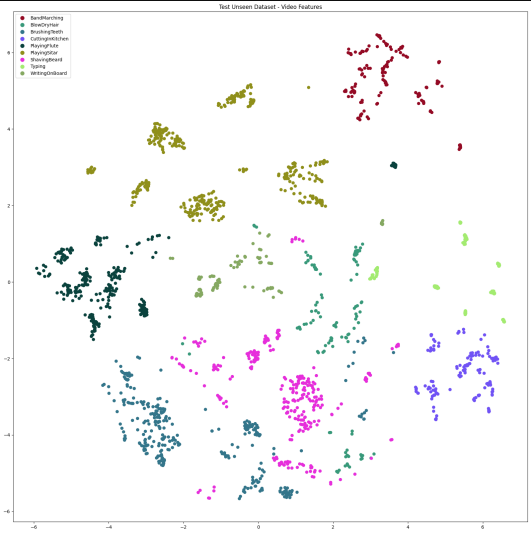
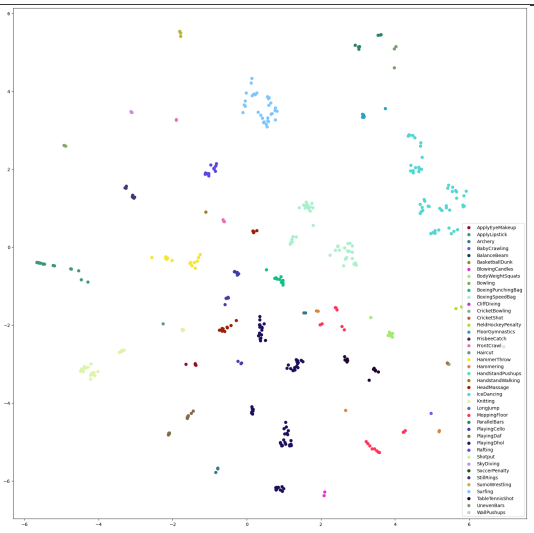
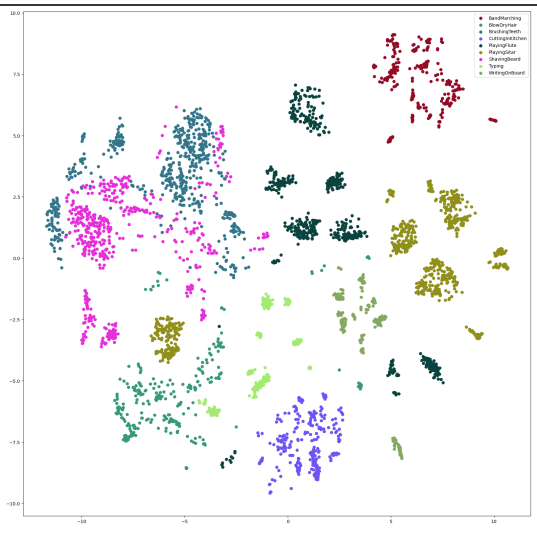
Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. 2018. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification.

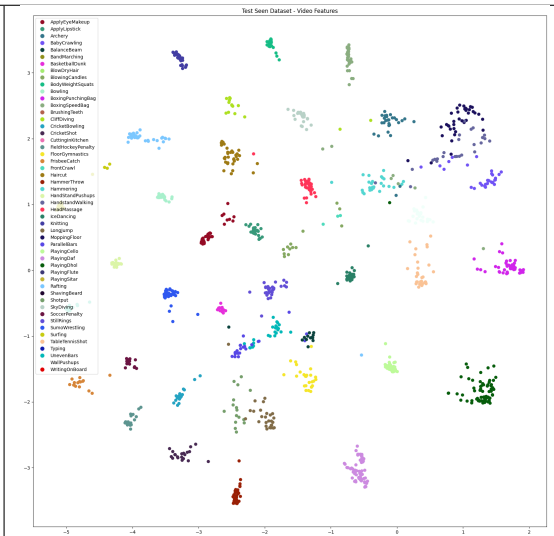
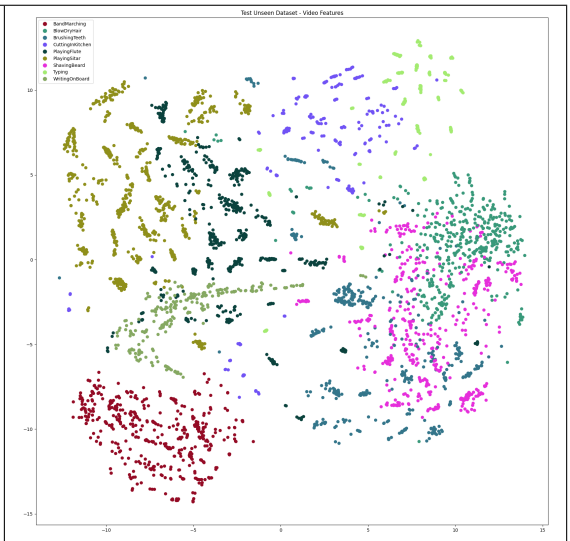
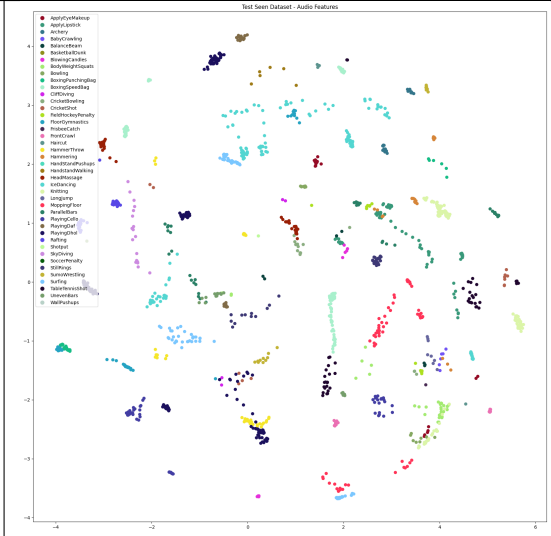
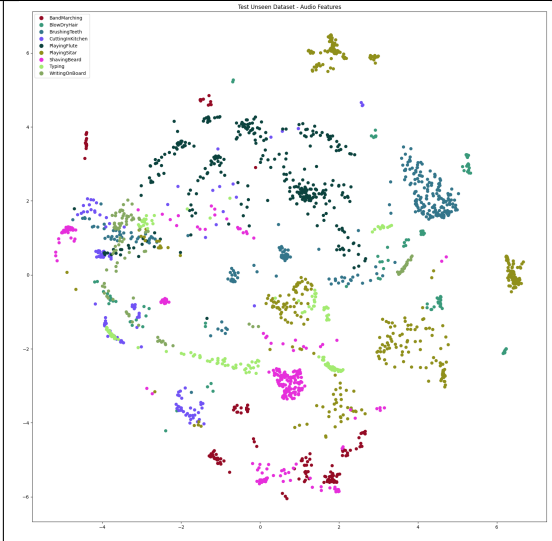
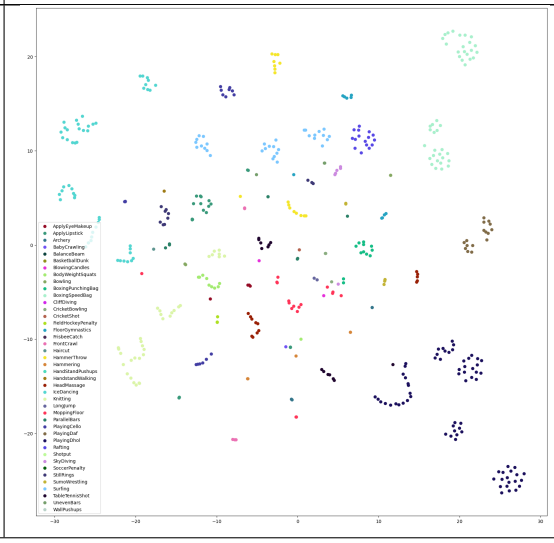
Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor fusion network for multimodal sentiment analysis.

