Journal of Korea Multimedia Society Vol. 23, No. 12, December 2020(pp. 1540-1551)
https://doi.org/10.9717/kmms.2020.23.12.1540

Human Activity Recognition Based on 3D Residual Dense Network

Jin-Ho Park[†], Eung-Joo Lee[††]

ABSTRACT

Aiming at the problem that the existing human behavior recognition algorithm cannot fully utilize the multi-level spatio-temporal information of the network, a human behavior recognition algorithm based on a dense three-dimensional residual network is proposed. First, the proposed algorithm uses a dense block of three-dimensional residuals as the basic module of the network. The module extracts the hierarchical features of human behavior through densely connected convolutional layers; Secondly, the local feature aggregation adaptive method is used to learn the local dense features of human behavior; Then, the residual connection module is applied to promote the flow of feature information and reduced the difficulty of training; Finally, the multi-layer local feature extraction of the network is realized by cascading multiple three-dimensional residual dense blocks, and use the global feature aggregation adaptive method to learn the features of all network layers to realize human behavior recognition. A large number of experimental results on benchmark datasets KTH show that the recognition rate (top-l accuracy) of the proposed algorithm reaches 93.52%. Compared with the three-dimensional convolutional neural network (C3D) algorithm, it has improved by 3.93 percentage points. The proposed algorithm framework has good robustness and transfer learning ability, and can effectively handle a variety of video behavior recognition tasks.

Key words: Human Activity Recognition, Video Classification, 3D Residual Dense Network, Deep Learning, Feature Fusion

1. INTRODUCTION

Video understanding is a very challenging task in the field of computer vision. Human behavior recognition in video is an important branch. With the deepening of research, great progress has been made. According to the different ways of extracting features from video sequences, literature [1-3] proposed that behavior recognition methods can be divided into two categories: methods based on construction features and methods based on automatic learning features. Among the many deep learning network structures, Convolutional Neural Network (CNN) is the most widely used. CNN has achieved great success in the field of static images [4-8]. It also has great advantages in researching video processing. In order to encode both spatial and temporal information in the deep convolution model, Hu et al. [9] aimed at the problem of complex feature extraction and low accuracy in human action recognition, this paper proposed a network structure combining batch normalization algorithm with GoogLeNet network model. Ji et al. [10] extended the 2D convolutional network simply and effectively, and proposed a 3D convolutional neural network model for learning dynamic continuous video sequences and deep learning of spatio-temporal features. Tran et al. [11] found the optimal

※ Corresponding Author : Eung-Joo Lee, Address: (48520) 428, Sinseon-ro, Nam-gu, Busan, Korea, TEL : +82-51-629-1143, FAX : +82-51-629-1143, E-mail : ejlee@tu.ac.kr
Receipt date : Oct. 7, 2020, Revision date : Nov. 9, 2020
Approval date : Nov. 10, 2020

[†] Dept. of Information and Communication Engineering, Tongmyong University
(E-mail : ejlee@tu.ac.kr)
[††] Dept. of Information and Communication Engineering, Tongmyong University

convolution kernel size of 3D convolutional neural network through systematic research, and proposed a three-dimensional convolutional neural network (C3D) suitable for large-scale datasets, and using C3D to extract video spatio-temporal features, the extracted features are very versatile and computationally efficient. In addition, Tran et al. [12] carried out three-dimensional convolutional neural network improvement in the framework of deep residual network and proposed Res3D network. The improved network is superior to C3D in terms of operation speed and recognition accuracy. Hara et al. [13] proposed that the Kinetics dataset has sufficient data to train a deep three-dimensional convolutional network, and the simple three-dimensional architecture pre-trained by the Kinetics dataset is superior to the complex two-dimensional architecture. Tran et al. [14] decomposed the two-dimensional convolution operations in the three-dimensional convolutional network into two independent continuous operations: three-dimensional spatial convolution and one-dimensional time convolution, and proposed the R(2+1)D network. Compared with C3D, the network effectively increases the capacity of the model. And at the same time is conducive to the optimization of the network. The three-dimensional convolutional network has received extensive attention and application due to its simple and effective strategy, but the network also has some defects. For example: due to its huge network parameters, it is very difficult to converge the network. Chen et al. proposed a lightweight multi-fiber network architecture, which significantly reduces the computational complexity of the 3D network and improves the recognition performance of the model. Yang et al. [15] proposed an asymmetric three-dimensional convolutional neural network model. In addition to reducing the amount of parameters and computational cost, multi-source enhanced input and multi-scale 3D convolution branches were introduced to process the convolution features of different scales in the video. Effective information fusion of RGB and optical flow frames significantly improves the expressive ability of the model. In addition, the 3D convolutional network has a certain gap with the most advanced baseline method in terms of space-time feature modeling. Diba et al. [16] introduced a Spatio-temporal Channel correlation block (STC) as their new residual module in the infrastructure such as ResNet, which can effectively capture spatio-temporal channel correlation information in the entire network layer. Hussein et al. [17] focused on time cues in behavior recognition. In order to model complex actions in the long-term range, a 3D convolutional network with improved Timeception convolutional layer was proposed, which used multi-scale time convolution, focus on short-term details to learn long-term dependencies.

