FunnyNet: Audiovisual Learning of Funny Moments in Videos

Zhi-Song Liu, Robin Courant, Vicky Kalogeiton

FunnyNet: Audiovisual Learning of Funny Moments in Videos

Zhi-Song Liu*,[1], Robin Courant*,[2], and Vicky Kalogeiton[3][0000−0002−7368−6993]

[1] Caritas Institute of Higher Education, Hong Kong
zhisong.liu@connect.polyu.hk
[2] VISTA, LIX, Ecole Polytechnique, IP Paris
{robin.courant, vicky.kalogeiton}@lix.polytechnique.fr
http://www.lix.polytechnique.fr/vista/projects/2022_accv_liu

Abstract. Automatically understanding funny moments (i.e., the moments that make people laugh) when watching comedy is challenging, as they relate to various features, such as facial expression, body language, dialogues and culture. In this paper, we propose FunnyNet, a model that relies on cross- and self-attention for both visual and audio data to predict funny moments in videos. Unlike most methods that focus on text with or without visual data to identify funny moments, in this work in addition to visual cues, we exploit audio. Audio comes naturally with videos, and moreover it contains higher-level cues associated with funny moments, such as intonation, pitch and pauses. To acquire labels for training, we propose an unsupervised approach that spots and labels funny audio moments. We provide experiments on five datasets: the sitcoms TBBT, MHD, MUStARD, Friends, and the TED talk UR-Funny. Extensive experiments and analysis show that FunnyNet successfully exploits visual and auditory cues to identify funny moments, while our findings corroborate our claim that audio is more suitable than text for funny moment prediction. FunnyNet sets the new state of the art for laughter detection with audiovisual or multimodal cues on all datasets.

1 Introduction

We understand the world by using our senses, especially in multimedia area. All signals can stimulate one's feelings and reactions. Funniness is universal and timeless: in 1900 BC Sumerians wrote the first joke and it is still funny nowadays. However, whereas humans can easily understand funny moments, even from different cultures and eras, machines do not. Even though the number of interactions between humans and machines is growing fast, identifying funniness is still a brake on making these interactions spontaneous. Actually, understanding funny moments is a complex concept since they can be purely visual, purely auditory, or they can mix both cues: there is no recipe for the perfect joke.

Recently, some works try to understand what is a joke, humor and funny moments [1,2]. These rely solely on text and only a couple of them use also

* These authors contributed equally to this work.

Mia: Three tomatoes are walking down the street -- a poppa tomato, a momma tomato, and a little baby tomato. Baby tomato starts lagging behind. Poppa tomato gets angry, goes over to the baby tomato, and squishes him... and says, ... "Catch up."

Fig. 1: What is funny? Audio cue along with visual frame and facial data are a rich source of information for identifying funny moments in videos. Video scene from Pulp fiction, 1994, source video https://www.youtube.com/watch?v=4L5LjjYVsHQ

videos [3,4]; in such cases, video is always combined with text. However, these works are limited as text (transcripts, subtitles) does not come naturally with videos, but it depends on external pipelines, which depend themselves on other factors such as accents, sound quality, simultaneous audios, language, annotation. Thus, such works lack flexibility and can hardly be used in the wild.

In contrast, audio comes naturally with videos, and it contains crucial and complementary cues, such as tones, pauses, pitch, pronunciation, background noises [5,6]. Indeed, when people talk, not only what they say matters, but also how they say it. In turn, the visual content is also very important. For instance, depending on the context, the very same phrase said by the same person can be funny or sad (Figure 1). Yet, facial expressions, body gestures and scene context help better understand the sense of a phrase, thus impacting funniness.

Therefore, in this paper, we introduce FunnyNet, an audiovisual model for funny moment prediction. It consists of three encoders: (a) visual that encompasses the global context information of a scene, (b) face for representing the facial expressions of individuals, and (c) audio that captures voice and language effects; and the Cross Attention Fusion (CAF) module, i.e., a new module that learns cross-modality correlations hierarchically so that features from different modalities can be combined to form a unified feature for prediction. Thus, FunnyNet is trained to learn to embed all cross-attention features in the same space via self-supervised contrastive learning [7], in addition to classifying clips as funny or not-funny. To obtain labelled data, we exploit the laughter that naturally exists in sitcom TV shows. We define as 'funny-moment' any n-second clip followed by laughter; and 'not-funny' the clips not followed by laughter. To extract laughter, we propose an unsupervised labelling approach that clusters audio segments into laughter, music, voice and empty, based on their waveform difference[3]. Moreover, we enrich the Friends dataset with laughter annotations.

Our extensive experimentation and analysis show that combining visual with audio cues is suitable for funny-moment detection; specifically, our findings

[3] Note that we use the laughter solely as indicator for data labelling, but the laughter is not the included in the audio segments of FunnyNet. Once FunnyNet is trained, it can detect funny moments in any video, with or without laughter.

demonstrate that audio results in superior performances than language, hence revealing its superiority for the task and supporting our intuition that audio captures higher-level cues than subtitles. Moreover, we compare FunnyNet to the state of the art on five datasets including sitcoms (TBBT, MHD, MUStARD, and Friends) and TED talks (UR-Funny), and show that it outperforms all other methods for all metrics and input configurations. We also apply FunnyNet on data from other domains, i.e., movies, stand-up comedies, and audiobooks. For quantitative evaluation, we apply FunnyNet on a sitcom without canned laughter manually annotated. It shows that FunnyNet predicts funny moments without fine-tuning, revealing its flexibility for funny-moment detection in the wild.

Our contributions are: (1) We introduce FunnyNet, an audiovisual model for funny moment detection. It combines features from various modalities using the proposed CAF module relying on cross and self-attention; (2) Extensive experiments and analysis highlight that FunnyNet successfully exploits audiovisual cues, and show that audio is better suited than text for funny-moment detection; (3) FunnyNet achieves the new state of the art on five datasets, and we also demonstrate its flexibility by applying it to in-the-wild videos.

2 Related Work

Sarcasm and Humor Detection. Sarcasm and humor share similar styles (irony, exaggeration and twist) but also differ from each other in terms of representation. Sarcasm usually relates to dialogues; hence, most methods detect sarcasm by processing language using human efforts. For instance, [8] collect a speech dataset from social media using the hashtag and manual labeling, while others [9,10] study the acoustic patterns related to sarcasm, like slower speaking rates or higher volumes of voice. In contrast, a humorous moment is defined as the moment before laughter [6,11]. Hence, such methods [12,6,11,4,13] process audios to extract laughter for labeling. Nevertheless, for prediction, most such approaches focus solely on language models [1,2] or on multiple cues including text [11,13]. For instance, LaughMachine [4] propose vision and language attention mechanisms, while MSAM [3] combine self-attention blocks and LSTMs to encode vision and text. [14] use first an advanced BERT [15] model to process long-term textual correlation and then vision for the prediction. Following this, [16] propose a Multimodal Adaptation Gate to efficiently leverage textual cues to explore better representation for sentiment analysis. Few methods also explore audio. For instance, MUStARD [6] and URFUNNY [11] process text, audio and frames using LSTM to explore long-term correlations, while HKT [13] classifies language (context and punchline) and non-verbal cues (audio and frame) to learn cross-attention correlations for humor prediction. They combine audio with other information (video and texts) in a simple feature fusion process without investigating the inter-correlations in depth. Specifically, they stack multimodal features to learn the global weighting parameters without considering the biases in different domains. In contrast, we believe that funny scenes can be triggered by mutual signal from multimodalities; hence, we explore the cross-domain agree-

ment of cues with contrastive training. Moreover, instead of text, FunnyNet relies solely on audiovisual cues, as audio comes naturally with videos, and it contains all essential cues for funny moment prediction.

