Proceedings of Machine Learning Research vol 278:1–10, 2025 2025 2nd International Conference on Machine Learning and Intelligent Computing

Event-Based Binary Neural Networks for Efficient and Accurate Lip Reading

Xueyi Zhang
National University of Defense Technology, China

Jialu Sun
The University of Hong Kong, China

Peiyin Zhu
National University of Singapore, Singapore

Bowen Wang
National University of Defense Technology, China

Mingrui Lao* LAOMINGRUI@VIP.SINA
National University of Defense Technology, China
**Corresponding author*

Yanming Guo
National University of Defense Technology, China

Editors: Nianyin Zeng, Ram Bilas Pachori and Dongshu Wang

Abstract

Event cameras provide exceptional temporal resolution and consume minimal power, making them highly suitable for lip reading tasks. However, traditional methods struggle with the high computational costs of processing asynchronous event streams. We propose STCNet, a spatio-temporal convolutional network optimized for event-driven lip reading, and its binary counterpart B-STCNet,which results in a substantial reduction in computational and memory resource requirements. B-STCNet introduces Kernel-Specific Scaling Factors to bridge the performance gap induced by binarization and adopts quantization-aware training to enhance model stability. Evaluated on the DVS-Lip dataset, B-STCNet achieves state-of-the-art accuracy with over 90% reduction in parameters and 50% fewer FLOPs, demonstrating its potential for deployment on resource-constrained edge devices.

Keywords: Event Cameras, Lip Reading, Binary Neural Networks, Quantization-Aware Training

1. Introduction

Lip reading is critical in enhancing communication for individuals with hearing impairments and in improving biometric verification systems. The advancement of event-based vision sensors, capable of recording fine-grained temporal dynamics with ultra-low latency and energy efficiency, offers new possibilities for applications requiring fine-grained temporal analysis such as lip reading.

Traditional optical cameras face challenges such as data redundancy, low temporal resolution, and dynamic range limitations in capturing lip movements. In contrast, event cameras, by recording changes in pixel-level brightness, exhibit superior performance. They have shown significant progress in temporal modeling tasks such as action recognition and gesture recognition, with microsecond-level temporal resolution enabling precise capture of subtle lip movements, significantly enhancing the accuracy and robustness of lip reading. Furthermore, their high dynamic

range allows accurate data capture even under extreme lighting conditions, and their low power consumption makes them ideal for long-duration operation on wearable and mobile devices.

Lip reading's practical application necessitates highly lightweight systems to accommodate the constraints of edge devices. Event cameras, with their efficient data capture method, have already shown significant advantages in data input lightweighting. Despite their potential, the practical deployment of event-based lip reading technologies has been hindered by the significant computational resources required by traditional models (Liu et al., 2024). While these cameras offer efficient data capture, making them advantageous for lightweight applications, achieving an optimal balance between system performance and resource constraints on edge devices remains a challenge (Wang et al., 2025). To address this, it is essential to optimize model design, focusing on reducing parameter counts and computational complexity without compromising performance. This streamlined approach is crucial for the efficient operation of lip reading systems in resource-limited settings.

To address these challenges, we introduce a binary neural network designed specifically for event-driven lip reading, named B-STCNet (Binary Spatio-Temporal Convolutional Network). Unlike traditional floating-point-based neural networks, B-STCNet, by binarizing network weights (i.e., weights are only -1 and +1), greatly reduces the model's storage requirements and computational complexity. Specifically, the binarization process eliminates the need for multiplication operations during model inference, requiring only addition operations, thereby significantly boosting computational efficiency.

