

**文献翻译报告**

**（ 2021 届本科 ）**

**学 院**：  **信息学院**

**专 业： 软件工程**

**班 级： 17软工2班**

**姓 名：**  **骆春江**

**学 号：**  **1759231**

**指导教师： 王甘霖**

## Research on Two Types of Typical Speech Processing Problems Based on Deep Learning

**Chapter One Introduction**

* 1. **Research background**

**1.1.1Deep learning concepts**

The 1981 Nobel Prize in Medicine/Physiology was awarded to David Hubel, Torsten Wiesel, and Roger Sperry. The main research results of the first two are the discovery of the information processing mechanism of the visual system and the discovery of the brain’s the visual cortex is graded [1] , as shown in Figure 1.1. The visual signal received by the human retina in the V1 area.It is a simple visual form, such as edges, corners, etc.; the V2 and V4 areas process these corners into intermediate visual forms, such as feature groups; in the AIT area, the brain can form high-level object descriptions, such as faces and objects. David and Torsten believes that the human visual function is abstraction and iteration. Abstraction is to take concrete image elements such as original. The pixel information is abstracted into meaningful concepts. These meaningful concepts will iterate upwards and generate more abstract, human perceivable concept.

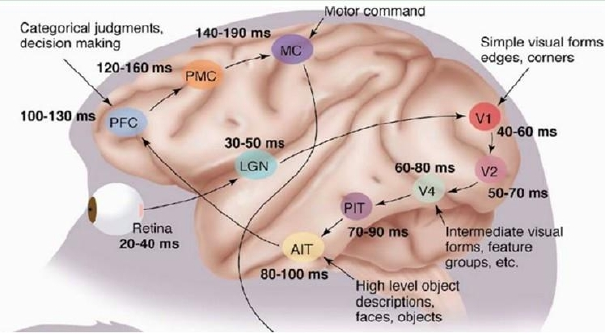


Figure 1.1 the hierarchical nature of the visual cortex of the brain [1]

Deep learning (Deep Learning, DL) originated from the research of artificial neural networks, which is the field of machine learning a new subject. Deep learning simulates the mechanism of the human brain to understand images, speech, text and other data, and its motivation. It is to build a neural network that simulates the human brain for analysis and learning. Deep learning expresses by combining simple concepts. For more complex concepts, the process of identifying faces is also an abstract iterative process, as shown in Figure 1.2. Input, the pixel information is similar to the original signal received by the human retina, and the edge information of the second layer is similar to the V1 region of the human brain. The edge detection of a certain part of the face of the third layer is similar to the initial shape detection of the human brain V2 area, the final face model is similar to the high-level visual abstraction of the V4 region of the human brain.

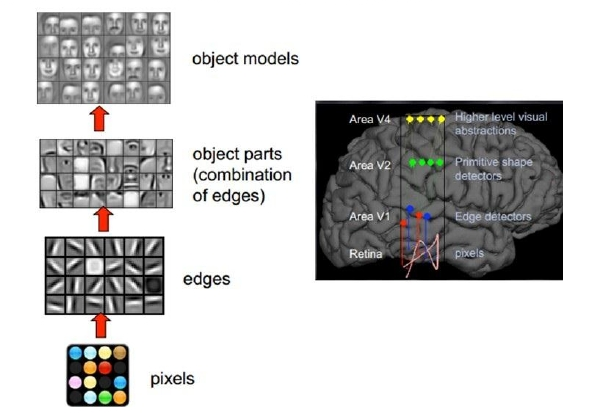


Figure 1.2 Face recognition process[122]

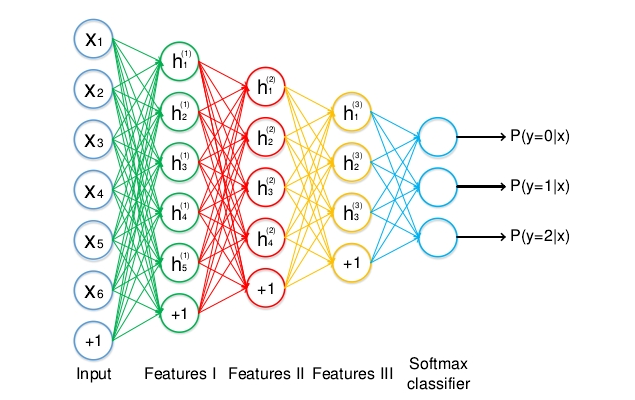


Figure 1.3 Schematic diagram of feed forward deep neural network

A typical deep learning model is a feed forward deep neural network (Deep Neural Network, DNN) or Called Multilayer Perceptron (MLP), as shown in Figure 1.3, where xi represents the input layer, hi(1) hi(2) hi(3) represents the hidden layer, and the last layer is the hidden layer, which represents the probability that the input belongs to each class. MLP, it is a mathematical function that maps the input set to the output set. This function is obtained by combining many simple functions to. For input, it can be considered that different mathematical functions can be used to obtain different output expressions

* 1. **The status quo of speech processing based on deep learning**

**1.2.1 Speech recognition**

For a long time, the dominant method of speech recognition is Gaussian mixture model-hidden Markov model (Gaussian Mixture Model-Hidden Markov Model, GMM-HMM), the method is based on context-sensitive. The generative model of GMM and HMM. Neural network was once a popular method for speech recognition, but it was effective. It is not as good as GMM-HMM.

After close cooperation between academia and industry in 2010, deep learning began to have an impact in the field of speech recognition[70] [71] . The cooperation began with the task of phoneme recognition, and the recognition results showed the DNN architecture and subsequent convolutional networks. The ability to structure, recursive network architecture. Their work also demonstrates the transition from the widely used MFCC features. To the lower level of the importance of the original speech spectral features. Their cooperation was also achieved on the task of large vocabulary recognition a good effect [72]. The main reasons for DNN's success in large vocabulary speech recognition tasks are: similar to. For the speech unit in GMM-HMM, DNN uses a large-scale output layer structure. The principle of using this structure. Because speech researchers expect to use context-sensitive phoneme modeling techniques, which are used in GMM-HMM Very effective; at the same time, this structure can be changed as little as possible for the high-efficiency decoder software developed by GMM-HMM Piece architecture. DNN's work in large vocabulary speech recognition also shows that if a large amount of labeled data can be used, the pre-training process similar to the deep belief network is unnecessary. Deep learning in the industry of speech recognition and academic circles have achieved success, this success is mainly due to three factors: (1) and the best GMM-HMM before compared with the system, deep learning significantly reduces the speech recognition error rate; (2) Because of the use of phonemes as the DNN output, the deployment of a DNN-based speech recognizer requires only a small amount of changes to the decoder; (3) DNN the powerful modeling ability reduces the complexity of the speech recognition system.

