Repetitive Activity Counting by Sight and Sound

Yunhua Zhang[1] Ling Shao[2] Cees G. M. Snoek[1]
[1]University of Amsterdam [2]Inception Institute of Artificial Intelligence

Abstract

This paper strives for repetitive activity counting in videos. Different from existing works, which all analyze the visual video content only, we incorporate for the first time the corresponding sound into the repetition counting process. This benefits accuracy in challenging vision conditions such as occlusion, dramatic camera view changes, low resolution, etc. We propose a model that starts with analyzing the sight and sound streams separately. Then an audiovisual temporal stride decision module and a reliability estimation module are introduced to exploit cross-modal temporal interaction. For learning and evaluation, an existing dataset is repurposed and reorganized to allow for repetition counting with sight and sound. We also introduce a variant of this dataset for repetition counting under challenging vision conditions. Experiments demonstrate the benefit of sound, as well as the other introduced modules, for repetition counting. Our sight-only model already outperforms the state-of-the-art by itself, when we add sound, results improve notably, especially under harsh vision conditions. The code and datasets are available at https://github.com/xiaobai1217/RepetitionCounting.

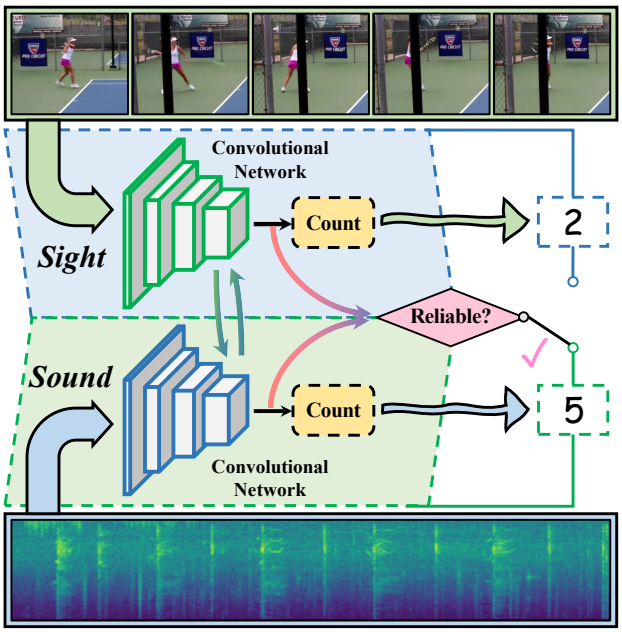

Figure 1: From sight and sound, as well as their cross-modal interaction, we predict the number of repetitions for an (unknown) activity happening in a video. This is especially beneficial in challenging vision conditions with occlusions and low illumination.

1. Introduction

The goal of this paper is to count in video the repetitions of (unknown) activities, like bouncing on a trampoline, slicing an onion or playing ping pong. The computer vision solutions to this challenging problem have a long tradition. Early work emphasized on repetitive motion estimation by Fourier analysis, *e.g.*, [4, 10, 24, 36], and more recently by a continuous wavelet transform [28, 29]. State-of-the-art solutions rely on convolutional neural networks [11, 21, 42] and large-scale count-annotated datasets [11, 42] to learn to predict the number of repetitions in a video. Albeit successful, all existing works focus exclusively on the visual modality, and could fail in poor sight conditions such as low illumination, occlusion, camera view changes, *etc*. Different from existing works, we propose in this paper the first repetitive activity counting method based on sight *and* sound.

Analyzing sound has recently proven advantageous in a variety of computer vision challenges, such as representation learning by audio-visual synchronization [1, 3, 19, 23], video captioning [25, 34, 39], sound source localization [27, 30], to name a few. Correspondingly, several mechanisms for fusing both modalities have been introduced. In works for action recognition by previewing the audio track [20] and talking-face generation [31, 38], the audio network usually works independently and the predictions guide the inference process of the visual counterpart. In contrast, feature multiplication and concatenation operations, as well as cross-modal attention mechanisms, are widely adopted for fusion in tasks like audio-visual synchronization [2, 19, 27] and video captioning [25, 34, 39]. We also combine sight and sound, but observe that for some activities, like playing ping pong, humans can count the number of repetitions by just listening. This gives us an incentive that sound could be an important

cue by itself. Hence, an intelligent repetition counting system should be able to judge when the sight condition is poor and therefore utilize complementary information from sound.

The first and foremost contribution of this paper is addressing video repetition estimation from a new perspective based on not only the sight but also the sound signals. As a second contribution, we propose an audiovisual model with a sight and a sound stream, where each stream facilitates each modality to predict the number of repetitions. As the repetition cycle lengths may vary in different videos, we further propose a temporal stride decision module to select the best sample rate for each video based on both visual and audio features. Our reliability estimation module finally exploits cross-modal temporal interaction to decide which modality-specific prediction is more reliable. Since existing works focus on visual repetition counting only, our third contribution entails two sight and sound datasets that we derive from Countix [11] and VGGsound [8]. One of our datasets is for supervised learning and evaluation and the other for assessing audiovisual counting in various challenging vision conditions. Finally, our experiments demonstrate the benefit of sound, as well as the other introduced network modules, for repetition counting. Our sight-only model already outperforms the state-of-the-art by itself, and when we add sound, the results improve further, especially under harsh vision conditions. Before detailing our model, as summarized in Figure 1, we first discuss related work.