Among the many behavior recognition methods based on deep learning, this research focuses on the behavior recognition architecture based on 3D convolutional neural network. 3D convolution can be used to extract universal and reliable spatio-temporal features directly from the original video, and the performance is immediately effective; However, the traditional 3D convolutional neural network algorithm lacks the full use of the network's multi-level convolution features, which affects the generalization performance of the network. Combining the idea of residual network and dense network, this research proposes a 3D-Residual Dense Network (3D-RDNet), the network can make full use of the hierarchical features of all convolutional layers, and uses 3D-Residual Dense Block (3D-RDB) as a building block. The features of each convolutional layer in the 3D-RDB can be directly transferred to all subsequent layers, and then local dense feature aggregation is used to adaptively retain beneficial information, and then local residual learning is performed on the input and output feature aggregation. The output of the 3D-RDB module after sampling will directly access all the layers in the next 3D-RDB module, forming a state of continuous transmission and multiplexing of features. At the same time, each 3D-RDB module's feature output after convolution sampling will be spliced together, and a variety of hierarchical features will be adaptively retained in a global manner to complete global feature aggregation. In order to verify the effectiveness of the proposed algorithm and apply it to real scenes, this research has been trained and tested on the KTH datasets, and the recognition rates of the proposed algorithms have reached 93.52%, respectively. Compared with traditional algorithms, C3D, 3D-ResNet and 3D-DenseNet, higher recognition accuracy is achieved. Experimental results show that the three-dimensional dense network proposed in this research can effectively identify human behavior in video.

2. 3D RESIDUAL DENSE NETWORK

2.1 3D Convolutional Neural Network

The three-dimensional convolution in the three-dimensional convolutional neural network is essentially a three-dimensional convolution kernel operation on a cube formed by stacking multiple video frames, since each feature map in the convolutional layer is connected to multiple adjacent consecutive frames in the previous layer, motion information can be captured [9]. A three-dimensional convolution operation can be described as $C(n,d,f)$, which means the input of a convolutional layer of size $n \times n \times n$ and d feature maps of size $f \times f \times f$, the output at the position of (x,y,z) on the m-th feature map of the 3D convolution l-layer Formally expressed as:

$$v_{lm}^{xyz} = b_{lm} + \sum_{q} \sum_{i=0}^{f-1} \sum_{j=0}^{f-1} \sum_{k=0}^{f-1} w_{lmq}^{ijk} v_{(l-1)q}^{(x+i)(y+j)} \qquad (1)$$

Among them: b_{lm} is the offset of the feature map; q is the feature map of the $(l-1)$-layer; $w^{ijk_{lmq}}$ is the weight at the kernel position (i,j,k) of the q-th feature map, and the weight and deviation will be obtained through training. C3D based on three-dimensional convolution is widely used in the field of video behavior recognition. The features extracted by C3D also have strong recognition in other tasks. Such as behavior recognition, time series behavior detection, gesture recognition. Compared with C3D, the improved 3D convolutional neural network based on ResNet and DenseNet architecture, such as 3D-ResNet and 3D-DenseNet can significantly improve the effect of video behavior recognition tasks. The following is a network constructed based on three-dimensional convolution of the experiment in this research. They are C3D in literature [11], improved 3D-ResNet of C3D in literature [12], and 3D-DenseNet constructed in this research. The network structure parameters are showed in Table 1, with a step size of $2 \times 2 \times 2$.

In order to simplify the description, the number of channels of the output size feature and the number of filters of each convolution layer is omitted from the table. The input and output of the network and the size of the convolution kernel is a three-dimensional tensor of $L \times H \times W$, among them L, H, W represent the length of time, height and width. In order to reduce data redundancy, even-numbered frames are skipped on the network input, and the input size of all networks is, where the maximum GPU memory limit is adapted and the appropriate batch size is retained, in addition, all three networks use the same data enhancement and data preprocessing methods. The C3D used in this research has five convolutional layers and five downsampling layers. The layers are connected in series, and the size of the convolution kernel is $3 \times 3 \times 3$, finally, through two fully connected layers and softmax layer, 101 class probabilities are output. The 3D-ResNet network is extended by 2D-ResNet, the convolutional layer is expanded from $d \times d$ to $3 \times d \times d$, and the step size of the downsampling layer behind the convolutional layer other

Table 1. Network Structural Parameters of C3D, 3D-ResNet and 3D-DenseNet.