Sound Event Detection detects which and when sound events happen in audios. Most attempts either rely on annotated data [17] or use source separation [18]. The input plays a crucial role, and most methods use Mel spectrograms [19,20,21,22] instead of audio waveforms. *Laughter detection* focuses on one specific event: laughter. For this, some methods rely on physiological sensors [23,24], while others [25,26] follow the supervised paradigm to train detectors. In contrast, our laughter detector is an unsupervised, robust and straightforward labelling method that exploits multichannel audio specificities.

Multimodal Signal is processed with models like LSTM [27], GRN [28], ViT [29] and VQVAE [30] and is studied for various tasks. For instance, [31,32] recognize the facial movements to separate the speaker's voice in the audio. [33,34] temporally align the audio and video using attention to locate the speaker. Several methods extend this to other applications, including speech recognition [35], audio-image retrieval [36,37], audiovisual generation [38,39], video-text retrieval [40], human replacement [41], visual question answering [42], and affect analysis [43]. Recently, Video Transformers models [44,45] showed improved accuracy on various video tasks, in particular for classification [44,46,45,47]. Their self/cross- attention operation provides a natural mechanism to connect multimodal signals. Thus, some works exploit this to account for multiple modalities, such as inter and intra cross-attention in [48], contrastive cross-attention in [49], iterative cross-attention in [47,50] or bottlenecks in [51] for various tasks, such as summarization [52], retrieval [53,54], audiovisual classification [51], predicting goals [55]. [49,48] iteratively apply self and cross-attention to explore correlations among modalities. Instead, FunnyNet both fuses all modalities and in parallel learns the cross-correlation among different modalities; this avoids any biases caused by one dominant modality.

3 Method

Here, we present FunnyNet, its training process and losses (Sections 3.1-3.2). For training labels, we propose an unsupervised laughter detector (Section 3.3).

Overview. FunnyNet consists of (a) three encoders: visual with videos as input, face with face tracks as input, and audio with audios as input, and (b) the proposed Cross-Attention Fusion (CAF) module, which explores cross- and intra-modality correlations by using cross- and self-attentions in the encoders outputs. Then, the fused feature is fed to a binary classifier (Figure 2). For comparison, FunnyNet can take text as extra input with a text encoder. It is trained to embed all modalities in the same space via self-supervised contrastive loss and to classify clips as funny or not. For training, we exploit laughter that naturally exists in TV Shows: we define as 'funny-moment' any audiovisual snippet followed by laughter; and 'not-funny' any audiovisual snippet not followed by laughter.

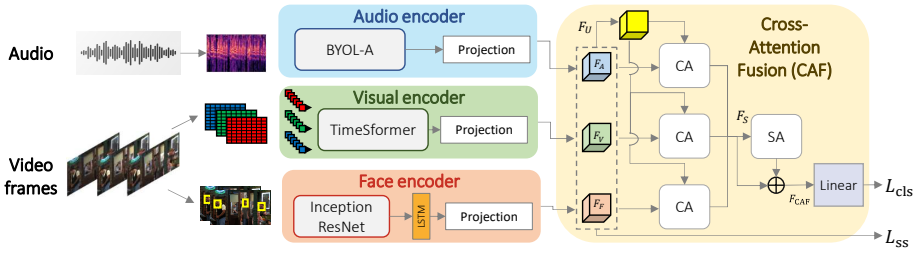

Fig. 2: FunnyNet. Given audio-visual clips, it predicts funny moments in videos. It consists of the audio (*blue*), visual (*green*), and face (*red*) encoders, whose outputs pass through the Cross Attention Fusion (*CAF*), which consists of cross-attention (*CA*) and self-attention (*SA*) for feature fusion. It is trained to embed all modalities in the same space via self-supervision (L_{ss}) and to classify clips as funny or not-funny (L_{cls})

3.1 FunnyNet Architecture

Audio Encoder. FunnyNet takes as input audio snippets X_{audio} in the form of Mel spectrogram[4]. It is fed into the audio encoder, i.e., BYOL-A [21] to obtain a 1D feature vector. Finally, we use a Projection module to map it to a 512-D vector for final prediction: $\mathbf{F}_{\text{A}} \in \mathbb{R}^{512}$.

Visual Encoder. It processes video frames with TimeSformer [45]. Its inputs are patches of size 16×16, partitioned from eight consecutive input frames X_{visual}. Unlike TimeSformer where the representation is obtained by the 'classification token', we obtain the representation by average pooling features from all patches, thus forming a 768-D vector; then, we use a Projection module to map it to a 512-D vector: $F_{\text{V}} \in \mathbb{R}^{512}$. Video context complements audio (or subtitles) to have richer content [11]. If there is no sound, hence no subtitle, visual cues can also provoke laughter.

Face Encoder. Face features capture local cues to enrich the visual representation. We use InceptionResNet [56,57] to extract up to eight faces per frame, that we then process with a LSTM to form a 512-D vector $\mathbf{F}_{\text{F}} \in \mathbb{R}^{512}$. Note, instead of more advanced models [58,59,60], we use InceptionResNet because of its robustness and efficiency [61].

Text Encoder. For a fair comparison to the state of the art, we also experiment with a text encoder that uses BERT [15] to extract key features and feeds them to a LSTM to model temporal correlations (more details in supplementary).

Projection Module. It consists of a linear layer followed by batch normalization, a tanH activation function and another linear layer. It takes input features from each encoder and projects them in a common 512-D feature space.

Cross-Attention Fusion (CAF) learns the cross-domain correlations among vision, audio and face (yellow box Figure 2). It consists of (a) three cross-attention (CA) and (b) one self-attention (SA) modules, described below:

[4] Mel spectrogram is a 2D acoustic time-frequency representation of sound.