To further adapt to binarization, we optimized the model architecture, incorporating a 3D convolutional ResNet-18 structure at the front-end and an MS-TCN at the back-end both of which are conducive to binarization compared to the GRU-based structures used in previous methods like MSTN (Tan et al., 2022), which are challenging to binarize. Additionally, to minimize the discrepancy between binary and full-precision weights, we introduce the Kernel-Specific Scaling Factors. Moreover, during the training process, we employed quantization-aware training methods to ensure the stability and accuracy of the model's performance. In a series of experiments, we evaluated both the designed model and its binarized version. Experimental findings demonstrate that our method outperforms others on the DVS-Lip dataset, with the binarized version maintaining comparable accuracy to the full-precision model while reducing parameters by over 90% and FLOPs by 50%. These achievements not only validate the effectiveness of our model architecture and binarization strategy but also demonstrate their potential application in resource-limited environments.

Contributions:

- We introduce the use of binary neural networks to the field of event camera-based lip reading, enhancing computational efficiency without compromising accuracy.
- We design STCNet and its binary version, B-STCNet, which incorporate a fully convolutional architecture that is more amenable to binarization than previous GRU-based models.
- We propose a Kernel-Specific Scaling Factor within B-STCNet to address the performance degradation usually associated with binary networks, ensuring robust model accuracy.
- Our results on the DVS-Lip dataset show that B-STCNet not only delivers top-tier accuracy but also cuts parameter count by over 90% and FLOPs by 50%, validating the effectiveness of our approach for deployment in resource-limited environments.

2. Methods

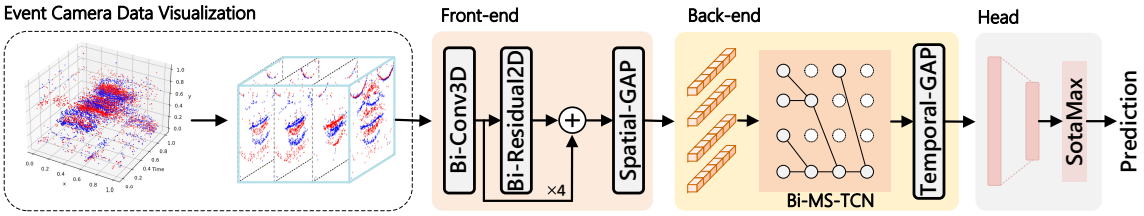

Figure 1: Overall Structure of STCNet: The event data is first voxelized and then processed by STCNet for feature extraction. STCNet is composed of a front-end for short-term dynamic modeling and a back-end for long-term context modeling. The 3D spatial-temporal feature map is reduced to a 1D temporal feature using Global Average Pooling (GAP) along the spatial dimension. This temporal feature is further condensed into a joint representation through GAP in the temporal dimension and is then fed into the classification head to compute class probabilities.

2.1. Event Representation

An event camera generates a high-frequency stream of events that captures the changes in brightness for each pixel. The event generation process may be formulated as:

$$\Delta I(x, y, t) = \log I(x, y, t) - \log I(x, y, t - \Delta t) \tag{1}$$

$$P(x, y, t) = \begin{cases} 1, & \text{if } \Delta I(x, y, t) \geq 0 \\ -1, & \text{if } \Delta I(x, y, t) < 0 \end{cases} \tag{2}$$

where $I(x, y, t)$ represents the brightness of the pixel located at x, y at time t, while $\Delta I(x, y, t)$ denotes the change in brightness over a time interval Δt. The term $P(x, y, t) \in \{1, -1\}$ indicates the event at x, y, t, where 1 represents an increase in brightness, and -1 represents a decrease. The event stream over a longer time range $T_0 < T < T_1$ can be expressed as:

$$\mathcal{E} = \{e_k\}_{k=1}^{\mathcal{N}} = \{(x_k, y_k, t_k, P(x_k, y_k, t_k))\}_{k=1}^{\mathcal{N}} \tag{3}$$

where $\mathcal{E}$ represents the set of the event stream. Each event is represented as a four-tuple consisting of the pixel position and time at which the event occurs, along with the indication of whether the brightness is increasing or decreasing. The event stream over this time interval will be projected and aggregated using a voxel grid, to facilitate feature extraction of the network. The defined voxel grid has three dimensions: T, W, H, which can be expressed as:

