RNAS-MER: A Refined Neural Architecture Search with Hybrid Spatiotemporal Operations for Micro-Expression Recognition

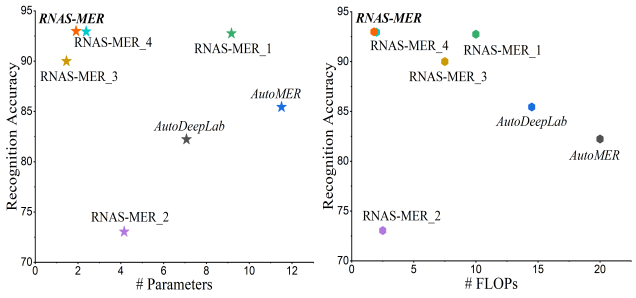
Monu Verma[1] Priyanka Lubal[2], Santosh Kumar Vipparthi[3], Mohamed Abdel-Mottaleb[1]
[1]Electrical and Computer Engineering, University of Miami, USA
[2]Vision Intelligence Lab, Malaviya National Institute of Technology, Jaipur, India
[3]CVPR Lab, Indian Institute of Technology, Ropar, India
monuverma.cv@gmail.com

Abstract

Existing neural architecture search (NAS) methods comprise linear connected convolution operations and use ample search space to search task-driven convolution neural networks (CNN). These CNN models are computationally expensive and diminish the quality of receptive fields for tasks like micro-expression recognition (MER) with limited training samples. Therefore, we propose a refined neural architecture search strategy to search for a tiny CNN architecture for MER. In addition, we introduced a refined hybrid module (RHM) for inner-level search space and an optimal path explore network (OPEN) for outer-level search space. The RHM focuses on discovering optimal cell structures by incorporating a multilateral hybrid spatiotemporal operation space. Also, spatiotemporal attention blocks are embedded to refine the aggregated cell features. The OPEN search space aims to trace an optimal path between the cells to generate a tiny spatiotemporal CNN architecture instead of covering all possible tracks. The aggregate mix of RHM and OPEN search space availed the NAS method to robustly search and design an effective and efficient framework for MER. Compared with contemporary works, experiments reveal that the RNAS-MER is capable of bridging the gap between NAS algorithms and MER tasks. Furthermore, RNAS-MER achieves new state-of-the-art performances on challenging MER benchmarks, including 0.8511%, 0.7620%, 0.9078% and 0.8235% UAR on COMPOSITE, SMIC, CASME-II and SAMM datasets respectively.

1. Introduction

Micro-expressions (MEs) are transient spontaneous and appear briefly on the facial regions but reveal enough visual cues for recognizing genuine human emotions. MEs arise in high-stakes situations when a person tries to hide their real emotions. Therefore, MEs can be instrumental in various psychological applications [23], e.g., lie detection, psychoanalysis, criminal interrogation, depression analysis, autism, and even negotiations.



(a) (b)

Figure 1: Recognition accuracy vs. model size (in terms of the total number of parameters and the total number of flops) on the CASME-II dataset. We plot the results of some recently proposed state-of-the-art NAS-based approaches: AutoDeepLab [14], AutoMER [26], variants of RNAS-MER, and the proposed RNAS-MER.

Due to the involuntary nature and subtle intensity variations in expressive regions, MEs are hard to detect by humans and by machines. In the literature, both traditional [18, 6, 37, 17, 5] and CNN-based algorithms [8, 11, 13, 32, 9, 29, 28, 27] have been effective for micro-expression recognition (MER). However, designing a CNN-based network for MER is a tedious task as it involves trial and error engineering, which requires a lot of effort and domain knowledge. Thus, there is a need for an optimum solution to automatically search and design the best possible CNN architecture instead of spending time and effort in manual CNN architecture designing for MER. NAS algorithms [46] made it possible to develop the best possible CNN architecture for domain-specific tasks by automating the search for optimum architecture search. Initially, NAS algorithms [15, 38] were limited in search of the inner cell structure (similar to auxiliary blocks in conventional CNN models)

only. Further, manual stacking of cells is required to develop deep networks for specific tasks.

In addition, these algorithms require human intervention and create an inconsistency between the cell-level and architecture search spaces[41]. Further, Liu et al. [14] introduced a hierarchical two-level search space for inner cell structure and architecture search for image segmentation to search for optimal architecture. Inspired by the two-level search strategies, Verma et al. [26] introduced a NAS-based solution: AutoMER for MER. However, these NAS algorithms face the problem of heavy computation, which hinders their application in real-world situations. Moreover, Xu et al. [38] developed a partial connection approach named PCDARTS to reduce the computational complexity of cell search-based NAS algorithms [15, 1] for the image classification task. Still, the PCDARTS has inconsistency between cell-level and architecture search. Moreover, state-of-the-art NAS algorithms have constrained search space with linearly connected convolution layers. Further, limited operations were defined to restrict the computational cost in search space. Thus, there is an immense need to develop an effective and efficient integrated inner and outer level search space to search best spatiotemporal CNN architecture for MER applications. These factors motivated us to propose a refined NAS algorithm: RNAS-MER for MER. The proposed RNAS-MER is designed to achieve an efficient and effective NAS-based algorithm by considering three main elements. First, the cell-level search aims to aggregate the multi-scale and complementary features to define local to global receptive semantic fields for MER. These semantic fields are estimated using task-specific fourteen hybrid spatiotemporal operations. In addition to refining the cell-level search space, we introduced a spatiotemporal attention block. Secondly, an optimal path explore network (OPEN) architecture search space is presented to locate an optimal path rather than searching for all possible pathways between cells. The optimal path is revealed by shrinking the search space size by ignoring the redundant paths between the cells to design a tiny spatiotemporal architecture for MER. Third, the memory consumption of NAS is observed and handled by imposing partial connections between hidden nodes and employing the hybrid operations set only on selected feature maps. To summarize, our main contributions are as follows:

1. We propose a refined hybrid module by incorporating multi-scale feature learning and spatiotemporal attention to the search for a robust cell structure for MER applications. Also, we use the partial connection approach between hidden nodes of the cell to reduce the memory footprints for searching.

2. We design a novel optimal path to explore the network, allowing various architecture designs by following random model architecture patterns to discover the best possible shallow and lightweight spatiotemporal CNN architecture design for MER.