Net Layer	Output Size Feature	Net Structural Parameters		
		C3D	3D-ResNet	3D-DenseNet
Conv1	8×112×112	$3\times3\times3, stride, 1\times2\times2$	$3\times7\times7, stride, 1\times2\times2$	$3\times7\times7, stride, 1\times2\times2$
Conv2_x	8×56×56	$3\times3\times3$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times2$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times4$
Conv3_x	4×28×28	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times1$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times2$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times4$
Conv4_x	2×14×14	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times1$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times2$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times4$
Conv5_x	1×7×7	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times1$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times2$	$\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}\times4$
----	1×1×1	global average pooling, 101-d fc, softmax		

than Conv1 is changed to 2×2×2 network Conv1 layer convolution kernel is 3×7×7, The convolution kernels of other layers are all 3×3×3. The Conv1, Conv2_x, Conv3_x, Conv4_x, and Conv5_x layers of the network all use residual connection to make the network easier to optimize. The 3D-DenseNet is constructed in a similar way to the 3D-ResNet. The difference is that the hierarchical connection method of each convolutional layer uses a dense connection. The network is composed of multiple dense blocks. Each layer in the same dense block reads information from all previous layers, finally concatenate, using the bottleneck layer in the same dense block, among them, 1×1×1 convolution operations are used to reduce the number of input feature maps, at the same time reduce the amount of calculation, and fuse the features of each channel. Dense blocks are connected by a transition layer, and the network finally concatenates the features obtained by multiple dense blocks. This connection method is conducive to the reuse of features, and it also strengthens the transfer of features and reduces the amount of calculation.

2.2 3D Residual Dense Network

The three-dimensional residual dense network (3D-RDNet) for video behavior recognition proposed in this research builds on the residual learning of ResNet and the dense connection mode of DenseNet network to construct a dense three-dimensional residual block, multi-level spatio-temporal features are extracted, and then feature aggregation is performed to combine the low-level features and high-level semantic features to improve the expressive ability of the model. The network structure is shown in Fig. 1.

As showed in Fig. 1, the dense three-dimensional residual network is divided into three parts, they are: shallow feature extraction layer, dense residual layer, and global feature aggregation layer. The shallow feature extraction layer (Part A) includes the two layers of 3D Conv shown; Residual dense layer (Part B) includes pooling layer (Maxpool), multiple residual dense blocks (3D-RDB), and the convolutional layers 3D Conv1 and 3D Conv2 used for convolution down sampling; The global feature aggregation layer (Part C) contains a concatenation layer for feature stitching and a convolutional layer for feature aggregation.

The input and output of the dense three-dimensional residual network are defined as P_{clip} and P_{cls}, respectively. The first two convolutional layers of the network are used to extract shallow features. Specifically, the process of extracting features from shallow layers to features can be described as:

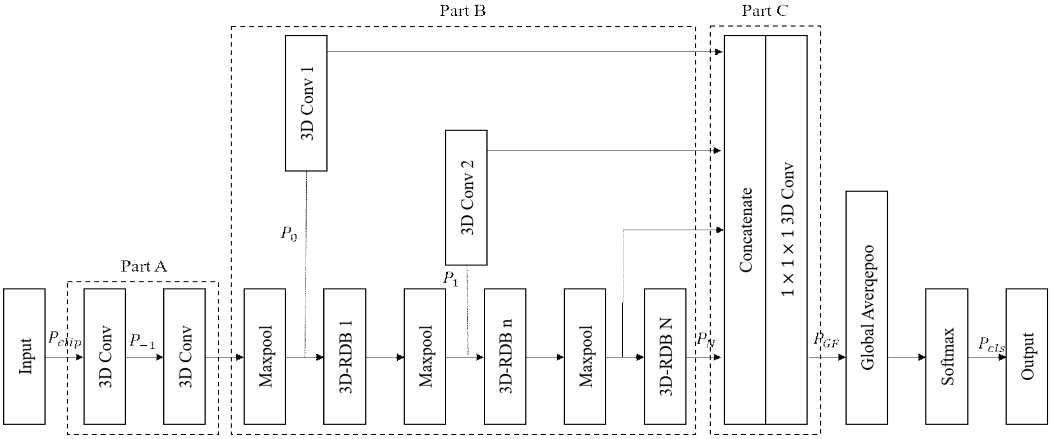


Fig. 1. Three-dimensional Residual Dense Network.

$$P_0 = G_{sh}(P_{clip}) \tag{2}$$

Among them: G_{sh} represents the composite function of the first two layers of convolution and downsampling operations; P_0 is the feature map extracted from the video clip, which is used for the input of the dense block of the first layer. Here, N residual dense blocks are set, the output of the nth residual dense block is P_n, and the calculation process is:

$$P_n = G_{3D-RDB,n}(G_{3D-RDB,n-1}(\cdots(G_{3D-RDB,1}(P_0))\cdots)) \tag{3}$$

In the formula: $G_{3D-RDB,n}$, n represent the calculation operation of the nth dense residual block (3D-RDB) and its downsampling (Maxpool), when n=N, $G_{3D-RDB,N}$ contains only the calculation operation of the dense residual block. $G_{3D-RDB,n}$ is a compound operation function, including multi-layer convolution and rectified linear units. Since P_n is generated by the operation of multiple convolutional layers in the nth residual dense block, P_n can be regarded as a local dense feature.