$$t_k^* = \frac{T - 1}{t_N - t_1}(t_k - t_1) \tag{4}$$

$$\mathcal{V}(x, y, t) = \sum_k p_k \max(0, 1 - |t - t_k^*|). \tag{5}$$

where the temporal information of events are normalized to t^*, and a single voxel grid $\mathcal{V}(x, y, t)$ only aggregates event information within its neighboring range, serving as the aggregated representation of that grid. After voxelization, the representation of events becomes similar to that of a conventional video camera. The format of the event data is $X_{\text{event}} \in \mathbb{R}^{T\times X\times Y\times 1}$, while the representation of a video is $X_{\text{video}} \in \mathbb{R}^{T\times X\times Y\times 3}$, where 1 represents the aggregation of events at each pixel, and 3 represents the RGB channels. The main difference between events and videos is that events can filter out most of the features unrelated to lip movements, whereas videos often contain a large amount of irrelevant background information and motion blur.

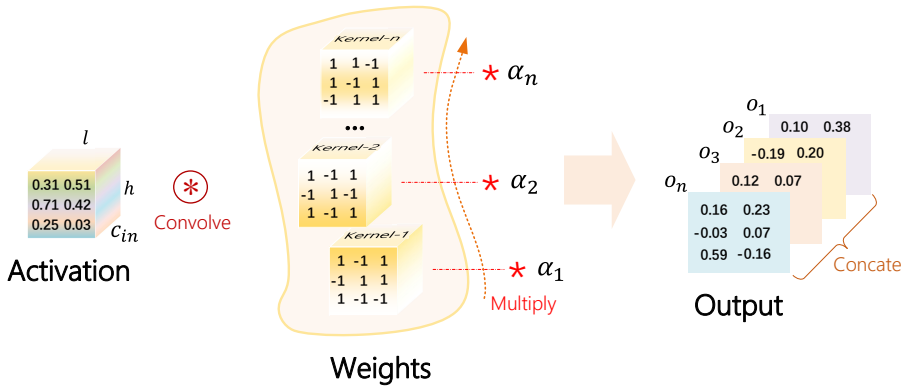

Figure 2: The convolution process employs the Kernel-Specific Scaling Factors strategy, where a unique scaling factor is calculated for each convolutional kernel. The results of the convolution are then concatenated to serve as the output of the convolution.

2.2. Overall of STCNet

Traditional video-based lip-reading methods typically employ a combination of a 3D convolutional neural network (CNN) layer followed by multiple 2D convolutional layers as the front-end, with a temporal modeling network as the back-end. This is a well-established network architecture that effectively captures both short-term lip movements and long-term contextual features. Inspired by this network design, our event-based lip-reading model will adopt a similar approach. The overall structure of our STCNet is shown in fig. 1. The event voxel grid will first utilize 3D convolutions to extract spatial-temporal features can be expressed as:

$$F^{3D} = Pool3D\left(C_{3D}BR(X)\right) \tag{6}$$

where CBR represents Convolution, Batch Normalization, and ReLU. The spatial-temporal features after pooling, $F^{3D} \in \mathbb{R}^{64\times T\times W/4\times H/4}$, are then further modeled to capture short-term temporal relationships using multiple 2D convolutional layers with residual connections and can be expressed as:

$$F_{i+1}^{S} = Block2D\left(F_i^S\right) \tag{7}$$

where F_0^S is initialized as F^{3D}. After passing through $i = 4$ layers of residual blocks, the feature map output from the front-end, $F_4^S \in \mathbb{R}^{512\times T\times \frac{W}{32}\times \frac{H}{32}}$, is obtained. This feature map is then average-pooled along the spatial dimensions to yield $F^{\text{front}} \in \mathbb{R}^{T\times 512}$. For the back-end network, we employ