3. The proposed RNAS-MER outperforms contemporary NAS as well as MER methods and demonstrates new state-of-the-art performance on COMPOSITE, SMIC, CASME-II, and SAMM datasets. Extensive experiments also show that our proposed RNAS-MER approach is computation efficient, and can perform favourably against existing approaches (see Figure 1).

2. Related Work

The MER approaches can broadly be divided into traditional machine-learning methods and CNN-based deep learning methods. Traditional machine-learning methods utilize descriptors to encode the visual features and forward them to traditional classifiers. Many descriptors [18, 6, 37, 17, 5] have been particularly successful in the handcrafted category. In contrast, CNN-based deep learning methods learn the visual feature and classify MEs. Many CNN-based models [11, 11, 13, 30, 32, 9, 35, 20, 29, 28, 27] have been proposed in past years. The CNN-based MER approaches achieve promising performance. However, designing a robust CNN architecture requires high-level domain knowledge and expertise. Thus, the use of NAS to automatically discover the best CNN architecture attracted the attention of researchers. Initially, reinforcement learning (RL) based on NAS was proposed for image classification. Zoph and Le [46] introduced NAS in RNN to search CNN-LSTM architectures. The RL-based NAS approach directly search the whole network architecture [46]. This approach require expensive computation overheads (e.g., thousands of GPU days) and hinder its applications in real-world scenarios. To alleviate the complexity issue, researchers proposed restricted search spaces. NASNet [47] first introduced a cell-based search space. Specifically, NASNet focuses on the cell structure instead of the whole CNN architecture. In the same line, Liu et al. [15] proposed a differential architecture search (DARTS) to discover the best suitable architecture in a continuous domain. Various search-based [1, 24, 2] approaches were designed to make improvements in performance as well as computation complexity for resource-constrained platforms, such as mobile phones. Further, Wang et al. [31] proposed a Direct Sparse Optimization NAS to prune the useless connections from the searched architecture by imposing sparse regularization. To reduce the GPU days in searching for optimal architecture, Singh et al. [22] introduced a NAS-based approach to reduce the computation complexity by using an augmented search space and super-kernels. Furthermore, to reduce power consumption and redundancy in exploring the network space, Xu et al. [38] introduced partially-connected DARTS by sampling a selected small part of the super-network. However, the above NAS approaches deal only with the inner

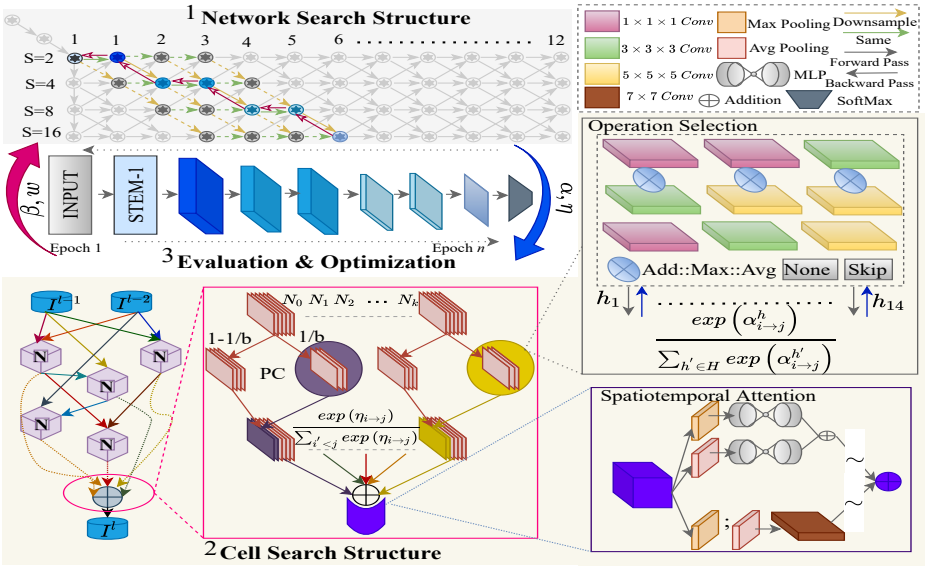


Figure 2: The proposed RNAS-MER framework. The 1^{st} part (gray box) of the framework represents the optimum path explore network (OPEN) search. The light gray nodes and connections imply the abandoned nodes and no possible path for exploration. While 2^{nd} part (yellow box) presents the refine hybrid module (RHM). The 3^{rd} part is depicted in the architecture evaluation and optimization stage. Network and Cell search generates a CNN architecture, and then evaluation and optimization take place to compute the outcome. All 1^{st}, 2^{nd}, and 3^{rd} parts of the framework work simultaneously for each epoch and create a Final CNN architecture with minimum loss. *Here, N and b, represent the hidden nodes and partial connections* $b = 4$.

cell structure search. The final architecture is developed manually by stacking the searched cells. Liu et al. [14] proposed a fully automatic NAS by incorporating two-level search spaces for inner cell and outer architecture, respectively. Recently, Verma et al. [26] proposed a two-level hierarchical search space-based NAS to discover a CNN architecture for micro-expression recognition.

In this paper, we analyse the potential of the hierarchical two-stage search space for exploring the best suitable CNN architecture. Also, we observe the influence of shallow networks and hybrid convolutional layers on MER approaches, respectively. Thus, we incorporate their properties in NAS to improve its effectiveness as well as efficiency.

3. Methodology

This section introduces the essential components of the proposed RNAS-MER (Figure 2) for both inner-level (cell) and outer-level (architecture) search space in detail.

Initially, we introduce a refined hybrid module (RHM) to search for and design an attentive feature cell with compact memory footprints. The compact footprint is embedded through partial connections in resultant feature maps of the nodes for selecting the operations from hybrid spatiotemporal operation sets. Similarly, an optimal path exploring network module is introduced to search the robust and tiny architectures for MER applications. Furthermore, we derived continuously differentiable representations of the cell-level and architecture-level search space.

3.1. The proposed Search Space

3.1.1 Inner-Level Search

The inner level search aims to search for the best suitable cell structure for a given task. The cell is a direct acyclic graph with an elementary structure in a search space representing a collection of available convolution operations. The structure of the cell and optimum selection of cell stacking plays a vital role in selecting more profound or tiny architecture in NAS. Also, the correlation between the cells and convolution operation decides the computational need of the architecture. Inspired by these factors, we propose a refined hybrid module to explore the best cell structure and spatiotemporal operations for the MER task.