2. Related Work

Repetitive activity counting. Existing approaches for repetition estimation in video rely on visual content only. Early works [4, 10, 24, 36] compress the motion field of video into one-dimensional signals and count repetitive activities by Fourier analysis [4, 10, 24, 36], peak detection [33] or singular value decomposition [9]. Burghouts and Geusebroek [5] propose a spatiotemporal filter bank, which works online but needs manual adjustment. Levy and Wolf [21] design a classification network able to learn from synthetic data. Their network is designed to extract features from an input video with a predefined sampling-rate, which cannot handle repetitions with various period lengths. The synthetic dataset is also less suitable for usage in the wild. All of the above methods assume the repetitions are periodic, so they can cope with stationary situations only.

Recently, algorithms for non-stationary repetitive action counting have been proposed. Runia *et al.* [28, 29] are the first to address non-stationary situations. They leverage the wavelet transform based on the flow field and collect a dataset containing 100 videos including non-stationary repetitions, but the videos do not contain an audio track. Zhang *et al.* [42] propose a context-aware framework based on a 3D convolution network, and introduce a new activity repetition counting dataset based on UCF101 [32]. While effective, the temporal length of every two repetitions is predicted by an iterative refinement, making the approach less appealing from a computational perspective. Dwibedi *et al.* [11] collect a large-scale dataset from YouTube, named Countix, containing more than 6,000 videos with activity repetition counts. Their method utilizes temporal self-similarity between video frames for repetition estimation. It chooses the frame rate sampling the input video by picking the one with the maximum periodicity classification score. While appealing, such a rate selection scheme is not optimal for accurate counting, as it is prone to select high frame rates leading to omissions.

Different from all these existing methods, we propose to address repetitive activity counting by sight and sound. Our network contains a temporal stride decision module able to choose the most suitable frame rate for counting, based on features from both modalities. To facilitate our investigation, we reorganize and supplement the Countix [11] dataset, to arrive at two audiovisual datasets for repetitive activity counting by sight and sound.

Learning by sight and sound. Many have demonstrated the benefit of audio signals for various computer vision challenges, *e.g.*, action recognition [14, 20], audiovisual event localization [40] and self-supervised learning [1, 3, 19, 23]. As processing audio signals is much faster than video frames, both Korbar *et al.* [20] and Gao *et al.* [14] reduce the computational cost by previewing the audio track for video analysis. However, the Kinetics-Sound [1] used for training and evaluation in [1, 14, 20] is simply formed by all the videos in the Kinetics dataset [6] covering 34 human action classes, which are potentially manifested visually and aurally. As a result, the audio track of many videos is full of background music, which introduces noise for training and fair evaluation. Recent talking-face generation works exploit sound for creating photo-realistic videos [31, 38]. While audio features are used to generate expression parameters in [31], Wang *et al.* [38] to map the audio to lip movements. There are also numerous works [7, 18, 19, 30, 37, 41] that consider the interaction between both modalities. Some simply integrate features by concatenation for tasks like saliency detection [37] and self-supervised learning [2, 19, 27]. Cartas *et al.* [7] combine multi-modal predictions by averaging or training a fully connected layer independently, for egocentric action recognition. Works for sound source localization [27, 30] and separation [12, 13, 41, 43] also commonly generate cross-modal attention maps. Hu *et al.* [17] use audio features to modulate the visual features for more accurate crowd counting.

Exploiting sound for activity repetition counting is still unexplored and to obtain audiovisual datasets facilitating our research, we also select videos with usable audio track from a large scale visual-only dataset manually to reduce label noise. To cope with various 'in the wild' conditions, we further introduce a novel scheme to explicitly estimate the reliability of the predictions from sight and sound.

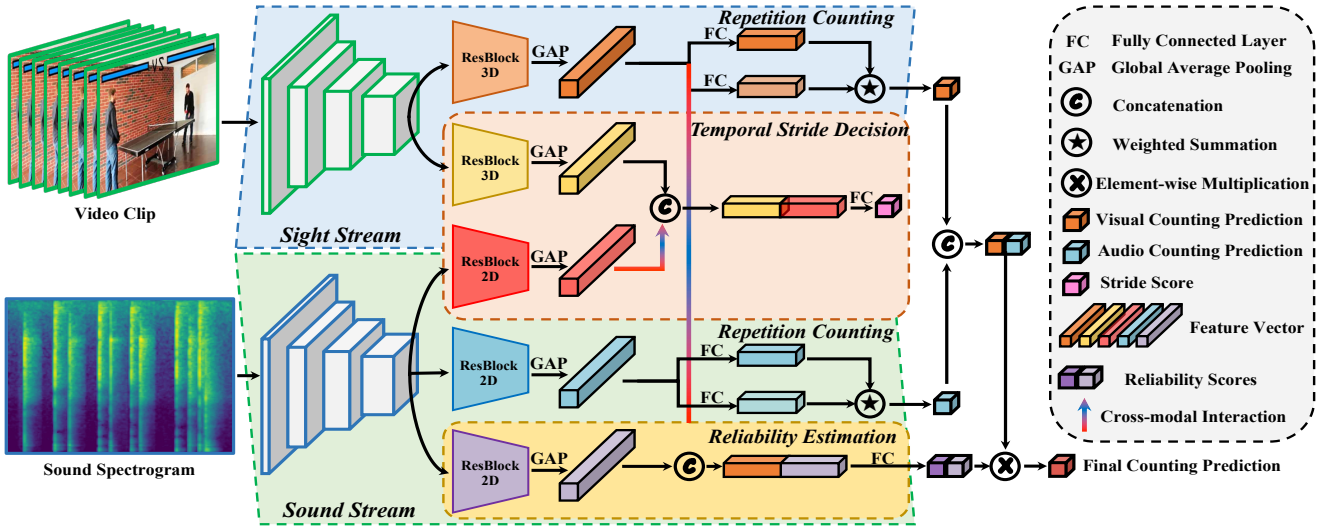

Figure 2: Proposed class-agnostic activity repetition counting model. Our model contains four components (1) sight stream, (2) sound stream, (3) temporal stride decision module and (4) reliability estimation module. Both streams contain a backbone network that outputs a modality-specific counting prediction. The temporal stride decision module takes audio and visual features as inputs and outputs the frame sample rate for the input video. Finally, the reliability estimation module decides what prediction from which modality to use.