the MS-TCN architecture, which has demonstrated excellent accuracy in video-based lip-reading. The MSTCN structure, with its multi-scale and dilated convolutions, is capable of modeling rich and long-term contextual dependencies. The architecture also consists of multiple multi-scale residual TCN blocks stacked together for feature extraction and can be expressed as:

$$F_{i+1}^{T} = BlockTCN\left(F_{i}^{T}\right) \tag{8}$$

where $BlockTCN$ represents a TCN block that contains multiple convolutional branches with different scales. Let the set of scales be denoted as $S = \{s_j\}_{j=0}^{J}$. The features from the previous layer are extracted through these multi-scale branches to capture multi-scale features can be expressed as:

$$F_{i+1}^{T} = cat\left(\left\{Conv1D\left(F_{i}^{T}, s_j, d_j\right)\right\}_{j=0}^{J}\right) \tag{9}$$

where $Conv1D(\cdot, s, d)$ denotes a multi-scale convolution with a dilation factor, and cat represents the concatenation along the feature dimension. Our set of scales is defined as $S = \{3, 5, 7\}$, and the dilation factor for each convolution layer is set as $d_i = 2^{i-1}$. The final output features, $F_4^T \in \mathbb{R}^{T\times 512}$, are obtained by applying average pooling along the temporal dimension to get the feature representation of the sample, which is then fed into a linear layer followed by a SoftMax layer to compute the classification probabilities. The entire network is optimized using cross-entropy loss minimization.

2.3. Binarization Strategy

Departing from existing quantization methods, the binarization strategy represents the most extreme form of quantization, typically employing the sign function to directly map full-precision weights to either +1 or -1. Although this approach significantly enhances the compression ratio, its restricted representational capacity substantially impacts the performance of neural networks. To address this limitation, we introduced Kernel-Specific Scaling Factors, which adjust the scale of binarized weights to better approximate their full-precision counterparts. Additionally, we replaced the commonly used PReLU activation function with RPReLU to further enhance model performance and stability under quantization.

2.3.1. KERNEL-SPECIFIC SCALING FACTORS

To reduce the discrepancy between full-precision weights and binary weights, a scaling factor $\alpha \in \mathbb{R}^{+}$ is introduced. Let $\mathbf{W} \in R^{c_{out}\times c_{in}\times w\times h}$ and $\mathbf{B} \in \{+1, -1\}^{c_{out}\times c_{in}\times w\times h}$ represent the full-precision and binary weights. To find the optimal estimate $\mathbf{W} \approx \alpha\mathbf{B}$, an optimization problem is defined as follows:

$$J(\mathbf{B}, \alpha) = \|\mathbf{W} - \alpha\mathbf{B}\|^2, \tag{10}$$

$$(\alpha^*, \mathbf{B}^*) = \arg\min_{\alpha,\mathbf{B}} J(\mathbf{B}, \alpha). \tag{11}$$

The error between the real weights $\mathbf{W}$ and the binary weights $\mathbf{B}$ is minimized when:

$$\alpha^* = \frac{\mathbf{W}^{\top}\operatorname{sign}(\mathbf{W})}{n} = \frac{\sum|\mathbf{W}_i|}{n} = \frac{1}{n}\|\mathbf{W}\|_{\ell_1}. \tag{12}$$

For 1D convolutions, the relatively modest parameter count of the kernels leads to correspondingly smaller values of the objective function $J(\mathbf{B}, \alpha)$. However, this relationship becomes increasingly strained as the dimensionality of the kernel expands. Despite the method for computing α remaining consistent, the error associated with this computation exhibits a tendency to escalate exponentially with increasing kernel dimensions.

To mitigate this challenge in both 2D and 3D convolutions, we deviate from the conventional approach of employing a uniform scaling factor across all kernels. Instead, we adopt a Kernel-Specific Scaling Factor strategy, where an individual scaling factor is computed for each kernel, thereby aiming to mitigate the cumulative error across the network.

