IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2021)

Global2Local: Efficient Structure Search for Video Action Segmentation

Shang-Hua Gao[1]* Qi Han[1]* Zhong-Yu Li[1]
Pai Peng[2] Liang Wang[3] Ming-Ming Cheng[1] †
TKLNDST, CS, Nankai University[1] Tencent[2] NLPR[3]
http://mmcheng.net/g2lsearch

Abstract

Temporal receptive fields of models play an important role in action segmentation. Large receptive fields facilitate the long-term relations among video clips while small receptive fields help capture the local details. Existing methods construct models with hand-designed receptive fields in layers. Can we effectively search for receptive field combinations to replace hand-designed patterns? To answer this question, we propose to find better receptive field combinations through a global-to-local search scheme. Our search scheme exploits both global search to find the coarse combinations and local search to get the refined receptive field combination patterns further. The global search finds possible coarse combinations other than human-designed patterns. On top of the global search, we propose an expectation guided iterative local search scheme to refine combinations effectively. Our global-to-local search can be plugged into existing action segmentation methods to achieve state-of-the-art performance. The source code is publicly available on http://mmcheng.net/g2lsearch.

1. Introduction

Action recognition segments the action of each video frame, playing an important role in computer vision applications such as clips tagging [59], video surveillance [8, 9], and anomaly detection [54]. While conventional works [4, 17, 18, 56] have continuously refresh the recognition performance of short trimmed videos containing a single activity, segmenting each frame densely in long untrimmed videos remains challenging as those videos contain many activities with different temporal lengths. Temporal convolutional networks (TCN) [12, 16, 35, 40, 65] are widely adapted in action segmentation tasks with their ability to capture both long-term and short-term information. Appropriate receptive fields in layers are crucial for TCN as large receptive fields contribute to long-term dependencies while small receptive fields benefit the local details. State-of-the-art (SOTA) methods [5, 29, 39, 40, 65] rely on human-designed receptive field combinations, *i.e.,* dilation rate or pooling size in each layer, to make the trade-off between capturing long and short term dependencies. Questions have raised: Are there other effective receptive field combinations that perform comparable or better than hand-designed patterns? Will the receptive field combinations vary among different datasets? To answer those questions, we propose to find the possible receptive field combinations in a coarse-to-fine scheme through the global-to-local search.

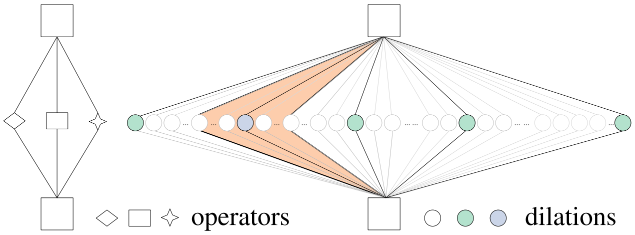


Figure 1. Search space comparison between searching for network architecture and receptive field combinations. Left: Network architecture search mostly search for several operations with different functions. Right: The search space of receptive field combinations is huge. The white, green, blue nodes and orange shade represent the dilation rate candidates, the sparse search space in global search, one of the global searched results, and the local search space.

As shown in Fig. 1, unlike the existing network architecture search spaces [3, 27, 43] that only contain several operation options within a layer, the available search space of receptive field combinations could be huge. Suppose a TCN has L convolutional layers and D possible receptive fields in each layer. There are D^L possible combinations, *i.e.,* the number of possible receptive field combinations in MS-TCN [12] is 1024^{40}. Directly apply network architecture searching algorithms [27, 41, 43, 66] to such a huge search space is impractical. For example, conventional reward-based searching methods [42, 47, 66] are not suitable for CNN-based models with a huge search

*Equal contribution

†M.M. Cheng (cmm@nankai.edu.cn) is the corresponding author.

space. The model training and performance evaluation of each possible combination are too costly. Differentiable architecture searching methods (DARTS) [3, 41, 43] rely on shared big networks to save training time, thus only supporting several operators within a layer due to the model size constraint. Moreover, they heavily dependent on the initial combination and fail to find new combinations with a huge difference from the initial one. While our goal is to explore effective receptive field combinations other than human-designed patterns in the huge search space, those algorithms are either too costly or cannot support the large search space.

To explore the search space with low cost, we exploit both a genetic-based global search to find the coarse receptive field combinations and an expectation guided iterative (EGI) local search to get the refined combinations. Specifically, we follow the MS-TCN [12] to use dilation rates to determine layers' receptive fields. A genetic-based global search scheme is proposed to find coarse combinations within a sparsely sampled search space at an affordable cost. The global search discovers various combinations that achieve even better performance than human designings but have completely different patterns. Based on the global-searched coarse combinations, we propose the local search to determine fine-grained dilation rates. Our proposed convolutional weight-sharing scheme enforces learned dilation weights to approximate the probability mass distribution for calculating the expectation of dilation rates. The expectation guided searching transfer the discrete dilation rates into a distribution, allowing fine-grained dilation rates searching. With an iteratively searching process, the local search gradually finds more effective fine-grained receptive field combinations with low cost. Our proposed global-to-local search scheme can be plugged into existing models, surpassing human-designed structures with impressive performance gain. In summary, we make two major contributions:

- The expectation guided iterative local search scheme enables searching fine-grained receptive field combinations in the dense search space.
- The global-to-local search discovers effective receptive field combinations with better performance than hand-designed patterns.

2. Related Work

2.1. Action Segmentation

Many approaches have been proposed for modeling dependencies for action segmentation. Early works [13–15] mostly model the changing state of appearance and actions with sliding windows [2, 30, 51]. Thus they mainly focus on short-term dependencies. Capturing both short-term and long-term dependencies then gradually becomes the focus of action segmentation.