(a) Cross-attention is used in cross-domain knowledge transfer to learn across-cue correlations by attending the features from one domain to another [62,48,63]. In CAF, it models the relationship among vision, audio, and face features. We stack all features as $\mathbf{F}_{\mathrm{U}} \in \mathbb{R}^{3\times 512}$, and then feed $\mathbf{F}_{\mathrm{U}}$ into three cross-attention modules to attend to vision, face, and audio, respectively (Figure 2). Next, the scaled attention per modality is computed as $\sigma\left(\frac{\mathbf{Q}_{\mathrm{U}}\mathbf{K}_{\mathbf{i}}^{\mathbf{T}}}{\sqrt{d}}\right)\mathbf{V}_{\mathbf{i}}$, where $i=\{V,F,A\}$ for {vision, face, audio}, and σ the softmax. The query comes from the stacked features: $\mathbf{Q}_{\mathrm{U}}=\mathbf{W}^{\mathrm{Q}_{\mathrm{U}}}\mathbf{F}_{\mathrm{U}}$, while the key and value come from a single modality as $\mathbf{K}_{\mathbf{i}}=\mathbf{W}^{\mathrm{K}_i}\mathbf{F}_{\mathbf{i}}$, and $\mathbf{V}_{\mathbf{i}}=\mathbf{W}^{\mathrm{V}_i}\mathbf{F}_{\mathbf{i}}$. Next, we obtain three cross-attentions and sum them to a unified feature $\mathbf{F}_{\mathrm{S}}$ as:

$$\mathrm{F}_{\mathrm{S}} = \sum_{i\in\{V,F,A\}} \sigma\left(\frac{\mathbf{Q}_{\mathrm{U}}\mathbf{K}_{\mathbf{i}}^{\mathbf{T}}}{\sqrt{d}}\right)\mathbf{V}_{\mathbf{i}} \quad . \tag{1}$$

(b) Self-attention computes the intra-correlation of the F_{S} features, which are further summed with a residual F_{S} as:

$$\mathrm{F}_{\mathrm{CAF}} = \mathbf{F}_{\mathrm{S}} + \sigma\left(\frac{\mathbf{Q}_{\mathrm{S}}\mathbf{K}_{\mathrm{S}}^{\mathbf{T}}}{\sqrt{d}}\right)\mathbf{V}_{\mathrm{S}} \quad , \tag{2}$$

where $\mathbf{Q}_{\mathrm{S}}=\mathbf{W}^{\mathrm{Q}_{\mathrm{S}}}\mathbf{F}_{\mathrm{S}}$, $\mathbf{K}_{\mathrm{S}}=\mathbf{W}^{\mathrm{K}_{\mathrm{S}}}\mathbf{F}_{\mathbf{S}}$, $\mathbf{V}_{\mathrm{S}}=\mathbf{W}^{\mathrm{V}_{\mathrm{S}}}\mathbf{F}_{\mathbf{S}}$. Finally, we average F_{CAF} tokens and feed it to a classification layer.

Discussion. CAF differs to existing methods [62,48] in the computation of the cross attention. Using stacked features F_{U} to attend to each modality Q_{U} brings three benefits: (a) it is order-agnostic: for any modality pair we compute cross-attention once, instead of twice by interchanging queries and keys/values; this results in reduced computation; (b) each modality serves as a query to search for tokens in other modalities; this brings rich feature fusion; and (c) it generalizes to any number of modalities, resulting in scalability.

3.2 Training Model and Loss Functions

Positive and Negative Samples. To create samples, we exploit the laughter that naturally exists in episodes. We define as 'funny' any n-sec clip followed by laughter; 'not-funny' any n-sec clip not followed by laughter. More formally, given a laughter at timestep (t_s, t_e), we extract a n-sec clip at $(t_s{-}n, t_s)$ and we split it into audio and video. For each video, we sample n frames (1 FPS). For the audio, we resample it at 16000 Hz and transform it to Mel spectrogram. Thus, each sample corresponds to n sec and consists of a Mel spectrogram for the audio and a n-frame long video. In practice, we use 8-sec clips as the average time between two canned laughters, and it also leads to better performances (ablations of n-sec clips and n-frames per clip in supplementary). Note that we clip the audio based on the starting time of the laughter so the positive samples do not include any laughter.

Self-Supervised Contrastive Loss. To capture 'mutual' audiovisual information, we solve a self-supervised synchronization task [64,65,66]: we encourage visual features to be correlated with true audios and uncorrelated with audios

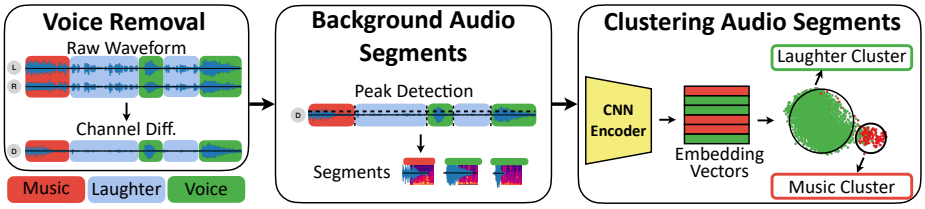

Fig. 3: Proposed laughter detector. It takes raw waveforms as input and consists of (i) removing voices by subtracting channels (here, the audio is stereo with 2 channels), (ii) detecting peaks, and (iii) clustering audios to music and laughter

from other videos. Given the i-th pair of visual v^i and true audio features a^i and N other audios from the same batch: $a_1, ..., a_N$ we minimize the loss [7,67,68]:

$$L_{\text{cotrs}} = -\log \frac{\exp(\text{S}(v^i, a^i)/\tau)}{\sum_{j=1}^{N} \exp(\text{S}(v^i, a^j)/\tau)} \quad , \tag{3}$$

where S the cosine similarity and τ the temperature factor. Equation 3 accounts for audio and visual features. Here, we compute the contrastive loss between all three modalities, i.e., visual-audio, face-audio, and visual-face. Thus, our self-supervised loss is $L_{\text{ss}} = -\frac{1}{3}\left(L_{\text{cotrs}}^{v^i,a^i} + L_{\text{cotrs}}^{v^i,f^i} + L_{\text{cotrs}}^{f^i,a^i}\right)$.

Final Loss. FunnyNet is trained with a Softmax loss Y_{cls} to predict if the input is funny or not, and the L_{ss} to learn 'mutual' information across modalities. Thus, the final loss is: $L{=}\lambda_{\text{ss}}L_{\text{ss}} + \lambda_{\text{cls}}L_{\text{cls}}$, where λ_{ss}, λ_{cls} the weighting parameters that control the importance of each loss.

3.3 Unsupervised Laughter Detection

To detect funny moments automatically, we design an unsupervised laughter detector consisting of three steps (Figure 3). **(i) Remove Voices.** Background audios include sounds, music, laughter; instead, voice (speech) is part of the foreground audio. We remove voices from audios by exploiting multichannel audio specificities. Given raw waveform audios, when the audio is stereo (two channels), the voices are centered and are common in both channels [69]; hence, by subtracting the channels, we remove the voice and keep the background audio. In surround tracks (six channels), we remove the voice channel [69] and keep the background ones. **(ii) Background Audios.** The waveforms from (i) are mostly empty with sparse peaks that correspond to audio: laughter and music. To split them into background and empty segments, we use an energy-based peak detector[5] that detects peaks based on the computed waveform energy. Then, we keep background segments and convert them to log-scaled Mel spectrograms. **(iii) Cluster Audio Segments.** For each laughter and music segment, we extract features using a self-supervised pre-trained encoder. Then, we cluster all audio segments using K-means to distinguish the laughter from the music ones.

