Learning Representations from Audio-Visual Spatial Alignment

Yi Li* Pedro Morgado* Nuno Vasconcelos
UC San Diego
{yil898,pmaravil,nuno}@eng.ucsd.edu
https://github.com/UCSD-SVCL/AVSpatialAlignment

Abstract

We introduce a novel self-supervised pretext task for learning representations from audio-visual content. Prior work on audio-visual representation learning leverages correspondences at the video level. Approaches based on audio-visual correspondence (AVC) predict whether audio and video clips originate from the same or different video instances. Audio-visual temporal synchronization (AVTS) further discriminates negative pairs originated from the same video instance but at different moments in time. While these approaches learn high-quality representations for downstream tasks such as action recognition, their training objectives disregard spatial cues naturally occurring in audio and visual signals. To learn from these spatial cues, we tasked a network to perform contrastive audio-visual spatial alignment of 360° video and spatial audio. The ability to perform spatial alignment is enhanced by reasoning over the full spatial content of the 360° video using a transformer architecture to combine representations from multiple viewpoints. The advantages of the proposed pretext task are demonstrated on a variety of audio and visual downstream tasks, including audio-visual correspondence, spatial alignment, action recognition and video semantic segmentation.

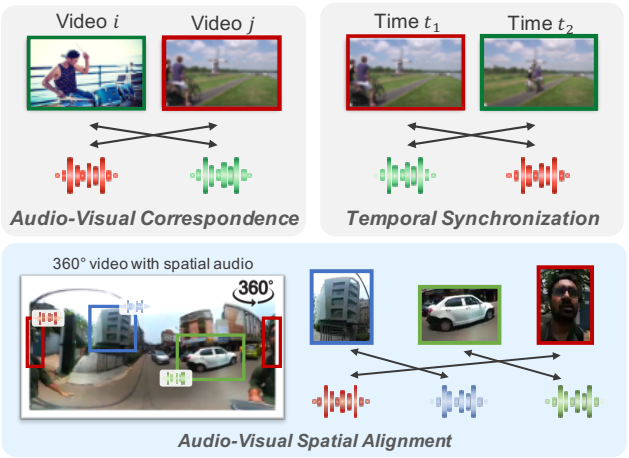

Figure 1: Audio-visual spatial alignment. Prior work on audio-visual representation learning leverages correspondences at the video level. Audio-visual correspondence (AVC) [1, 12] predicts whether a pair of audio and video clips originate from the same video (positive) or different videos (negative). Audio-visual temporal synchronization (AVTS) [14, 7] discriminates negative pairs that are sampled from the same video but different moments in time. However, prior work ignores the spatial cues of audio-visual signals. Instead, we learn representations by performing audio-visual spatial alignment (AVSA) of 360°video and spatial audio. This is accomplished by training a model to distinguish audio and video clips extracted from different viewpoints.

1. Introduction

In computer vision, the natural co-occurrence of audio and video has been extensively studied. Prior work has shown that this co-occurrence can be leveraged to learn representations in a self-supervised manner, i.e., without human annotations. A common approach is to learn to match audio and video clips of the same video instance [1, 12]. Prior work has also demonstrated the value of temporal synchronization between audio and video clips for learning representations for downstream tasks such as action recognition [7, 14]. Since these methods do not need to localize sound sources, they struggle to discriminate visual concepts that often co-occur. For example, the sound of a car can be quite distinctive, and thus it is a good target description for the "car" visual concept. However, current approaches use this audio as a descriptor for the whole video clip, as opposed to the region containing the car. Since cars and roads often co-occur, there is an inherent ambiguity about which of the two produce the sound. This makes it is hard to learn good representations for visual concepts like "cars", distinguishable from co-occurring objects like "roads" by pure audio-visual correspondence or temporal synchronization.

To address this issue, we learn representations by training deep neural networks with 1) 360°video data that contain audio-visual signals with strong spatial cues and 2) a pretext task to conduct audio-visual spatial alignment (AVSA, Figure 1). Unlike regular videos with mono audio recordings, 360°video data and spatial audio formats like ambisonics fully capture the spatial layout of audio and visual content within a scene. To learn from this spatial information, we collected a large 360°video dataset, five times

*Equal contribution.