Fig. 2 illustrates this advancement in scaling factor computation. Traditionally, a single scaling factor is shared across all convolution kernels. This conventional approach calculates a universal scaling factor, applying it uniformly to the results of each convolution kernel. In contrast, our refined method assigns a distinct scaling factor to each convolution kernel, allowing for independent computation. The outputs from these convolutions are then concatenated, incorporating the individually adjusted results. This strategy introduces only $(C_{\text{out}} - 1)$ additional parameters, effectively balancing computational efficiency with significant error reduction.

2.3.2. RPRELU

In order to augment the expressiveness of our network, we incorporated the RPReLU activation function. This approach introduces three learnable parameters: γ, ζ, and β, which augment the capabilities of the PReLU function. The formulation of the RPReLU function is given by:

$$f(x_i) = \begin{cases} x_i - \gamma_i + \zeta_i, & \text{if } x_i > \gamma_i \\ \beta_i(x_i - \gamma_i) + \zeta_i, & \text{if } x_i \leq \gamma_i \end{cases} \tag{13}$$

Here, x_i represents the signal fed into the RPReLU unit for channel i. The parameters γ and ζ are trainable shifts influencing the input's distribution, while β controls the gradient of the negative phase, enhancing the nonlinearity's adaptability across various activation patterns. This modification is visualized in Fig. 3.

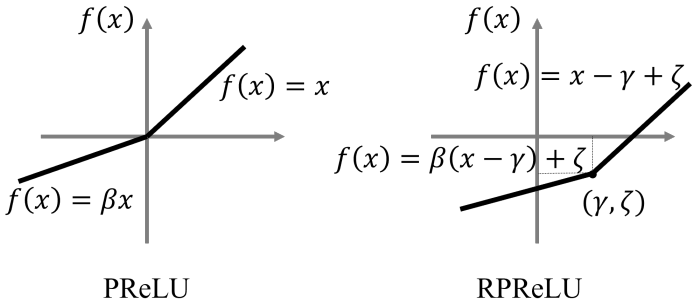

Figure 3: PReLU vs. RPReLU

2.4. Training with Quantization Awareness

In the context of quantized neural networks, there are predominantly two training strategies: quantization-aware training and post-training quantization. With the post-training approach, conventional neural

networks are initially trained using floating-point arithmetic. Subsequently, it is converted to a quantized representation by finding suitable fixed-point formats to approximate the quantization as close to the original network as possible. This approach often leads to significant decreases in accuracy, particularly for binary quantization. To mitigate this, we employ a quantization-aware training strategy, where the imprecision of 1-bit quantization is modeled during the training process. We utilize the straight-through gradient estimator (STE) technique. Specifically, during model inference, weights are binarized directly using the following sign function:

$$\text{sign}(w) = \begin{cases} 1 & \text{if } w > 0 \\ -1 & \text{otherwise} \end{cases} \tag{14}$$

Here, w denotes the convolutional kernel parameters. Since the sign function lacks differentiability, during backpropagation, the sign function is omitted and replaced with the following clip function:

$$\text{clip}(w) = \begin{cases} 1 & \text{if } w > 1 \\ w & \text{if } -1 \leq w \leq 1 \\ -1 & \text{if } w < -1 \end{cases} \tag{15}$$

The derivative during the backward propagation is calculated as follows:

$$\frac{\partial \text{clip}(w)}{\partial w} = \begin{cases} 1 & \text{if } |w| \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

In summary, we propose a binarization strategy that enhances model expressiveness and stability under low-bit precision. To address the limited representational capacity of binary weights, we introduce kernel-specific scaling factors that better approximate full-precision weights. We further employ the adaptive RPReLU activation to improve non-linear modeling capability. During training, quantization-aware training with a straight-through estimator is used to simulate binarization behavior, ensuring alignment between training and inference. These techniques effectively mitigate the performance degradation typically associated with aggressive quantization.