The Refined Hybrid module : The refined hybrid module aims to explore an optimal and robust cell structure using the available spatiotemporal operations between the hidden nodes in a cell. The RHM consists of a hybrid operator space (H) with several combinations of convolution operations for the MER task, as shown in Figure 2 (Operation Selection). The details of convolution operations are listed in Table 1. The H aims to aggregate multi-scale and complimentary hybrid receptive responses to define subtle changes in MEs. Also, it allows a cell to estimate inter and intra-emotion class variations of MEs with the pro-

1. Add (ν_1, ν_3)	2. Add (ν_3, ν_5)	3. Add (ν_1, ν_5)	4. Max (ν_1, ν_3)
5. Max (ν_3, ν_5)	6. Max (ν_1, ν_5)	7. Avg (ν_1, ν_3)	8. Avg (ν_3, ν_5)
9. Avg (ν_1, ν_5)	10. ν_1	11. ν_3	12. ν_5
13. Skip	14. None	$\nu_e = (e \times e \times e)$	$3D\ Conv$

Table 1: The hybrid operator set. *e represents the kernel size of the 3D convolution operation.*

posed task-specific fourteen hybrid spatiotemporal operations compared to standard eight operations in the existing NAS approaches [15, 14]. In addition, we observed the inner/cell level search space prone to invade more memory footprints with the size of convolution operations and the number of stacking of cells between the nodes. As a result, the hybrid operator space causes a high GPU memory consumption problem. To overcome the problem of memory inefficiency, we adopted the partial channel connection strategy from the PC-DARTS [38]. Like PC-DARTS, we employed edge normalization over each connection to mitigate the side effects of biasing.

Similarly, the inner level search space's power is refined by adding attention to the cell with spatial and channel axes before fetching it to the following cells. Therefore, we propose spatiotemporal attention by incorporating channel, and spatial attention modules [33]. These modules are employed to aggregate feature responses of all hidden nodes in a decoupled manner. The spatiotemporal attention block refines the cell structure across the network by ignoring the irrelevant or redundant features for MEs.

3.1.2 Outer-Level search

An optimal path exploring network (OPEN) structured search space is proposed to design MER's final spatiotemporal CNN architecture. The OPEN aims to discover a robust and optimal path rather than search for all possible pathways between the cells in [14]. The optimal path is located by shrinking the network search space by reducing the number of layers and paths between cells. The proposed search space structure also benefited from discovering a shallow and light-weighted architecture, which is well proven in the literature [13, 29] for MER. The proposed search space is designed to explore only six cells with a maximum of 16 and a minimum of 2 downsampling factors, as shown in the 1^{st} part of Figure 2. Furthermore, inspired by the literature [29, 27], we exploit the stride convolution operation instead of the pooling operation to reduce the resolution factor for reduction cells.

3.2. Differentiable Representation

Continuous relaxation is described as the discrete architecture for continuous representation. We employ the continuous relaxation over both inner- and outer-level search spaces to derive a fully differentiable search space so that the stochastic gradient descent method could be applicable to discover the best promising spatiotemporal architectures for MER.

3.2.1 Inner-Level Search

In inner-level search, we assign three parameters w, α, and η to search for the best task-driven cell. Specifically, parameter w is used to compute the feature weights. The parameter α is used to select the operation between connections. For example, there is a connection from i to j, and we define a channel sampling mask $M_{i\to j}\{0, 1\}$, where 0 and 1 imply the masked and selected channels. The $\alpha^h_{i\to j}$ is assigned to each hybrid operator $h \in H$ over selected channels. In contrast, the masked channels bypass the hybrid operations and are concatenated directly to the output channels. To compute the architecture weight of a particular operation, we employ the softmax over all possible operations:

$$Aw_{i\to j}\left(I^l_i; M_{i\to j}\right) = \sum_{h\in H} \frac{exp\left(\alpha^h_{i\to j}\right)}{\sum_{h'\in H} exp\left(\alpha^{h'}_{i\to j}\right)} \bullet h\left(M_{i\to j} * I^l_i || \left(1 - M_{i\to j}\right) * I^l_i\right) \quad (1)$$

where, I^l_i represents the feature maps (initially all input frames of a video) of hidden layer $i \to j$ in a l cell. The $\bullet$ and $||$ represents the multiplication concatenation operations, respectively. The Eq 1 contains two parts: $(M_{i\to j}) * I^l_i$ and $(1 - M_{i\to j}) * I^l_i$, which denote the selected and masked channels, respectively. The parameter η is responsible for computing the weight of each edge $i \to j$ to normalize the edges in the architecture like in PC DARTS [38]. The output of I is computed by using Eq. 2.

$$I^l_j = \sum_{i<j} \frac{exp\left(\eta_{i\to j}\right)}{\sum_{i'<j} exp\left(\eta_{i'\to j}\right)} \bullet Aw_{i\to j}\left(I^l_j; M_{i\to j}\right) \quad (2)$$

As discussed in Section 3.1.1, the final outcome of the cell is computed by employing the spatiotemporal attention module [33] on the aggregated feature responses of all hidden nodes (N) using Eq.3:

$$I^l_i = ST\{\sum_{k=1}^{N} I^l_i\{k\}\} \quad (3)$$

where, ST refer to the spatiotemporal attention and is computed using Eq. 4:

$$ST(F) = \sigma\left(MLP^{n_0,n_1}\left(AP\{F\}\right) + MLP^{n_0,n_1}\left(MP\{F\}\right)\right) + \sigma\left(C^{7\times 7}\left(AP\{F\}; MP\{F\}\right)\right) \quad (4)$$

where MLP^{n_0,n_1}, AP, and MP represents the multi-layer perceptron with n_0 and n_1 weights similar to [33], average pooling, and max pooling, respectively. While $C^{7\times 7}$ refers to the convolution operation with 7×7 kernel size. Finally, the cell level can be upgraded through Eq. 5

$$I^l_i = Cell\left(I^{l-1}_{i,s}, I^{l-2}_{i,s}; \alpha, \eta\right) \quad (5)$$