Sequential Model. Sequential models capture long-short term dependencies in an iterative form. Vo and Bobick [64] apply the Bayes network to segment actions represented with the stochastic context-free grammar. Tang *et al.* [63] use a hidden Markov model to model transitions between states and durations. Later, hidden Markov models are combined with context-free grammar [32], Gaussian mixture model [33], and recurrent networks [34, 50] to model long-term action dependencies. Cheng *et al.* [7] apply the sequence memorizer to capture long-range dependencies in visual words learned from the video. However, these sequential models are inflexible in parallel modeling long-term dependencies and usually suffer from information forgetting [12, 40].

Multi-stream Architecture. Some researchers [10, 49, 57, 58] utilize multi-stream models to model dependencies from both the long and short term. Richard and Gall employ [49] dynamic programming to inference models composed of length model, language model, and action classifier. Singh *et al.* [57] learn short video chunks representation with a two-stream network and pass these chunks to a bi-directional network to predict action segmentation results sequentially. A three-stream architecture is proposed in [58], which contains egocentric cues, spatial and temporal streams. Tricornet [10] utilizes a hybrid temporal convolutional and recurrent network to capture local motion and memorize long-term action dependencies. CoupledGAN [19] uses a GAN model to utilize multi-modal data to better model human actions' evolution. Capturing long-short term information with multiple streams increases the computational redundancy.

Temporal Convolutional Network. Recently, temporal convolutional networks (TCN) are introduced to model dependencies of different ranges within a unified structure by adjusting receptive fields and can process long videos in parallel. Lea *et al.* [35] propose the encoder-decoder style TCN for action segmentation to capture long-range temporal patterns and apply the dilated convolution to enlarge the receptive field. TDRN [37] further introduces the deformable convolution to process the full-resolution residual stream and low-resolution pooled stream. MS-TCN [12, 40] utilizes multi-stage dilated TCNs with hand-designed dilation rate combinations to capture information from various temporal receptive fields. However, the adjustment of receptive fields still relies on human design, which may not be appropriate. Our proposed efficient receptive field combinations searching scheme can automatically discover more efficient structures, improving these TCN based methods.

Complementary Techniques. Instead of capturing long-term and short-term information, some works [11, 65] further improve the action segmentation performance with boundary refinement. Li *et al.* [11] utilize an iterative training procedure with transcript refinement and soft boundary assignment. Wang *et al.* [65] leverage semantic boundary information to refine the prediction results. Other researchers focus on action segmentation under the weakly supervised [11,33,50] or unsupervised [55] settings. These works still rely on the efficient TCN to model the action dependencies, thus complementing the proposed method.

2.2. Network Architecture Search

The genetic algorithm [45] has achieved remarkable performance on a wide range of applications. Many genetic-based methods are recently introduced for the neural networks architecture search of vision tasks [42,44,47,61,66]. An evolutionary coding scheme is proposed in Genetic CNN [66] to encode the network architecture to a binary string. A hierarchical representation is presented by Liu *et al.* [42] to constrain the search space. Real *et al.* [47] regularize the evolution by an age property selection operation. Sun *et al.* [61] introduce a variable-length encoding method for effective architecture designing. However, the genetic algorithm requires the training of each candidate, consuming too much computational cost when faced with a huge search space.

Differentiable architecture search [43] saves the training time by introducing a large network containing sub-networks with different searching options. The importance of searched blocks is determined by gradient backpropagation [53]. This differentiable search idea is further extended [67] to deal with semantic segmentation [41] and other tasks beyond image classification [3]. However, these network architecture search methods are designed for finding a limited number of operations such as convolution, ReLU, batch normalization, short connection, *etc*. Thus, they cannot handle the huge receptive field combinations search space. In this paper, we propose a global search to handle the huge search space with sparse sampling. The expectation guided iterative local search then transfers the sparse search space of receptive fields into the dense one for fine-level searching.

3. Method

The pipeline of our proposed global-to-local search method has two components: (i) a genetic-based global search algorithm that produces coarse but competitive combinations of the receptive fields; (ii) an expectation guided iterative local search scheme that locally refines the global-searched coarse structures.

3.1. Description

Our objective is to efficiently search for optimal receptive field combinations for the given dataset. The receptive field can be represented with multiple forms, such as the dilation rate, kernel size, pooling size, stride, and the stack number of layers. In this work, we mainly follow the MS-TCN [12] to formulate the receptive fields using the combinations of dilation rates in layers and propose to evolve these combinations during the searching process. Note that other receptive field representations can also be applied to the proposed global-to-local search with some minor adjustments.

Suppose a TCN has L convolutional layers and $D = \{d_1, d_2, ..., d_N\}$ is the possible dilation-rates/receptive-fields in each layer. The combination of receptive fields is represented with $C = \{c_1, ..., c_l, ..., c_L\}$, where $l \in [1, L]$ is the index of layers with dilated convolutions, and $c_l \in D$ is the receptive field of each layer. There are $|D|^L$ possible combinations of receptive fields, *i.e.,* the possible receptive field combinations in MS-TCN [12] is 1024^{40} when dilation rates ranging from 1 to 1024. Directly searching for effective combinations in such a large search space is impractical. We thus decompose the searching process into the global and local search to find the combination in a coarse-to-fine manner.

3.2. Global Search

The objective of the global search is to find the coarse receptive field combinations with affordable cost. Therefore, we reduce the search space by sparsely sampling the dilation rates within layers. Multiple sparse discrete sampling strategies such as uniform sampling, gradually sparse sampling, and gradually dense sampling can be applied to sparse the search space. A gradually sparse sampling scheme from small to large dilation rates is appropriate for the action segmentation task. Because small receptive fields benefit the extraction of precise local details while large receptive fields contribute to coarse long-term dependencies of video sequences. Therefore we formulate the receptive field space in global search as:

$$D_g = \{d_i = k^i, i \in [0, 1, \cdots T]\}, \tag{1}$$

where k is the controller of the search space sparsity, and T determines the largest available receptive field. With the same maximum receptive field, $|D_g| \ll |D|$. The search space is greatly reduced. *i.e.,* when set $k = 2$, and set the maximum receptive field to 1024 as in MS-TCN, the search space is reduced from 1024^{40} to 11^{40}.