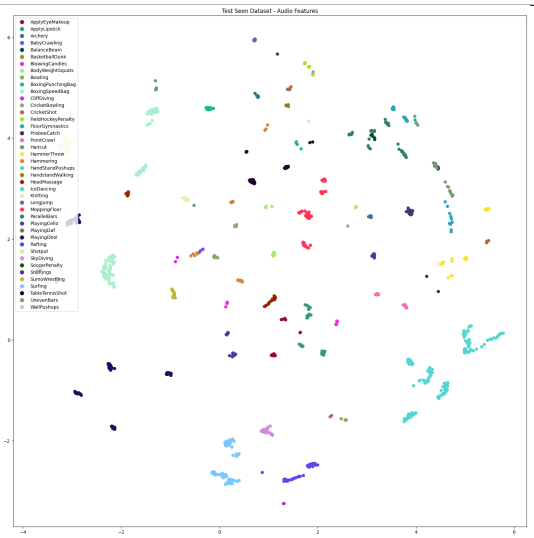
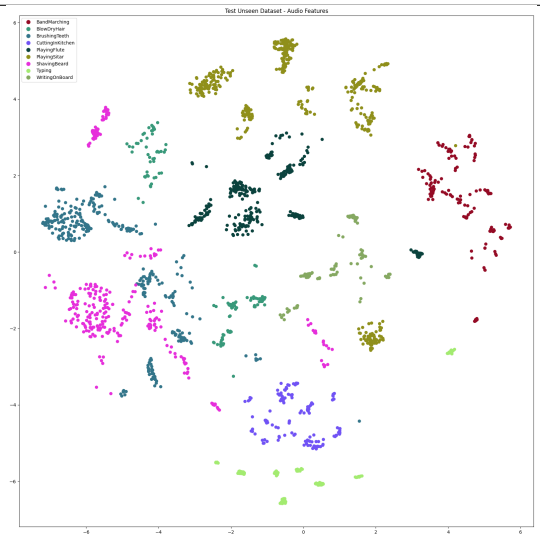
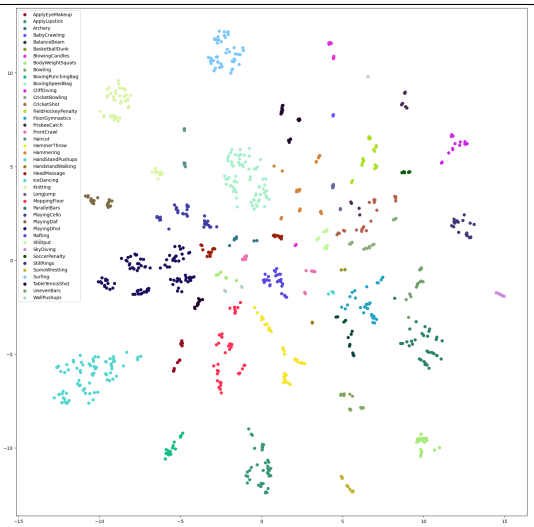
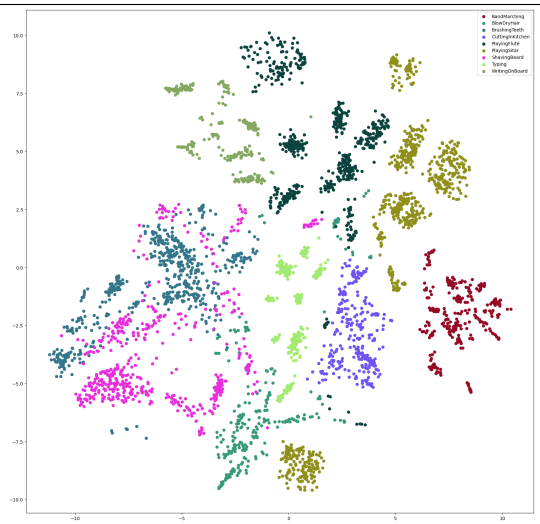
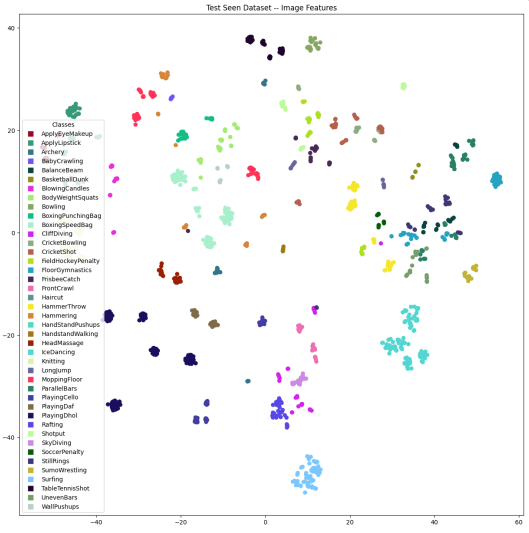
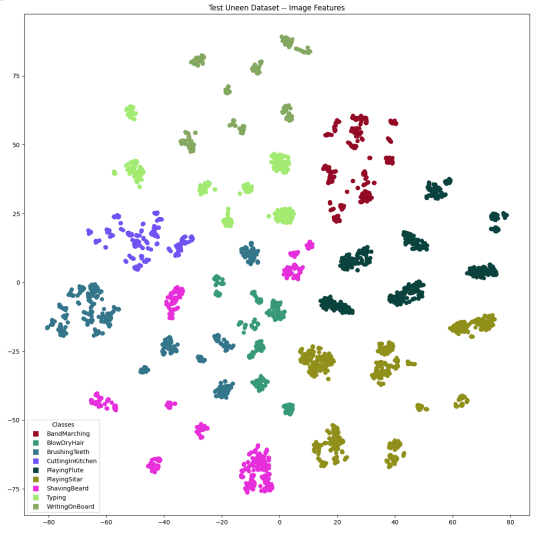
Appendix