3.2.2 Outer-Level Search

The outer-level OPEN search space is designed to discover light-weighted compact spatiotemporal architecture for MER. As defined in Eq. 1 and 2, the resolution within the cell (inner-level) should be same to enable the summation. While the OPEN search space allows for exploration of the different resolution paths as shown in the 1^{st} part of the Figure 2. The proposed OPEN search space holds four different network states N_S with four resolution sizes $\{2, 4, 8, 16\}$. To discover the best architecture, we assigned a parameter β to each connection between network states (represented by the arrow in Figure 2). The architecture search is explored using Eq. 7.

$$I_{i,s}^{l} = \beta_{\frac{s}{2}\to s}^{l}\left[Cell\left(I_{i,\frac{s}{2}\to s}^{l-1}, I_{i,s\to s}^{l-2}; \alpha, \eta\right)\right] + \beta_{s\to s}^{l}\left[Cell\left(I_{i,s\to s}^{l-1}, I_{i,s\to s}^{l-2}; \alpha, \eta\right)\right] \quad (6)$$

The normalization parameter β should meet the conditions as follows:

$$\beta_{s\to\frac{s}{2}}^{l} + \beta_{s\to s}^{l} = 1 \qquad \forall l, s \quad (7)$$

$$\beta_{s\to\frac{s}{2}}^{l} \geq 0 \qquad \beta_{s\to s}^{l} \geq 0 \qquad \forall l, s \quad (8)$$

To optimize the parameters: α and β, we adopted the bi-level optimization method using gradient descent [15]. Furthermore, the evaluation and optimization for searching and training are performed by following Algorithm 1. While decoding of the optimum path in OPEN architecture search space is performed by the Algorithm 2.

Algorithm 1 Evaluation and Optimization Algorithm

Data → Training Dataset
Outcome→ RNAS-MER with optimized $\alpha, \eta, and \beta$.
1. ***Search Stage:***
Create a search space, including OPEN (outer-level) and RHM (inner-level) search structure for MER task.
Devide the dataset into two parts Train A and Train B.
while not coverage **do**
$w \leftarrow w - \nabla(w)\, Loss_{Train\ A}(K)$;
$\alpha, \eta, \beta \leftarrow \alpha, \eta, \beta - \nabla(\alpha, \eta, \beta)\, Loss_{Train\ B}(K)$
$K = (w, \alpha, \eta, \beta)$
end while
α parameter is responsible for hybrid operation selection and applied only on selected channels; Divide the channels into two parts, $\frac{1}{b}$ are used as selected channels and remaining are considered as masked channels.
η parameter is responsible for edge normalization over hidden nodes connections in inner-cell space.
β parameter is responsible for optimum path selection in OPEN outer-level space.
2. ***Decode Stage:***
Decode the spatiotemporal CNN architecture for MER based on optimized α, η, β parameters.
3. ***Training Stage:***
Arrange the dataset in according to subject IDs.
for all $c \in Subject\ IDs$ **do**
while not coverage **do**
$w \leftarrow w - \nabla(w)\, Loss_{Train}(w)$
Train ← Subject IDs -c
end while
end for
Generate outcome as spatiotemporal CNN model for MER from RNAS-MER with trained weight w parameters for all subjects.

4. Experimental Results and Analysis

This section discusses the dataset and evaluation strategies. Further, a comparative study of the proposed RNAS-MER and state-of-the-art approaches is presented. We carry out the ablation study and computational analysis in the following subsection. Moreover, the implementation settings of the architecture search and training are detailed in the supplementary document.

4.1. Datasets

Recent research in the micro-expression community is influenced by the MEGC-19 challenge [21] and uses SMIC [12], CASME-II [39], and SAMM [4] databases and their composite version as evaluation standards for MER [13, 40]. Therefore, to make a fair comparison with state-of-the-art approaches, we adopt the same evaluation standard for the composite dataset. Specifically, the CASME-II dataset contains five MEs classes, i.e., happiness, surprise, disgust, repression, and others. The SMIC dataset comprises three MEs classes, i. e., negative, positive, and surprise. The SAMM dataset includes seven MEs classes, happiness, surprise, disgust, repression, anger, fear, and contempt. To establish the compatibility between all three datasets, we have merged the classes and annotated them with new MEs classes as positive (happy), negative (disgust, sad, fear), and surprise, like the MEGC-19 challenge. All experiments are conducted over a leave-one-subject-out validation strategy. Moreover, all datasets comprise a varied number of image sequences. Thus, to come up with a uniform composition of all samples in a dataset, we adopted a temporal interpolation model (TIM) [45] to normalize the video sequences into equal lengths. Moreover, to aid more visibility of MEs' involuntary changes, we utilized the magnified videos [19] in our experiments.

4.2. Experimental Results Comparison

The performance of the existing MER methods, NAS-based methods, and proposed RNAS-MER is evaluated in terms of recognition accuracy, unweighted average recall

Methods	Pub-Yr	COMPOSITE			SMIC		CASME-II		SAMM	
		Acc	UAR	UF1	UAR	UF1	UAR	UF1	UAR	UF1
Dual_Inc^{OF} [44]	FG-19	N/A	0.7278	0.7322	0.6726	0.6645	0.8560	0.8621	0.5663	0.5868
STSTNet^{OF} [13]	FG-19	0.7692	0.7605	0.7353	0.7013	0.6801	0.8686	0.8382	0.6810	0.6588
NMER^{OF}[16]	FG-19	N/A	0.7824	0.7885	0.7530	0.7461	0.8209	0.8293	0.7152	0.7754
CapsuleNet [25]	FG-19	N/A	0.6506	0.6520	0.5877	0.5820	0.7018	0.7068	0.5989	0.6209
MTMNet^{MaM}[34]	ACMMM-20	N/A	**0.8570**	**0.8640**	**0.8610**	**0.8640**	0.8720	0.8700	0.8190	0.8250
ICE-GAN [40]	Arxiv-20	N/A	0.8410	0.8450	0.7910	0.7900	0.8680	0.8760	0.8230	0.8450
CLFM^{LF} [3]	IEEE-Acc-20	N/A	0.7200	0.750	0.7100	0.7100	0.7700	0.7200	0.5100	0.6500
RCN-F^{OF} [36]	IEEE-TIP-20	N/A	0.7052	0.7164	0.5980	0.5991	0.8087	0.8563	0.6771	0.6976
FR^{OF} [43]	PR-21	N/A	0.7832	0.7838	0.7083	0.7011	0.8873	0.8915	0.7155	0.7372
Two-stage MER^{OF} [42]	Neu. Comp-21	N/A	0.7986	0.8068	0.7598	0.7356	0.8763	0.8818	0.7280	0.7475
Graph-CNN [10]	CVPR-W-21	N/A	0.7933	0.7914	0.7215	0.7192	0.8710	0.8798	0.7890	0.7751
AutoDeepLab* (3D) [14]	CVPR-19	0.8083	0.7372	0.6993	0.6433	0.5920	0.7240	0.7157	0.7609	0.7192
AutoMER* (3D) [26]	TNNLS-21	0.7991	0.7210	0.6858	0.7225	0.6725	0.7583	0.7334	0.7077	0.6508
RNAS-MER	***Proposed***	***0.9029***	***0.8511***	***0.8302***	***0.7620***	***0.7443***	***0.9078***	***0.8985***	***0.8235***	***0.7880***