3. Model

Given a video, containing a visual stream and its corresponding audio stream, our goal is to count the number of repetitions of (unknown) activities happening in the content. To achieve this, we propose a model that contains four modules. (*i*) The sight stream adopts a 3D convolutional network as the backbone. It takes video clips as inputs and outputs the counting result for each clip. (*ii*) For the sound stream, we rely on a 2D convolutional network, which takes the sound spectrogram generated by the short-time Fourier transform as input and outputs the counting result in the same way as the sight stream. (*iii*) A temporal stride decision module is designed to select the best temporal stride per video for the sight stream based on both visual and audio features. (*iv*) Finally, the reliability estimation module decides what prediction to use. The overall model is summarized in Figure 2 and detailed per module next.

3.1. Repetition Counting by Sight

The sight stream uses an S3D [35] architecture and the final classification layer is replaced by two separate fully connected layers, as shown in Figure 2. Given a video clip V_i of size $T\times H\times W\times 3$, visual features are extracted with the following equation:

$$\mathbf{v}_{i,feat} = \mathcal{V}_{CNN}(V_i), \tag{1}$$

where $\mathbf{v}_{i,feat} \in \mathbb{R}^{512}$. Intuitively, a fully connected layer with one output unit could suffice to output the counting result. However, this setting leads to inferior repetition counts since different types of movements should not be counted in the same way, and each action class cannot be simply regarded as one unique repetition class. For example, different videos of doing aerobics contain various repetitive motions, while bouncing on a bouncy castle or a trampoline contains similar movements despite belonging to different action classes. Therefore, in our work, two fully connected layers work in tandem, with one f_v^1 outputting the counting result of each repetition class and the other one f_v^2 classifying which repetition class the input belongs to:

$$\begin{aligned} \mathbf{C}'_{i,v} &= f_v^1(\mathbf{v}_{i,feat}), \mathbf{C}'_{i,v} \in \mathbb{R}^P, \\ \mathbf{T}_{i,v} &= softmax(f_v^2(\mathbf{v}_{i,feat})), \mathbf{T}_{i,v} \in \mathbb{R}^P, \end{aligned} \tag{2}$$

where P is the number of repetition classes, $\mathbf{C}'_{i,v}$ is the counting result of each class, and $\mathbf{T}_{i,v}$ is the classification result by the softmax operation. Here, we assume that there are roughly P classes of repetitive motion patterns, and the network learns to classify the training videos into those P classes automatically during training. Then the final counting result $\mathbf{C}_{i,v}$ from the visual content is obtained by:

$$C_{i,v} = \sum_{k=1}^{P} \mathbf{C}'_{i,v}(k)\mathbf{T}_{i,v}(k). \tag{3}$$

For training the repetition counting, we define the loss function as follows:

$$L' = \frac{1}{N}\sum_{i=1}^{N} L_2(C_{i,v}, l_i) + \lambda_1^v \frac{|C_{i,v} - l_i|}{l_i}, \tag{4}$$

where N is the batch size, L_2 is the L2 loss [26], l_i is the groundtruth count label of the ith sample and λ_1^v is a hy-

perparameter for the second term. Note that when using the L2 loss only, the model tends to predict samples with groundtruth counts of large values accurately, due to higher losses, while for videos with a few repetitions, the predicted counts tend to be unreliable. Therefore, we add a second term here to let the model pay more attention to such data.

Besides, we expect the output units of f_v^2 to focus on different repetition classes given various videos. However, without constraint, f_v^2 could simply output a high response via the same unit. To avoid such degenerated cases, we add a diversity loss [22] based on the cosine similarity:

$$L_{i,v}^{div} = \sum_{q=1}^{P-1} \sum_{j=q+1}^{P} \frac{T_{i,v}^q \cdot T_{i,v}^j}{||T_{i,v}^q|| ||T_{i,v}^j||}, \quad (5)$$

where T_v^q and T_v^j are the qth and jth units of the classification outputs. By minimizing such a diversity loss, the output T_v in the same batch are encouraged to produce different activations on different types of repetitive motions. Then the total loss function is:

$$L_v = \frac{1}{N} \sum_{i=1}^{N} L_2(C_{i,v}, l_i) + \lambda_1^v \frac{|C_{i,v} - l_i|}{l_i} + \lambda_2^v L_{i,v}^{div}, \quad (6)$$

where λ_2^v is a hyperparameter.

3.2. Repetition Counting by Sound

The sound stream adopts a ResNet-18 [15] as the backbone. Following [8, 16], we first transform the raw audio clip into a spectrogram and then divide it into a series of 257×500 spectrograms, which become the inputs to our network. Similar to the sight stream, we also replace the final classification layer by two separate fully connected layers, with one classifying the input and the other one outputting the corresponding counting result of each repetition class. We use the same loss function as the sight stream:

$$L_a = \frac{1}{N} \sum_{i=1}^{N} L_2(C_{i,a}, l_i) + \lambda_1^a \frac{|C_{i,a} - l_i|}{l_i} + \lambda_2^a L_{i,a}^{div}, \quad (7)$$

where $C_{i,a}$ is the counting result from the audio track, and λ_1^a and λ_2^a are hyperparameters.