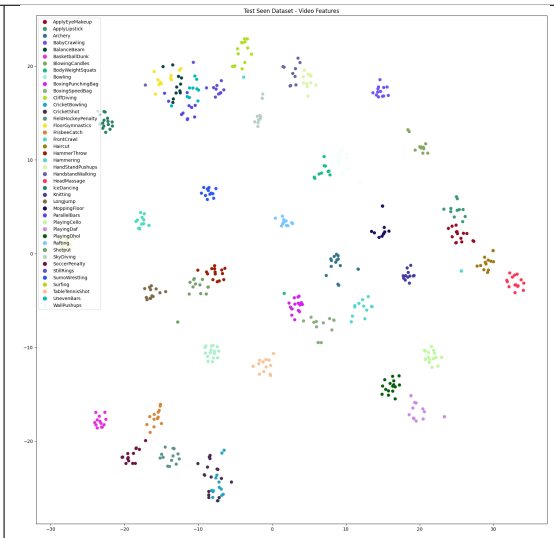
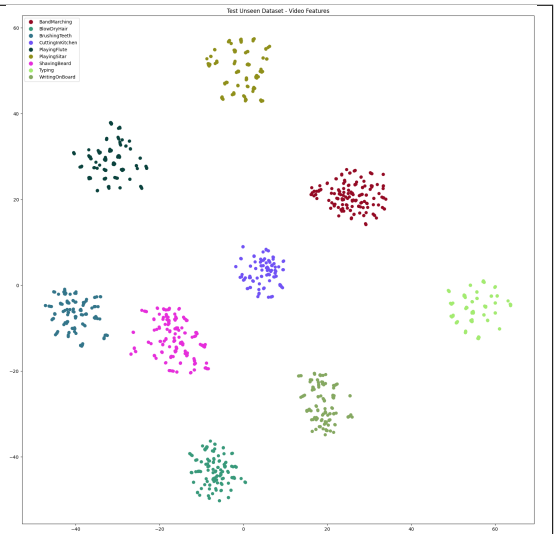
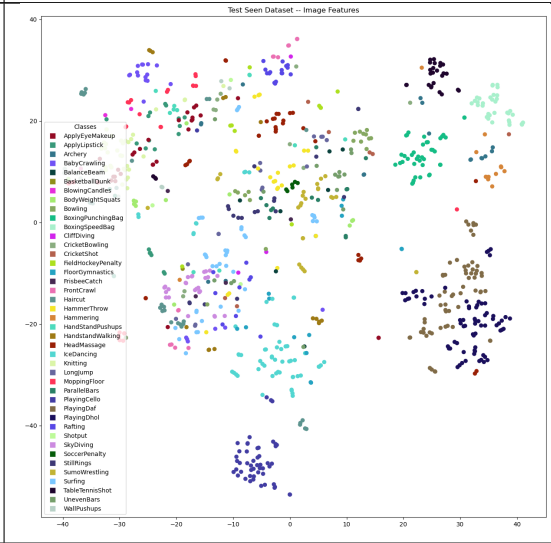
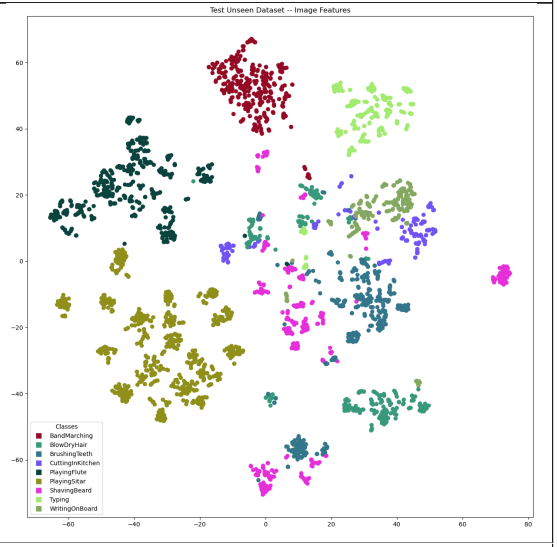
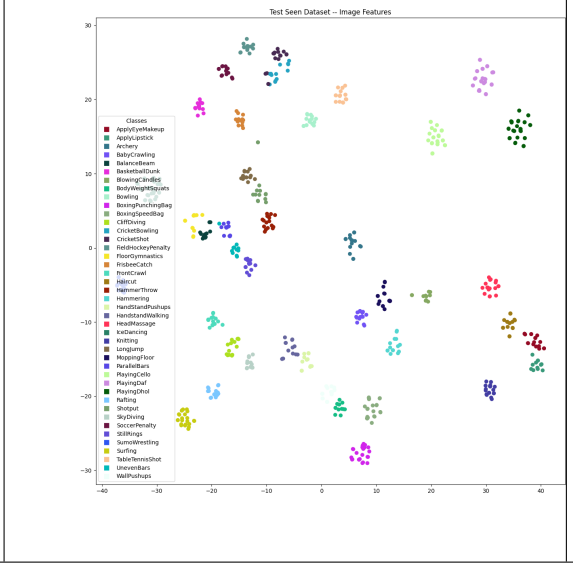
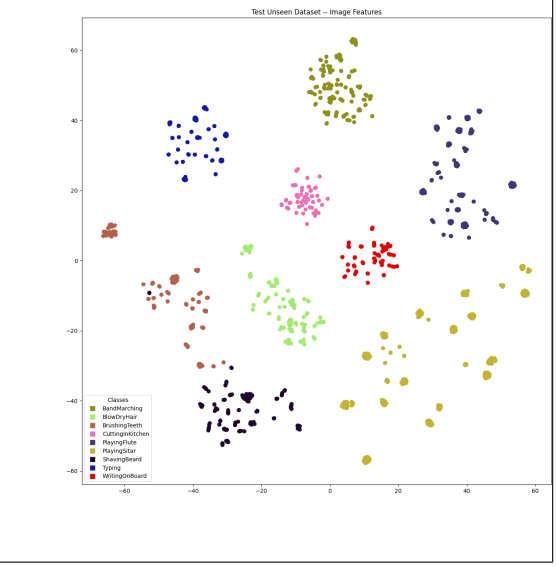
Fusion Methods	Top-1 ↑ Accuracy % Seen	Improvement
Unimodal		
(V) S3D	88.81	0
(A) ResNet-18 on FBank	71.61	-
Multimodal fusion strategy		
(A+V)(LF) S3D pretrained + Resnet-18 on FBank		
Summation	93.07	+4.26
Concatenation	94.33	+5.52
Gated Fusion (Arevalo et al., 2017)	92.91	+4.10
Tensor Fusion (Zadeh et al., 2017)	89.92	+1.11
Low Rank Fusion (Liu et al., 2018)	**94.48**	**+5.67**