However, the reduced space of receptive field combinations can still be huge, unaffordable for a brute force search. We propose a genetic algorithm [45] based method to find coarse combinations that are competitive or even better than

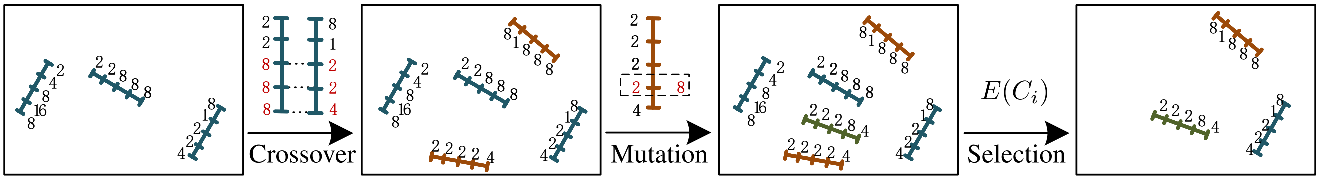


Figure 2. Illustration of one iteration in our genetic-based global search algorithm.

human designing. We illustrate one iteration of our proposed genetic-based global search in Fig. 2. We now detail the selection, crossover, mutation process within our proposed global search method.

Selection. The population of receptive field combinations can be described as a group of candidate structures $P = \{C_i, i \in [1, M]\}$, where C_i is the candidate structure in the global search space, and M is the number of individuals in the population. The selection operation selects individuals to be kept in P based on the estimated performance of each structure C_i, denoted by $E(C_i)$:

$$E(C_i) = f(V|C_i, \theta_n), \tag{2}$$

where $f(\cdot)$ is the evaluation metric detailed in Sec. 4, and V, θ_n are the cross-validation set and model trained with n epochs, respectively.

Crossover. This operation generates new samples of receptive field combinations. Every two combinations in the population are exchanged to born new patterns of the combination while maintaining the local structures. Each C_i will be selected for the crossover operation with probability $p(C_i)$:

$$p(C_i) = \frac{E(C_i)}{\sum_i^M E(C_i)}. \tag{3}$$

Instead of randomly exchanging individual points, we choose to exchange random segments of the receptive field combination since the representation ability lies in the combination patterns. Specifically, we randomly choose two anchors and exchange combinations within anchors to generate new samples.

Mutation. The mutation operation avoids getting stuck in local optimal results by choosing an individual with probability p_m and randomly changing a value within the selected combination.

The global search process can be summarised as Algorithm (1), and a simple example is given in Fig. 2. With the coarse search space and the global search method, we can find receptive field combinations with different patterns than human-designed structures while having similar or even better performance. We further propose the local search to locally find the more efficient combinations on top of the global-searched structures. We show in Tab. 5 that local search heavily relies on the initial structure, revealing the importance of global search.

Algorithm 1 Global Search.

```
Input: Iterations $N$, training epoch $n$, randomly initialized
  $P$, mutation probability $p_m$, and population size $M$;
  for iter in $[1, N]$ do
      Select individuals for crossover based on Eqn. (3) and
  crossover for every two selected individuals;
      Mutate the new individuals with probability $p_m$;
      Training each individual with $n$ epochs;
      Select the top $M$ individuals based on Eqn. (2) as the
  new population $P$;
  end for
  return $P$.
```

3.3. Expectation Guided Iterative Local Search

The local search aims to find more efficient receptive field combinations in a fine-grained level at a low cost. A naive approach is to sample finer-grained dilation rates near the initial dilation rate searched by the global search and apply existing DARTS algorithms [3, 43] to choose for the proper one. However, even with the good initial structure provided by the global search, the available range of fine-grained dilation rates is still large. Existing search algorithms are designed for searching sparse operators with several choices in each layer, thus cannot handle dilation rates with hundreds of choices. While too sparsely sampling is in conflict with our goal of searching for the finer-grained receptive fields. Also, DARTS methods search operators with different functionality [43], while the searching on receptive fields only contains one functional dimension. Different subsets in the dataset sometimes prefer different searching options. Searching within a functional dimension enables us to determine dilation rates with the expectation of all subsets instead of choosing the option required by one majority subset. Therefore, we propose an expectation guided iterative (EGI) local search scheme to determine the finer-level dilation rates on top of the global-searched structures.

Suppose that the receptive field of a layer l is D_l. For a dataset, once we get the probability mass distribution of dilation rates around D_l, we can obtain the expected dilation rate with the weighted average of the dilation rates required by all subsets. However, the probability mass of di-

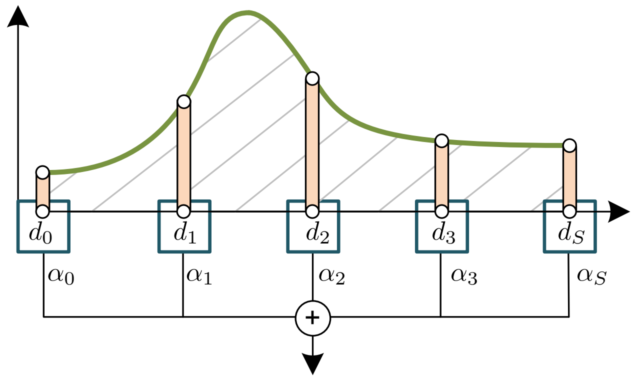
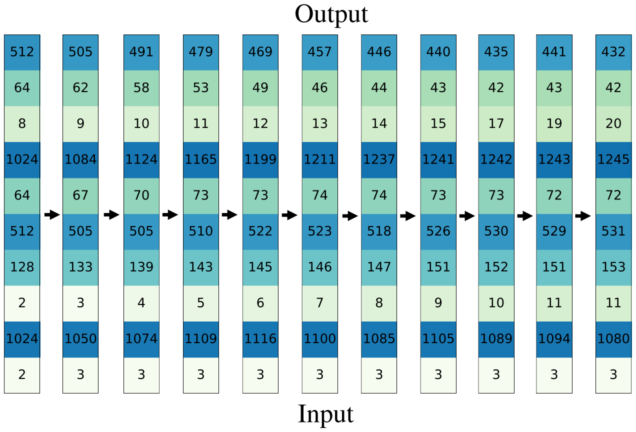