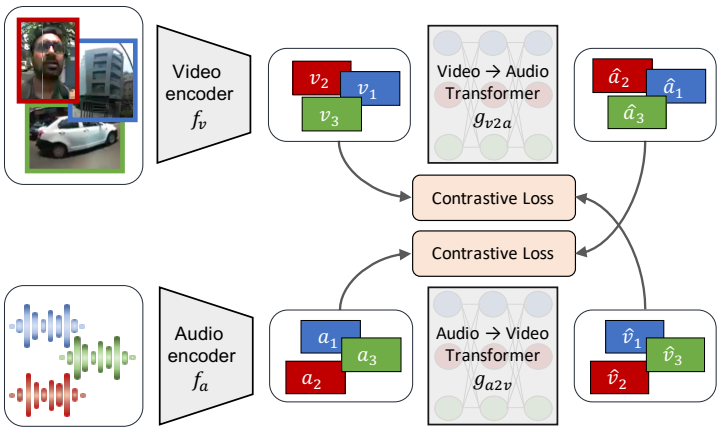

Figure 2: Architecture overview for contrastive audio-visual spatial alignment.

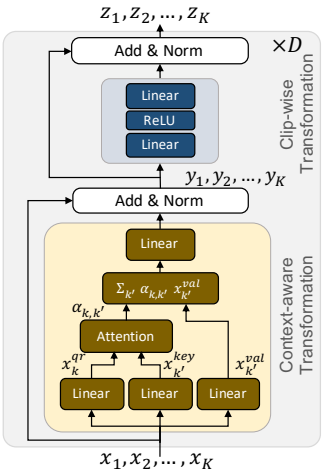

Figure 3: Transformer architecture for context-aware video-to-audio and audio-to-video feature translation.

larger than currently available datasets. We also designed a pretext task where audio and video clips are sampled from different viewpoints within a 360°video, and spatially misaligned audio/video clips are treated as negatives examples for contrastive learning. To enhance the learned representations, two modifications to the standard contrastive learning setup are proposed. First, the ability to perform spatial alignment is boosted using a curriculum learning strategy that initially focus on learning audio-visual correspondences at the video level. Second, we propose to reason over the full spatial content of the 360°video by combining representations from multiple viewpoints using a transformer network. We show the benefits of the AVSA pretext task on a variety of audio and visual downstream tasks, including audio-visual correspondence and spatial alignment, action recognition and video semantic segmentation.

2. Audio-visual spatial alignment

We learn audio-visual representations by leveraging spatial cues in 360°media. 360°video and spatial audio encode visual and audio signals arriving from all directions (θ, ϕ) around the recording location, where θ denotes the longitude (or horizontal) angle, ϕ the latitude (or elevation) angle. We adopt the equi-rectangular projection as the 360°video format and first-order ambisonics [4] for the spatial audio. Both formats can be easily rotated and/or decoded into viewpoint specific clips.

2.1. Contrastive AVSA

Inspired by recent advances in contrastive learning [13, 12] , we propose to solve the audio-visual spatial alignment task in a contrastive fashion. As shown in Figure 1, given a 360°audio-video sample (v_i, a_i), K video and audio clips $\{(v_i^k, a_i^k)\}_{k=1}^K$ are extracted from K randomly sampled viewing directions $\{(\theta_k, \phi_k)\}_{k=1}^K$. Video clips v_i^k are obtained by extracting normal field-of-view (NFOV) crops using a Gnomonic projection centered around (θ_k, ϕ_k), and audio clips a_i^k by realigning the global frame of reference of the ambisonics signal such that the frontal direction points towards (θ_k, ϕ_k) [8]. Audio-visual spatial alignment is then encouraged by tasking a network to predict the correct correspondence between the K video $\{v_i^k\}_{k=1}^K$ and the K audio $\{a_i^k\}_{k=1}^K$ signals.