Table 22: S3D + Resnet-18 Late Fusion results

Model	t-SNE on Seen	t-SNE on Unseen
(V) ResNet-18 + Self-attention		
(V) S3D		

(V) Video-MAE Pre-trained		
(A) Self-Attention on FBank		
(A) ResNet-18 on FBank		N/A

(A+V)(EF) Self + Cross-modal Atten-tion		
(A+V)(LF) S3D Pre-trained + ResNet-18 on FBank		
(V) Pre-trained CLIP		

(V+T) Video Fine-tuned CLIP		
(A) Pre-trained CLAP		
M2CFv2		

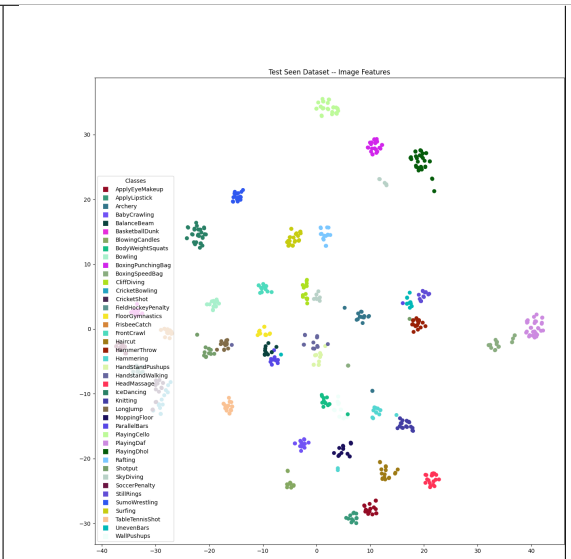
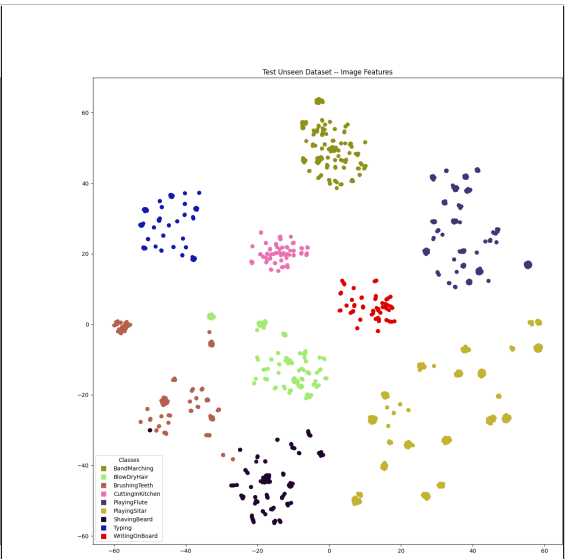
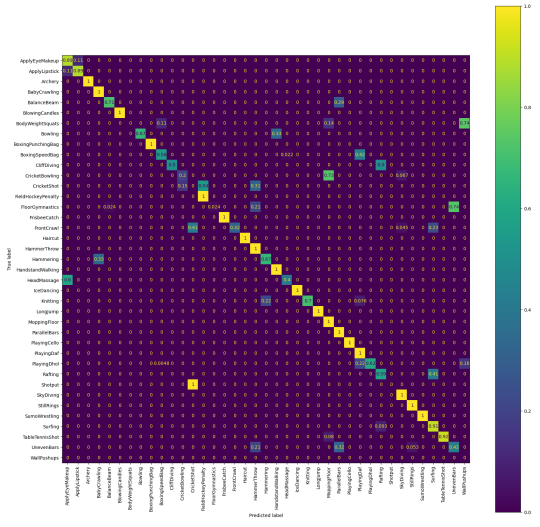
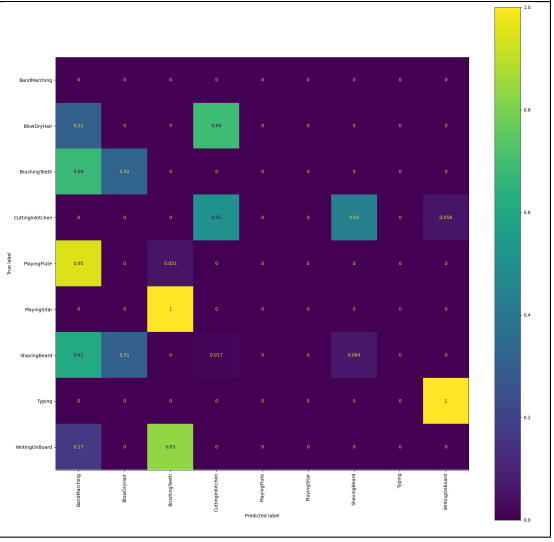
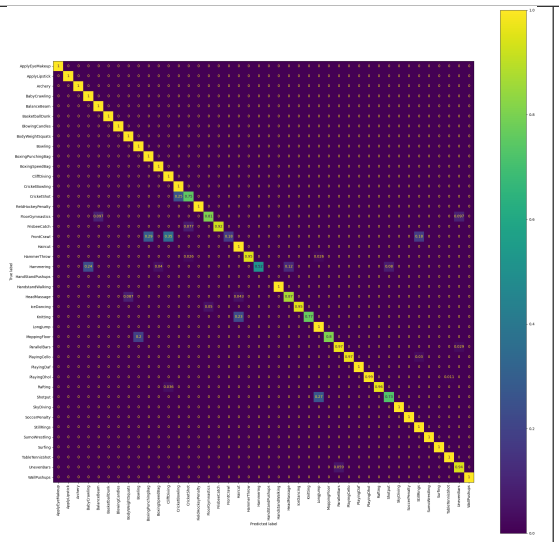
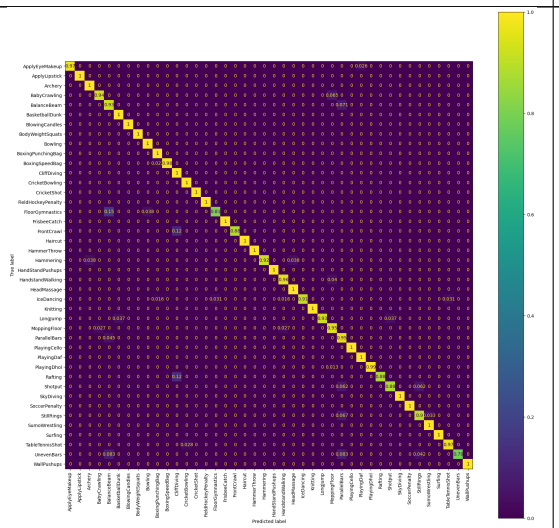
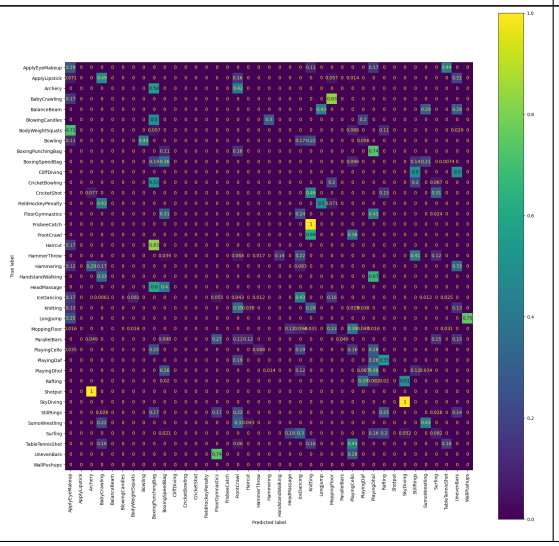
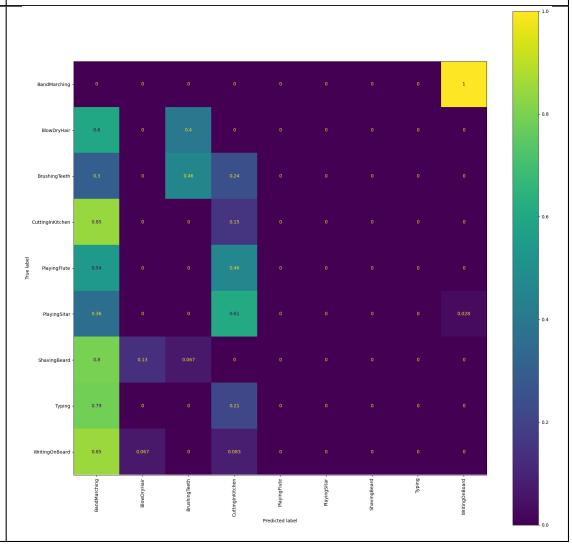
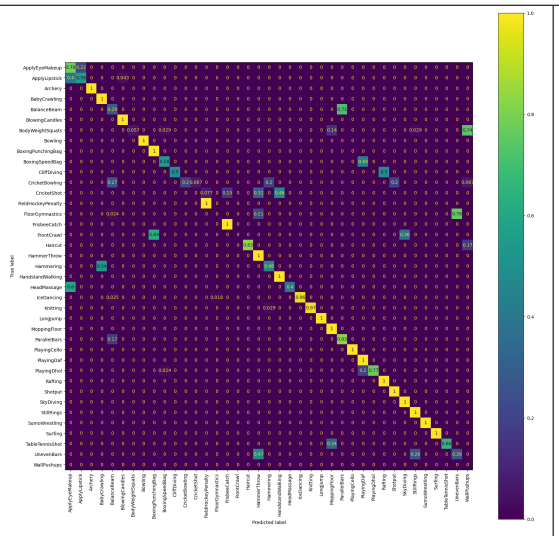
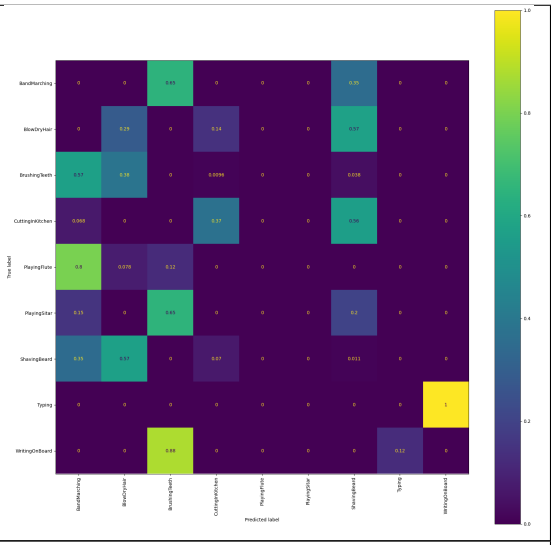