Figure 3. The approximated probability mass function of dilation rates is determined by the multi-dilated convolutional layer with shared weights. d_i is the dilation rate and α_i is the PMF in Eqn. (4).

lation rates for the dataset is inaccessible. Therefore, we utilize a convolutional weight-sharing scheme to enforce the learned importance weights of dilation rates to approximate the probability mass. To get the approximated probability mass function of dilation rates, we first evenly sample S dilation rates near the initial dilation rate D_l within the range of $[D_l \pm \Delta D_l]$. The set of available dilation rates within this layer is $T_l = \{d_i | i \in [1, S]\}$, where $d_i = D_l - \Delta D_l + (i-1) \cdot 2\Delta D_l / (S-1)$. ΔD_l is the finer controller of the search space that is smaller than the sampling sparsity in the global search.

Algorithm 2 Expectation Guided Iterative Local Search.

```
Input: Iterations $N$, initial receptive fields $D$;
  Initialize model using given $D$;
  for iter in $[1, N]$ do
      Construct $T_l$ for each layer based on $D$;
      Train model to get the $PMF$ in Eqn. (4);
      Obtain new dilation rates through Eqn. (6);
      Update $D$;
  end for
  return local-searched $D$.
```


Output
Input

512	505	491	479	469	457	446	440	435	441	432
64	62	58	53	49	46	44	43	42	43	42
8	9	10	11	12	13	14	15	17	19	20
1024	1084	1124	1165	1199	1211	1237	1241	1242	1243	1245
64	67	70	73	73	74	74	73	73	72	72
512	505	505	510	522	523	518	526	530	529	531
128	133	139	143	145	146	147	151	152	151	153
2	3	4	5	6	7	8	9	10	11	11
1024	1050	1074	1109	1116	1100	1085	1105	1089	1094	1080
2	3	3	3	3	3	3	3	3	3	3

Figure 4. Visualization of receptive field combinations changes during the EGI local searching process.

With the dilation rates set T_l, we propose a multi-dilated layer composed of a shared convolutional weight and multiple branches with different dilation rates, as shown in Fig. 3. Each branch has a unique weight to determine the importance of the dilation rate. During the searching process, the weights are updated with the gradient backpropagation to reflect the receptive field requirements of the dataset. Existing DARTS schemes [43, 67] have separated weights in each branch. In contrast, our convolutional weight-sharing strategy forces the model to learn the approximated probability of receptive fields and ease the model convergence. Specifically, the dilation rates in the multi-dilated convolutional layer are set to T_l. Apart from the shared convolutional θ, the multi-dilated layer contains weight $W = \{w_1, w_2, ..., w_i, i \in [1, S]\}$ to determine the importance of the dilation rates. W is unbounded, thus cannot be directly used to determine the dilation rates probability. Therefore, we propose a normalization function to get the approximated probability mass function $PMF(d_i)$ of dilation rates through normalizing w_i:

$$PMF(d_i) = \alpha_i = \frac{|w_i|}{\sum_i^S |w_i|}. \quad (4)$$

With the probability mass function, given the input feature x, the output y of the multi-dilated convolutional layer can be written as follows:

$$y = \sum_i^S \alpha_i \Psi(x, d_i, \theta), \quad (5)$$

where $\Psi(x, d_i, \theta)$ is the convolutional operation with the shared weight θ and dilation rate d_i. α_i is updated with gradient optimization. Once we get the probability mass function, the newly searched dilation rate $D_l^{'}$ is obtained with the expectation:

$$D_l^{'} = \lfloor \sum_{d_i \in T_l} PMF(d_i) \cdot d_i \rfloor. \quad (6)$$

To reduce the computational cost during the local search process, we reduce the number of dilation rates in T_l to 3 by default and apply the iterative search scheme to find the more suitable dilation rate based on the $D_l^{'}$ from the last iteration. The local search process can be summarised as Algorithm (2). Furthermore, Fig. 4 visualizes the dilation rates changes during the local searching process.

4. Experiments

We introduce the implementation details, verify the effectiveness, and analyze the property of our proposed global-to-local search scheme in this section.

	BreakFast					50Salads					GTEA				
	F@{10,25,50}			Edit	Acc	F@{10,25,50}			Edit	Acc	F@{10,25,50}			Edit	Acc
MS-TCN [12]	52.6	48.1	37.9	61.7	66.3	76.3	74.0	64.5	67.9	80.7	87.5	85.4	74.6	81.4	79.2
Reproduce	69.1	63.7	50.1	69.9	67.3	78.8	75.3	64.4	71.4	77.8	87.1	83.6	70.4	81.1	75.5
Global	72.2	66.0	51.5	71.0	69.2	79.3	76.5	68.1	71.9	81.2	89.1	87.1	74.4	84.2	**78.6**
Local	**74.9**	**69.0**	**55.2**	**73.3**	**70.7**	**80.3**	**78.0**	**69.8**	**73.4**	**82.2**	**89.9**	**87.3**	**75.8**	**84.6**	78.5

Table 1. Performance of the global and local searching stages of our global-to-local searching method using MS-TCN [12] as the baseline.

	#Cls	#Vid	#Frame	Scene
GTEA [15]	11	28	1115	daily activities
50Salads [60]	17	50	11552	preparing salads
BreakFast [31]	48	1712	2097	cooking breakfast

Table 2. Details of three action segmentation datasets. #Cls and #Vid are the numbers of classes and videos, respectively. #Frame is the average frames of videos.