2.2. Architecture

Figure 2 summarizes the architecture used to solve the spatial alignment task. First, video and audio encoders, f_v and f_a, extract feature representations from each clip independently, $\mathbf{v}_i^k = f_v(v_i^k)$ and $\mathbf{a}_i^k = f_a(a_i^k)$. These representations are then converted between the two modalities using audio-to-video g_{a2v} and video-to-audio g_{v2a} feature translation networks

$$\bar{\mathbf{v}}_i^1, \ldots, \bar{\mathbf{v}}_i^K = g_{a2v}(\mathbf{a}_i^1, \ldots, \mathbf{a}_i^K), \quad (1)$$

$$\bar{\mathbf{a}}_i^1, \ldots, \bar{\mathbf{a}}_i^K = g_{v2a}(\mathbf{v}_i^1, \ldots, \mathbf{v}_i^K). \quad (2)$$

One important distinction between audio and video is the spatial localization of the signals. Unlike video, any sound source can be heard regardless of the listening angle. In other words, while an audio clip a_i^k sampled at position (θ_k, ϕ_k) contains audio from all sound sources present in a scene, only those physically located around (θ_k, ϕ_k) can be seen on the video clip v_i^k. This implies that, to enable accurate feature translation, networks g_{v2a} and g_{a2v} should combine features from all sampled locations. This is ac-

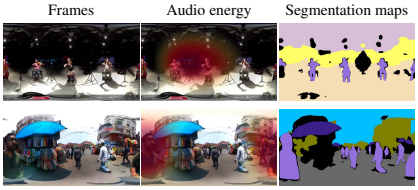

Figure 4: Examples from YT-360 dataset.

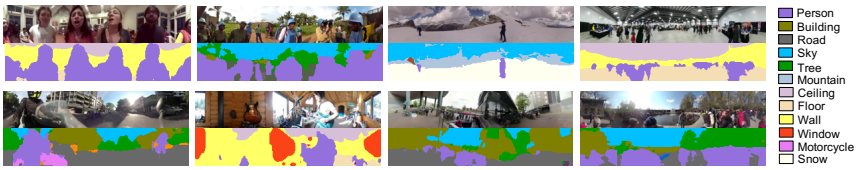

Figure 5: Segmentation results from an AVSA pre-trained model on the YT-360 test set.

complished by a translation network similar to the transformer of [17]. As shown in Fig. 3, given a set of K features $\{\mathbf{x}_k\}_{k=1}^K$, a transformer of depth D alternates D times between two modules. The first module combines the K features $\mathbf{x}_k$ using attention

$$\{\alpha_{k,l}\}_{l=1}^K = \mathit{Softmax}\left(\left\{\frac{\langle W_{key}^T\mathbf{x}_k, W_{qr}^T\mathbf{x}_l\rangle}{\sqrt{d}}\right\}_{l=1}^K\right), \quad (3)$$

$$\mathbf{y}_k = \mathit{Norm}\left(\mathbf{x}_k + W_0^T\textstyle\sum_l \alpha_{k,l} W_{val}^T\mathbf{x}_l\right). \quad (4)$$

The second module computes a simple clip-wise feed-forward transformation

$$\mathbf{z}_k = \mathit{Norm}\left(\mathbf{y}_k + W_2^T\max(W_1^T\mathbf{y}_k, 0)\right). \quad (5)$$

In (3)-(5), $W_{key}, W_{qr}, W_{val}, W_0, W_1$ and W_2 are learnable weights and *Norm* is layer normalization [2]. We omit the biases of linear transformations for simplicity of notation.

2.3. Learning strategy

AVSA is a difficult task to optimize since it requires discriminating between various crops from the same video. To enhance learning, we employed a curriculum learning strategy [3]. In the first phase, the network is trained to identify audio-visual correspondences (AVC) [1, 12] at the video level. This is accomplished by extracting a single crop (v_i, a_i) for each video i from a randomly drawn viewing angle. The visual and audio encoders, f_v and f_a, are then trained to minimize

$$L_{AVC} = \sum_i L_{InfoNCE}\left(\mathbf{v}_i, \mathbf{a}_i, \{\mathbf{a}_j\}_{j=1}^N\right) + L_{InfoNCE}\left(\mathbf{a}_i, \mathbf{v}_i, \{\mathbf{v}_j\}_{j=1}^N\right) \quad (6)$$