After the DNN-HMM system has achieved success in speech recognition, researchers have proposed many new architectures and the non-linear unit is used to improve the accuracy of speech recognition.

Yu et al. [73]used dual projection layers and tensor layers to replace one or more layers in the traditional DNN, and improved the tensor version of DNN is released. The dual projection layer projects each input vector into two nonlinear subspaces. In Zhang

In the measurement layer, two subspace projections interact with each other and jointly predict the next layer of the entire depth architecture. Researcher when proposed a method to map the tensor layer to the traditional sigmoid layer, so the tensor layer can be the same as the sigmoid layer train in a similar way.

The idea of ​​time-domain convolution is derived from the time-delay neural network (TDNN), as this kind of shallow network is used in early speech recognition. Recently researchers used deep convolutional neural networks for phonemes when identifying tasks, it is found that weight sharing in the frequency domain is more effective than weight sharing in the time domain [74]. A study, the report also pointed out that convolutional neural networks can help large vocabulary continuous speech recognition tasks, using a large number of convolution kernels or the multi-layer convolutional neural network of feature maps will have a greater performance improvement [75]. Sainath et al. [76] explored the depth volume a large number of variants of the convolutional neural network, it was found that when combined with several new methods, the deep convolutional neural network these large vocabulary speech recognition tasks have achieved the best results.

In speech recognition tasks, the most noteworthy deep structure is the recurrent neural network and its deep version[77][78]. Although RNN was the first to succeed in phoneme recognition, due to the complexity of training RNN, RNN difficult to extend to larger speech recognition tasks. Since then, the learning algorithm of RNN has been improved, using RNN better results have been achieved on some tasks, especially the use of bidirectional LSTM RNN[79].

In addition to innovating in deep learning models for speech recognition, a lot of work development and realization of better non-linear units. Sigmoid function and tanh function are the most commonly used nonlinearities in DNN Function, but both have limitations. For example, when the neuron node is close to saturation, the error function is relatively because the gradient value of the parameter is very small, the network training speed is slow at this time. To overcome the sigmoid function and tanh function for the shortcomings, Jaitly and Hinton [80] first used ReLU in speech recognition. Excitation function represented by ReLU for f(x)-max (0, x).

Another effective unit for speech recognition is the maxout unit, which is used to build a deep maxout network[81]. The deep maxout network performs maximum operation or maxout operation on a fixed number of weighted inputs. Operate to generate hidden layer activation value. This operation is the same as the maximum pooling operation in convolutional neural networks. These most the larger value is the output of the previous layer. Afterwards, Zhang et al. [82] generalized the maxout unit into two new types. The first is to use the soft-max function to replace the soft-max unit of the original max function, and the second is to use y=||x||p the p- norm unit of. Experiments show that using p =2 a p-norm unit than maxout unit, tanh the effect of unit and ReLU unit are good.

**1.2.2 Audio and music processing**

In the field of audio and music processing, the research involving deep learning is mainly about music signal processing and music information. Information retrieval [87][88] . Music audio signal is a time sequence organized by music time instead of real time. Change with rhythm and emotion. The influencing factors of music audio signal include music tradition, style, composition and deduction. Music audio signals are complex and changeable, and the high-level abstractions provided by deep learning are very suitable for the characterization of music audio signals.

In order to process high-dimensional audio signals, Lee et al. [89][90]used restricted Boltzmann combined with convolutional structure the machine built a convolutional deep belief network. The convolution operation shares the weight in the time dimension between hidden layer nodes, using in the detection time invariance characteristics. Convolutional deep belief networks have been applied to many tasks in audio and voice data.Business, including classification of music artists and music genres, speaker recognition, speaker gender recognition, phoneme classification, etc., and achieved good results.

Due to the powerful dynamic system modeling capabilities of RNN, researchers have recently applied RNN to music processing

In application [91] , and use ReLU as the excitation function instead of logistic function or tanh function. RNN,it is mainly used to automatically recognize chords from music, which is an active topic in the field of music information retrieval. Experiment the results show that the RNN-based automatic chord recognition system is comparable to the best existing methods. RNN can learn acquire basic musical attributes, such as time continuity, harmony, and time dynamics.

Deep learning can also be applied to content-based music recommendation systems [92]. Automatic music recommendation has become it is a very important technology in daily life. Many recommendation systems rely on collaborative filtering, this kind of algorithm exists the cold start problem, that is, the algorithm will fail when there is no data available. Therefore, collaborative filtering algorithms are recommending new songs and unpopular songs are not very effective. Deep learning uses the latent factor model for recommendation. If not obtain the potential factors from the available data, and predict the potential factors from the music audio. The experimental results show that the deep degree learning can achieve good results in content-based music recommendation.

* 1. **The main work of the paper**

**1.3.1 Voice matching**

Voice matching automatically retrieves the same content as the query voice segment from the given voice database All voice clips of. Voice matching is a type of content-based voice retrieval application. It is used in music retrieval, song recommendations, voice intelligence analysis, etc. are widely used. At the same time, voice matching is a kind of unsupervised learning task, suitable for the technology used for speech matching can be applied to other unsupervised learning tasks in the field of machine learning to study speech the matching algorithm has important academic value. Therefore, the voice matching task is a typical voice processing task.

The key to speech matching is the extraction of speech features. Poor generalization ability for traditional speech feature extraction algorithms the shortcomings of this paper, this paper proposes to use the convolutional deep belief network for speech feature extraction, and based on the convolution depth Based on the binary features extracted by the belief network, a fast voice feature matching algorithm is proposed.

**1.3.2 Multimodal speech recognition**

Human-computer interaction interfaces for smart machines, such as smart phones, home robots, and self-driving cars it becomes more and more common in daily life. Speech recognition robust to noise is the key to effective human-computer interaction. Multi-modal speech recognition is considered to be one of the effective solutions for robust speech recognition. In the human-computer interaction system, in addition to receiving the operator’s voice signal, the machine can also observe the operator’s behavior information, such as the body movement and mouth shape changes. These behavioral information can help the machine recognize the operator’s voice signal and study multi-modal speech recognition has important application value in human-computer interaction systems. In addition, multi-modal speech recognition it is a supervised learning task that involves the fusion of multi-source information and has important academic research value. Multimode State speech recognition is also a typical speech processing task.