4.1. Implementation Details

Structure Searching and Training. Our proposed method is implemented with the PyTorch [46], MindSpore [1], and Jittor [28] frameworks. Following existing works [12,40], features are first extracted from videos using the I3D network [4] and then passed to action segmentation models to get the temporal segmentation. Since our proposed global-to-local search scheme is model-agnostic, the training settings for model evaluation, *i.e.,* training epochs, optimizer, learning rate, batch size, keep the same with the cooperation methods [5,40,65]. In the global search stage, we set the total iterations $N = 100$, $k = 2$ in Eqn. (1), the initialized population size $M = 50$, and mutation probability $p_m = 0.2$. The T in Eqn. (1) is set to 10, indicating the maximum dilation rate of the global search space is 1024. We observe that 5 epochs of training can reflect the structure performance, and therefore models are trained with 5 epochs for evaluation. In the EGI local search stage, ΔD_l and S are set to be $0.1D_l$ and 3, respectively. We train the model for 30 epochs during local search and update the structure every 3 epochs.

Datasets. Following [5, 12, 40, 65], we evaluate our proposed method on three popular action segmentation datasets: Breakfast [31], 50Salads [60], and GTEA [15]. The details of the three datasets are summarised in Tab. 2. As far as we know, the Breakfast dataset is the largest public dataset for action segmentation task, which has a larger number of categories and samples compared with the other two datasets. So we perform our ablations mainly on the Breakfast dataset if not otherwise stated. Following common settings [5, 12, 40, 65], we perform 4-fold cross-validation for the Breakfast and GTEA dataset and 5-fold cross-validation for the 50Salads dataset.

BreakFast	F@0.1	F@0.25	F@0.5	Edit	Acc
ED-TCN [35]	-	-	-	-	43.3
HTK (64) [32]	-	-	-	-	52.0
TCFPN [11]	-	-	-	-	56.3
GRU [50]	-	-	-	-	60.6
GTRM [29]	57.5	54.0	43.3	58.7	65.0
MS-TCN [12]	52.6	48.1	37.9	61.7	66.3
Ours-MS-TCN	74.9	69.0	55.2	73.3	70.7
MS-TCN++ [40]	64.1	58.6	45.9	65.6	67.6
Ours†-MS-TCN++	72.4	66.8	53.5	70.2	69.6
BCN [65]	68.7	65.5	55.0	66.2	70.4
Ours†-BCN	72.5	69.9	60.2	69.0	72.9
SSTDA [5]	75.0	69.1	55.2	73.7	70.2
Ours‡-SSTDA	76.3	69.9	54.6	74.5	70.8

Table 3. Cooperating with SOTA methods. We perform the whole search pipeline based on MS-TCN [12]. Because of the limited computing resources, we only perform the EGI local search on MS-TCN++ [40] and BCN [65], denoted by †. SSTDA [5] uses MS-TCN [12] as a backbone, so we directly add our searched structure to SSTDA, denoted by ‡.

Evaluation Metrics. We follow previous works [5,12,40, 65] to use the frame-wise accuracy (Acc), segmental edit score (Edit) [35], and segmental F1 score [38] at temporal intersection over union with thresholds 0.1, 0.25, 0.5 (F@0.1, F@0.25, F@0.5) as our evaluation metrics.

4.2. Performance Evaluation

Global2Local Search. Our proposed global-to-local search aims to find new combinations of receptive fields better than human designings. We mainly take MS-TCN [12] as our baseline architecture to perform the global-to-local search. When testing the MS-TCN on the Breakfast dataset, we train all models with the batch size 8 to save training time. The reproduced results shown in Tab. 1 indicates that large batch size achieves much better performance. Tab. 1 shows that global-to-local searched structures achieve considerable performance improvements than human-designed baselines, *i.e.,* the searched structure surpasses the reproduced baseline with 5.8% in terms of F@0.1. The global-to-local search focuses on the receptive field combinations, thus can cooperate with existing SOTA action segmentation methods to further improve their performance. As shown in Tab. 3, on the large scale BreakFast dataset, global-to-local search consistently boosts the performance of MS-

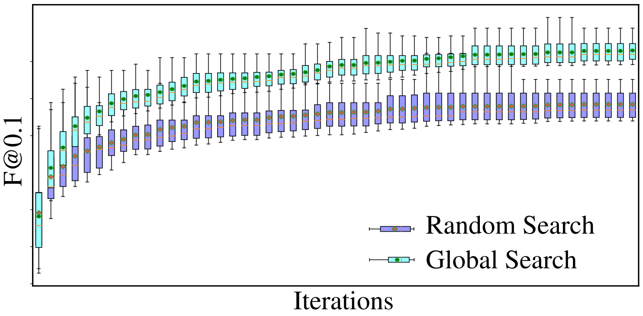


Figure 5. Performance comparison between our proposed genetic-based search and random search during the global search stage.

BreakFast	F@0.1	F@0.25	F@0.5	Edit	Acc
DARTS	73.8	67.6	52.8	72.8	69.3
Ours	**74.9**	**69.0**	**55.2**	**73.3**	**70.7**

Table 4. Performance of our proposed EGI local search and previous DARTS [43].

TCN++ [40], BCN [65], and SSTDA [5]. Also, we give comparisons on two small scale datasets, 50Salads and GTEA dataset in Tab. 10 and supplementary, proving the effectiveness of our proposed global-to-local search.

Global Search. Global search reduces the computational cost with the sparse search space and our proposed genetic-based searching scheme. Fig. 5 shows the performance change of models during the global searching process. Compared with the random search, the genetic-based global search convergences faster. The standard division of model performance searched by genetic-based search is smaller than the random search, showing the stability of our proposed search scheme. The visualized well-performed global-searched structures shown in the supplementary prove that the global search discovers various structures completely different from human-designed patterns. Tab. 5 also shows that the local search heavily relies on global-searched structures to achieve better performance.