*Here, the suffix * refers to the re-evaluated results for existing methods. OF, MaM, and LF represents the optical flow, macro to micro feature adaption, and landmark feature maps (extra features utilize by the MER approaches).*

Table 2: Performance comparison of existing deep learning as well as NAS based MER approaches and proposed RNAS-MER on leave-one-subject-out (LOSO) validation setup.

Algorithm 2 Decoding OPEN Architecture Structure

Data $\rightarrow$ path weights parameters.
Outcome$\rightarrow$ The optimum path with maximum probability.
$O_{max}\left(o_1, ...o_l, ...o_L\right), L = 6.$
Initialize the starting node with the $P_{L=0}^{s=2}$ probability.
Initialize the 4 paths $\{O^2, O^4, O^8, O^{16}\}$.
while l¡L **do**
  for s in $\{2, 4, 8, 16\}$ **do**
    $P_l^s \leftarrow max\{P_{l-1}^{\frac{s}{2}}\beta_l^{\frac{s}{2}\rightarrow s}, P_{l-1}^{s\beta_l^s\rightarrow s}\};$
    Update the current path O^s
  end for
end while
Generate the max probability for optimum path $O_{max} \leftarrow O^s$.

Methods	Par. Norm.	PC	D_Op	H_Op	ST_Att.
RNAS-MER_1	α, β	✗	✓	✗	✗
RNAS-MER_2	α, β, η	✓	✓	✗	✓
RNAS-MER_3	α, β, η	✓	✗	✓	✗
RNAS-MER_4	α, β	✗	✗	✓	✓
RNAS-MER	α, β, η	✓	✗	✓	✓

Here, PC, D_Op, H_Op, and ST_Att. stands for partial connections, darts operations, hybrid operations, and spatiotemporal attention block, respectively.

Table 3: Experimental settings for ablation models and the proposed RNAS-MER.

(UAR), and unweighted F1-score (UF1). The UAR and UF1 score are used to validate the performance of the proposed ME-NAS concerning the imbalanced expression distribution. The quantitative results are tabulated in Table 2. From Table 2 it is clear that, the proposed RNAS-MER achieves 0.1139%, 0.1301% and 0.1309%, 0.1444% more UAR and UF1 as compared to AutoDeepLab and AutoMER over the COMPOSITE dataset, respectively. Similarly, for SMIC, the proposed RNAS-MER gains 0.1187%, 0.0395% and 0.1523%, 0.0718%, an improvement over AutoDeepLab and AutoMER in terms of UAR and UF1-Score, respectively. Whereas, the RNAS-MER outperformed the existing AutoDeepLab and AutoMER by 0.1921%, 0.1495% and 0.1828%, 0.1651%, UAR and UF1, respectively, for CASME-II. Furthermore, for SAMM the proposed method achieves 0.0626%, 0.1158% and 0.0688%, 0.1372%, improvement in UAR and UF1, respectively, as compared to AutoDeepLab and AutoMER. Moreover, the final search space and cell structure for CASME-II generated by AutoDeepLab, AutoMER, and the proposed RNAS-MER are shown in Fig. 3. From the above discussion, it is evident that the proposed RNAS-MER achieved impressive performance compared to state-of-the-art MER and NAS methods for almost all datasets. The results show that the state-of-the-art MER approach [34] is achieving better results than the proposed RNAS-MER for COMPOSITE and SMIC datasets. We observe that Xia et al. [34] used extra features of macro-expressions to aid the guidance for MEs, which allows the network to learn sufficient features and improve the performance of the MER. However, [34] requires auxiliary macro-expression data samples and also follows the two-stage network, which is not recommendable in real-time applications.

Methods	Type	COMPOSITE			SMIC		CASME-II		SAMM	
		Acc	UAR	UF1	UAR	UF1	UAR	UF1	UAR	UF1
RNAS-MER_1	Ablation	0.8760	0.7817	0.7619	0.5941	0.5393	0.9015	0.8887	0.7923	0.7564
RNAS-MER_2	Ablation	0.8371	0.7645	0.7217	0.6916	0.6492	0.6202	0.5806	0.7596	0.7157
RNAS-MER_3	Ablation	0.7987	0.7555	0.7204	0.6407	0.5929	0.8353	0.8255	0.6407	0.5929
RNAS-MER_4	Ablation	0.8392	0.7762	0.7472	0.6493	0.5847	0.8797	0.8754	0.7819	0.7514
RNAS-MER	***Proposed***	***0.9029***	***0.8511***	***0.8302***	***0.7620***	***0.7443***	***0.9078***	***0.8985***	***0.8235***	***0.7880***

Table 4: Performance comparison of ablation study and proposed RNAS-MER on LOSO validation setup.