Based on the recurrent neural network, this paper proposes a speech recognition framework combining audio and video: multi-modal recursive neural network. It includes the auditory part that processes audio, the visual part of video, and the auditory part the part where the part and the visual part merge. There are many variants of each part, which can be based on the specific task configure the network architecture appropriately.

**Chapter 2 Background Knowledge**

**2.1 Deep Belief Network**

The deep belief network was proposed in 2006 by Geoffrey Hinton, a well-known deep learning scholar [27], which is a deep significant progress in the field of learning. DBN is a probabilistic generative model, which contains several layers of hidden variables. Hidden variable it is often binary, and the visible unit can be binary or real. There is no connection between nodes in the layer. Through often, each node of each layer is connected to all nodes of the adjacent layer, although it can be constructed more sparsely sparsely connected deep confidence network.

DBN is constructed by stacking restricted Boltzmann machines. DBN is first initialized by a layer-by-layer greedy training strategy the network then uses the desired output to jointly tune all the parameters of the network. DBN's layer-by-layer greedy training strategy, there are two advantages: one is that an appropriate initial value can be generated for the network, because inappropriate parameter selection can be can make the network converge to a relatively poor local optimal solution, so this solves the problem of parameter selection to a certain extent difficulty; the second is that the training process is unsupervised, so there is no need for a class label for training data.

**2.1.1 Restricted Boltzmann machine**

RBM is a two-layer network, called visible layer V and hidden layer H respectively. Visible layer unit v and hidden layer list the joint probability p of element h is defined by the energy function E:

(2.1)

The segmentation function Z is obtained by accumulating all possible visible layer and hidden layer pairs:

(2.2)

If the visible layer unit is binary, define the energy function E:

(2.3)

Where vi,hj visible layer units i and hidden layer units j state, ai,bj is its offset value, wij to connect it their weight value. If the visible layer unit is real-valued, define the energy function E:

(2.4)

RBM is usually trained using contrast divergence [93]. Given the state of the visible layer unit, the hidden layer unit state the probability p that hj is set to 1 is:

(2.5)

WhereIs the sigmoid function 1 / (1+exp(-x)) .After the binary state of the hidden layer unit is determined, you can "Fictitious" visible layer. The "fictitious" visible layer unit state the probability p that vi is set to 1 is:

(2.6)

The status of the hidden layer unit is updated again to represent the characteristics of the "fictional" visible layer. Weight update w through the following formula:

(2.7)

Where ε is the learning rate, is the product of the visible layer unit and the hidden layer unit of the original data, for "Fictitious" data is the product of visible layer units and hidden layer units. The update of the offset value uses a more simplified learning the rule of thumb is not to use the product of paired unit states, but to use the state of individual units.

**2.2 Convolutional Neural Network**

Convolutional neural network is one of the most famous deep learning models, which is used in a variety of computer vision tasks. Got the best results. The general CNN structure is shown in Figure 2.1, which shows the application of CNN to images identification, which contains an input layer, two convolutional layers C1 and C3, a pooling layer S2, and a fully connected layer, and an output layer.

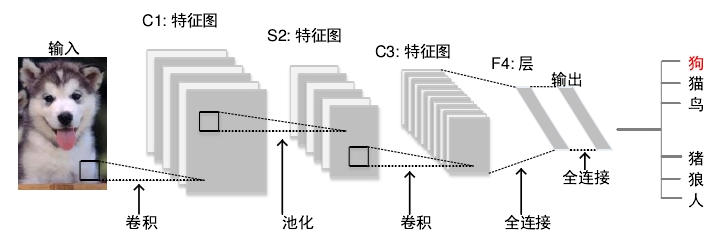


Figure 2.1 Convolutional neural network structure

Usually CNN contains three types of layers, namely convolutional layer, pooling layer, and fully connected layer. Different types of layers Play different roles. Next, this section first introduces the functions of the various layers and the recent research progress, then give CNN training algorithm, and finally give a typical CNN model.

**2.2.1 Types of layers**

CNN is a hierarchical neural network, in which convolutional layers and pooling layers are alternately connected, and finally some fully connected layer. This section gives the functions of the three layers and briefly reviews recent research progress on these layers.

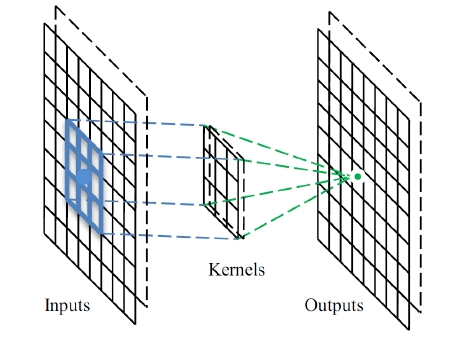


Figure 2.2 Convolutional layer operation [123]

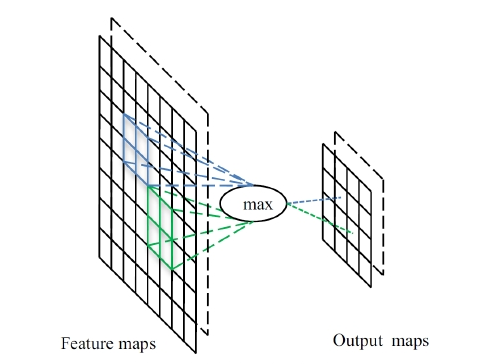


Figure 2.3 Maximum pooling layer operation [123]

**2.2.1.1 Convolutional layer**

In the convolutional layer, CNN uses different convolution kernels and the entire input image and the feature map generated in the middle to do the convolution product operation and generate different feature maps, as shown in Figure 2.2, where two convolution kernels and the input feature map are used for convolution. The product operation generates two output feature maps. Convolution operation has three main advantages: 1) The weight sharing mechanism in the same feature map reduces the network number of parameters in the middle; 2) Local connection learns the associated information of adjacent pixels; 3) The position of the object in the image it is immutable.