[5] https://github.com/amsehili/auditok.

4 Datasets and Metrics

Datasets. We use five datasets (more details in suppl.). **The Big Bang Theory (TBBT)** dataset [4] contains 228 episodes of *TBBT* TV show: (183,23,22) for (train,val,test). All episodes come with video, audio and subtitles, labelled as humor (or non) if followed (or not) by laughter. **Multimodal Humor Dataset (MHD)** [3] contains episodes from *TBBT*, with 110 episodes split (84,6,20) for (train,val,test) (disjoint splits to TBBT). It contains multiple modalities; the subtitles are tagged as humor (or not). **MUStARD** [6] contains 690 segments from 4 TV shows with video-audio-transcript labelled as sarcastic or not. **UR-Funny** [11] contains 1866 TED-talk segments with video-audio-transcript labelled as funny or not. **Friends** [70,71] contains 25 episodes from the third season of *Friends* TV show, split as (1-15,16-20,21-25) for (train,val,test). Each episode contains video, audio, and face tracks. Here, we enrich it with manually annotated laughter time-codes, i.e., starting-ending time of laughter.
Metrics. To evaluate **FunnyNet**, we use classification accuracy (Acc) and F1 score (F1). For **laughter detector**, we use sample-scale at detection level and frame-scale at temporal level to compute precision (Pre), recall (Rec) and F1.

5 Experiments

We provide experiments for FunnyNet. In supplementary, we include more results on feature modalities, modules, impacts, time windows of inputs (n-sec inputs, n frames), losses, automatic/manual laughter, datasets, effect of fine-tuning, metrics, complexity, discussion of laughter detector, and videos.
Implementation Details. We train FunnyNet using Adam optimizer with a learning rate of 10^{-4}, batch size of 32 and Pytorch [72]. At training, we use data augmentation: for frames, we randomly apply rotation and horizontal/vertical flipping, and randomly set the sampling rate to 8 frames; for audios, we apply random forward/backward time shifts and random Gaussian noises. **Setting.** In our experiments, we train FunnyNet on Friends. For MUStARD/UR-Funny, we fine-tune FunnyNet on their respective train sets. For TBBT/MHD, we fine-tune it only with a subset of the training set from TBBT (32 random episodes).

5.1 Comparison to the State of the Art

Here, we evaluate FunnyNet on five datasets: TBBT, MHD, MUStARD, UR-Funny and Friends and compare it to the state of the art: MUStARD [6], MSAM [3], MISA [14], HKT [13] and LaughM [4]. Table 1 reports the results (including random, positive and negative baselines) for both metrics. We indicate the modalities each method uses as V: video, F: face, T: text and A: audio.

Overall, we observe that the proposed audiovisual FunnyNet (V+F+A) outperforms all methods for all metrics on all five datasets. Fsor TBBT it outperforms the LaughM by a notable margin of +5% for F1 and Acc, while for MHD it outperforms MSAM by 3% in F1 and 7% in Acc and LaughM by 3% in Acc.

Table 1: Comparison to the state of the art on five datasets. Modalities used per method A: audio, V: visual frames, F: Face, T: text. †Reproduced results: we use the exact model as in [4], pre-train it on Friends and fine-tune it on the other datasets

Method / Metrics	TBBT		MHD		MUStARD		UR-Funny		Friends	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
Random	46.3	50.0	56.1	50.9	48.3	48.7	50.2	50.2	51.0	51.0
All positive	60.3	43.2	75.6	60.8	66.7	50.0	75.4	50.7	66.7	50.0
All negative	0.0	56.8	0.0	39.2	0.0	50.0	0.0	49.3	0.0	50.0
MUStARD 2019 (V+A+T) [6]	-	-	-	-	71.7	71.8	-	-	-	-
MSAM 2021 (V+T) [3]	-	-	81.3	72.4	-	-	-	-	-	-
MISA 2020 (V+A+T) [14]	-	-	-	-	-	66.2	-	69.8	-	-
HKT 2021 (V+A+T) [13]	-	-	-	-	-	79.4	-	77.4	-	-
LaughM† 2021 (T) [4]	64.2	70.5	**86.5**	76.3	68.6	68.7	71.9	67.6	74.7	59.8
FunnyNet: V+F+A	**69.6**	**74.0**	84.0	**79.3**	**81.4**	**81.0**	**83.7**	**78.0**	**86.8**	**84.8**
FunnyNet: V+A+T	73.8	75.8	83.4	78.6	79.5	79.9	84.1	79.9	88.2	85.8
FunnyNet: V+F+T	**76.0**	69.5	75.9	69.8	75.2	76.3	82.3	73.0	81.3	76.2
FunnyNet: V+F+A+T	75.9	**78.3**	85.2	**79.6**	**83.2**	**82.0**	**84.4**	**80.2**	**88.8**	**86.4**

For MUStARD and UR-Funny, the results are more conclusive as we compare against several methods that use different modalities; in all cases, FunnyNet outperforms MUStARD, MISA, HKT, LaughM by 10-12% in F1 and 2-15% in Acc for MUStARD and 20% in F1 and 1-10% in UF-Funny. For Friends, we observe similar patterns, where we outperform LaughM by 11% in F1 and 25% in Acc. These results confirm the effectiveness of FunnyNet compared to other methods.

Our remarks are: First, FunnyNet performs best among all methods that leverage audio (MUStARD, MISA, HKT), even without using text. Second, the performance in the out-of-domain UR-Funny is significantly high. Third, for TBBT and MHD our results are much less optimized than the ones from LaughM or MSAM, as we do not have access to the exact same test videos as either work, so inevitably there are some time shifts or wrong labels[6] and we use much fewer training data (32 vs 183 episodes in LaughM vs 84 episodes in MHD). These highlight that FunnyNet is an effective model for funny moment detection.

FunnyNet Using Text. For fair comparison, we explore a FunnyNet version that leverages subtitles in addition to audiovisual cues (FunnyNet: V+A+T). We compare it against MUStARD, MISA, and HKT that use the same modalities and observe that FunnyNet (V+A+T) outperforms them all by a large margin (1-14% for all metrics and datasets). This shows that the performance boost from FunnyNet stems from the superior architecture and the adequate modality fusion, rather than the difference in input modalities.

5.2 Analysis of Unsupervised Laughter Detector

We compare our laughter detector to the state-of-the-art LD [25] used in [6] and RLD [26]. Table 2 reports the results on Friends. We observe that overall, our detector outperforms both supervised ones. We also examine the efficiency

[6] The label time shift is 0.3-1s on TBBT and 0.3-2s on v2.