3.3. Temporal Stride Decision

Repetitions have various period lengths for different videos. For the sound stream, we can simply resize the spectrogram along the time dimension to ensure each 257×500 segment to have at least two repetitions. However, for the sight stream, we cannot roughly resize the video frames along the time dimension. Therefore, for each video, we need to use a specific temporal stride (*i.e.* frame rate) to form video clips of T frames as the inputs. This is important as video clips with small temporal strides may fail to include at least two repetitions, while too large temporal strides lead the network to ignore some repetitions. Therefore, we add an additional temporal stride decision module to select the best temporal stride for each video. It has two parallel residual blocks, processing visual and audio features from the third residual block of the two streams, with the same structure as those of the backbones. Then we concat the output features (as shown in Figure 2) and send them into a fully connected layer, which outputs a single unit representing the score of the current temporal stride. We use a max-margin ranking loss for training this module:

$$L_s = \frac{1}{N} \sum_{i=1}^{N} max(0, s_i^- - s_i^+ + m), \quad (8)$$

where m is the margin, s_i^- and s_i^+ are the scores from negative and positive strides. During inference, we send a series of clips from the same video with different strides into the network, and select the stride with the highest score.

Training details. For each training video, the trained visual model predicts the counting result with a series of temporal strides, *i.e.* $s = 1, ..., S_k, ..., S_K$, where S_K is the maximum stride we use. Then we can obtain corresponding predictions $C_{i,v}^1, ..., C_{i,v}^{S_K}$. First, we select the temporal strides that cover less than two repetitions as negative strides. Then, we choose the smallest stride that is enough to contain at least two repetitions as the positive temporal stride S^*. Correspondingly, for the remaining strides, we quantitatively compute their prediction deviations from the prediction of the positive stride by:

$$\delta_n = \frac{C_{i,v}^* - C_{i,v}^k}{C_{i,v}^*}, \quad (9)$$

where $C_{i,v}^*$ and $C_{i,v}^k$ are the counting predictions from the best stride and a selected stride, and δ_n is the computed deviation. Finally, we select strides with $\delta_n > \theta_s$ (θ_s is a predefined threshold) as negative strides, since for these strides, the network begins to omit certain repetitions. During training, for each video, its S^* is used to form a positive video clip outputting s_i^+, while we randomly select one from the negative strides to generate the clip outputting s_i^-.

3.4. Reliability Estimation

Depending on the sensory video recording conditions, the reliability of the sight and sound predictions may vary. To compensate for this variability, we introduce a reliability estimation module to decide what prediction from which modality is more reliable for the current input. As shown in Figure 2, it contains one residual block for processing the audio feature and one fully connected layer taking features from both modalities as inputs. The output is a single unit processed by a sigmoid function and represents the

confidence γ of the audio modality. Correspondingly, the confidence of the visual modality is $1-\gamma$. Then the final counting result is obtained by:

$$C_i = C_{i,v} * (1-\gamma) + C_{i,a} * \gamma. \tag{10}$$

As C_i is expected to be close to the groundtruth counting label, the loss function we use for training is:

$$L_r = \frac{1}{N}\sum_{i=1}^{N}\frac{|C_i - l_i|}{l_i}. \tag{11}$$

Training details. During training, for each video, the accuracy of $C_{i,v}$ and $C_{i,a}$ in Eq. 10 is expected to indicate the reliability of the corresponding modality content. To get $C_{i,v}$ and $C_{i,a}$, one simple approach is to directly use the predictions from the trained models. However, we empirically observe that such a manner suffers from severe over-fitting, since the learned models could overfit to recent training samples with poor modality content. As a result, the obtained $C_{i,v}$ and $C_{i,a}$ cannot represent the reliability effectively. However, for one modality of a video, if the corresponding model predicts inaccurately most of the time during training, then intuitively the content may be too noisy or poor for learning. Therefore, instead of $C_{i,v}$ and $C_{i,a}$ from the final models, we use the average prediction of each stream at different training stages. Here, we take the sight stream as an example. After each training epoch, if the loss computed by $\frac{1}{\mathcal{N}}\sum_{i=1}^{\mathcal{N}}\frac{|C_{i,v}-l_i|}{l_i}$, $\mathcal{N}$ is the number of videos, over the validation set is below a threshold θ_r^v (*i.e*. current model parameters have competitive performance), we record the predictions of the model over the training videos. Once the training is finished, we can obtain the average prediction (*i.e*. empirical prediction) of each training video by recordings correspondingly. The empirical prediction of the sound stream is computed in the same way, with a threshold θ_r^a for the validation loss. Finally, our reliability estimation module uses those empirical predictions for Eq. 10 during training and learns to switch between sight and sound models for effective late fusion.

4. Experimental Setup

4.1. Datasets

Existing datasets for repetition counting [11, 21, 28, 42] focus on counting by visual content only. Thus, the videos have either no audio information at all, or at best a few only. Nonetheless, we evaluate our (sight) model on the two largest existing visual-only datasets, *i.e*. UCFRep and Countix. As we focus on counting by sight and sound, we also repurpose, reorganize and supplement one of those two datasets.

UCFRep. The UCFRep dataset by Zhang *et al.* [42] contains 526 videos of 23 categories selected from UCF101 [32], a widely used benchmark for action recognition, with 420 and 106 videos for training and validation. Particularly, it has boundary annotations for each repetition along the time dimension. However, the large majority of videos do not have any associated audio track.

Countix. The Countix dataset by Dwibedi *et al.* [11] serves as the largest dataset for video repetition counting in the wild. It is a subset of the Kinetics [6] dataset annotated with segments of repeated actions and corresponding counts. The dataset contains 8,757 videos in total of 45 categories, with 4,588, 1,450 and 2,719 for training, validation and testing.