One way to deal with the convolutional layer is Network in Network [94] , the main idea is to use a small scale the multi-layer perceptron replaces the traditional convolutional layer. The multilayer perceptron has multiple fully connected layers and uses non-linear the sexual activation function, in this way, replaces the linear convolution kernel with a nonlinear neural network. The method is shown in the figure good results have been achieved on image classification tasks.

**2.2.1.2 Pooling layer**

Generally, the pooling layer follows the convolutional layer, which can reduce the dimensionality of the feature map and the number of network parameters item. Because the calculation of the pooling layer takes adjacent pixels into consideration, similar to the convolutional layer, the pooling layer also has there is offset invariance. Mean pooling and maximum pooling are the most commonly used strategies. Figure 2.3 shows the maximum pooling examples of operations. For 8\*8 feature map, using size 2\*2 pooling operator with 2 as the step size, then the output graph is reduced to 4\*4. For the maximum pooling layer and average pooling layer, Boueau et al. [95] detailed their performance theoretical analysis. Scherer et al. [96] further compared these two operations and found that the maximum pooling the layer has a faster convergence rate, can choose better features with invariance, and improve the generalization of the model ability. In the implementation of CNN in recent years, most of them use the maximum pooling strategy. The pooling layer is the most studied of the three layers. Regarding the pooling layer, there are three different for different purposes strategy.

**• Random pooling**

One disadvantage of maximum pooling is that it is easy to over fit the training set, making it difficult to generalize well to the test set. Trial set. To solve this problem, Zeiler et al. [97] proposed a random pooling method. Pass in each pool the activation value is randomly selected according to the polynomial distribution function in the transformation area, and random pooling is replaced by a random process it replaces the traditional deterministic pooling operation. Random pooling operation is equivalent to the standard maximum pooling operation there are multiple copies of the input image, and each copy has a small local deformation. Random pooling operation the machine characteristics help prevent over fitting.

**• Spatial pyramid pooling**

CNN usually requires the input image to be a fixed size. This restriction may reduce images of any size the recognition accuracy rate. In order to eliminate this limitation, He et al. [98] used the spatial pyramid layer (Spatial pyramid pooling, SPP) replaces the last pooling layer of traditional CNN. The SPP layer can be from any extracting fixed-length features from the image provides flexibility for processing images of different sizes and aspect ratios solution. Applying SPP to CNN can improve the performance of CNN.

**• Deformation constraint pooling**

In computer vision, especially in object detection tasks, dealing with deformation is a fundamental challenge. Maximum value pooling and mean pooling can be used to deal with deformation, but they cannot learn the deformation constraints and geometry of local objects what model. In order to deal with deformation more effectively, Ouyang et al. [99] proposed a deformation constrained pooling layer, which enrich the depth model by learning the deformation of the visual pattern. The deformation constrained pooling layer can be used to replace transmission any pooling layer of CNN.

**2.2.1.3 Fully connected layer**

In CNN, the last pooling layer is usually followed by several fully connected layers. Fully connected layer combines two-dimensional features the graph is converted into a one-dimensional feature vector to facilitate subsequent feature representation. The fully connected layer is similar to a multilayer perceptron, as shown in Figure 1.3 in Chapter 1. The fully connected layer performs calculations similar to traditional DNN, and its parameters account for 90% of all CNN parameters. The structure of the fully connected layer is usually not changed. However, in transfer learning, a common practice is to keep the network parameters learned on the ImageNet dataset, and use two new fully connected layers to replace the trained network the last fully connected layer of the network makes the new network suitable for new visual recognition tasks. The lack of fully connected layer the point is that it contains a large number of parameters, which brings huge computational overhead to the training of CNN.

**2.2.3 Classic model**

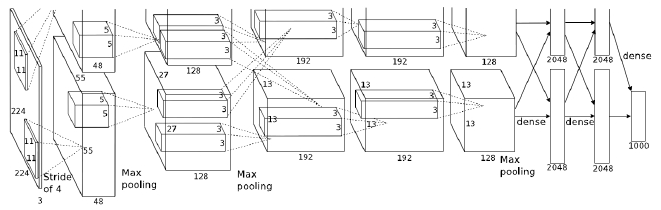


Figure 2.4 Architecture diagram of AlexNet [48]

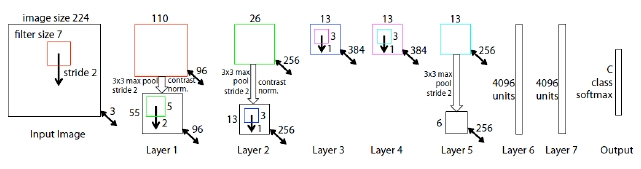


Figure 2.5 Clarifai [100] architecture diagram

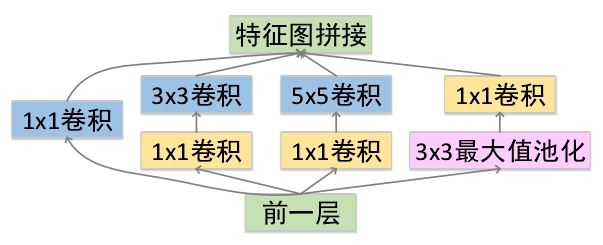


Figure 2.6 Inception module of GoogLeNet [50]