Table 2: Laughter detection evaluation on Friends. We compare 'Ours' to two audio feature extractors

	Temporal				**Det** IoU = 0.3			**Det** IoU = 0.7		
	Acc	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
LD [25]	43.64	35.70	**98.95**	52.28	25.69	22.09	23.35	4.02	3.73	3.82
RLD [26]	74.46	58.91	61.98	59.69	66.15	53.71	59.10	18.45	15.04	16.52
Ours Wav2CLIP[73]	77.56	64.49	63.66	63.70	91.25	61.23	73.07	49.74	33.45	39.89
Ours BYOL-A[21]	**85.97**	**76.94**	79.38	**77.81**	**94.57**	**82.25**	**87.83**	**54.07**	**47.11**	**50.27**

Table 3: Ablation of modalities of FunnyNet on the test set of all five datasets (A: audio, V: visual frames, F: face)

Modalities			**TBBT**		**MHD**		**MUStard**		**URFunny**		**Friends**	
A	**V**	**F**	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
✓	-	-	68.9	63.2	64.9	67.5	65.2	63.8	68.2	64.7	73.7	66.7
-	✓	-	65.0	58.9	66.3	66.5	64.6	60.2	67.5	60.4	72.9	63.4
-	-	✓	64.8	59.4	66.8	67.7	64.7	61.1	67.2	61.2	72.9	62.1
✓	✓	-	**70.3**	72.8	79.9	73.1	79.0	77.5	80.9	76.8	84.2	81.1
✓	-	✓	**70.3**	72.9	80.5	73.9	80.4	78.9	82.9	**79.4**	83.9	82.2
✓	✓	✓	69.6	**74.0**	**84.0**	**79.3**	**81.4**	**81.0**	**83.7**	78.0	**86.8**	**84.8**

Table 4: Ablation of CAF of FunnyNet on Friends test set

CAF		**A+V**		**A+F**		**A+V+F**	
Self	**Cross**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**
-	-	**84.51**	80.65	82.62	78.99	85.97	84.02
✓	-	84.10	80.64	**84.01**	81.01	86.57	84.13
-	✓	83.13	81.05	83.71	81.33	86.61	84.52
✓	✓	84.19	**81.14**	83.91	82.22	**86.79**	**84.75**
MMCA [48]		83.56	81.05	83.44	82.06	86.71	84.36
CoMMA [49]		83.71	81.08	83.79	**82.24**	86.69	84.63

of BYOL-A [21] and Wav2CLIP [73] encoders, where we observe that using BYOL-A outperforms Wav2CLIP, due to the richer audio representation capacity. From the laughters, we make three notes: (a) most false positives are unfiltered sounds not well separable by K-means; (b) most false negatives are intra-diegetic laughter, which is less loud, and hence, less detectable; (c) when the music is superposed with the laughter (e.g. party) the peak detector fails.

5.3 Ablation of FunnyNet

Modalities. Table 3 reports the ablation of all modalities of FunnyNet on five datasets. Using audio alone produces better results than any visual modality alone, underlying that audio is more suitable than visual cues for our task, as it encompasses the way of speaking (tone, pauses). Combining modalities outperforms using single ones: combining audio and visual increases the F1 by 3-12% and the Acc by 6-17%. This is expected as modalities bring complementarity and their combination helps discriminate funny moments. Audio+face leads to smaller boosts than audio+visual, as frames capture better than (low-level) faces global information. Overall, using all modalities achieves the best performance.
Cross-Attention Fusion (CAF). Table 4 reports results with various cross- and self-attention fusions in CAF. We observe that including either self- or cross-attention (second, third rows) brings improvements over not having any (first row), indicating that they enhance the feature representation. The fourth row shows that using them both for feature fusion leads to the best performance. For completeness, we also compare CAF against the state of the art MMCA [48] and CoMMA [49]. All CAF, MMCA and CoMMA use self and cross-attentions jointly

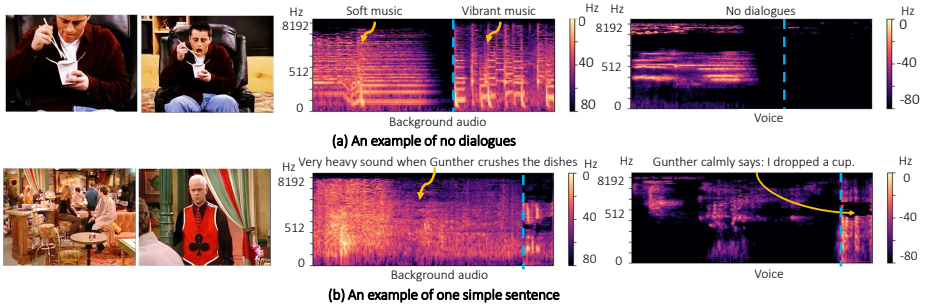

Fig. 4: Audio vs Text for funny-moment detection on Friends. Relying solely on voices (subtitles) fails when nobody is speaking; the audio, however, may succeed. (a) humorous background music without voices, (b) abrupt background sound (plates smashing) accompanied by a simple dialogue 'I dropped a cup'

for feature extraction. Their main difference is that both MMCA and CoMMA first use self-attention to individually process each modality, then concatenate all modalities together and process them using cross-attention to output the final feature representation. Instead, CAF uses cross-attention to gradually fuse one modality with the rest of modalities to fully explore cross-modal correlations. The results (fifth, last rows) show that CAF outperforms MMCA [48] and CoMMA [49] by 0.1$\sim$ 0.4 in F1 score and 0.03$\sim$ 0.2 in accuracy. This reveals the importance of the gradual modality fusion and hence the superiority of CAF.

6 Analysis of FunnyNet

6.1 Audio vs Subtitles

Instead of subtitles [4], FunnyNet relies on audio, as it (1) encodes mutual information to the text, (2) does not encode only words, but also the way of speaking (pauses, pitch and so on), and (3) can succeed when a scene is funny, yet nobody is speaking, by exploiting background sounds.

Quantitative Analysis. Table 1 reports FunnyNet results with the combination of text and audiovisual modalities. By comparing visual-textual to visual-audio features (V+F+T vs V+F+A) we observe that in most cases audiovisual cues perform better than the visual-textual ones (4-8% in both metrics for all datasets); this corroborates our claim that audio identifies better than text funny moments in videos. The last row reports results when combining all modalities (V+F+A+T). This further outperforms all other FunnyNet results, indicating that FunnyNet can efficiently exploit all sources of signals for funny prediction.

Qualitative Analysis. Figure 4 shows some scene frames from Friends and their audio, separated into background (middle) and vocal/voice (bottom) [74]. (a) Joey is trying to use chopsticks to pick up the food but end up dropping it. There is no voice, but vibrant and active music with a simpler rhythm to Joey's action.