where $\mathbf{v}_i = f_v(v_i)$ and $\mathbf{a}_i = f_a(a_i)$ are the video and audio representations. $L_{InfoNCE}$ is the InfoNCE loss [13]

$$L_{InfoNCE}(\mathbf{x}, \mathbf{x}_t, \mathcal{P}_\mathbf{x}) = -\log\frac{\exp(h(\mathbf{x}_t, \mathbf{x})/\tau)}{\sum_{\mathbf{x}_p\in\mathcal{P}_\mathbf{x}}\exp(h(\mathbf{x}_p, \mathbf{x})/\tau)} \quad (7)$$

where $h(\mathbf{x}, \mathbf{x}_t)$ is a prediction head that computes the cosine similarity between $\mathbf{x}$ and $\mathbf{x}_t$ after linear projection into a low-dimensional space, and τ is a temperature hyper-parameter. In the case of AVC, the target representation $\mathbf{x}_t$ is the feature from the crop of same video but opposing modality, and the proposal distribution $\mathcal{P}_\mathbf{x}$ is composed by the target feature representations of all videos in the batch.

In the second phase, the network is trained on the more challenging task of matching audio and video at the crop level, i.e. matching representations in the presence of multiple crops per video. This is accomplished by augmenting the proposal set $\mathcal{P}_\mathbf{x}$ to include representations from multiple randomly sampled viewing angles $\{(v_i^k, a_i^k)\}_{k=1}^K$ from the same video. In this phase, we also introduce the feature translation networks g_{v2a} and g_{a2v} and require the translated features ($\bar{\mathbf{v}}_i^k$ and $\bar{\mathbf{a}}_i^k$) to match the encoder outputs ($\mathbf{v}_i^k$ and $\mathbf{a}_i^k$) obtained for the corresponding viewing angle k. Encoders f_v and f_a and feature translation networks g_{v2a} and g_{a2v} are jointly trained to minimize

$$L_{AVSA} = \sum_i\sum_k L_{InfoNCE}\left(\bar{\mathbf{v}}_i^k, \mathbf{v}_i^k, \{\mathbf{v}_j^l\}_{j,l=1}^{N,K}\right) + L_{InfoNCE}\left(\bar{\mathbf{a}}_i^k, \mathbf{a}_i^k, \{\mathbf{a}_j^l\}_{j,l=1}^{N,K}\right). \quad (8)$$

2.4. YouTube-360 dataset

We collected a dataset of 360°video with spatial audio from YouTube, containing clips from a diverse set of topics such as musical performances, vlogs, sports, and others. Search results were cleaned by removing videos that 1) did not contain valid ambisonics, 2) only contain still images, or 3) contain a significant amount of post-production sounds such as voice-overs and background music. The resulting dataset, denoted YouTube-360 (YT-360), contains a total of 5 506 videos, which was split into 4 506 videos for training and 1 000 for testing. The videos are processed into 88 733 clips of roughly 10s each (246 hours of video content), with periods of silence skipped. We also generated segmetation maps for YT-360 using the state-of-the-art ResNet101 Panoptic FPN model [6] trained on the MS-COCO dataset [11]. Examples from the YT-360 dataset are shown in Figure 4 together with the predicted segmentation maps and heatmaps of spatial audio volume.

3. Experiments

We evaluate the representations learned by AVSA pre-training on several downstream tasks.

Evaluation Task		AVC-Bin		AVSA-Bin	
# Viewpoints		1	4	1	4
AVC	no transf.	79.82	82.68	59.48	59.25
	transf.	–	83.87	–	61.20
AVTS	no transf.	80.08	82.77	59.78	60.37
	transf.	–	83.77	–	60.73
AVSA	no transf.	86.19	91.67	64.97	68.87
	transf.	–	89.83	–	69.97

Table 1: Accuracy of binary AVC and AVSA predictions on YT-360 test set.