Local Search. Based on the global-searched structures, our proposed EGI local search aims to fine-tune the receptive field in a finer search space. Compared with the DARTS [43] method that only supports several search options, the EGI local search iteratively finds the accurate dilations in a dense space, obtaining structures with better performance, as shown in Tab. 4. As shown in Tab. 6, EGI local search is insensitive to the number of sampling dilation rates S, as it searches dilation rates with the expectation. Tab. 5 shows that the EGI local search can boost the performance of randomly generated, human-designed, and global-searched structures. Still, the performance of the local-searched structures is related to the initial structures, as local search focuses on searching for receptive fields within a finer local search space. We visualize the searching process of the iterative local search in Fig. 4. The dilation rates for each layer gradually converge to a suitable state during the iterative searching process. Tab. 7 verifies different ways to get the approximated probability mass function $PMF(d_i)$ from weight w. Eqn. (4) is more superior than the sigmoid function and softmax function as it maintains the probability distribution while the other two functions change the distribution non-linearly.

BreakFast	F@0.1	F@0.25	F@0.5	Edit	Acc
random	67.7	61.8	48.3	68.4	67.0
random + local	73.6	67.8	53.7	72.3	69.9
baseline [12]	69.1	63.7	50.1	71.0	69.2
baseline + local	74.1	68.5	55.3	72.3	70.2
global	72.2	66.0	51.8	71.5	69.4
global + local	74.9	69.0	55.2	73.3	70.7

Table 5. Performance of the EGI local search initialized by different structures.

BreakFast	F@0.1	F@0.25	F@0.5	Edit	Acc
$S = 2$	74.8	68.9	55.0	73.4	70.4
$S = 3$	74.9	69.0	55.2	73.3	70.7
$S = 4$	74.9	68.8	55.1	73.3	70.9

Table 6. Ablation of the value of S in the EGI local search.

BreakFast	F@0.1	F@0.25	F@0.5	Edit	Acc
sigmoid	72.7	66.9	52.7	71.8	69.4
softmax	73.2	67.2	52.0	71.6	69.7
Eqn. (4)	**74.9**	**69.0**	**55.2**	**73.3**	**70.7**

Table 7. Ablation of possible probability mass functions in EGI local search.

4.3. Observations

In this section, we try to exploit the common knowledge contained in the global-to-local searched structures.

Connections between Receptive Fields and Data. We want to know if receptive field combinations vary among data. Therefore, we evaluate the generalization ability of the searched structures on the subsets of the same dataset and different datasets, respectively. Within the BreakFast dataset, we perform the global-to-local search on one fold and then evaluate the searched structures on other folds. Tab. 9 shows that there is almost no obvious performance gap on different folds, indicates that receptive field combinations almost have no difference within a dataset. However, when search and evaluate structures across different datasets, different structures searched on different datasets have a large performance gap as shown in Tab. 8. We can conclude that different data distribution will result in different receptive field combinations. We visualize the structures searched from different datasets in Fig. 6. The structure searched on 50Salads dataset trends to have larger re-

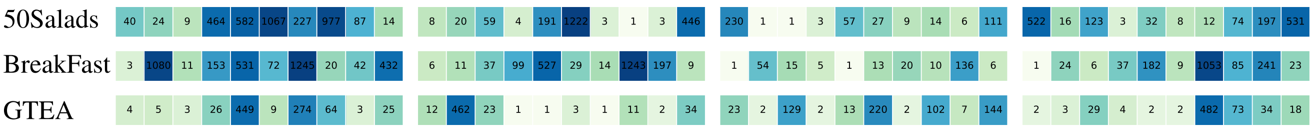


Figure 6. Visualization of the global-to-local searched structures of three datasets with the MS-TCN baseline. Each row represents the dilations of one structure, which contains four stages.

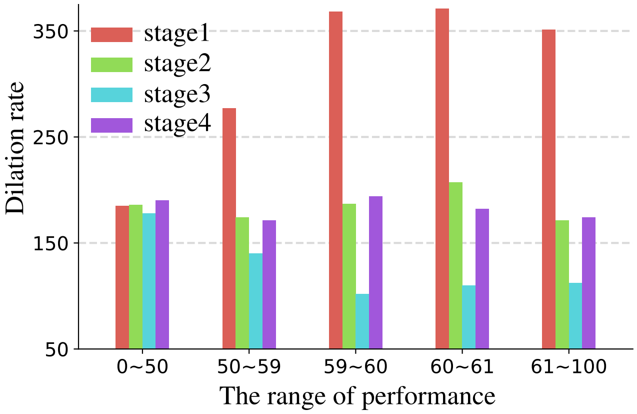


Figure 7. Visualization of average dilation rates in each stage and the range of performance of global-searched structures.

ceptive fields, while the structure searched on the GTEA dataset has smaller receptive fields. The number of video frames shown in Tab. 2 is positively correlated with receptive fields. Longer videos need larger receptive fields to capture the context. We also show more searched structures in the supplementary.

Receptive Fields for Different Stages. Our global-to-local search is based on MS-TCN. MS-TCN contains four stages, and all stages share the same receptive field combination in human designing. The visualized searched structures shown in Fig. 6 demonstrate that different stages have different receptive field combinations, which conflicts with human designing. We further count the average receptive fields of each stage among all individuals. The range of performance and the average dilation rates of each stage are shown in Fig. 7. The average dilation rate in the first stage of MS-TCN tends to be large on high-performance structures. In contrast, the average dilation rate in the third stage of MS-TCN is relatively small on high-performance structures. We assume that the first stage of MS-TCN requires large receptive fields to get the long-term context for coarse prediction, while the following stages need small receptive fields to refine the results locally.

	MS-TCN	**Arch-50Salads**	**Arch-GTEA**	**Arch-BF**
50Salads	67.1	**75.4**	68.8	72.6
GTEA	83.8	82.4	**88.9**	85.6
BF	69.9	75.1	72.5	**76.4**

Table 8. Cross-validation performance (F@0.1) of searched structures among the fold 1 of different datasets. Arch-dataset indicates the structure is searched on which dataset.

BreakFast	Arch-1	Arch-2	Arch-3	Arch-4
fold1	76.4	76.3	76.2	75.7
fold2	74.1	75.3	75.1	74.6
fold3	76.1	76.6	76.1	75.4
fold4	71.7	72.1	72.0	71.8

Table 9. Cross-validation performance (F@0.1) of searched structures among different folds of the BreakFast dataset. Arch-n means the structure is searched on fold n.