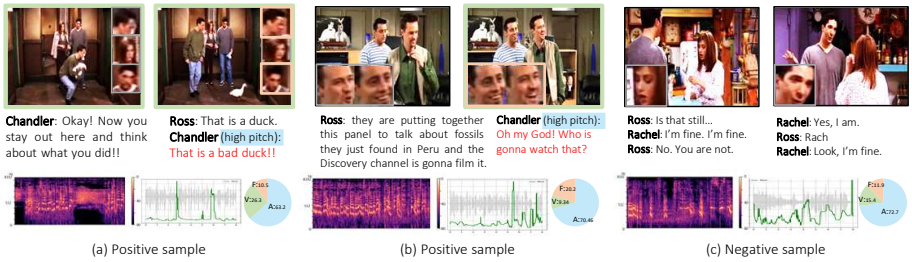

Fig. 5: Visualization of (*a*,*b*) funny, (*c*) non-funny predictions on Friends. We show the audio and visual (frame and faces) inputs, the learned average weights of cross-attentions from CAF (*pie chart*), and the subtitles (for better understanding)

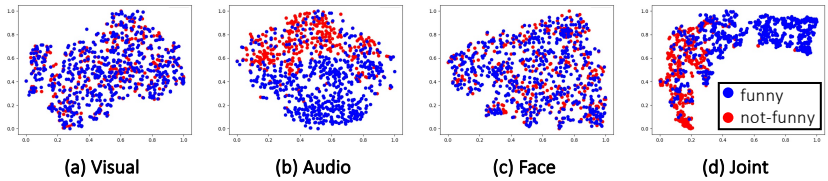

Fig. 6: t-SNE visualization of embeddings on Friends for (*a*) visual, (*b*) audio, (*c*) face, (*d*) all modalities. We show positive (*blue*) and negative samples (*red*)

FunnyNet correctly predicts the scene as funny by leveraging audio. (b) The extreme loud sound of several smashing dishes is followed by Gunther appearing and calmly saying "I drop A cup". Unlike text, the audio correctly detects the contradiction between the smashing sound and the calm words, hence predicting the scene as funny. In both cases, using text results in incorrect predictions, whereas audio successfully leverages no verbal cues and correctly predicts the scenes as funny, showing that audio better than text addresses such cases.

6.2 FunnyNet Architecture

Modality Impact. To visualize the impact of modalities, we compute the average attention values on the three CA modules (CA boxes in Figure 2) and then, show the average weights for each modality in the pie chart of Figure 5. For this, we show two positive and one negative samples on Friends with frames, face (on frames), and audio spectrogram (left) and pitch (right). Note, FunnyNet does not use subtitles; we show dialogues only for a better understanding. We observe that the contribution of each modality varies; the commonality though is that audio contributes more than half, followed by visual and finally face. When characters smile ('Chandler' in b), the contribution of face increases, indicating the importance of facial expressions, whereas the 'over the shoulder' shot of (c) shows that face posture play a small role. Moreover, the dramatic peaks of the audio pitch in (a,b) show that they are associated with funny punchline ('That is a bad duck'), whereas the smooth ones in (c) imply a 'normal scene'.

Fig. 7: Top-5 predicted positive and negative faces of 'Chandler' and 'Rachel' from Friends

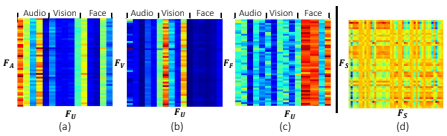

Fig. 8: CAF attention maps on Friends. (a,b,c) CA between F_U and audio, vision and face; (d) SA on F_U

Fig. 9: Failure cases on Friends

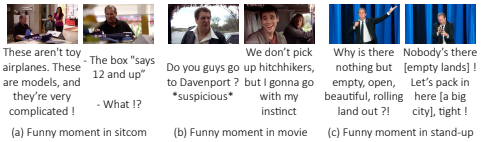

Fig. 10: Funny moments in the wild

Feature Visualization. Figure 6 shows the t-SNE [75] visualization of features: (a-c) visual, audio, face, (d) all with blue for funny and red for not-funny samples. All single features, and in particular the visual and facial ones, scattered around the centre of the 2D space without clear boundaries between positives and negatives. However, the joint embedding shows clear separation between funny and not-funny, thus revealing the effectiveness of FunnyNet.

Impact of Face Encoder. To examine its effect, in Figure 7 we depict the top-5 detected faces from positive (top) and negative (bottom) samples of two characters from Friends. We observe that all positives have rich expressions (yell, stare, smile, open mouth), while the negatives are either of bad quality (no useful features) or show neutral expression, thus indicating scenes without funniness.

Impact of CAF Module. To examine the effect of CAF, we visualize in Figure 8 the learned attention maps: red indicates higher, and blue lower attention. (a,b,c) display the cross-attention between fused F_{U} and (a) audio, (b) visual, (c) face features. Since F_{U} is stacked from audio, vision, face, we observe that each modality highly attends to itself. We also observe in (a), both the visual and the face in F_{U} fire with the audio, thus indicating that FunnyNet captures correlations between audio and visual expressions or movements (e.g. character laughing). Then, in (c), all modalities attend to the face features, thus revealing their mutual information. Finally, (c) displays the self-attention map between F_{S}, where we observe that F_{S} attends to all tokens with different weights.

Failure Analysis. We note two groups of failure cases. First, when characters laugh sarcastically is not always funny; but, all modalities incorrectly, yet confidently tag them as funny. Figure 9-(a) shows this, where 'Rachel' laughs sarcastically, which is not funny (subs 'ha ha'). We wrongly predict it as positive. Second, visual cues fail in dark scenes; thus, we rely on audio. Figure 9-(b) shows a night scene with no clear faces and dark frames, where FunnyNet uses mostly the non-discriminative audio; hence, we wrongly predict the scene as negative.

6.3 Funny Scene Detection in the Wild

We show applications of FunnyNet in videos from other domains (more in suppl.). **1. Sitcoms without Canned Laughter.** We collect 9 episodes of the first season (∼180 minutes) of *Modern Family* (Lloyd and Levitan, 2009)[7] without canned laughter. We manually annotate as positive every punchline that could lead to laughter, resulting in 453 positives (we will make them available). We apply FunnyNet on the 8-secs preceding funny moments, resulting in an accuracy of 55.4% vs 50% for random. Our remarks are: (i) FunnyNet is not fine-tuned on this data, and (ii), *Modern Family* differs from other sitcoms with live audience as characters are not reacting to punchlines. Thus, our results indicate that FunnyNet is capable of detecting funny-moments in out-of-domain cases. Figure 10 (a) shows a correctly predicted funny moment between two characters who vary their speech rhythm and tones. **2. Movies with Diverse Funny Styles.** Figure 10 (b) depicts such an example from the *Dumb and Dumber* film (Farrelly, 1994). FunnyNet correctly detects funny moments followed by silence or a speaker's change of tone. **3. Stand-Up Comedies** contain several punchlines that make audiences laugh. We experiment on the Jerry Seinfeld *23 Hours to Kill* stand-up comedy. Figure 10 (c) shows that FunnyNet detects funny moments correctly and confidently as Jerry is highly expressive (expressions, gestures). **4. Audio-Only.** As audio is the most discriminative cue, we examine its impact on out-of-domain audios: narrating jokes and reading books. FunnyNet detects funny punchlines from jokes, mostly when they are accompanied by a change of pitch or pause; for the audiobook, it successfully detects funny moments when the reader's voice imitates a character.