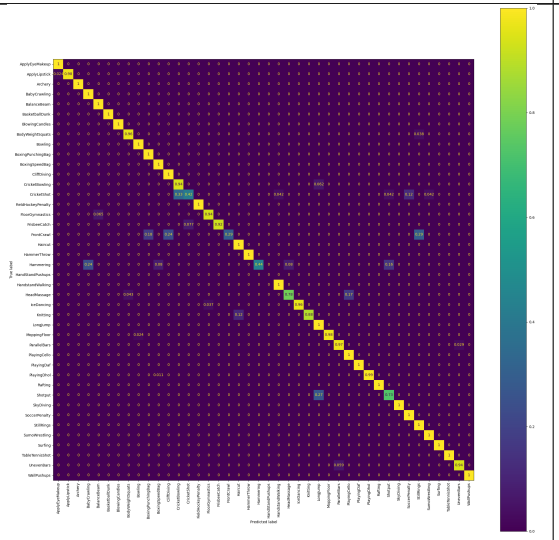
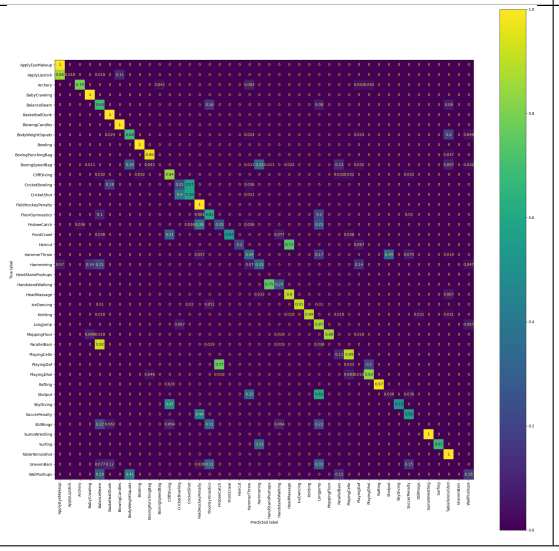
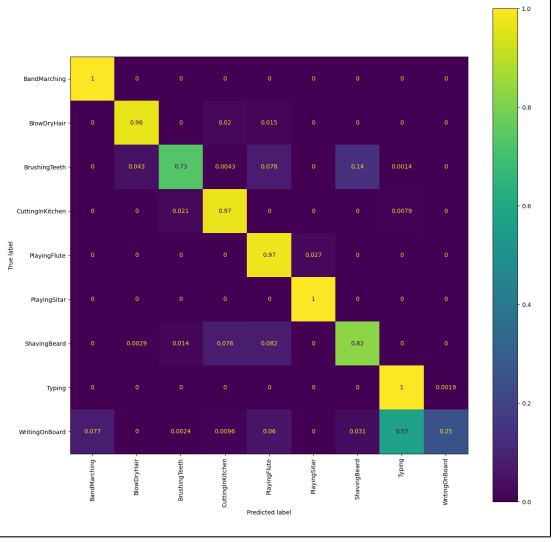
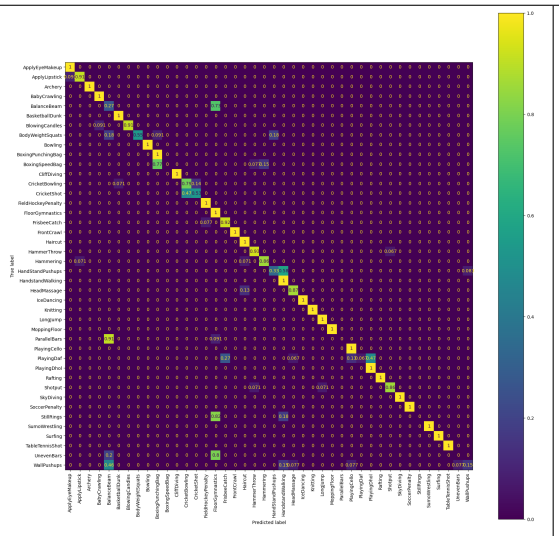
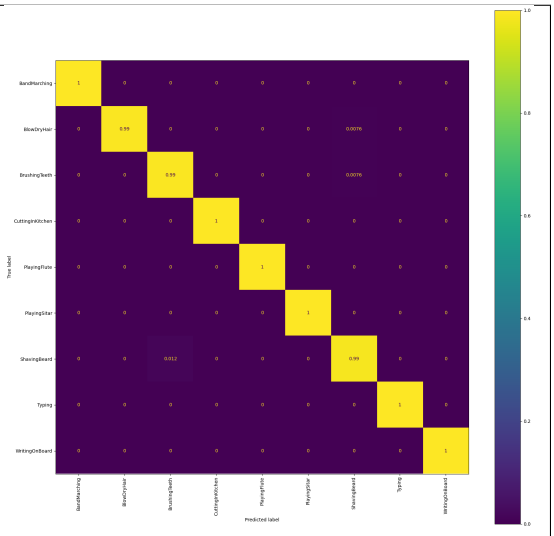
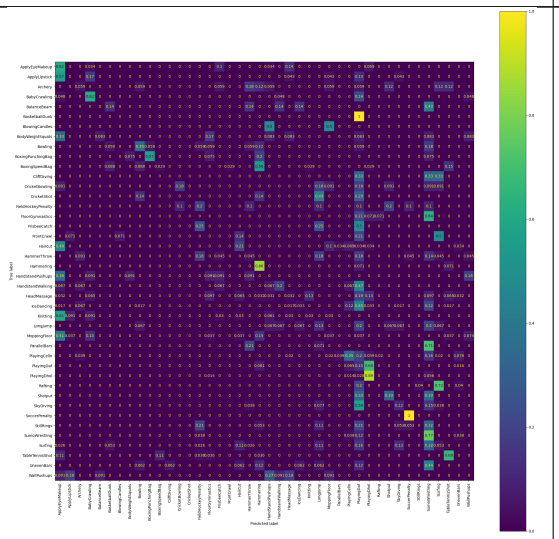
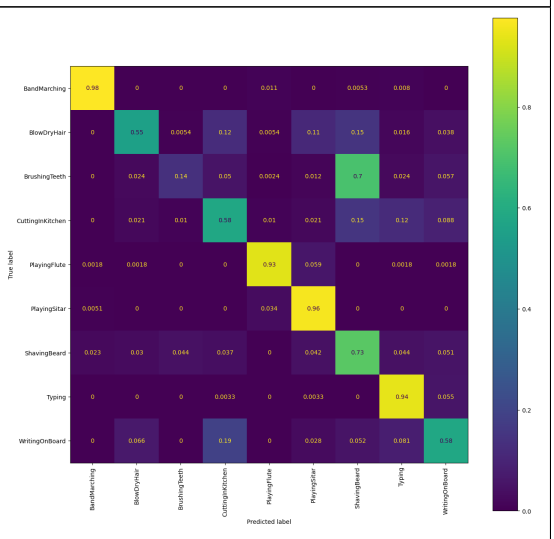
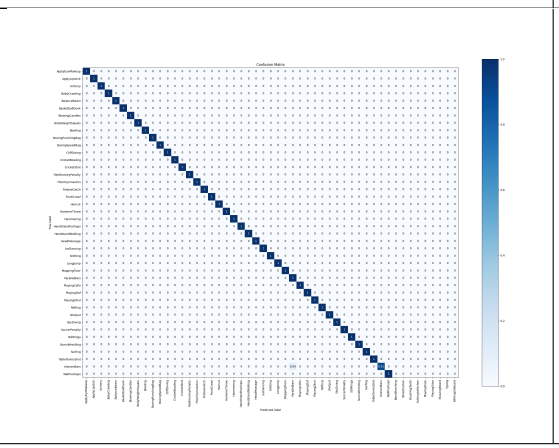
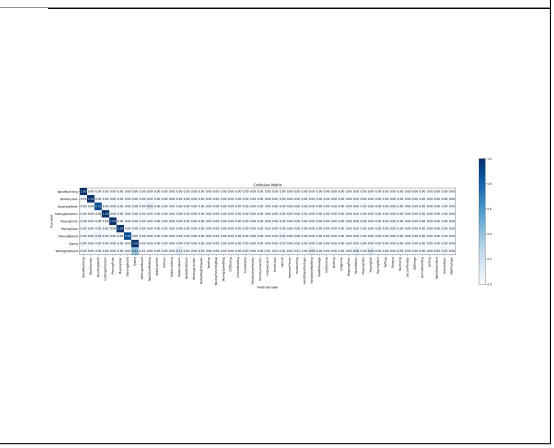
M2CFv3		