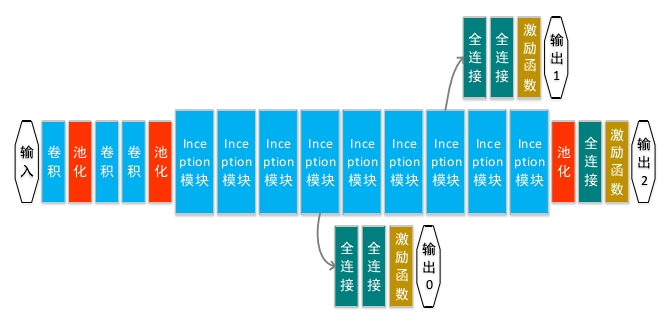


Figure 2.7 GoogLeNet [50] architecture diagram

AlexNet [48] is a well-known CNN architecture, which contains 5 convolutional layers (the first, second, and fifth convolution after the layer there is a pooling layer) and 3 fully connected layers, 650,000 neurons, 60 million parameters, and the input is a fixed size (224 224), as shown in Figure 2.5. AlexNet is trained on the ImageNet data set, the 1.2 million high-resolution training images are divided into 1000 different classes. AlexNet uses multiple generalizations techniques, including data increase, dropout, etc. The network won the ILSVRC2012 competition, the image classification the Top-5 error rate was 15.3%. In 2013, Zeiler et al. [100] proposed a novel visualization technique that allows them to analyze the internal working principle of the middle feature layer of AlexNet, and based on this, a new CNN architecture called Clarifai. Clarifai is similar to AlexNet, including 5 convolutional layers and 3 fully connected layers, but two networks the size of the convolution kernel is different from the step size. The architecture of Clarifai is shown in Figure 2.6. Clarifai on ImageNet it surpassed AlexNet in the classification task and won the ILSVRC2013 competition in the ImageNet classification task the Top-5 error rate dropped to 11.2%. Unlike AlexNet, GoogLeNet [50] increases the depth of the network by adding more convolutional layers, and use very small convolution kernels in all layers. GoogLeNet reduces the number of fully connected layers, because fully connected the layer occupies most of the parameters of CNN, which makes the network parameters of GoogLeNet only 5 million. GoogLeNet introduces the Inception module, as shown in Figure 2.6, to use convolution kernels of different sizes. GoogLeNet contains 9 Inception modules, and the entire network has 21 convolutional layers and 1 fully connected layer. its architecture is shown in Figure 2.7. GoogLeNet won the ILSVRC2014 competition, in the ImageNet image score the Top-5 error rate on similar tasks dropped to 6.67%.

**Chapter 3 Speech Matching Based on Convolutional Deep Belief Network**

**4.1 Introduction**

With the development of multimedia and network technology, people can freely play and download interesting music files pieces. Therefore, an effective and simple search method is particularly important for obtaining music files and related information. Currently, most music search engine mechanisms rely on the label information of music files. These marked information are usually actors, song name, album name, or other information. The most typical message is to use the name of the composer as a keyword, such as "Beethoven" (as shown in Figure 3.1(a)). These music search engines often require users to master basic music music information, that is, the traditional text annotation retrieval system can only search for content that the user already knows. To overcome the target-based note the limitations of the retrieval system, the researchers proposed a retrieval system based on music tags. Music tags usually contains the music type or genre, as well as the mood, beat and other information conveyed in the song. Figure 3.1(b) shows for the "Beethoven" tag cloud, the most significant tag is "classical". When users pay attention to specific music when it comes to sexual music, tag information becomes very important. However, obtaining these label information is both labor-intensive and big time-consuming project. In particular, to obtain a credible, consistent, and musically meaningful label requires musical professional knowledge guidance. Recently, retrieval strategies based on the content of music itself (as shown in Figure 3.1(c)) have shown great potential. The retrieval strategy based on music content does not rely on any text annotations or manually generated tags, but just based on the voice itself. A typical scenario is that a user records a favorite song in a shopping mall. Without knowing the song name and artist and other information, the user can use the recorded song fragment as the query voice retrieve the song. content-based speech retrieval and analysis is a very challenging task in the field of signal processing. Based on content voice retrieval tasks, voice matching aims to retrieve and query voice from a given voice database. Fragment All speech fragments with the same content. A typical scenario for voice matching is how much the same song is a singing version, such as a 10-second song of "We will rock you" sung by a given Queen the goal of voice matching is to find all the segments in the database that have the same content as the 10-second segment. Including the repetition of the 10-second segment of the song from "We will rock you" performed by the queen, as well as other singers such as Britney Spears, Russian Red, The Park sang "We will rock you" and these 10-second fragments have the same content fragments.

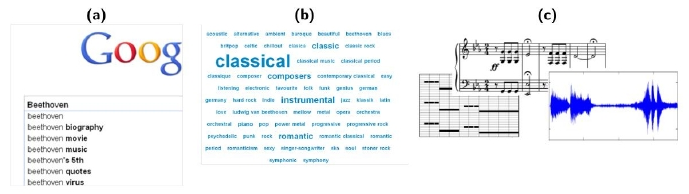


Figure 3.1 Different retrieval methods

Traditional voice matching relies on the results of voice identification. Voice identification identified from the voice database contains query the complete voice of the voice fragment. At present, speech identification algorithms have made great progress, even if the query the sound has noise, compression marks, or slight time distortion. However, these algorithms cannot handle strong non-linear time distortion or spectral deviation. In order to overcome the shortcomings of speech matching algorithms based on speech identification, Kurth and Muller et al. [101][102] proposed the chrominance energy normalized statistical feature CENS is used for speech matching. Voice matching based on CENS features the algorithm flow is: firstly, the query voice segment and all voice files in the database are converted into CNES features sequence, and then calculate the query voice CENS feature sequence and the voice file CENS feature sequence in the database the distance function of any subsequence, and finally the distance function is used to determine the best match for the query voice in the database voice snippets. The key of this algorithm is the extraction of CNES features. The CENS feature is first extracted from the speech signal take the chroma feature (chroma feature), and then blur the chroma feature in time to obtain a robustness to the local beat difference great, and finally standardize the features obtained in the previous step to achieve invariance to dynamic deviations. CENS feature pairs deviations in state, sound quality, etc. and local beat differences have a certain degree of robustness. However, CNES features the characteristics of manual design have poor generalization ability. Experiments show that the voice matching algorithm based on CENS features in many cases, it is difficult to achieve satisfactory results. In order to overcome the shortcomings of CENS features, this chapter uses CDBN to unsupervisedly extract features from speech data. Sign for voice matching. CDBN combines the advantages of both CNN and DBN, which can unsupervised the features extracted from the dimensional data are consistent with the characteristics of the voice matching task. Further, based on Kurth and Muller the speech matching algorithm process proposed by others, this chapter proposes a more efficient speech feature matching algorithm. the method is designed for binary characteristics. This chapter evaluates the basis on the TIMIT database and a simulated music database. For CDBN's speech matching algorithm, the experimental results show the effectiveness of the proposed algorithm.

**3.2 Convolutional deep belief network model**

When DBN is applied to computer vision tasks, its disadvantage is that it does not consider the two-dimensional structure of the input image information. To overcome this problem, Lee et al. proposed CDBN [103]. Due to the introduction of convolution operations and pooling.

**3.2.1 Convolution restricted Boltzmann machine**

For the convenience of writing, this section makes some simplified assumptions. Suppose the input of CDBN is Nv × Nv image, and it is assumed that all units are binary (can be directly extended to real-valued visible layer units, refer to section 2.1.1). In this section, use \* to denote convolution operations, and use • to denote corresponding elements to be multiplied and then added, namely A·B=trATB, and use A indicates that the array A is flipped horizontally and vertically.