7 Conclusions

We introduced FunnyNet, an audiovisual model for funny moment detection. In contrast to works that rely on text, FunnyNet exploits audio that comes naturally with videos and contains high-level cues (pauses, tones, etc). Our findings show audio is the dominant cue for signaling funny situations, while video offers complementary information. Extensive analysis and visualizations also support our finding that audio is better than text (in the form of subtitles) when it comes to scenes with no or simple dialogue but with hilarious acting or funny background sounds. Our results show the effectiveness of each component of FunnyNet, which outperforms the state of the art on the TBBT, MUStARD, MHD, UR-Funny and Friends. Future work includes analyzing the contribution of audio cues (pitch, tone, etc).

8 Acknowledgements

We would like to thank Dim P. Papadopoulos and Xi Wang for proofeading and the anonymous reviewers for their feedback. This work was supported by a DIM RFSI grant, and the ANR projects WhyBehindScenes and APATE.

[7] https://www.youtube.com/playlist?list=PL8v3aNB88WMM0iwOUeLpgFf3pHH9uxz7_.

References

1. Annamoradnejad, I., Zoghi, G.: Colbert: Using bert sentence embedding for humor detection. arXiv preprint arXiv:2004.12765 (2020) 1, 3
2. Weller, O., Seppi, K.: The rjokes dataset: a large scale humor collection. In: LREC. (2020) 1, 3
3. Patro, B.N., Lunayach, M., Srivastava, D., Sarvesh, S., Singh, H., Namboodiri, V.P.: Multimodal humor dataset: Predicting laughter tracks for sitcoms. In: WACV. (2021) 2, 3, 8, 9
4. Kayatani, Y., Yang, Z., Otani, M., Garcia, N., Chu, C., Nakashima, Y., Takemura, H.: The laughing machine: Predicting humor in video. In: WACV. (2021) 2, 3, 8, 9, 11
5. Zadeh, A., Liang, P.P., Mazumder, N., Poria, S., Cambria, E., Morency, L.P.: Memory fusion network for multi-view sequential learning. In: AAAI. (2018) 2
6. Castro, S., Hazarika, D., Pérez-Rosas, V., Zimmermann, R., Mihalcea, R., Poria, S.: Towards multimodal sarcasm detection (an *Obviously* perfect paper). In: ACL. (2019) 2, 3, 8, 9
7. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: Proc. ICML. (2020) 2, 7
8. Davidov, D., Tsur, O., Rappoport, A.: Semi-supervised recognition of sarcastic sentences in twitter and amazon. In: ACL. (2010) 3
9. Rockwell, P.: Lower, slower, louder: Vocal cues of sarcasm. Journal of Psycholinguistic research (2000) 3
10. Tepperman, J., Traum, D., Narayanan, S.S.: 'yeah right': Sarcasm recognition for spoken dialogue systems. In: INTERSPEECH. (2006) 3
11. Hasan, M.K., Rahman, W., Bagher Zadeh, A., Zhong, J., Tanveer, M.I., Morency, L.P., Hoque, M.E.: UR-FUNNY: A multimodal language dataset for understanding humor. In: EMNLP-IJCNLP. (2019) 3, 5, 8
12. Bertero, D., Fung, P.: Deep learning of audio and language features for humor prediction. In: LREC. (2016) 3
13. Hasan, M.K., Lee, S., Rahman, W., Zadeh, A., Mihalcea, R., Morency, L.P., Hoque, E.: Humor knowledge enriched transformer for understanding multimodal humor. In: AAAI. (2021) 3, 8, 9
14. Hazarika, D., Zimmermann, R., Poria, S.: Misa: Modality-invariant and-specific representations for multimodal sentiment analysis. Proceedings of the 28th ACM International Conference on Multimedia (2020) 3, 8, 9
15. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL. (2019) 3, 5
16. Rahman, W., Hasan, M.K., Lee, S., Bagher Zadeh, A., Mao, C., Morency, L.P., Hoque, E.: Integrating multimodal information in large pretrained transformers. In: ACL. (2020) 3
17. Mesaros, A., Heittola, T., Virtanen, T.: Tut database for acoustic scene classification and sound event detection. In: 24th European Signal Processing Conference (EUSIPCO). (2016) 4
18. Défossez, A., Usunier, N., Bottou, L., Bach, F.: Music source separation in the waveform domain. arXiv preprint arXiv:1911.13254 (2019) 4
19. Mesaros, A., Heittola, T., Benetos, E., Foster, P., Lagrange, M., Virtanen, T., Plumbley, M.D.: Detection and classification of acoustic scenes and events: Outcome of the dcase 2016 challenge. ACM Transactions on Audio, Speech, and Language Processing (2017) 4