	Video only		+Audio		+Audio+Context	
	Pix Acc	mIoU	Pix Acc	mIoU	Pix Acc	mIoU
AVC	71.16	32.85	71.07	32.69	–	–
AVTS	73.24	34.88	72.97	**34.88**	–	–
AVSA	**73.44**	**35.11**	**73.11**	34.63	**73.85**	**35.83**
AVSA (no curr.)	71.95	33.66	71.49	33.23	72.06	34.30
AVSA (mlp)	73.10	35.02	73.21	34.83	72.68	34.35
Kinetics (sup)	75.47	36.91	–	–	–	–
End-to-end	77.37	41.05	77.93	42.00	79.65	43.21

Table 2: Pixel accuracy and mean IoU of semantic segmentation predictions on YT-360 test set.

	UCF		HMDB	
	Clip@1	Video@1	Clip@1	Video@1
Scratch	54.85	59.95	27.40	31.10
Kinetics Sup.	78.50	83.43	46.45	51.90
AVC	64.63	69.68	31.33	34.58
AVTS	65.65	70.34	32.29	35.89
AVSA	**68.52**	**73.80**	**32.96**	**37.66**

Table 3: Action recognition performance on UCF and HMDB.

3.1. Experimental setting

Video pre-processing We sampled $K = 4$ crops per video at different viewing angles. Normal field-of-view (NFOV) crops are extracted using a Gnomonic projection with random angular coverage between 25°and 90°wide for data augmentation. Following NFOV projection, video clips are resized into 112×112 resolution. Random horizontal flipping, color jittering and Z normalization are applied. Each video clip is $0.5s$ long and is extracted at 16fps.

Audio pre-processing First-order ambisonics (FOA) are used for spatial audio. Audio clips for the different viewing angles are generated by simply rotating the ambisonics [8]. One second of audio is extracted at 24kHz, and four channels (FOA) of normalized log mel-spectrograms are used as the input to the audio encoder.

Architecture and optimization The video encoder f_v is a 18-layer R(2+1)D model [16], and the audio encoder f_a is a 9-layer 2D convolutional neural network operating on the time-frequency domain. The translation networks, g_{v2a} and g_{a2v}, are instantiated with depth $D = 2$. Training is conducted using the Adam optimizer [5] with a batch size of 28 distributed over 2 GPUs, learning rate of $1e-4$, weight decay of $1e-5$ and default momentum parameters $(\beta_1, \beta_2) = (0.9, 0.999)$. Both curriculum learning phases are trained for 50 epochs. Models trained only on the first or second phases are trained for 100 epochs.

Baseline pre-training methods We compare AVSA to Audio-Visual Correspondence (AVC) [1, 12] and Audio-Visual Temporal Synchronization (AVTS) [7, 14] training on the YouTube-360 dataset. AVC is trained to optimize the loss of (6), which only uses negatives from different videos. Note that (6) is similar to the loss used in [1] but considers multiple negatives simultaneously. This has actually been shown to improve generalization in [12]. To implement AVTS, we augment the proposal set $\mathcal{P}_{\mathbf{x}}$ of the InfoNCE loss of (7) with clips sampled from different moments in time. In the base AVC and AVTS implementations, we directly match the audio and visual features computed by the encoders f_v and f_a directly, as done in the original papers [1, 12, 7, 14]. However, to control for the number of seen crops, we also conduct AVC and AVTS pre-training using multiple crops of the same video and the feature translation networks g_{a2v} and g_{v2a}. Since AVC requires predictions at the video level (not for each individual clip), clip representations are combined by max-pooling.

3.2. Results

Audio-visual spatial alignment. We start by evaluating the performance on binary AVC and AVSA tasks, where a classification head on top of audio and visual features is trained to predict if the audio-visual pair are sampled from the same video instance (AVC) or at the same spatial direction (AVSA). For the binary AVSA task, negative pairs are generated by artificially rotating the ambisonic audio of a positive pair. We also study the performance improvements on both tasks by averaging outputs over four viewpoints.

Table 1 shows that the proposed AVSA pretext training mechanism significantly outperforms AVC and AVTS on both evaluation tasks, increasing the binary AVC accuracy by 6% and AVSA by 5% using a single input clip. By learning representations that are discriminative of different viewpoints, AVSA learns a more diverse set of features, resulting an even more significant gain when all 4 viewpoints are used (>8% on AVSA-Bin). We also observe improvements by using the transformer architecture in 5 out of 6 configurations, showing its effectiveness at combining information

from different viewpoints.