After 3D-RDNet extracts multi-level local dense features through multiple 3D-RDBs, it further performs Global Feature Aggregation (GFA). GFA makes full use of the features of all previous layers. Specifically, the input features of different levels are convoluted and sampled into feature maps, and the norm is normalized, then use the concatenation layer (Concatenate) to stitch together the local dense features from different levels, then use the convolution of 1×1×1 to perform feature aggregation and channel adjustment to obtain a feature map of global feature aggregation. The process of stitching local dense features can be described as:

$$P_{GFA} = G_{GFA}([X_0, X_1, ..., X_N]) \tag{4}$$

Among them: PGFA is a feature map output by global feature aggregation; G_{GFA} is a composite function of 1×1×1 convolution. It is used to adaptively fuse features from different layers; $[X_0, X_1, ..., X_N]$ refers to the stitching of N feature maps after three-dimensional dense block and convolution sampling.

Combining the above operations, the network extracts shallow features from the input clip, then through multiple dense blocks of residuals to obtain rich local features, and then through global feature aggregation to obtain global features, finally, various categories of scores are obtained through the softmax classifier. The entire network 3D-RDNet calculation process can be expressed as:

$$P_{cls} = G_{RDNet}(P_{clip}) \tag{5}$$

Among them: G_{RDNet} is the operation of the entire network of 3D-RDNet; P_{cls} is the output of the network.

2.3 3D Residual Dense Block

The three-dimensional residual dense network consists of multiple three-dimensional residual dense blocks. Fig. 2 shows the network structure of the three-dimensional residual dense block (3D-RDB). 3 D-RDB mainly includes dense connection layer, Local Feature Aggregation (LFA) and Local Residual Learning (LRL), which enables the network to fully learn multi-layer convolutional features.

2.3.1 Dense Connection Mode

The 3D-RDB module is composed of multiple convolutional layers, a rectified linear unit (ReLU) and a batch normalization layer. The features learned in the previous 3D-RDB are passed directly to each layer in the current 3D-RDB, at the same time, there is a direct connection between each layer inside the module, this dense connection makes the transfer of features and gradients more effective, promotes feature reuse, retains the characteristics of forward propagation, and extracts locally dense features. Here, P_{n-1} and P_n are defined as the input of the n-th and (n+1)-th 3D-RDB, then the output of the α-th Conv layer of the n-th 3D-RDB can be expressed as:

$$P_{n,\alpha} = \sigma(W_{n,\alpha}[P_{n-1}, P_{n,1}, P_{n,2}, \cdots, P_{n,\alpha-1}]) \qquad (6)$$

Among them: σ indicates that the kernel is the activation function of ReLU; $W_{n,\alpha}$ is the weight of the α-th convolutional layer, and the offset term is omitted here for simplicity. Suppose $P_{n,\alpha}$ is composed of multiple feature maps, and $[P_{n-1}, P_{n,1}, P_{n,2}, \cdots, P_{n,\alpha-1}]$ refers to the stitching of the feature maps output by the (n-1)-th 3D-RDB and the convolutional layer 1, 2, ..., (α- 1) in the n-th 3D-RDB.

2.3.2 Local Feature Fusion

After learning the multi-level spatio-temporal features through the dense connection mode, 3D-RDB fuse the local dense features, specifically, by extracting a series of convolutional layer features from the previous 3D-RDB and the current 3D-RDB, then stitch them together, 1×1×1 convolution layer is introduced to adaptively fuse a series of features with different levels, and this operation is named Local Feature Aggregation (LFA). The calculation process can be described as following:

$$P_{d,LF} = G^{n}_{LFA}([P_{n-1}, P_{n,1}, P_{n,2}, \cdots, P_{n,\alpha}, \cdots, P_{n,A}]) \qquad (7)$$

Where: G^{n}_{LFA} represents the compound operation of the 1×1×1 convolution layer in the nth 3D-RDB, it can reduce the number of feature maps, reduce the amount of calculation, simultaneously