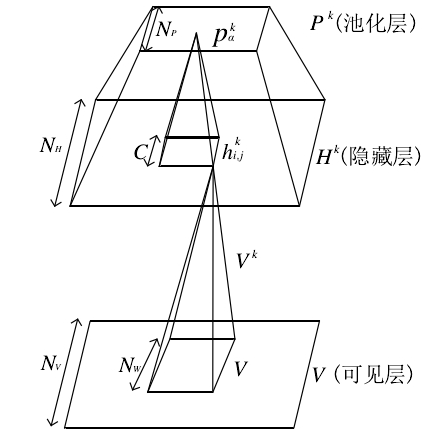


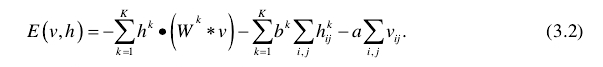
Figure 3.2 Convolution restricted Boltzmann machine with maximum probability pooling layer

Intuitively, CRBM is similar to RBM, but in CRBM, the weight between the visible layer and the hidden layer all positions in the input image are shared. The basic CRBM consists of two layers: input layer V and hidden layer H (corresponding to the lower two layers in Figure 3.2). Input layer contains Nv × Nv Two binary units, the hidden layer contains K groups, each group contains Nh × Nh Binary units, total NH2K hidden layer units. Each group of K hidden layers associate one Nw × Nw filter (Nw=Nv –NH +1); All hidden layer units in the same group have been shared filter weight. In addition, each group of hidden layer units hk corresponds to an offset value bk , and all visible layer units v share an offset value a .

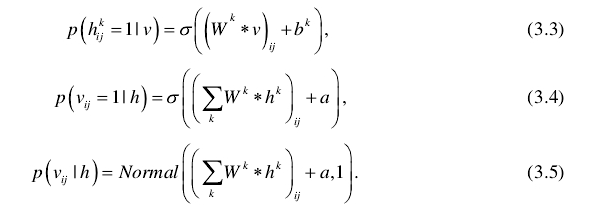
Define the energy function E :



Use the previously defined operator,



Similar to RBM, the following conditional probability distribution p can be obtained :

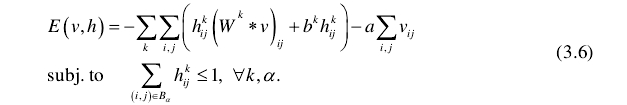


Where σ is the sigmoid function and Normal is the normal distribution function. Equation 3.4 corresponds to the binary visible layer unit, equation 3.5 corresponds to the real-valued visible layer unit.

**3.2.2 Maximum probability pooling**

Generally, the higher the level of feature detectors, the more information about the input area is needed. Translation invariant sexual representations, such as CNN, often alternately contain two types of layers: feature detection layer and pooling layer. Detect the response of the layer is calculated by convolving the feature detector with the previous layer, and the pooling layer passes a constant the factor reduces the characterization information of the detection layer, more specifically, each unit of the pooling layer calculates a cell of the detection layer the maximum value in the domain. Through the maximum pooling operation to reduce the characterization information, on the one hand, the high-level characterization can be the input local small offset is invariant; on the other hand, it can greatly reduce the calculation pressure. Maximum pooling operation only it is designed for the feed-forward structure, so in order to generate the model (including both top-down inference, but also the inference to the top) contains operations similar to maximum pooling. Lee et al. [103] proposed maximum probability pooling. Consider that as shown in Figure 3.2, there is a visible layer V , a feature detection layer (hidden layer) H , and a pooling layer P 's model. The detection layer and the pooling layer have K groups of units, and each group of the pooling layer has Np × Np a binary unit. For each k, Pooling layer Pk reduces the detection layer k along each dimension by a factor C characterization of H , here C is a small integer, such as 2 or 3. In other words, the detection layer HK is divided into size C × C of small blocks, each small block is connected to the only binary unit k of the pooling layer pαk ,therefore NP=NH/C. Formally Bα=｛( I , j ) | hij∈ block α｝Can be defined.

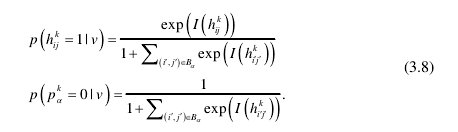
Block Bα the detection layer unit and the pooling layer unit pα connect with a single possibility and satisfy such the lower limit: at most one unit of the detection layer is open, and the pooling layer is open if and only if One detection layer unit is open. Equivalently, these C2+1 units as a single random variable, the random variable can be C2+1 possible values: C2 values ​​correspond to each detection layer the element is in the open state, and another value corresponds to all the elements in the closed state. Lee et al. defined the simplified energy function E of CRBM with maximum probability pooling according to the following formula :



Based on this energy function, we discuss how to sample the feature detection layer H and the pooling layer P when the visible layer V is given . The first k Group the bottom-up signal received from visible layer V is:



Lee et al. independently sample each block with a polynomial function input to each block. Suppose hijk is included in the block the hidden layer unit in α (（i，j）∈ Bα ), open unit hijk the energy increase caused by h is  *-I（hijk）*, conditions the rate is given by:



After the hidden layer H is given , the sampling of the visible layer V is the same as Section 3.2.1.

Similar to DBN by stacking RBM, CDBN alternately stacking CRBM and maximum probability pool the structure of the chemical layer is used to obtain a generative model with offset invariance. CDBN also greedily trains the party layer by layer formula to optimize: that is, after the training of the current given layer is completed, its weight will not change, and the activation value of this layer will be used as the input into the next floor. In addition to image data, CDBN has also achieved success in speech processing tasks [90].

**3.3 Algorithm flow**

The flowchart of CDBN-based voice matching algorithm is shown in Figure 3.3. First, pre-process the voice data the processing is mainly to reduce the dimensionality of the voice data, and then use CDBN to extract the voice data from the preprocessed voice data. Take the features, and finally execute the voice feature matching algorithm based on the voice features extracted by CDBN.