Countix-AV. We repurpose and reorganize the Countix dataset for our goal of counting repetitive activities by sight and sound. We first select 19 categories for which the repetitive action has a clear sound, such as *clapping*, *playing tennis*, etc. As several videos contain artificially added background music or have no audio track at all, they are less suited for assessing the impact of the sound-stream, and the sight and sound combination. Therefore, we manually filter out such videos so that the videos preserved are guaranteed to contain the environmental sound only, be it they may also include realistic background noise or unclear repetition sounds. This results in the Countix-AV dataset consisting of 1,863 videos, with 987, 311 and 565 for training, validation and testing. We maintain the original count annotations from Countix and keep the same split (*i.e*. training, validation, or testing) for each video. The dataset is detailed in the appendix.

Extreme Countix-AV. For most videos in Countix-AV, the ongoing action is both visible and audible. As the audio signal is expected to play a vital role when the visual content is not reliable, we further introduce the Extreme Countix-AV dataset to quantitatively evaluate the benefits brought by audiovisual counting under various extreme sight conditions. For data collection, we first define 7 vision challenges, according to which we select videos. Then, 156 videos from Countix-AV are selected and to enlarge this selection, we choose and label another 58 videos from the VGGSound dataset by Chen *et al.* [8]. The overall dataset and challenges are summarized in Table 1, with video examples depicted in Figure 3. More details are provided in the appendix.

Vision challenge	**Number of videos**
Camera viewpoint changes	69
Cluttered background	36
Low illumination	13
Fast motion	31
Disappearing activity	25
Scale variation	24
Low resolution	29
Overall	214

Table 1: Extreme Countix-AV dataset statistics.

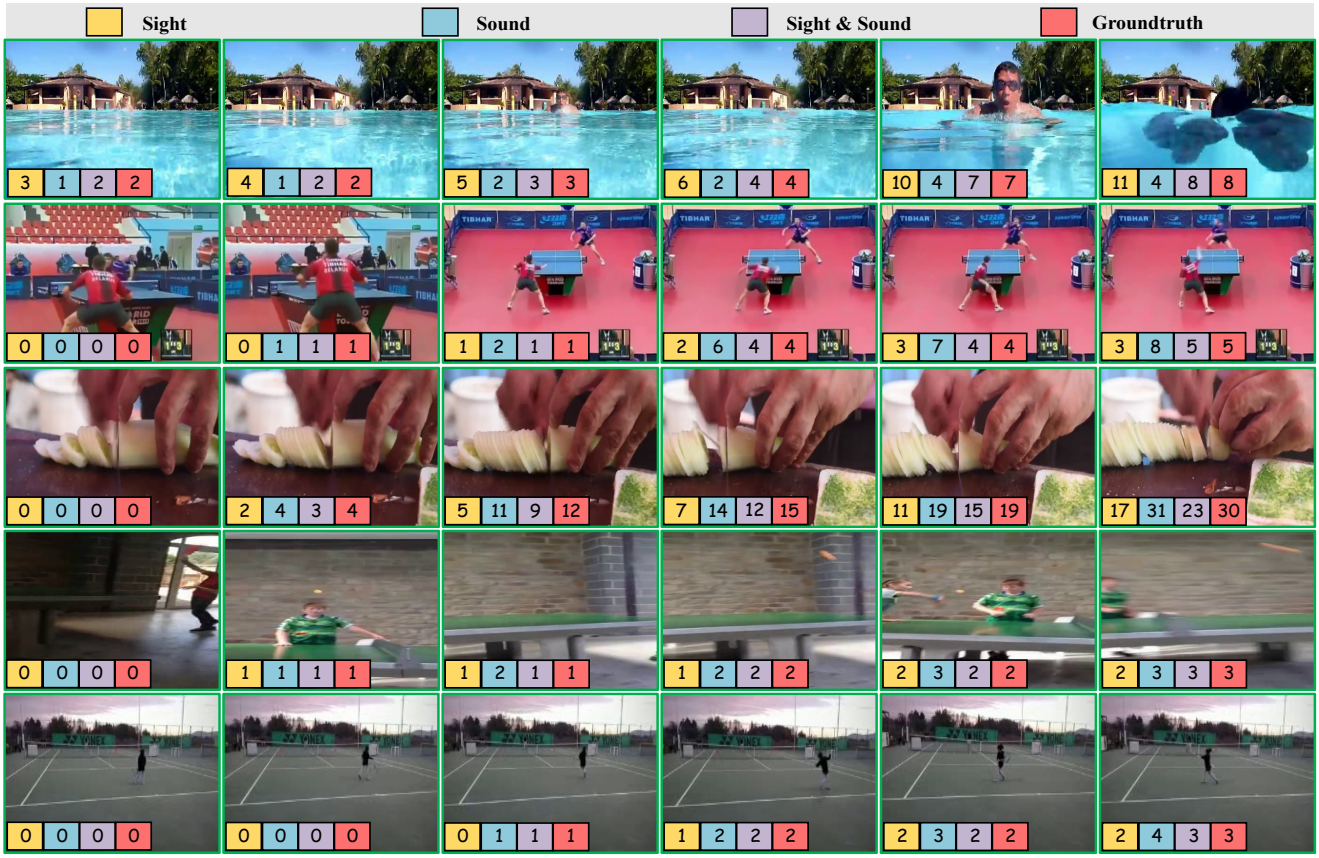

Figure 3: Example videos from the Extreme Countix-AV dataset. From top-row to bottom-row are videos with scale variation, camera viewpoint change, fast motion, a disappearing activity and low resolution, with the numbers in colored boxes indicating counting results and the corresponding groundtruth.