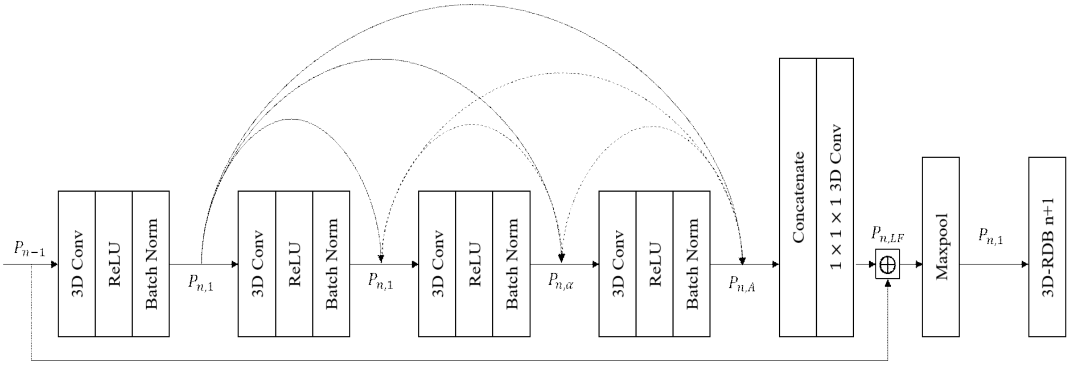


Fig. 2. Three-dimensional Residual Dense Block.

Table 2. Network Structural Parameters of 3D-RDNet

Net Layer	Conv1	Conv2_x	Conv3_x	Conv4_x	Conv5_x	---
Output Size Feature	8×112×112	8×56×56	4×28×28	2×14×14	1×7×7	1×1×1
Net Structure Parameters	3×3×3, *stride*, 1×2×2	3×3×3	$\begin{bmatrix}1\times1\times1\\3\times3\times3\end{bmatrix}\times4$	$\begin{bmatrix}1\times1\times1\\3\times3\times3\end{bmatrix}\times4$	$\begin{bmatrix}1\times1\times1\\3\times3\times3\end{bmatrix}\times4$	global average pooling, 101-d fc, softmax

merge the various channels. As the growth rate of dense networks becomes larger, LFA will contribute to very dense network training.

2.3.3 Local Residual Learning

In a sufficiently deep network structure, in order to ensure the maximum information flow between the various layers in the network, the 3D-RDB uses the jump connection method of the residual network, it connects feature maps with the same feature map size, so that the output of each layer is directly connected to the input of subsequent layers. This jump in connection from the previous layer to the subsequent layer alleviates the problem of network gradient disappearance, enhances feature propagation, promotes feature reuse, and retains the characteristics of forward propagation. The output of the n-th 3D-RDB can be expressed as:

$$P_n = P_{n-1} + P_{n,LF} \tag{8}$$

Use of Local Residual Leaming (LRL) can improve the expression ability of the network, and the network effect is better. Due to the dense connection mode and local residual learning, this modular system is called a three-dimensional residual dense block (3D-RDB).

2.3.4 Algorithm Implementation Details

In the 3D-RDNet network proposed in this research, except for the size of the convolution kernel in the local and global feature aggregation is 1×1×1, the other layers of convolution kernels are set to 3×3×3. The first layer of the shallow layer of the network is equipped with 96 jet filters, the filter of the global feature aggregation convolution layer is set to 512, and the rest of the network is equipped with 128 filters. In addition, the number of 3D residual dense blocks in the 3D-RDNet network in this research is set to 3, and the number of dense layers inside the 3D residual dense block is set to 4. In addition to 3D Conv1 and 3D Conv2 convolutional layers such as convolutional matrix sampling in the dense residual layer on the 3D-RDNet network, the remaining structural parameters are showed in Table 2 and the step size is 2×2×2.

Through analysis of Table 1 and Table 2, four network model parameters (Parameters, Params) and calculations (FLOPs) can be obtained, as showed in Table 3.

Table 3. Parameters and Calculations of Different Models

Network	Params/10^6	FLOPs/10^9
C3D	11.7	6.4
3D-ResNet	33.2	19.3
3D-DenseNet	17.6	8.2
Proposed	14.9	7.4

3. EXPERIMENT AND RESULT ANALYSIS

In this research, the experimental tools, original data sets, experimental conditions and other related experimental information are shown in Table 4. Refer to the following chapters for detailed instructions.

3.1 Dataset

The experimental data in this research is KTH

Table 4. Experimental Environment

Experimental Tools	All training and testing was conducted out on the NVIDIA GeForce GTX 1080 graphics card which has 8GB of memory. The OS used was Windows 10, the Python version was 3.6, and the TensorFlow version was 1.9.
Data Sets	KTH Data set and real scenes data set
Experimental Setup	The input of the model is a clip composed of a continuous 16-frame video sequence. The sampling rate of the clip is set to 2, and the input image is de-averaged to speed up the convergence of the model.
Experimental Method for KTH	After data enhancement and preprocessing, the input size of the video frame of the KTH dataset is 8×112×112, the input batch size is set to 16, the Adam optimizer is used, the parameters beta_1=0.9, beta_2=0.999, the initial learning rate is 10^{-4}, the loss function uses multiple types of cross-picking functions, and the training duration is 25 cycles.
Experimental Method for real scenes	The input of the network is a continuous 16-frame video clip extracted from each video. The video frame width and height are resized to 171×128. After data preprocessing, the input size is cropped to 8×112×112. In terms of network optimization, the stochastic gradient descent method with momentum is adopted. The initial learning rate of the network is set to 0.01, the momentum parameter is 0.9, the learning rate decay rate is 10^{-4}, the objective function is the cross annoying loss function, and the network training period is 25. The batch size is set to 16.