Semantic segmentation. AVSA representations are also evaluated on semantic segmentation. We extract features from the video encoder f_v at multiple scales, which were combined using a feature pyramid network (FPN) [10] for semantic segmentation. To measure the value added by audio inputs, we concatenate the features from the audio encoder f_a at the start of the top-down pathway of the FPN head. Similarly, to measure the benefits of combining features from multiple viewpoints, we concatenate the context-aware representations computed by the feature translation modules g_{v2a} and g_{a2v}. Since the goal is to evaluate the pretext representations, networks trained on the pretext task were kept frozen. To provide an upper bound on the performance, we trained the whole system end-to-end.

Table 2 shows the pixel accuracy and mean IoU scores obtained using video features alone, or their combination with audio and context features. Examples of segmentation maps obtained with the AVSA model with context features are also shown in Figure 5. AVSA learns significantly better visual features for semantic segmentation than AVC. This is likely due to the fine-grained nature of the AVSA task which requires discrimination of multiple crops within the same video frame. As a result, AVSA improves the most upon AVC on background classes such as rocks (34.7% accuracy vs. 27.7%), pavement (36.8% vs. 33.3%), sand (42.1% vs. 38.8%), sea (50.1% vs. 46.8%) and road (47.1% vs. 45.1%). When context features from four viewpoints are combined, using the translation networks g_{v2a} and g_{a2v}, AVSA yields a 3% mIoU improvement over AVC and 1% over AVTS. Without curriculum learning, AVSA achieved 1.5% worse mIoU. Similar loss is observed when replacing the transformer architecture of g_{v2a} and g_{a2v} with a similarly sized multi-layer perceptron, confirming the benefit of modeling spatial context for semantic segmentation.

Action recognition. Following standard practices, we finetuned the pretext models either on the UCF [15] or the HMDB [9] datasets, and measure the top-1 accuracies obtained for a single clip or by averaging predictions over 25 clips per video. For comparison, we also provide the performance of our model trained on UCF and HMDB from a random initialization (Scratch), or finetuned from a fully supervised model trained on Kinetics [18] (Kinetics Sup.). The results shown in Table 3 show once more the benefits of AVSA pretext training. AVSA dense predictions outperform AVC by 4% on UCF and 3% on HMDB, and outperform AVTS by 3.5% on UCF and 2% on HMDB.

References

[1] R. Arandjelovic and A. Zisserman. Look, listen and learn. In *International Conference on Computer Vision (ICCV)*, 2017. 1, 3, 4

[2] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 3

[3] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *26th annual international conference on machine learning*, pages 41–48, 2009. 3

[4] M. A. Gerzon. Periphony: With-height sound reproduction. *Journal of the audio engineering society*, 21(1):2–10, 1973. 2

[5] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[6] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic segmentation. In *IEEE conference on computer vision and pattern recognition*, pages 9404–9413, 2019. 3

[7] B. Korbar, D. Tran, and L. Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1, 4

[8] M. Kronlachner and F. Zotter. Spatial transformations for the enhancement of ambisonic recordings. In *2nd International Conference on Spatial Audio, Erlangen*, 2014. 2, 4

[9] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *2011 International Conference on Computer Vision (ICCV)*. IEEE, 2011. 5

[10] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 5

[11] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*. 2014. 3

[12] P. Morgado, N. Vasconcelos, and I. Misra. Audio-visual instance discrimination with cross-modal agreement. *arXiv preprint arXiv:2004.12943*, 2020. 1, 2, 3, 4

[13] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 3

[14] A. Owens and A. A. Efros. Audio-visual scene analysis with self-supervised multisensory features. In *European Conference on Computer Vision (ECCV)*, 2018. 1, 4

[15] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. Technical Report CRCV-TR-12-01, 2012. 5

[16] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. A closer look at spatiotemporal convolutions for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3

[18] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman. The kinetics human action video dataset. *arXiv:1705.06950*, 2017. 5