4.2. Evaluation Criteria

We adopt the same evaluation metrics as previous works [11, 24, 28, 29, 42], *i.e.* the mean absolute error (MAE) and off-by-one accuracy (OBO), defined as follows:

$$MAE = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \frac{|\hat{c}_i - l_i|}{l_i}, \tag{12}$$

$$OBO = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} [|\hat{c}_i - l_i| \leq 1], \tag{13}$$

where $\mathcal{N}$ is the total number of videos, $\hat{c}_i$ is the model prediction of the ith video and l_i is the groundtruth. Specifically, for the Extreme Countix-AV, we report MAE only, as those videos have more repetitions than other datasets and OBO cannot evaluate the performance effectively.

4.3. Implementation Details

We implement our method using PyTorch with two NVIDIA GTX1080Ti GPUs. We provide training details below and inference procedures in the appendix.

Sight and sound models. For the sight stream, all input video frames are resized to 112×112, and we form each clip of 64 frames with its temporal stride S^* defined in Section 3.3. We initialize the backbone with weights from a Kinetics [6] pre-trained checkpoint. The training of the sight model is on the original Countix training set [11] and takes 8 epochs by SGD with a fixed learning rate of 10^{-4} and batch size of 8. λ_v^1, λ_v^2, λ_a^1 and λ_a^2 are all set to 10. The sound model is trained with the same setting as the sight stream but using our Countix-AV training set for 20 epochs.

Temporal stride decision module. The training takes 5 epochs with a learning rate of 10^{-3} after obtaining the negative strides of each video. Here, we provide two options. First, it can be trained with the visual modality only, *i.e.* without the audio feature, using the original Countix [11] dataset so that the sight model can work independently. The other option is our full setting (as shown in Figure 2) trained on Countix-AV with the audio modality. Margin m in Eq. 8 is set to 2.9 and S_K is set to 8. In experiments, we find the value of θ_s does not influence results too much, and θ_s=0.29 works best (see appendix for ablation).

Model components	*MAE* ↓	*OBO* ↑
Sight stream	0.331	0.431
Sound stream	0.375	0.377
Sight with temporal stride	0.314	0.459
Averaging predictions	0.300	0.439
Full sight and sound model	0.291	0.479

Table 2: Benefit of model components on Countix-AV. All modules matter and reliability estimation is preferred over simple averaging of sight and sound predictions.

	Sight		Sound	
Loss term	*MAE* ↓	*OBO* ↑	*MAE* ↓	*OBO* ↑
L_2	0.371	0.424	0.471	0.338
$L_2 + L_{div}$	0.324	0.478	0.410	0.343
$L_{div} + L_{mae}$	0.356	0.446	0.447	0.310
$L_2 + L_{mae}$	0.370	0.421	0.426	0.340
$L_2 + L_{div} + L_{mae}$	0.314	0.498	0.375	0.377

Table 3: Influence of loss function terms on Countix-AV. Each term lowers the counting error, while L_{div} is indispensable.

Reliability estimation module. We first collect the empirical predictions before training, and θ_r^v and θ_r^a are set to 0.36 and 0.40. Then, this module is trained on Countix-AV for 20 epochs with a learning rate of 10^{-4} and batch size of 8.

5. Results

Benefit of model components. Our model consists of four main components: the sight and sound counting models, the temporal stride decision module and the reliability estimation module. We evaluate the performance of several network variants on Countix-AV to validate the efficacy of each component. The results are shown in Table 2. Note that for "Sight stream" in the first row, its temporal stride decision module takes the visual modality only as input. In isolation, the sight stream performs better than the sound stream. When we incorporate audio features into the temporal stride decision module, denoted as "Sight with temporal stride", the MAE of the sight stream is further reduced from 0.331 to 0.314. This demonstrates the audio signals provide useful temporal information. Simply averaging the predictions from both modalities results in higher accuracy than either modality alone. However, when we further reweigh the predictions by our reliability estimation module, we obtain the best result with an MAE of 0.291 and an OBO of 0.479.

Influence of loss function terms. The loss function used for training the visual and audio models consists of three terms. We perform an ablation on different term combinations to further understand their contributions. Results in Table 3 indicate both L_{div} and L_{mae} reduce the counting error, especially on the sound stream. We observe adding L_{div} contributes to performance improvements because it allows the units in the classification layer to affect each other during training. It prevents this layer from converging to a degenerated solution, in which all videos are assigned to the same repetition class. Combining all loss terms during training produces best results for both modalities.

Effect of repetition classes. As detailed in Eq. 2 and 3, our counting models for both modalities involve a parameter P, *i.e.* the number of repetition classes. We evaluate its effect on both the sight and sound models. To this end, we fix the backbone architectures and train them by setting P from 1 to 50 with the same setting described in Section 4.3. Here, we do not include cross-modal interactions, *i.e.* the sight and sound models are trained and evaluated on Countix and our Countix-AV datasets, separately. The results are shown in Figure 4. The performances of both models are inferior when P has a low value, demonstrating the need to model various types of repetitions. The performance fluctuates only slightly when P is between 20 and 50. We observe that there is a spike when P=36 for both streams, at where the networks converge to local minimum and this can be eliminated by more training epochs. In particular, the sight and sound models obtain their best results at P=41 and P=43, the values we use for all other experiments, while Countix [11] and our Countix-AV cover 45 and 19 action categories. Thus, the repetition types do not simply correspond to the number of action categories. For instance, in Figure 3, the second and the last rows have similar repetition classification distributions, while dissimilar to the fourth row. We show more examples in the supplementary material.