**3.3.1 Pretreatment**

In the preprocessing stage, the time domain signal is first converted into spectral data. The frequency spectrum has a window size of 20ms. And has an overlap of 10ms. However, the dimensionality of the spectrum is relatively high, so this section uses PCA whitening to reduce the frequency. The dimension of the spectrum. The final voice data input to CDBN contains nc one-dimensional vector of n channels, where nc is PCA the number of ingredients (set to 80 in the experiments in this chapter).

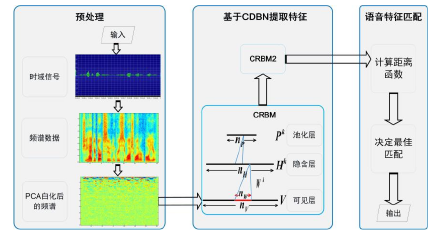


Figure 3.3 Flow chart of voice matching algorithm based on convolutional deep belief network

**3.3.2 Speech feature extraction**

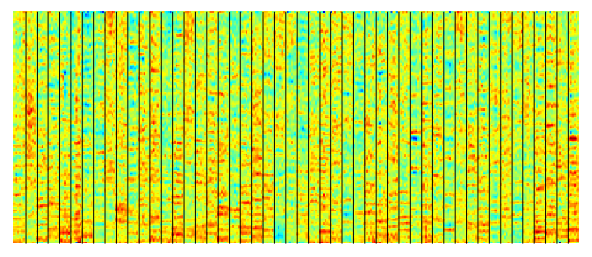


Figure 3.4 Visualization of the feature detector randomly selected in the first layer of the convolutional deep belief network

This chapter uses CDBN to extract speech features. However, it should be noted that the common CDBN is used to two-dimensional image data, this chapter uses CDBN to process one-dimensional speech vectors. The CDBN used to extract speech features consists of two CRBMs. The configuration of the two CRBMs is the same: Each contains 300 convolution kernels, the size of the convolution kernel is 6, and the maximum pooling ratio (local neighbor size) is

3. The features learned by CDBN can be visualized. This section multiplies each feature detector use the inverse of PCA whitening to visualize the first layer of feature detectors, as shown in Figure 3.4. As you can see, different the feature detector has learned different voice features. The energy ratio of some feature detectors in the high frequency region of the spectrum more concentrated, some feature detectors are more concentrated in the low frequency region, and some feature detectors are more uniform distributed in the entire spectrum band.

**3.3.3 Voice feature matching**

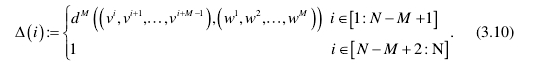
Using the voice features extracted by CDBN, according to the voice matching framework proposed by Kurth and Muller, this section proposes a more efficient speech feature matching algorithm. The voice database contains a set of voice records, which these voices may be the same sentence spoken by different people or the same song sung by different singers. By putting all separate voice records are spliced ​​together. In this section, a large file D is used to represent the voice database, and Q is used to represents a short query voice segment. All voice recording files and query voice files are preprocessed and characterized after extraction, all have C channels. DefiniteΩC:=｛x=(x1,…,xc)T|xi∈｛0,1｝｝. In the feature extraction stage, convert the spliced ​​voice file D and query fragment Q into a binary feature sequence. This section uses F[D]=(v1,…,vN) and F[Q]=(w1,…,wM) represents these characteristic sequences, in which all n ∈[1:N] have vn∈ΩC . For all m∈[1:M] have wm ∈ΩC.

Matched target speech is spliced into a voice file D identify all voice query Q having

Fragments of the same content. In this section, the feature sequence F[Q] and F[D] contains M continuous vector subsequences line comparison. Specifically, let X=(x1,…,xM) ∈ ΩCM，Y=(y1,…,yM)∈ΩCM and define ,among them x∈ΩC, y∈ΩC ,  is defined as:



Then, set this section dM (x，y):=1-, where X∈, Y∈. Note dM meaning the value range is [0,1] ⊂ ℜ , and the value is 0 when X and Y coincide. Next, for F[D] and F[Q], this section defines the distance function △：[1:N]→[0,1]:



△(i) describes F[Q] and F[D] in one sub-sequence from the sequence starting at position i and comprises M number continuous vector.

The calculation of ∆ is shown in Figure 3.5, sliding with a step of the window of size M in F[D] , and the window mouth and query voice clips F[Q] compared.

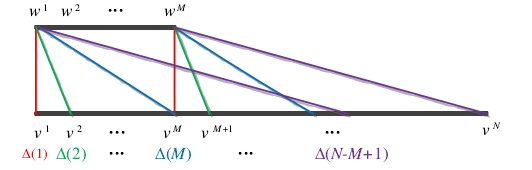


Figure 3.5 Diagram of distance function calculation

The best match of Q in D can be determined by continuously selecting the minimum value of the distance function ∆ : in the first step, determine the index i ∈[1: N] that minimizes ∆, indicating that it corresponds to the characteristic sequence (vi,…,vi+M-1) speech fragment is the best match. Then the sequence of length M near the best match is excluded from further consideration. In order to avoid a large overlap between subsequent matches and the current best match. In the second step, make sure to minimize the modification the feature index of the distance function to get the second best matching result. The process will continue until to retrieve a predefined number of matching voice fragments or the distance between the retrieved matching voice and the query voice exceeds pass a certain threshold.

The main overhead of the feature matching algorithm is calculation of dM. Since the speech features extracted by CDBN are binary, using this feature, this section rewrites dM the calculation formula is as follows:



The XOR calculation in this formula is compared with the vector inner product operation of the feature matching algorithm proposed by Kurth and Muller. calculate faster.

**3.4 Experiment**

In this section, MATLAB is used to implement the CDBN-based voice matching algorithm, and the TIMIT database and its performance is evaluated on an analog music database containing songs collected from the Internet. To train CDBN, this section uses the unlabeled speech database TIMIT[104].

**3.4.1 Voice matching results on TIMIT database**

The TIMIT database contains a total of 6,300 sentences, including 8 major dialect regions (dr1—dr8) in the United States. 630 speakers, each with 10 sentences. In each dialect area, there are more than 20 speakers narrated 2 sentences (sa1, sa2), sentences narrated by only one speaker (si--), and sentences narrated by 1 to 5 speakers child (sx--). In each dialect area, this section splices all the voice files into one large voice file to get 8 spliced ​​voice files, one for each dialect region. In each dialect area, select 8 short voice files (sa1, sa2, 3 si--, 3 sx--) as the query voice, and the corresponding spliced ​​voice these query voices are retrieved from the file. The query voice files selected for each dialect area and these query voices the number of times the file is repeated in the corresponding spliced ​​voice file is shown in Table 3.1.