50Salads	F@0.1	F@0.25	F@0.5	Edit	Acc
Spatial CNN [36]	32.3	27.1	18.9	24.8	54.9
Bi-LSTM [57]	62.6	58.3	47.0	55.6	55.7
Dilated TCN [35]	52.2	47.6	37.4	43.1	59.3
ST-CNN [36]	55.9	49.6	37.1	45.9	59.4
TUnet [52]	59.3	55.6	44.8	50.6	60.6
ED-TCN [35]	68.0	63.9	52.6	59.8	64.7
TResNet [26]	69.2	65.0	54.4	60.5	66.0
TricorNet [10]	70.1	67.2	56.6	62.8	67.5
TRN [37]	70.2	65.4	56.3	63.7	66.9
TDRN [37]	72.9	68.5	57.2	66.0	68.1
MS-TCN [12]	76.3	74.0	64.5	67.9	80.7
Ours-MS-TCN	**80.3**	**78.0**	**69.8**	**73.4**	**82.2**

Table 10. Comparison with SOTA on the 50Salads dataset.

5. Conclusion

We propose a global-to-local search scheme to search for effective receptive field combinations in a coarse-to-fine scheme. The global search discovers effective receptive field combinations with better performance than hand designings but completely different patterns. The expectation guided iterative local search scheme enables searching fine-grained receptive field combinations in the dense search space. Our proposed global-to-local search can be plugged into multiple tasks, *i.e.,* action segmentation, probabilistic forecasting [6], classification [20, 21, 24], segmentation [22, 23, 62], detection [25, 48] methods to further boost the performance.

Acknowledgement This research was supported by the Major Project for New Generation of AI under Grant No. 2018AAA0100400, NSFC (61922046), and S&T innovation project from Chinese Ministry of Education. We thank MindSpore [1] for the partial support of this work.

References

[1] Mindspore. http://www.mindspore.cn, 2020. 6, 8

[2] Subhabrata Bhattacharya, Mahdi M Kalayeh, Rahul Sukthankar, and Mubarak Shah. Recognition of complex events: Exploiting temporal dynamics between underlying concepts. In *CVPR*, pages 2235–2242, 2014. 2

[3] Han Cai, Ligeng Zhu, and Song Han. ProxylessNAS: Direct neural architecture search on target task and hardware. In *ICLR*, 2019. 1, 2, 3, 4

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. 1, 6

[5] Min-Hung Chen, Baopu Li, Yingze Bao, Ghassan AlRegib, and Zsolt Kira. Action segmentation with joint self-supervised temporal domain adaptation. In *CVPR*, pages 9454–9463, 2020. 1, 6, 7

[6] Yitian Chen, Yanfei Kang, Yixiong Chen, and Zizhuo Wang. Probabilistic forecasting with temporal convolutional neural network. *Neurocomputing*, 399:491–501, 2020. 8

[7] Yu Cheng, Quanfu Fan, Sharath Pankanti, and Alok Choudhary. Temporal sequence modeling for video event detection. In *CVPR*, pages 2227–2234, 2014. 2

[8] Robert T. Collins, Alan J Lipton, and Takeo Kanade. Introduction to the special section on video surveillance. *IEEE TPAMI*, 22(8):745–746, 2000. 1

[9] Robert T Collins, Alan J Lipton, Takeo Kanade, Hironobu Fujiyoshi, David Duggins, Yanghai Tsin, David Tolliver, Nobuyoshi Enomoto, Osamu Hasegawa, Peter Burt, et al. A system for video surveillance and monitoring. *VSAM final report*, 2000:1–68, 2000. 1

[10] Li Ding and Chenliang Xu. Tricornet: A hybrid temporal convolutional and recurrent network for video action segmentation. *arXiv preprint arXiv:1705.07818*, 2017. 2, 8

[11] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. In *CVPR*, pages 6508–6516, 2018. 3, 6

[12] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *CVPR*, pages 3575–3584, 2019. 1, 2, 3, 6, 7, 8

[13] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *ICCV*, pages 407–414. IEEE, 2011. 2

[14] Alireza Fathi and James M Rehg. Modeling actions through state changes. In *CVPR*, pages 2579–2586, 2013. 2

[15] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR*, pages 3281–3288. IEEE, 2011. 2, 6

[16] Mohsen Fayyaz and Jurgen Gall. Sct: Set constrained temporal transformer for set supervised action segmentation. In *CVPR*, pages 501–510, 2020. 1

[17] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. 1

[18] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *CVPR*, pages 4768–4777, 2017. 1

[19] Harshala Gammulle, Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Coupled generative adversarial network for continuous fine-grained action segmentation. In *IEEE WACV*, pages 200–209. IEEE, 2019. 2

[20] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. 8

[21] Shang-Hua Gao, Qi Han, Duo Li, Pai Peng, Ming-Ming Cheng, and Pai Peng. Representative batch normalization with feature calibration. In *CVPR*, 2021. 8

[22] Shang-Hua Gao, Yong-Qiang Tan, Ming-Ming Cheng, Chengze Lu, Yunpeng Chen, and Shuicheng Yan. Highly efficient salient object detection with 100k parameters. In *ECCV*, 2020. 8

[23] Yuchao Gu, Lijuan Wang, Ziqin Wang, Yun Liu, Ming-Ming Cheng, and Shao-Ping Lu. Pyramid constrained self-attention network for fast video salient object detection. In *AAAI*, pages 10869–10876, 2020. 8

[24] Yu-Chao Gu, Li-Juan Wang, Yun Liu, Yi Yang, Yu-Huan Wu, Shao-Ping Lu, and Ming-Ming Cheng. Dots: Decoupling operation and topology in differentiable architecture search. In *CVPR*, 2021. 8

[25] Qi Han, Kai Zhao, Jun Xu, and Ming-Ming Cheng. Deep hough transform for semantic line detection. In *ECCV*, 2020. 8

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 8

[27] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *ICCV*, pages 1314–1324, 2019. 1