dataset, and the dataset collected and produced in this research. Among them, KTH is the most commonly used dataset in the field of computer vision behavior recognition. The KTH dataset is completed by 25 people performing 6 types of actions in 4 different scenarios, for a total of 600 video samples. Among them: behavior categories include boxing, tempo, waving, jogging, running and walking; Four scenes include different lighting conditions, clothing changes and background scale changes. However, the background is relatively simple, the behavior categories are few, and the camera shooting angle is also fixed. The experiment in this research uses behavior videos of 16 people as training and the remaining 9 people as test. The six types of actions in the KTH dataset are showed in Fig. 3.

In this research, a dataset of real scenes is established for the needs of patrol robot security tasks. The video data are taken from the vicinity of the building entrance control, and the four types of actions such as swiping, wandering, walking, and standing are completed by the moving people entering and leaving the door. Each type of action in the dataset includes 100 video segments, a total of 400 video samples. The video shooting angle is relatively fixed, and the lighting conditions of the video data include the daytime and night lighting conditions. In this research, 2/3 of the behavior data is used as the training set, and the remaining 113 behavior data are used as the test set. Examples of four types of actions in real scenes are showed in Fig. 4.

3.2 Experimental Setup

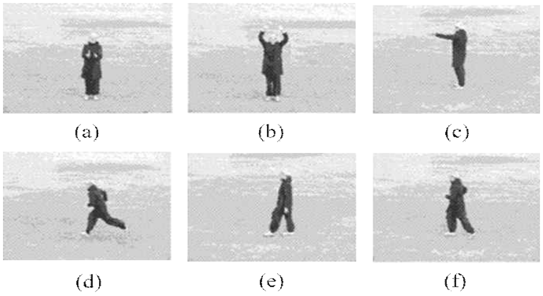


Fig. 3. Six Types of Behavior Examples in KTH dataset. (a) Hand Clapping; (b) Waving; (c) Boxing; (d) Running; (e) Walking; (f) Jogging.

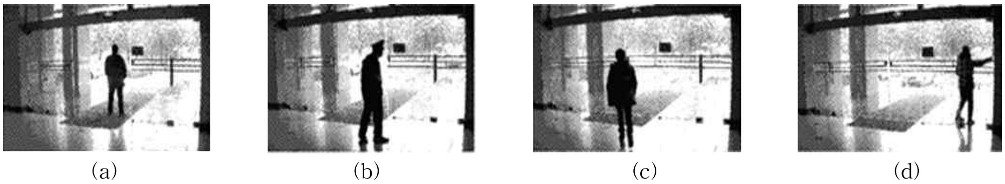


Fig. 4. Four Types of Behavior Examples in Real Scene. (a) Standing; (b) Hover; (c) Walking; (d) Swiping Card.

In this research, the input of the experimental model is a clip composed of 16 consecutive video sequences. The sampling rate of the clip is set to 2, and the input image is de-averaged to accelerate the model convergence speed. During training, skip even-numbered frames on the network input, and use random cropping, random flipping, and random rotation or fixed angle of the picture to increase the diversity of training samples. During the test, input video clips are preprocessed in the same way as the training phase, and then the trained model is used to estimate the behavior classification of each video clip sequence. If it is necessary to obtain the classification result of the entire video level, select multiple clips of the current video to obtain the classification result, and then average to obtain the final behavior classification in the video.

3.3 Experimental Results of KTH dataset

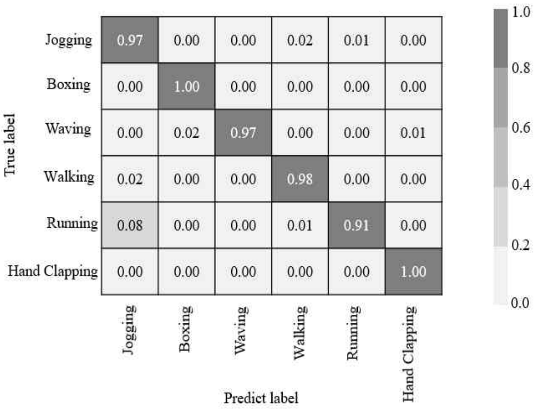
In order to verify the effectiveness of the three-dimensional dense network proposed in this research, experiments were performed on the KTH dataset and real scene data set. First, an experiment was conducted on a small dataset KTH, and the correctness of video behavior recognition of four network models was tested. During training, after data enhancement and preprocessing, the input size of the video frame of the KTH dataset is $8\times112\times112$, the input batch size is set to 16, the Adam optimizer is used, the parameters beta_1=0.9, beta_2=0.999, the initial learning rate is 10^{-4}, the loss function uses multiple types of cross-picking functions, and the training duration is 25 cycles. The performance results of human behavior recognition algorithms such as C3D, 3D-ResNet, 3D-DenseNet, 3D-RDNet network, and literature [18] model, literature [19] model, literature [20] model are showed in Table 4. Among them, the behavior recognition accuracy rate is calculated based on the video-level classification results, that is, video top-1. Schuldt et al. [18] proposed the KTH dataset and introduced local spatio-temporal features. The use of Support Vector Machine (SVM) method for classification has achieved good results. Dollar et al. [19] used sparse spatio-temporal feature points to recognize human behavior. Taylor et al. [20] proposed the use of convolutional networks to learn the spatio-temporal features of human behavior, and tested the best results on public dataset.