3. Experiment

3.1. Datasets

The DVS-Lip corpus represents the earliest attempt at constructing a lip-reading dataset using event-driven cameras. It was collected using the DAVIS346 event camera, which outputs both event streams and grayscale frames at a resolution of 346 × 260 pixels. The dataset comprises 19,817 instances recorded by 40 volunteers (equally divided between males and females) in indoor settings. Volunteers were instructed to read aloud sequences of five words containing all the words from a lexicon, with each word randomized to prevent repetitive reading patterns. The vocabulary was divided into two parts for the study: one part consisting of 25 pairs of visually similar words, and another part consisting of 50 randomly selected common words. The Montreal Forced Aligner tool was used to determine the start and end times of each word based on the corresponding audio to segment the audio data into word-level samples. Additionally, a face detection tool was employed

to extract 128×128 pixel crops centered around the mouth from the output of the image by the event camera. The design of the dataset takes full advantage of the event camera's ability to capture subtle motion changes, making it suitable for studying lip-reading recognition, an application that requires high temporal resolution and low energy consumption.

3.2. Experimental Setup

In this study, we evaluate STCNet and its binarized version, B-STCNet, in processing event camera data over 60 frames to measure computational efficiency and processing speed in realistic settings. For B-STCNet, full-precision weights were maintained only within the initial convolutional block and the final dense layers, while other layers were quantized to binary values of 1 and -1. The models were trained on an NVIDIA A800 GPU using Adam optimization with a 3e-4 learning rate, 1e-4 regularization strength, and label smoothing as the criterion.

3.3. Experimental Results

Table 1: Evaluation on DVS-Lip. Accuracy1/Accuracy2 refer to the two test partitions; Accuracy is the overall score.

Methods	Accuracy1 (%)	Accuracy2 (%)	Accuracy (%)
TANet (Liu et al., 2021)	58.36	79.17	68.74
ACTION-Net (Wang et al., 2021)	58.32	79.41	68.84
MSTP (Tan et al., 2022)	62.17	82.07	72.10
SSGRU (Dampfhoffer and Mesquida, 2024)			75.3
STCNet (Ours)	**65.86**	**87.43**	**76.62**
B-STCNet (Ours)	**66.03**	**86.26**	**76.15**

The accuracy of different models is present in Tab. 1. Our proposed STCNet demonstrated exceptional performance on the DVS-Lip dataset, achieving state-of-the-art results in both test segments. Specifically, STCNet achieved an accuracy of 65.86% in the first test segment and an even higher accuracy of 87.43% in the second, achieving 76.62% average accuracy. This significantly surpasses the previous best result of 75.3%, showcasing our model's robust capability in processing event-driven lip reading data.

Furthermore, we explored the impact of binarizing the weights of STCNet to assess whether the model could maintain high efficiency while reducing computational resource consumption. Encouragingly, the binarized version, B-STCNet, sustained nearly the same level of performance as the full-precision model. B-STCNet achieved an accuracy of 66.03% in the first test segment and 86.26% in the second, culminating in an average accuracy of 76.15%, closely matching the original non-binarized model.

In addition to reporting the accuracy of STCNet and B-STCNet on the DVS-Lip dataset, we also present a comparison of the parameter count and FLOPs for several methods with high accuracy. Tab. 2 presents a comparative overview of several models, highlighting their architectures, accuracies, parameter counts, and computational costs measured in FLOPs. The STCNet achieves a

commendable accuracy of 76.62% with significantly lower parameter counts (36.06M) and FLOPs (20.97G) compared to other models like MSTP and SSGRU. Notably, the binarized version, B-STCNet, almost matches the accuracy of the full-precision model at 76.15%, while drastically reducing the parameter footprint to just 3.54M and cutting the computational cost to 10.55G FLOPs. These results underscore the effectiveness of binarization in maintaining performance while enhancing computational efficiency, making B-STCNet particularly suited for applications where resources are constrained.