Table 23: 2D t-SNE Features on Embeddings Generated by Different Baselines

Model	Confusion Matrix on Seen	Confusion Matrix on Unseen
(V) ResNet-18 + Self-attention		

(V) S3D Pretrained		N/A
(V) VideoMAE Pretrained		N/A
(A) Self-Attention on FBank		

(A+V)(EF) Self + Cross-modal Attention		
(A+V)(LF) 3D Pretrained + ResNet-18 on FBank		N/A
(V) Pretrained CLIP		

(V+T) Video Finetuned CLIP		
(A) Pretrained CLAP		
M2CFv2		

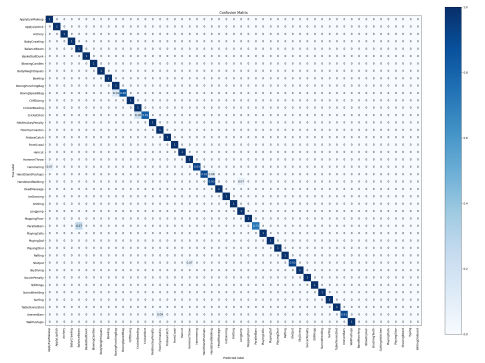
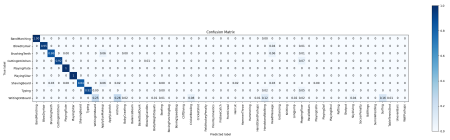
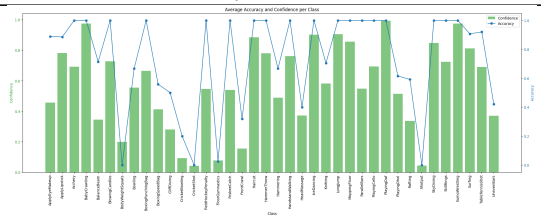
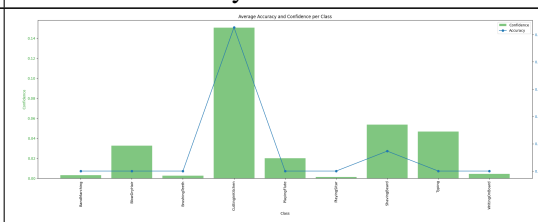
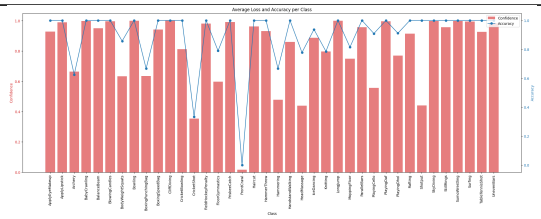
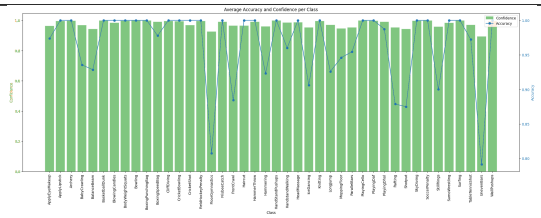
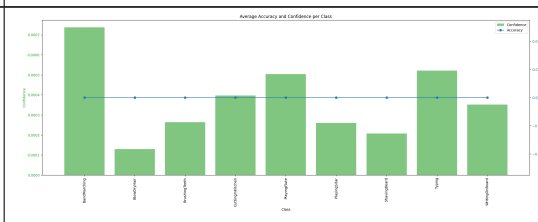
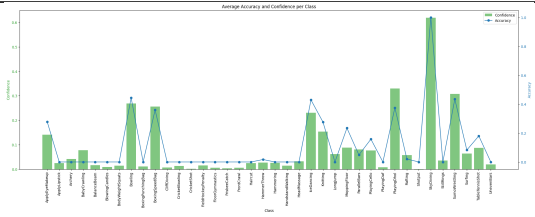
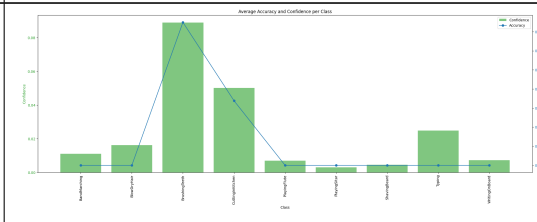
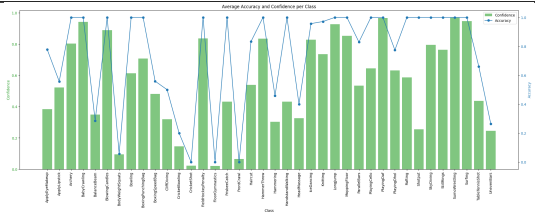
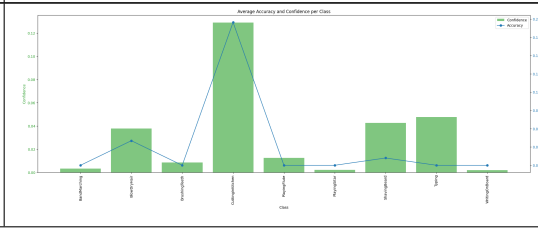
M2CFv3		

Table 24: Confusion Matrices for Different Baselines