It can be seen from Table 5: The accuracy on the KTH dataset, the 3D convolutional network has a greater advantage than other algorithms; At the same time, the improved networks of 3D convolution, 3D-ResNet and 3D-DenseNet, have achieved better results than C3D, and the accuracy rates are 1.61 percentage points and 2.08 percentage points higher than C3D, respectively; Moreover, the three-dimensional dense network proposed in this research is 3.93 percentage points higher than C3D.

The model trained on the KTH training set is tested on the entire dataset to obtain the confusion

Table 5. Accuracy Comparison of Different Models on KTH dataset (%)

Net Model	Model[18]	Model[19]	Model[20]	C3D	3D-ResNet	3D-Densenet	Proposed
Rate	71.70	81.20	90.00	89.59	91.20	91.67	93.52



True label \ Predict label	Jogging	Boxing	Waving	Walking	Running	Hand Clapping
Jogging	0.97	0.00	0.00	0.02	0.01	0.00
Boxing	0.00	1.00	0.00	0.00	0.00	0.00
Waving	0.00	0.02	0.97	0.00	0.00	0.01
Walking	0.02	0.00	0.00	0.98	0.00	0.00
Running	0.08	0.00	0.00	0.01	0.91	0.00
Hand Clapping	0.00	0.00	0.00	0.00	0.00	1.00

Fig. 5. Confusion Matrix of 3D-RDNet on KTH dataset.

matrix showed in Fig. 5, among them, the color scale on the right of the picture represents the meaning that the darker the color (the closer to 1.0), the higher the accuracy of behavior classification. It can be seen from the confusion matrix that the overall recognition rate of the 3D-RDNet network on the K four dataset is very high, but the model is not very good at distinguishing behaviors like running, jogging and walking, on the one hand, because the similarity of these actions is high, on the other hand, the resolution of the video itself is low, which is easy to cause false judgments. Overall, the 3D-RDNet network has a good recognition effect on the KTH dataset. The model trained on the KTH training set tests the entire dataset and obtains a recognition rate of 97.2%.

3.4 Experimental Results in Real Scene

This research also tested the established real scene data set. The settings of the two data sets during the experiment are basically the same. The experiment is trained from scratch. The input of the network is a continuous 16-frame video clip extracted from each video. The video frame width and height are resized to 171×128. After data preprocessing, the input size is cropped to 8×112×112. In terms of network optimization, the stochastic gradient descent method with momentum is adopted. The initial learning rate of the network is set to 0.01, the momentum parameter is 0.9, the learning rate decay rate is 10^{-4}, the objective function is the cross annoying loss function, and the network training period is 25. The batch size is set to 16.

On the real scene data set, the recognition effect of this model is also better than other networks, achieving a recognition rate of 94.66%. The accuracy rate is calculated based on the clip of 16 consecutive frames of video, namely Clip top-1. In summary, it is shown that the 3D-RDNet network is still capable of tasks in real scenarios, and the network has good robustness and migration learning capabilities.

4. CONCLUSION

Aiming at the problem that the traditional 3D convolutional neural network algorithm lacks the full use of the multi-level convolutional features of the network, this research proposes a three-dimensional residual dense network architecture based on video-based human behavior recognition, which is based on public dataset and real scene data. The effectiveness of the algorithm is verified on the set. The main work of this research lies in: 1) Improve the three-dimensional convolutional neural network, and propose a two-dimensional dense residual network, which reduces the complexity of the model while ensuring the accuracy of the network; 2) A network building module, a three-dimensional dense block of residuals, is proposed. The dense connection mode, local feature aggregation and local residual learning strengthen the network's ability to fully learn multi-layer convolutional features, reducing the original video information in the risk of loss during network training; 3) The proposed algorithm uses multiple dense blocks of two-dimensional residuals to extract multi-level spatio-temporal features, and then aggregates global features to combine the underlying features with high-level semantic features to improve the expressive ability of the model. In addition, the preprocessing and data enhancement methods used in this research can also significantly prevent overfitting during network training. Experiments on public dataset and real scene dataset verify that the proposed algorithm is superior to most traditional algorithms and the same type of 3D convolution method significantly improves the accuracy of video behavior recognition tasks.