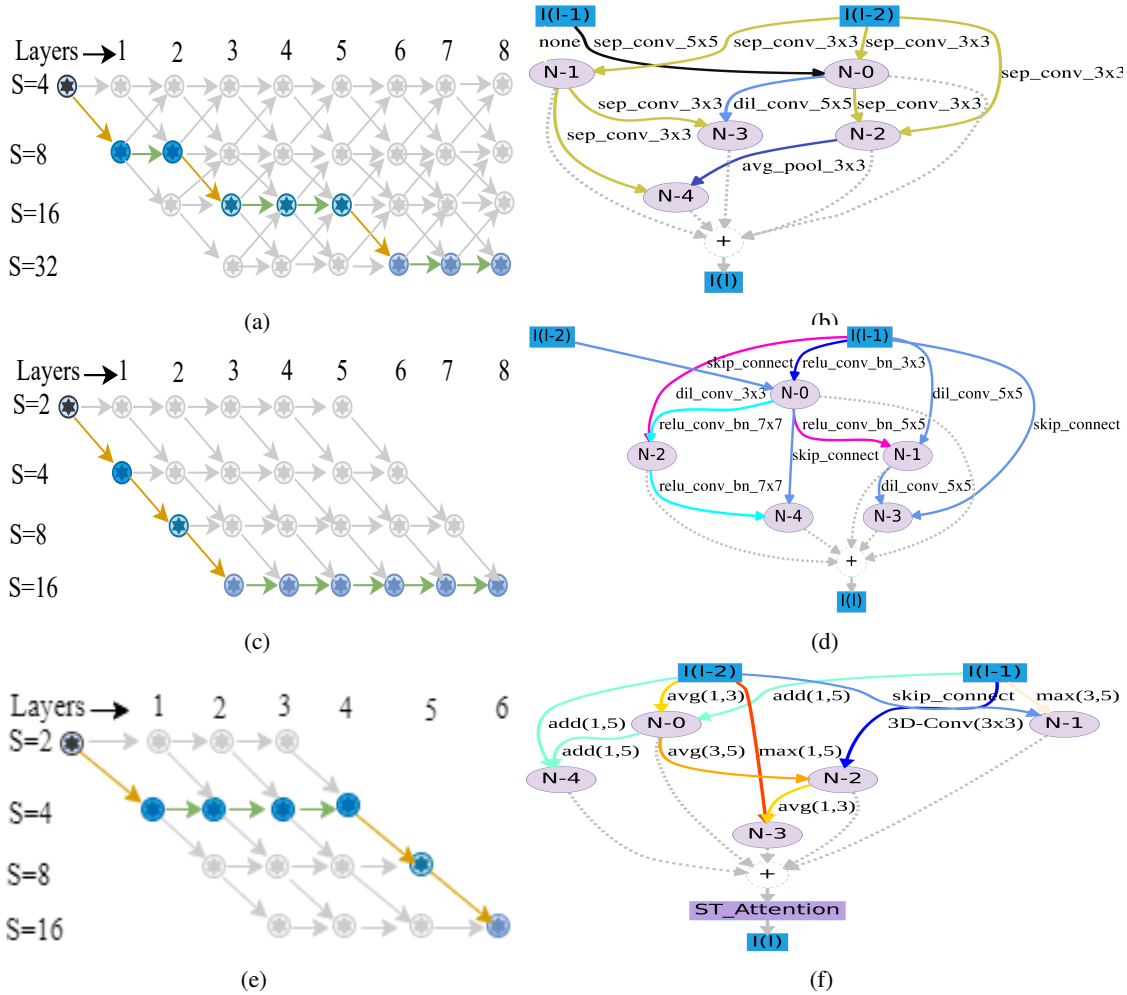


Figure 3: The final network path and cell structures discovered by the state-of-the-art: AutoDeepLab (a,b), AutoMER (c,d), and proposed RNAS (e,f), respectively, over the CASME-II dataset.

4.3. Ablation Study

In this section, we analyse the impact of each component of the RNAS-MER by conducting four supplementary experiments. The detailed experimental details of the ablation study are presented in Table 3.

Impact of Partial Connection and Hybrid Operation set: To validate the impact of partial connections and proposed Hybrid operation set, we have conducted an experiment with DARTS [15] (cell search space) and proposed OPEN outer-level search space in study 1 (RNAS-MER_1). The experimental results for RNAS-MER_1 are tabulated in Table 4 over four datasets: COMPOSITE, SMIC, CASME-II, and SAMM. From the results, it is evident that the existing search space and operations of [15] degrade the performance in MER as compared to the proposed RNAS-MER. Moreover, RNAS-MER_1 is computationally very expen-

Methods	Type	#Parameters	#Memory
LEARNet [29]	2D-CNN	1.8M	N/A
AffectNet [27]	2D-CNN	2.2M	8.30MB
ICE-GAN [40]	2D-CNN	6.7M	N/A
DualInc [7]	2D-CNN	6.48M	N/A
FR [43]	2D-CNN	10.24M	N/A
AutoDeepLab [14]	3D-CNN	7.07M	57.00MB
AutoMER [26]	3D-CNN	11.51M	92.40MB
RNAS-MER_1	3D-CNN	9.17M	73.70MB
RNAS-MER_2	3D-CNN	4.16M	33.50MB
RNAS-MER_3	3D-CNN	1.46M	11.90MB
RNAS-MER_4	3D-CNN	2.38M	19.20MB
RNAS-MER	***3D-CNN***	***1.91M***	***15.60MB***

Table 5: Computational Complexity Analysis of state-of-the-art MER approaches and proposed RNAS-MER with its four variants.

sive (requires 9.17Million parameters) as compared to the proposed RNAS-MER (requires 1.91Million parameters).

Impact of Hybrid operation set: To further investigate the effect of the Hybrid operation set in the proposed RNAS-MER, the Hybrid operations are replaced with existing operations [15] with proposed OPEN outer-level search space in study 2 (RNAS-MER_2). The quantitative results are tabulated in Table 4. From the results, it is clear that the proposed Hybrid operations play an important role in designing a robust architecture for MER.

Impact of spatiotemporal block: To validate the efficiency of spatiotemporal block in cell structure, we conducted the experiment without spatiotemporal block in study-3 (RNAS-MER_3). The performance of RNAS-MER_3 is tabulated in Table 4. From the table, it is clear that the proposed RNAS-MER outperform the RNAS-MER_3. Thus, we can conclude that the proposed spatiotemporal attention block can refine the aggregated features of the hidden nodes in the cell and enhances the performance of the proposed RNAS-MER.

Impact of edge normalization:To validate the effectiveness of the edge normalization with parameter η in the inner/cell search space of RNAS-MER, we evaluate the performance of the RNAS-MER without edge normalization in study-4 (RNAS-MER_4). The impact of RNAS-MER_4 in MER is tabulated in Table 4. From the table, it is evident that edge normalization plays an important role in handling the biasing and enhance the performance.