Model	Per-class Accuracy & Confidence on Seen	Per-class Accuracy & Confidence on Unseen
(V) ResNet-18 + Self-attention		
(V) S3D Pretrained		N/A
(V) VideoMAE Pretrained		
(A) Self-Attention on FBank		
(A+V)(EF) Self + Cross-modal Attention		

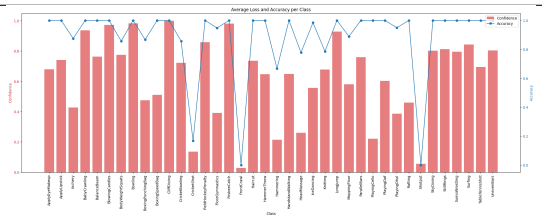
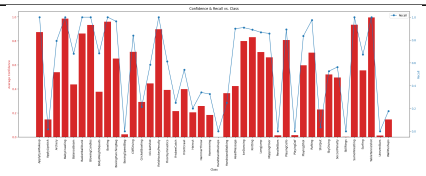
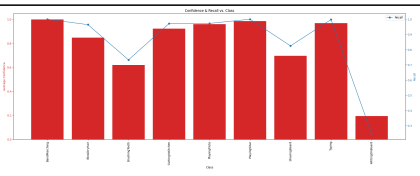
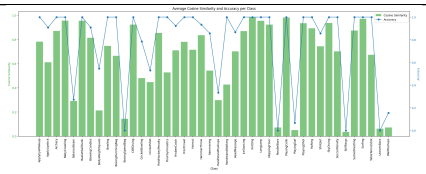
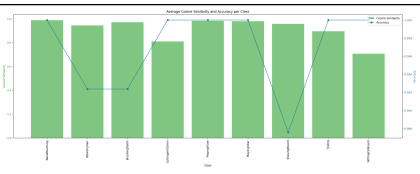
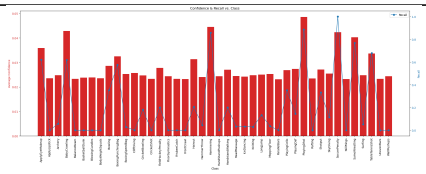
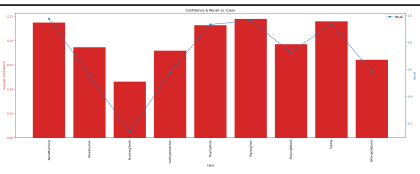
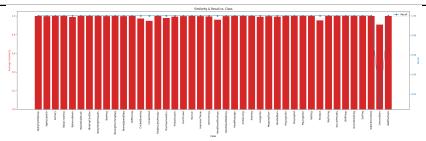
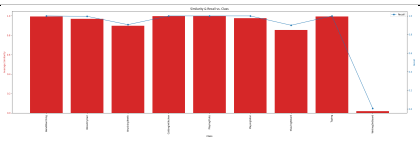
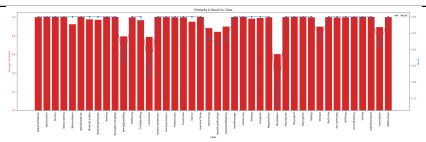
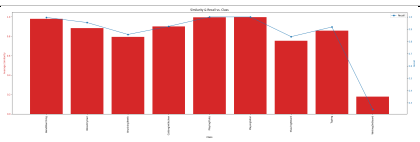
(A+V)(LF) 3D Pretrained + ResNet-18 on FBank		N/A
(V) Pretrained CLIP		
(V+T) Video Finetuned CLIP		
(A) Pretrained CLAP		
M2CFv2		
M2CFv3		

Table 25: Per-class Accuracy and Confidence for Different Baselines