REFERENCE

[1] F. Zhu, L. Sha, J. Xie, and Y. Fang, "From Handcrafted to Learned Representations for Human Action Recognition: A Survey," *Image and Vision Computing*, Vol. 55, No. 1, pp. 42-52, 2016.

[2] Y. Guangle, L. Tao, and Z. Jiandan, "A Review of Convolutional Neural Network Based Action Recognition," *Pattern Recognition Letters*, Vol. 118, No. 1, pp. 14-22, 2019.

[3] K. Wang, "A Survey of Human Body Action Recognition," *Pattern Recognition and Artificial Intelligence*, Vol. 27, No. 1, pp. 35-48, 2014.

[4] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Image-net Classification with Deep Convolutional Neural Networks," *Proceedings of the 25th International Conference on Neural Information Processing Systems*, pp. 1097-1105, 2012.

[5] Very Deep Convolutional Networks for Large-scale Image Recognition(2015), https://arxiv.org/pdf/1409.1556. pdf (accessed April 10, 2015).

[6] S. Christian, L. Wei, J. Yangqing, S. Pierre, R. Scott, A. Dragomir, et al., "Going Deeper with Convolutions," *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9, 2015.

[7] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, "Deep Residual Learning for Image Recognition," *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.

[8] H. Gao, L. Zhuang, M. Laurens van der, and W.Q. Kilian, "Densely Connected Convolutional Networks," *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition,* pp. 2261-2269, 2017.

[9] H. Zeyuan, P. Sange-yun, and L. Eung-joo, "Human Motion Recognition Based on Spatio-temporal Convolutional Neural Network," *Journal of Korea Multimedia Society*, Vol. 23, No. 8, pp. 977-985, 2020.

[10] J. Shuiwang, X. Wei, Y. Ming, and Y. Kai, "3D Convolutional Neural Networks for Human Action Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 1, pp. 221-231, 2013.

[11] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning Spatio-temporal Features with 3D Convolutional Networks," *Proceedings of the 2015 IEEE International Conference on Computer Vision,* pp. 4489-4497, 2015.

[12] D. Tran, J. Ray, Z. Shou, S. Chang, and M. Paluri, "ConvNet Architecture Search for Spatio-temporal Feature Learning," *Computer Vision and Pattern Recognition*, Vol. 17, No. 8, pp. 65-77, 2017.

[13] K. Hara, H. Kataoka, and Y. Satoh, "Can Spatio-temporal 3D CNNs Retrace the History of 2D CNNs and lmageNet?" *Proceeding of The 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition,* pp. 6546-6555, 2018.

[14] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A Closer Look at Spatio-temporal Convolutions for Action Recognition," *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6450-6459, 2018.

[15] C. Yunpeng, K. Yannis, L. Jianshu, Y. Shui-

cheng, and F. Jiashi, "Multi-fiber Networks for Video Recognition," *Proceedings of the 2018 European Conference on Computer Vision*, pp. 364-380, 2018.

[16] A. Diba, M. Fayyaz, V. Sharma, M.M. Arzani, R. Yousefzadeh, J. Gall, et al., "Spatio-temporal Channel Correlation Networks for Action Classification," *Proceedings of the 2018 European Conference on Computer Vision*, pp. 284-299, 2018.

[17] N. Hussein, E. Gavves, and A.W.M. Smeulders, "Timeception for Complex Action Recognition," *Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 254-263, 2019.

[18] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing Human Actions: A Local SVM Approach," *Proceedings of the 17th International Conference on Pattern Recognition*, pp. 32-36, 2014.

[19] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior Recognition via Sparse Spatio-temporal Features," *Proceedings of the 2015 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 65-72, 2005.

[20] G.W. Taylor, R. Fergus, Y. LeCun, and C. Bregler, "Convolutional Learning of Spatio-temporal Features," *Proceedings of the 2010 European Conference on Computer Vision*, pp. 140-153, 2010.

Jin-Ho Park

received his B. S. in Business Administration from Tongmyong University, Korea, in 2017; His M. S. in Information and Communication Engineering from Tongmyong University, Korea, in 2019. Currently, he is studying in Department of Information and Communication Engineering from Tongmyong University, Korea for doctoral degree. His main research areas are image processing and pattern recognition.

Eung-Joo Lee

received his B. S., M. S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Aug. 1996, respectively. Since 1997 he has worked with the Department of Information & Communications Engineering, Tongmyong University, Korea, where he is currently a professor. From 2005 to July 2006, he was a visiting professor in the Department of Computer and Information Engineering, Dalian Polytechnic University, and from Dec 2018 he was appointed honorary professor of Dalian Polytechnic University, China. His main research interests include biometrics, image processing, and computer vision.