Table 2: Comparison of the parameter count and FLOPs for several methods with high accuracy.

Model	Frontend	Backend	ACC. (%)	Nb.Params(M)	FLOPs(G)
MSTP (Tan et al., 2022)	ResNet-18	BiGRU	72.10	60.3	15.78
SSGRU (Dampfhoffer and Mesquida, 2024)	Spiking ResNet-18	Spiking BiGRU(SpikGRU2+)	75.30	58.6	23.73
STCNet (Ours)	ResNet-18	MS-TCN	76.62	36.06	20.97
B-STCNet (Ours)	Binary ResNet-18	Binary MS-TCN	76.15	**3.54**	10.55

3.4. Ablation Study

To gain a deeper understanding of the impact of individual components on the performance of our B-STCNet model, we performed systematic ablation studies, whose and the corresponding findings are presented in Table 3. Our reference variant, which utilized one single constant scaling factor for binarization and the PReLU activation function, achieved an accuracy of 74.99%. When we introduced the Kernel-Specific Scaling Factor into the baseline model to provide more detailed quantization control, the model's accuracy improved to 75. 58%. Additionally, replacing the activation function with RPReLU enhanced the model accuracy to 75.40%. Ultimately, employing a combination of multiple scaling factors and the RPReLU activation function enabled the model to reach an accuracy of 76.15%. These results demonstrate that each component individually enhances model performance, and their synergistic integration significantly maximizes the model's capability to process event-driven data.

Table 3: Ablation study results for B-STCNet showing the effect of incorporating Kernel-Specific Scaling Factor and RPReLU on the accuracy of the model.

Baseline	Kernel-Specific Scaling Factor	RPReLU	ACC. (%)
✓			74.99
✓	✓		75.58
✓		✓	75.40
✓	✓	✓	**76.15**

4. Conclusion

The significant potential of event cameras to enhance lip reading by leveraging their fine-grained temporal precision and energy-efficient operation is well-recognized. Our introduced Spatio-Temporal Convolutional Network (STCNet) and its binary counterpart, B-STCNet, have shown remarkable

effectiveness in utilizing the strengths of event cameras for this specific task. By introducing B-STCNet, which incorporates Kernel-Specific Scaling Factors and employs quantization-aware training methods, we have successfully mitigated the performance gap typically associated with binarized models. Our evaluation conducted on the DVS-Lip benchmark have validated the effectiveness of this approach, with B-STCNet achieving state-of-the-art accuracy while drastically substantially lowering model parameters by over 90% and floating-point operations by 50%. These results highlight the feasibility of deploying such models on resource-constrained edge devices, making event-based lip reading a practical reality.

References

Manon Dampfhoffer and Thomas Mesquida. Neuromorphic lip-reading with signed spiking gated recurrent units. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2141–2151, 2024.

Qianhui Liu, Meng Ge, and Haizhou Li. Intelligent event-based lip reading word classification with spiking neural networks using spatio-temporal attention features and triplet loss. *Information Sciences*, 675:120660, 2024.

Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. Tam: Temporal adaptive module for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13708–13718, 2021.

Ganchao Tan, Yang Wang, Han Han, Yang Cao, Feng Wu, and Zheng-Jun Zha. Multi-grained spatio-temporal features perceived network for event-based lip-reading. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20094–20103, 2022.

Xubin Wang, Zhiqing Tang, Jianxiong Guo, Tianhui Meng, Chenhao Wang, Tian Wang, and Weijia Jia. Empowering edge intelligence: A comprehensive survey on on-device ai models. *ACM Computing Surveys*, 2025.

Zhengwei Wang, Qi She, and Aljosa Smolic. Action-net: Multipath excitation for action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13214–13223, 2021.