20. Wang, L., Luc, P., Recasens, A., Alayrac, J.B., Oord, A.v.d.: Multimodal self-supervised learning of general audio representations. arXiv preprint arXiv:2104.12807 (2021) 4
21. Niizumi, D., Takeuchi, D., Ohishi, Y., Harada, N., Kashino, K.: Byol for audio: Self-supervised learning for general-purpose audio representation. In: 2021 International Joint Conference on Neural Networks (IJCNN). (2021) 4, 5, 10
22. Saeed, A., Grangier, D., Zeghidour, N.: Contrastive learning of general-purpose audio representations. In: ICASSP. (2021) 4
23. Barral, O., Kosunen, I., Jacucci, G.: No need to laugh out loud: Predicting humor appraisal of comic strips based on physiological signals in a realistic environment. ACM Transactions on Computer-Human Interaction (TOCHI) (2017) 4
24. Shimasaki, A., Ueoka, R.: Laugh log: E-textile bellyband interface for laugh logging. In: Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems. (2017) 4
25. Ryokai, K., Durán López, E., Howell, N., Gillick, J., Bamman, D.: Capturing, representing, and interacting with laughter. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. (2018) 4, 9, 10
26. Gillick, J., Deng, W., Ryokai, K., Bamman, D.: Robust laughter detection in noisy environments. INTERSPEECH (2021) 4, 9, 10
27. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation (1997) 4
28. Gao, Y., Glowacka, D.: Deep gate recurrent neural network. In: ACCV. (2016) 4
29. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR. (2020) 4
30. Walker, J., Razavi, A., Oord, A.v.d.: Predicting video with vqvae. arXiv preprint arXiv:2103.01950 (2021) 4
31. Gabbay, A., Ephrat, A., Halperin, T., Peleg, S.: Seeing through noise: Visually driven speaker separation and enhancement. In: ICASSP. (2018) 4
32. Afouras, T., Chung, J.S., Zisserman, A.: The conversation: Deep audio-visual speech enhancement. In: INTERSPEECH. (2020) 4
33. Senocak, A., Oh, T.H., Kim, J., Yang, M.H., Kweon, I.S.: Learning to localize sound source in visual scenes. In: CVPR. (2018) 4
34. Tian, Y., Shi, J., Li, B., Duan, Z., Xu, C.: Audio-visual event localization in unconstrained videos. In: ECCV. (2018) 4
35. Aytar, Y., Vondrick, C., Torralba, A.: See, hear, and read: Deep aligned representations. arXiv preprint arXiv:1706.00932 (2017) 4
36. Engel, J., Agrawal, K.K., Chen, S., Gulrajani, I., Donahue, C., Roberts, A.: Gansynth: Adversarial neural audio synthesis. In: ICLR. (2019) 4
37. Zhou, H., Sun, Y., Wu, W., Loy, C.C., Wang, X., Liu, Z.: Pose-controllable talking face generation by implicitly modularized audio-visual representation. In: CVPR. (2021) 4
38. Owens, A., Isola, P., McDermott, J., Torralba, A., Adelson, E.H., Freeman, W.T.: Visually indicated sounds. In: CVPR. (2016) 4
39. Zhou, H., Xu, X., Lin, D., Wang, X., Liu, Z.: Sep-stereo: Visually guided stereophonic audio generation by associating source separation. In: ECCV. (2020) 4
40. Dong, J., Li, X., Xu, C., Yang, X., Yang, G., Wang, X., Wang, M.: Dual encoding for video retrieval by text. IEEE TPAMI (2021) 4
41. Dufour, N., Picard, D., Kalogeiton, V.: Scam! transferring humans between images with semantic cross attention modulation. In: ECCV. (2022) 4

42. Liang, Z., Jiang, W., Hu, H., Zhu, J.: Learning to contrast the counterfactual samples for robust visual question answering. In: EMNLP. (2020) 4
43. Deng, D., Zhou, Y., Pi, J., Shi, B.E.: Multimodal utterance-level affect analysis using visual, audio and text features. arXiv preprint arXiv:1805.00625 (2018) 4
44. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: ICCV. (2021) 4
45. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? In: Proc. ICML. (2021) 4, 5
46. Ryoo, M.S., Piergiovanni, A., Arnab, A., Dehghani, M., Angelova, A.: Tokenlearner: What can 8 learned tokens do for images and videos? arXiv preprint arXiv:2106.11297 (2021) 4
47. Jaegle, A., Gimeno, F., Brock, A., Zisserman, A., Vinyals, O., Carreira, J.: Perceiver: General perception with iterative attention. Proc. ICML (2021) 4
48. Wei, X., Zhang, T., Li, Y., Zhang, Y., Wu, F.: Multi-modality cross attention network for image and sentence matching. In: CVPR. (2020) 4, 6, 10, 11
49. Tan, R., Plummer, B.A., Saenko, K., Jin, H., Russell, B.: Look at what i'm doing: Self-supervised spatial grounding of narrations in instructional videos. In: NeurIPS. (2021) 4, 10, 11
50. Lee, J.T., Jain, M., Park, H., Yun, S.: Cross-attentional audio-visual fusion for weakly-supervised action localization. In: ICLR. (2020) 4
51. Nagrani, A., Yang, S., Arnab, A., Jansen, A., Schmid, C., Sun, C.: Attention bottlenecks for multimodal fusion. In: NeurIPS. (2021) 4
52. Narasimhan, M., Rohrbach, A., Darrell, T.: Clip-it! language-guided video summarization. In: NeurIPS. (2021) 4
53. Gabeur, V., Sun, C., Alahari, K., Schmid, C.: Multi-modal transformer for video retrieval. In: ECCV. (2020) 4
54. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: A joint video and image encoder for end-to-end retrieval. In: ICCV. (2021) 4
55. Epstein, D., Vondrick, C.: Learning goals from failure. In: CVPR. (2021) 4
56. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR. (2015) 5
57. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: CVPR. (2016) 5
58. Wang, C., Fang, H., Zhong, Y., Deng, W.: Mlfw: A database for face recognition on masked faces. arXiv preprint arXiv:2109.05804 (2021) 5
59. Chrysos, G.G., Moschoglou, S., Bouritsas, G., Deng, J., Panagakis, Y., Zafeiriou, S.P.: Deep polynomial neural networks. IEEE TPAMI (2021) 5
60. Chou, H.R., Lee, J.H., Chan, Y.M., Chen, C.S.: Data-specific adaptive threshold for face recognition and authentication. In: 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). (2019) 5
61. Du, H., Shi, H., Zeng, D., Zhang, X.P., Mei, T.: The elements of end-to-end deep face recognition: A survey of recent advances. ACM Computing Surveys (CSUR) (2020) 5
62. Mohla, S., Pande, S., Banerjee, B., Chaudhuri, S.: Fusatnet: Dual attention based spectrospatial multimodal fusion network for hyperspectral and lidar classification. In: CVPRW. (2020) 6
63. Nam, H., Ha, J.W., Kim, J.: Dual attention networks for multimodal reasoning and matching. In: CVPR. (2017) 6
64. Chung, J.S., Zisserman, A.: Out of time: automated lip sync in the wild. In: ACCV. (2016) 6

65. Korbar, B.: Co-training of audio and video representations from self-supervised temporal synchronization. In: CoRR. (2018) 6
66. Owens, A., Efros, A.A.: Audio-visual scene analysis with self-supervised multisensory features. In: ECCV. (2018) 6
67. Chung, S.W., Chung, J.S., Kang, H.G.: Perfect match: Improved cross-modal embeddings for audio-visual synchronisation. In: ICASSP. (2019) 7
68. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018) 7
69. Huber, D.M., Runstein, R.: Modern recording techniques. Routledge (2012) 7
70. Kalogeiton, V., Zisserman, A.: Constrained video face clustering using 1nn relations. In: BMVC. (2020) 8
71. Brown, A., Kalogeiton, V., Zisserman, A.: Face, body, voice: Video person-clustering with multiple modalities. In: ICCV. (2021) 8
72. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: NeurIPS. (2019) 8
73. Wu, H.H., Seetharaman, P., Kumar, K., Bello, J.P.: Wav2clip: Learning robust audio representations from clip. arXiv preprint arXiv:2110.11499 (2021) 10
74. Hennequin, R., Khlif, A., Voituret, F., Moussallam, M.: Spleeter: a fast and efficient music source separation tool with pre-trained models. Journal of Open Source Software (2020) 11
75. Hinton, G., Roweis, S.: Stochastic neighbor embedding. In: NeurIPS. (2002) 13