4.4. Computation Complexity

The computation complexity of the proposed RNAS-MER, RNAS variants and state-of-the-art approaches is compared in terms of the number of parameters, number of flops, and memory needed for trained MER models. The total number of parameters, number of (floating point operations) flops, and memory engaged in each network are represented in Table 5 and Figure 1. From the results, it is clear that the proposed RNAS-MER requires a much smaller number of parameters and less memory compared to other state-of-the-art MER approaches as well as NAS-based approaches like AutoDeepLab and AutoMER. The RNAS-MER trained model requires only 1.91 million (M) parameters and 15.60 megabytes (MB) memory footprints, while AutoDeepLab and AutoMER NAS approaches need 7.07 M, 11.51 M, and 57.0 MB, 92.40 MB parameters and memory space, respectively. Moreover, from Figure. 1, we can see that the proposed RNAS-MER attain the highest performance with the lowest computation cost and a number of flops. *Thus, based on the results, we can conclude that the proposed RNAS-MER can discover and design effective and efficient spatiotemporal MER architecture.*

5. Conclusion

In this paper, we proposed RNAS-MER: refined neural architecture search strategy to search a tiny CNN architecture for MER. The RNAS-MER is designed by following a hierarchical two-level search by including inner-level and outer-level search spaces. For the inner level, we proposed a refined hybrid module. The RHM aims to discover optimal cell structures by incorporating a multilateral hybrid spatiotemporal operation space. Also, we proposed a spatiotemporal attention block to refine the aggregated cell features. For outer-level search space, we introduced an optimal path explore network (OPEN). The OPEN search space aims to trace an optimal path between the cells to generate a tiny spatiotemporal CNN architecture. Moreover, the proposed RHM and OPEN jointly promote searching the shallow CNN architecture required for reliable training with fewer data samples in MER. The performance of RNAS-MER is evaluated using four benchmark datasets: COMPOSITE, SMIC, CASME-II, and SAMM in terms of recognition accuracy, unweighted average recall, and unweighted F1-score, respectively. The extensive four ablation studies are executed to study the contribution of each component of the proposed RNAS-MER. The experimental results and the computational complexity analysis validate the effectiveness of RNAS-MER as compared to state-of-the-art MER methods. Moreover, we analyse that the inner/cell level search space prone to invade more memory footprints with the size of convolution operations and the number of stacking of cells between the nodes. As a result, the hybrid operator space with 14 convolution operations needs a high GPU memory consumption for inner/cell level searching. In addition, the proposed RNAS-MER is designed specifically for MER application. In the future, we will focus on designing a NAS-based algorithm for other computer vision applications with less computational time.

References

[1] Han Cai, Ligeng Zhu, and Song Han. Proxylessnas: Direct neural architecture search on target task and hardware. *arXiv preprint arXiv:1812.00332*, 2018.

[2] Xin Chen, Lingxi Xie, Jun Wu, and Qi Tian. Progressive differentiable architecture search: Bridging the depth gap between search and evaluation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1294–1303, 2019.

[3] Dong Yoon Choi and Byung Cheol Song. Facial micro-expression recognition using two-dimensional landmark feature maps. *IEEE Access*, 8:121549–121563, 2020.

[4] Adrian K Davison, Cliff Lansley, Nicholas Costen, Kevin Tan, and Moi Hoon Yap. Samm: A spontaneous micro-facial movement dataset. *IEEE transactions on affective computing*, 9(1):116–129, 2016.

[5] Xiaohua Huang, Su-Jing Wang, Xin Liu, Guoying Zhao, Xiaoyi Feng, and Matti Pietikäinen. Discriminative spatiotemporal local binary pattern with revisited integral projection for spontaneous facial micro-expression recognition. *IEEE Transactions on Affective Computing*, 10(1):32–47, 2017.

[6] Xiaohua Huang, Su-Jing Wang, Guoying Zhao, and Matti Piteikainen. Facial micro-expression recognition using spatiotemporal local binary pattern with integral projection. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 1–9, 2015.

[7] Huai-Qian Khor, John See, Sze-Teng Liong, Raphael CW Phan, and Weiyao Lin. Dual-stream shallow networks for facial micro-expression recognition. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 36–40. IEEE, 2019.

[8] Huai-Qian Khor, John See, Raphael Chung Wei Phan, and Weiyao Lin. Enriched long-term recurrent convolutional network for facial micro-expression recognition. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 667–674. IEEE, 2018.

[9] Dae Hoe Kim, Wissam J. Baddar, Jinhyeok Jang, and Yong Man Ro. Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition. In *IEEE Transactions on Affective Computing*, volume 10, pages 223–236, 2017.

[10] Ling Lei, Tong Chen, Shigang Li, and Jianfeng Li. Micro-expression recognition based on facial graph representation learning and facial action unit fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1571–1580, 2021.

[11] Jing Li, Yandan Wang, John See, and Wenbin Liu. Micro-expression recognition based on 3d flow convolutional neural network. *Pattern Analysis and Applications*, 22(4):1331–1339, 2019.

[12] Xiaobai Li, Tomas Pfister, Xiaohua Huang, Guoying Zhao, and Matti Pietikäinen. A spontaneous micro-expression database: Inducement, collection and baseline. In *2013 10th IEEE International Conference and Workshops on Automatic face and gesture recognition (fg)*, pages 1–6. IEEE, 2013.

[13] Sze-Teng Liong, Yee Siang Gan, John See, Huai-Qian Khor, and Yen-Chang Huang. Shallow triple stream three-dimensional cnn (ststnet) for micro-expression recognition. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–5. IEEE, 2019.

[14] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan L Yuille, and Li Fei-Fei. Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 82–92, 2019.

[15] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

[16] Yuchi Liu, Heming Du, Liang Zheng, and Tom Gedeon. A neural micro-expression recognizer. In *2019 14th IEEE international conference on automatic face & gesture recognition (FG 2019)*, pages 1–4. IEEE, 2019.

[17] Yong-Jin Liu, Bing-Jun Li, and Yu-Kun Lai. Sparse mdmo: Learning a discriminative feature for spontaneous micro-expression recognition. *IEEE Transactions on Affective Computing*, 2018.