[28] Shi-Min Hu, Dun Liang, Guo-Ye Yang, Guo-Wei Yang, and Wen-Yang Zhou. Jittor: a novel deep learning framework with meta-operators and unified graph execution. *Science China Information Sciences*, 63(12):1–21, 2020. 6

[29] Yifei Huang, Yusuke Sugano, and Yoichi Sato. Improving action segmentation via graph-based temporal reasoning. In *CVPR*, pages 14024–14034, 2020. 1, 6

[30] Svebor Karaman, Lorenzo Seidenari, and Alberto Del Bimbo. Fast saliency based pooling of fisher encoded dense trajectories. In *ECCV*, volume 1, page 5, 2014. 2

[31] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*, pages 780–787, 2014. 6

[32] Hilde Kuehne, Juergen Gall, and Thomas Serre. An end-to-end generative framework for video segmentation and recognition. In *IEEE WACV*, pages 1–8. IEEE, 2016. 2, 6

[33] Hilde Kuehne, Alexander Richard, and Juergen Gall. Weakly supervised learning of actions from transcripts. *Comput. Vis. and Image Understanding*, 163:78–89, 2017. 2, 3

[34] Hilde Kuehne, Alexander Richard, and Juergen Gall. A hybrid rnn-hmm approach for weakly supervised temporal action segmentation. *IEEE TPAMI*, 2018. 2

[35] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *CVPR*, pages 156–165, 2017. 1, 2, 6, 8

[36] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV*, pages 36–52. Springer, 2016. 8

[37] Peng Lei and Sinisa Todorovic. Temporal deformable residual networks for action segmentation in videos. In *CVPR*, pages 6742–6751, 2018. 2, 8

[38] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *CVPR*, pages 5325–5334, 2015. 6

[39] Jun Li and Sinisa Todorovic. Set-constrained viterbi for set-supervised action segmentation. In *CVPR*, pages 10820–10829, 2020. 1

[40] Shi-Jie Li, Yazan AbuFarha, Yun Liu, Ming-Ming Cheng, and Juergen Gall. Ms-tcn++: Multi-stage temporal convolutional network for action segmentation. *IEEE TPAMI*, pages 1–1, 2020. 1, 2, 6, 7

[41] 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 *CVPR*, pages 82–92, 2019. 1, 2, 3

[42] Hanxiao Liu, Karen Simonyan, Oriol Vinyals, Chrisantha Fernando, and Koray Kavukcuoglu. Hierarchical representations for efficient architecture search. In *ICLR*, 2018. 1, 3

[43] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. In *ICLR*, 2019. 1, 2, 3, 4, 5, 7

[44] Zhichao Lu, Ian Whalen, Vishnu Boddeti, Yashesh Dhebar, Kalyanmoy Deb, Erik Goodman, and Wolfgang Banzhaf. Nsga-net: neural architecture search using multi-objective genetic algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 419–427, 2019. 3

[45] Melanie Mitchell. *An introduction to genetic algorithms*. MIT press, 1998. 3

[46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8026–8037, 2019. 6

[47] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *AAAI*, volume 33, pages 4780–4789, 2019. 1, 3

[48] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015. 8

[49] Alexander Richard and Juergen Gall. Temporal action detection using a statistical language model. In *CVPR*, pages 3131–3140, 2016. 2

[50] Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *CVPR*, pages 754–763, 2017. 2, 3, 6

[51] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, pages 1194–1201. IEEE, 2012. 2

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Int. Conf. Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 8

[53] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986. 3

[54] Venkatesh Saligrama and Zhu Chen. Video anomaly detection based on local statistical aggregates. In *CVPR*, pages 2112–2119. IEEE, 2012. 1

[55] Fadime Sener and Angela Yao. Unsupervised learning and segmentation of complex activities from video. In *CVPR*, pages 8368–8376, 2018. 3

[56] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, pages 568–576, 2014. 1

[57] Bharat Singh, Tim K Marks, Michael Jones, Oncel Tuzel, and Ming Shao. A multi-stream bi-directional recurrent neural network for fine-grained action detection. In *CVPR*, pages 1961–1970, 2016. 2, 8

[58] Suriya Singh, Chetan Arora, and CV Jawahar. First person action recognition using deep learned descriptors. In *CVPR*, pages 2620–2628, 2016. 2

[59] Mohammad Soleymani, Jeroen Lichtenauer, Thierry Pun, and Maja Pantic. A multimodal database for affect recognition and implicit tagging. *IEEE Trans. Affect. Comput.*, 3(1):42–55, 2011. 1

[60] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *The ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 729–738, 2013. 6

[61] Yanan Sun, Bing Xue, Mengjie Zhang, Gary G Yen, and Jiancheng Lv. Automatically designing cnn architectures using the genetic algorithm for image classification. *IEEE Trans. Cybernetics*, 2020. 3

[62] Yong-Qiang Tan, Shang-Hua Gao, Xuan-Yi Li, Ming-Ming Cheng, and Bo Ren. Vecroad: Point-based iterative graph exploration for road graphs extraction. In *CVPR*, 2020. 8

[63] Kevin Tang, Li Fei-Fei, and Daphne Koller. Learning latent temporal structure for complex event detection. In *CVPR*, pages 1250–1257. IEEE, 2012. 2

[64] Nam N Vo and Aaron F Bobick. From stochastic grammar to bayes network: Probabilistic parsing of complex activity. In *CVPR*, pages 2641–2648, 2014. 2

[65] Zhenzhi Wang, Ziteng Gao, Limin Wang, Zhifeng Li, and Gangshan Wu. Boundary-aware cascade networks for temporal action segmentation. In *ECCV*, 2020. 1, 3, 6, 7

[66] Lingxi Xie and Alan Yuille. Genetic cnn. In *ICCV*, pages 1379–1388, 2017. 1, 3

[67] Arber Zela, Thomas Elsken, Tonmoy Saikia, Yassine Marrakchi, Thomas Brox, and Frank Hutter. Understanding and robustifying differentiable architecture search. In *ICLR*, 2020. 3, 5