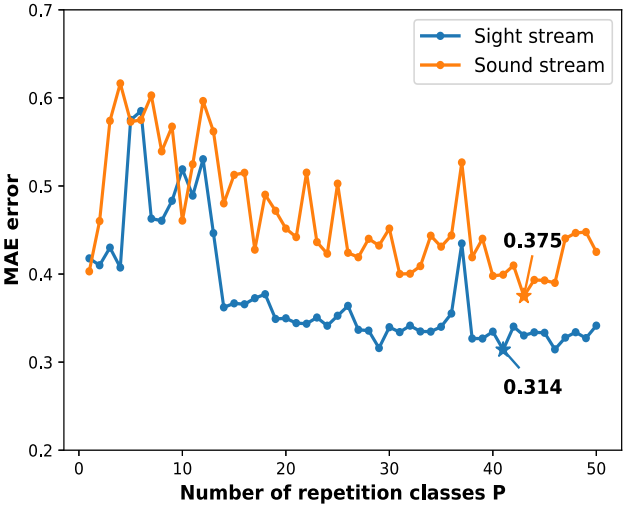

Figure 4: Effect of repetition classes. The performances of both streams fluctuates slightly when P is large enough, and achieves the best result when P=41 (sight) and P=43 (sound).

	UCFRep		Countix		Countix-AV		Extreme Countix-AV
	MAE ↓	*OBO* ↑	*MAE* ↓	*OBO* ↑	*MAE* ↓	*OBO* ↑	*MAE* ↓
Baseline†	0.474	0.371	0.525	0.289	0.503	0.269	0.620
Dwibedi *et al.* [11]	-	-	0.364	**0.697**	-	-	-
Levy and Wolf* [21]	0.286	0.680	-	-	-	-	-
Zhang *et al.* [42]	0.147	0.790	-	-	-	-	-
This paper: ***Sight***	**0.143**	**0.800**	0.314	0.498	0.331	0.431	0.392
This paper: ***Sound***	-	-	0.793	0.331	0.375	0.377	0.351
This paper: ***Sight & Sound***	-	-	**0.307**	0.511	**0.291**	**0.479**	**0.329**

† Sight-only model, pre-trained on Countix, publicly released by authors of [11].
* Sight-only model [21], reproduced and pre-trained on UCFRep by authors of [42].

Table 5: Comparison with state-of-the-art. Our sight model outperforms two recent state-of-the-art repetition counting algorithms in terms of MAE, while combining sight and sound shows further benefit in reducing counting error, especially in visually challenging settings.

Vision challenge	Sight	Sound	Sight & Sound
Camera viewpoint changes	0.384	0.376	0.331
Cluttered background	0.342	0.337	0.307
Low illumination	0.325	0.269	0.310
Fast motion	0.528	0.311	0.383
Disappearing activity	0.413	0.373	0.339
Scale variation	0.332	0.386	0.308
Low resolution	0.348	0.303	0.294
Overall	0.392	0.351	0.329

Table 4: MAE metric on hard cases in repetition counting. Sound tends to have lower MAE than sight. Combining sight and sound always outperforms sight only.

Effectiveness of temporal stride module. We also considered fixed temporal strides for the sight-stream on Countix. MAE varies from 0.607 to 0.378 (see appendix). Our temporal stride decision module obtains a better 0.314 MAE.

Hard cases in repetition counting. To quantitatively evaluate the contribution of sound information and how sensitive the sight stream is under different visually challenging environment, we test the sight, sound and full sight and sound model separately on the Extreme Countix-AV dataset. The results are listed in Table 4. Compared to the performance on Countix-AV dataset, which is dominated by videos with normal sight conditions, the MAE of the sight stream increases considerably. In contrast, the sound stream performs stably and is superior under visually challenging circumstances as expected, except for the scale variation challenge. This means that changes in image quality can easily affect the sight stream. Especially when activities are moving fast or disappearing due to occlusions, the value of the sound stream is prevalent. Combining sight and sound is always better than sight only, resulting in considerable MAE reductions on videos with camera view changes, disappearing activities, scale variation and cluttered background. For scale variation, the sound stream does not perform competitively compared to the sight stream, while the fused results do improve over the sight stream. This again indicates the effectiveness of our reliability estimation module. For low illumination and fast motion, the sight stream performs poor compared to the sound stream, and the combination cannot improve over the sound stream only. Overall, the integration of sight and sound is better than unimodal models and more stable when the imaging quality varies.

Comparison with state-of-the-art. We compare our method with two recent state-of-the-art (vision-only) repetition counting models [11, 42] and one early work by Levy and Wolf [21], as shown in Table 5. As the complete code of [11] is unavailable, we also report the performance of their released (vision-only) model as a baseline. Our sight-only stream already outperforms Dwibedi *et al.* [11] on their original Countix dataset with respect to the MAE metric, and achieves competitive performance on UCFRep [42]. Note the work by Zhang *et al.* [42] needs the training videos to have boundary annotations for each repetition, which are not provided for Countix [11]. As Countix is dominated by "silent" repetitions, our sound-only model performs inferior compared to the sight-only model. Nevertheless, our full sight and sound model sets a new state-of-the-art on all three Countix datasets in MAE and surpasses the released model of [11] by a large margin. Therefore, we conclude that when the original sound track of the video is available, audiovisual repetition counting is superior to sight-only models.

6. Conclusion

We propose to count repetitive activities in video by sight and sound using a novel audiovisual model. To facilitate further progress, we repurpose and reorganize an existing counting dataset for sight and sound analysis. Experiments show that sound can play a vital role, and combining both sight and sound with cross-modal temporal interaction is beneficial. Using sight only we already outperform the state-of-the-art in terms of MAE. When adding sound, results improve further, especially under harsh vision conditions.