[18] Yong-Jin Liu, Jin-Kai Zhang, Wen-Jing Yan, Su-Jing Wang, Guoying Zhao, and Xiaolan Fu. A main directional mean optical flow feature for spontaneous micro-expression recognition. *IEEE Transactions on Affective Computing*, 7(4):299–310, 2015.

[19] Tae-Hyun Oh, Ronnachai Jaroensri, Changil Kim, Mohamed Elgharib, Fr'edo Durand, William T Freeman, and Wojciech Matusik. Learning-based video motion magnification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 633–648, 2018.

[20] Sai Prasanna Teja Reddy, Surya Teja Karri, Shiv Ram Dubey, and Snehasis Mukherjee. Spontaneous facial micro-expression recognition using 3d spatiotemporal convolutional neural networks. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[21] John See, Moi Hoon Yap, Jingting Li, Xiaopeng Hong, and Su-Jing Wang. Megc 2019–the second facial micro-expressions grand challenge. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–5. IEEE, 2019.

[22] Shashank Singh, Ashish Khetan, and Zohar Karnin. Darc: Differentiable architecture compression. *arXiv preprint arXiv:1905.08170*, 2019.

[23] Madhumita Takalkar, Min Xu, Qiang Wu, and Zenon Chaczko. A survey: facial micro-expression recognition. *Multimedia Tools and Applications*, 77(15):19301–19325, 2018.

[24] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V Le. Mnasnet: Platform-aware neural architecture search for mobile. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2820–2828, 2019.

[25] Nguyen Van Quang, Jinhee Chun, and Takeshi Tokuyama. Capsulenet for micro-expression recognition. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–7. IEEE, 2019.

[26] Monu Verma, M Satish Kumar Reddy, Yashwanth Reddy Meedimale, Murari Mandal, and Santosh Kumar Vipparthi. Automer: Spatiotemporal neural architecture search for microexpression recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[27] Monu Verma, Santosh Kumar Vipparthi, and Girdhari Singh. Affectivenet: Affective-motion feature learning for microexpression recognition. *IEEE MultiMedia*, 28(1):17–27, 2020.

[28] Monu Verma, Santosh Kumar Vipparthi, and Girdhari Singh. Non-linearities improve originet based on active imaging for micro expression recognition. *arXiv preprint arXiv:2005.07991*, 2020.

[29] Monu Verma, Santosh Kumar Vipparthi, Girdhari Singh, and Subrahmanyam Murala. Learnet: Dynamic imaging network for micro expression recognition. *IEEE Transactions on Image Processing*, 29:1618–1627, 2019.

[30] Chongyang Wang, Min Peng, Tao Bi, and Tong Chen. Micro-attention for micro-expression recognition. *arXiv preprint arXiv:1811.02360*, 2018.

[31] Naiyan Wang, Shiming XIANG, Chunhong Pan, et al. You only search once: Single shot neural architecture search via direct sparse optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[32] Su-Jing Wang, Bing-Jun Li, Yong-Jin Liu, Wen-Jing Yan, Xinyu Ou, Xiaohua Huang, Feng Xu, and Xiaolan Fu. Micro-expression recognition with small sample size by transferring long-term convolutional neural network. In *Neurocomputing 312*, pages 251–262, 2018.

[33] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[34] Bin Xia, Weikang Wang, Shangfei Wang, and Enhong Chen. Learning from macro-expression: A micro-expression recognition framework. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2936–2944, 2020.

[35] Zhaoqiang Xia, Xiaopeng Hong, Xingyu Gao, Xiaoyi Feng, and Guoying Zhao. Spatiotemporal recurrent convolutional networks for recognizing spontaneous micro-expressions. *IEEE Transactions on Multimedia*, 22(3):626–640, 2019.

[36] Zhaoqiang Xia, Wei Peng, Huai-Qian Khor, Xiaoyi Feng, and Guoying Zhao. Revealing the invisible with model and data shrinking for composite-database micro-expression recognition. *IEEE Transactions on Image Processing*, 29:8590–8605, 2020.

[37] Feng Xu, Junping Zhang, and James Z Wang. Microexpression identification and categorization using a facial dynamics map. *IEEE Transactions on Affective Computing*, 8(2):254–267, 2017.

[38] Yuhui Xu, Lingxi Xie, Xiaopeng Zhang, Xin Chen, Guo-Jun Qi, Qi Tian, and Hongkai Xiong. Pc-darts: Partial channel connections for memory-efficient architecture search. *arXiv preprint arXiv:1907.05737*, 2019.

[39] Wen-Jing Yan, Xiaobai Li, Su-Jing Wang, Guoying Zhao, Yong-Jin Liu, Yu-Hsin Chen, and Xiaolan Fu. Casme ii: An improved spontaneous micro-expression database and the baseline evaluation. *PloS one*, 9(1):e86041, 2014.

[40] Jianhui Yu, Chaoyi Zhang, Yang Song, and Weidong Cai. Ice-gan: identity-aware and capsule-enhanced gan for micro-expression recognition and synthesis. 2020.

[41] Qihang Yu, Dong Yang, Holger Roth, Yutong Bai, Yixiao Zhang, Alan L Yuille, and Daguang Xu. C2fnas: Coarse-to-fine neural architecture search for 3d medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4126–4135, 2020.

[42] Sirui Zhao, Hanqing Tao, Yangsong Zhang, Tong Xu, Kun Zhang, Zhongkai Hao, and Enhong Chen. A two-stage 3d cnn based learning method for spontaneous micro-expression recognition. *Neurocomputing*, 448:276–289, 2021.

[43] Ling Zhou, Qirong Mao, Xiaohua Huang, Feifei Zhang, and Zhihong Zhang. Feature refinement: An expression-specific feature learning and fusion method for micro-expression recognition. *Pattern Recognition*, 122:108275, 2022.

[44] Ling Zhou, Qirong Mao, and Luoyang Xue. Dual-inception network for cross-database micro-expression recognition. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–5. IEEE, 2019.

[45] Ziheng Zhou, Guoying Zhao, and Matti Pietikäinen. Towards a practical lipreading system. In *CVPR 2011*, pages 137–144. IEEE, 2011.

[46] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.

[47] Barret Zoph, Vijay Vasudevan, Le Jonathon Shlens, and Quoc V. Learning transferable architectures for scalable image recognition. In *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018.