References

[1] Relja Arandjelović and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.

[2] Relja Arandjelović and Andrew Zisserman. Objects that sound. In *ECCV*, 2018.

[3] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.

[4] Ousman Azy and Narendra Ahuja. Segmentation of periodically moving objects. In *ICPR*, 2008.

[5] Gertjan J Burghouts and Jan-Mark Geusebroek. Quasi-periodic spatiotemporal filtering. *TIP*, 15(6):1572–1582, 2006.

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, 2017.

[7] Alejandro Cartas, Jordi Luque, Petia Radeva, Carlos Segura, and Mariella Dimiccoli. Seeing and hearing egocentric actions: How much can we learn? In *ICCVW*, 2019.

[8] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGGSound: a large-scale audio-visual dataset. In *ICASSP*, 2020.

[9] Dmitry Chetverikov and Sándor Fazekas. On motion periodicity of dynamic textures. In *BMVC*, 2006.

[10] Ross Cutler and Larry S. Davis. Robust real-time periodic motion detection, analysis, and applications. *TPAMI*, 22(8):781–796, 2000.

[11] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Counting out time: Class agnostic video repetition counting in the wild. In *CVPR*, 2020.

[12] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *ECCV*, 2018.

[13] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *ICCV*, 2019.

[14] Ruohan Gao, Tae-Hyun Oh, Kristen Grauman, and Lorenzo Torresani. Listen to look: Action recognition by previewing audio. In *CVPR*, 2020.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[16] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, Malcolm Slaney, Ron J Weiss, and Kevin Wilson. CNN architectures for large-scale audio classification. In *ICASSP*, 2017.

[17] Di Hu, Lichao Mou, Qingzhong Wang, Junyu Gao, Yuansheng Hua, Dejing Dou, and Xiao Xiang Zhu. Ambient sound helps: Audiovisual crowd counting in extreme conditions. *arXiv preprint arXiv:2005.07097*, 2020.

[18] Hamid Reza Vaezi Joze, Amirreza Shaban, Michael L Iuzzolino, and Kazuhito Koishida. Mmtm: multimodal transfer module for cnn fusion. In *CVPR*, 2020.

[19] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.

[20] Bruno Korbar, Du Tran, and Lorenzo Torresani. SCSampler: Sampling salient clips from video for efficient action recognition. In *ICCV*, 2019.

[21] Ofir Levy and Lior Wolf. Live repetition counting. In *ICCV*, 2015.

[22] Daochang Liu, Tingting Jiang, and Yizhou Wang. Completeness modeling and context separation for weakly supervised temporal action localization. In *CVPR*, 2019.

[23] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.

[24] Erik Pogalin, Arnold WM Smeulders, and Andrew HC Thean. Visual quasi-periodicity. In *CVPR*, 2008.

[25] Tanzila Rahman, Bicheng Xu, and Leonid Sigal. Watch, listen and tell: Multi-modal weakly supervised dense event captioning. In *ICCV*, 2019.

[26] Rasmus Rothe, Radu Timofte, and Luc Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126(2):144–157, 2018.

[27] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *ICASSP*, 2019.

[28] Tom FH Runia, Cees GM Snoek, and Arnold WM Smeulders. Real-world repetition estimation by div, grad and curl. In *CVPR*, 2018.

[29] Tom FH Runia, Cees GM Snoek, and Arnold WM Smeulders. Repetition estimation. *IJCV*, 127(9):1361–1383, 2019.

[30] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound sources in visual scenes: Analysis and applications. *arXiv preprint arXiv:1911.09649*, 2019.

[31] Linsen Song, Wayne Wu, Chen Qian, Ran He, and Chen Change Loy. Everybody's talkin': Let me talk as you want. *arXiv preprint arXiv:2001.05201*, 2020.

[32] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[33] Ashwin Thangali and Stan Sclaroff. Periodic motion detection and estimation via space-time sampling. In *WACV*, 2005.

[34] Yapeng Tian, Chenxiao Guan, Justin Goodman, Marc Moore, and Chenliang Xu. An attempt towards interpretable audio-visual video captioning. In *ICCV*, 2019.

[35] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[36] Ping-Sing Tsai, Mubarak Shah, Katharine Keiter, and Takis Kasparis. Cyclic motion detection for motion based recognition. *Pattern recognition*, 27(12):1591–1603, 1994.

[37] Antigoni Tsiami, Petros Koutras, and Petros Maragos. STAViS: Spatio-temporal audiovisual saliency network. In *CVPR*, 2020.

[38] Kaisiyuan Wang, Qianyi Wu, Linsen Song, Zhuoqian Yang, Wayne Wu, Chen Qian, Ran He, Yu Qiao, and Chen Change Loy. MEAD: A large-scale audio-visual dataset for emotional talking-face generation. In *ECCV*, 2020.

[39] Xin Wang, Yuan-Fang Wang, and William Yang Wang. Watch, listen, and describe: Globally and locally aligned cross-modal attentions for video captioning. *arXiv preprint arXiv:1804.05448*, 2018.

[40] Yu Wu, Linchao Zhu, Yan Yan, and Yi Yang. Dual attention matching for audio-visual event localization. In *ICCV*, 2019.

[41] Xudong Xu, Bo Dai, and Dahua Lin. Recursive visual sound separation using minus-plus net. In *ICCV*, 2019.

[42] Huaidong Zhang, Xuemiao Xu, Guoqiang Han, and Shengfeng He. Context-aware and scale-insensitive temporal repetition counting. In *CVPR*, 2020.

[43] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019.