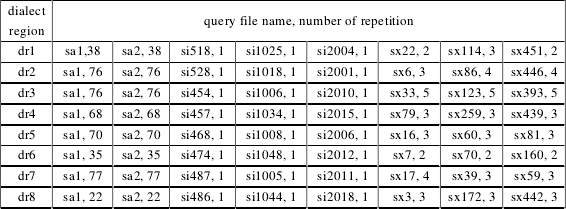


Table 3.1 Query voice files and the number of times each query voice is repeated

Next, we discuss the voice matching results of the proposed algorithm. For one in the database nq repeated occurrences query voice file, the voice feature matching process will return

nq indexes indicating where the query fragment appears. If the location predicted by the algorithm is exactly where the query fragment appears, it is called a "hit". Different phonetic text the number of “hits” of files is shown in Figure 3.5, where each sub-picture shows a dialect area: dr1 in the upper left corner, the upper right corner is dr2, and the lower right corner is dr8. In addition to the results based on CENS features, this section will also be based on CDBN the algorithm is compared with the voice matching algorithm based on MFCC features. Here, "CDBN L1" is the first layer features, "CDBN L2" is a feature of the second layer.

It can be seen from Figure 3.6 that the voice matching algorithm based on CDBN far exceeds that based on MFCC and CENS. Feature voice matching algorithm, whether it is for query voices with more than 20 repetitions in the database, or for there are only a limited number of repeated query voices. Specifically, for "si--" query voice files, based on CDBN matching the matching algorithm can successfully locate all query files in the corresponding voice files, but based on MFCC and the CENS feature algorithm cannot successfully locate these query files; for the "sx--" query voice file, the basic the method based on CDBN can locate part of the positions where these query voice files appear, but based on MFCC the algorithm with CENS feature cannot locate a position; for sa1 and sa2, the algorithm based on CDBN for most query voice files, more query voices can be located. In order to verify the robustness of the CDBN-based voice matching algorithm to noise, this section will add white Gaussian noise, and get 3 additional query voice files for each query voice file. The signal-to-noise ratio of the external query voice file is 10db, 20db, 30db. Original clean query voice file and the number of "hits" obtained by the three query voice files with white Gaussian noise are shown in Figure 3.7 and Figure respectively. As shown in 3.8. In Figure 3.7, the features of the first layer of CDBN are used, and in Figure 3.8, CDBN is used features of the second layer.

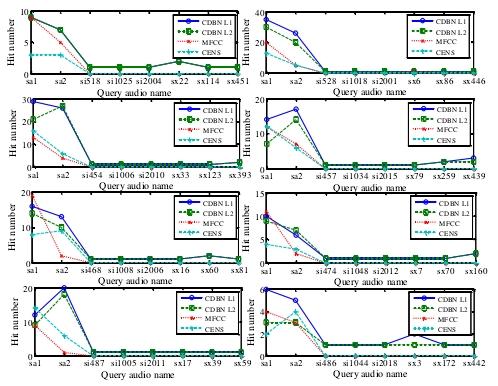


Figure 3.6 The number of hits for different query voice files under different characteristics

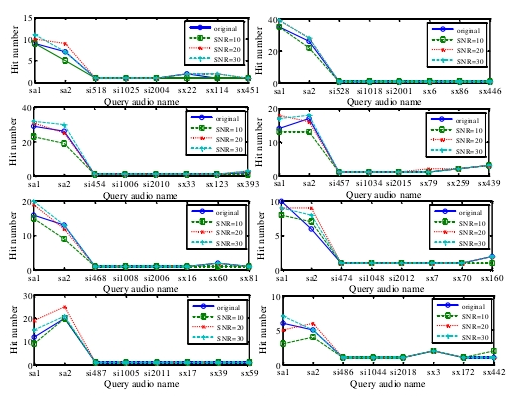


Figure 3.7 The number of hits on CDBN L1 for different SNR query voices

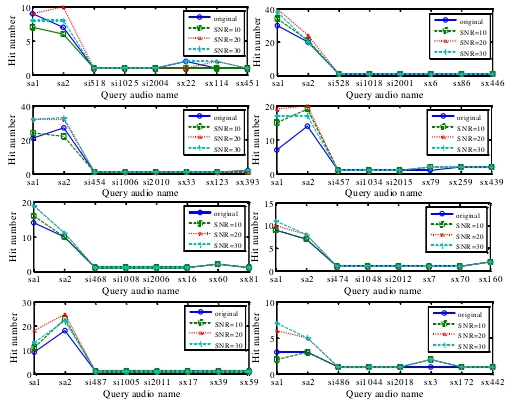


Figure 3.8 The number of hits on CDBN L2 for different SNR query voices

It can be seen from Figure 3.7 and Figure 3.8 that the number of “hits” in the query speech with white Gaussian noise added is not less than the corresponding original clean query voice file, whether it is using CDBN first-level features or using the second layer of CDBN features indicate the robustness of the CDBN-based voice matching algorithm to noise.

**3.4.2 Voice matching results on the music database**

This section also verifies the effectiveness of the proposed algorithm on a music database. The music database, collecting on the Internet, including the following songs: "Better man", "God is a girl", "Halo", "Rolling in the deep", "We will rock you", and "Yesterday once more", each song is performed by several singers sing. For each song, there are 3 query voice files with different lengths. Among all songs, among 3 query files the longest is 20 seconds to 30 seconds, the shortest is 2 seconds to 7 seconds, and the middle is 7 seconds to 17 seconds. For each the number of "hits" for each query voice file is shown in Table 3.2. Here, the length of time for each query file ( leng\_t ) and the number of occurrences (#) in the database are given in the second column. The name of the song from which the query voice clip is extracted it is given in the first column. Table 3.2 also shows that the performance of the voice matching algorithm based on CDBN far exceeds that based on voice matching algorithm with MFCC and CENS features.

**3.5 Summary**

This chapter proposes a new speech matching algorithm based on the features extracted by CDBN. Specifically, given a query voice fragments, voice matching needs to be automatically and effectively identified from the database all speech fragments with the same content. This chapter uses CDBN to extract speech features and is based on CDBN the extracted features propose an effective matching algorithm. In the TIMIT database and a collection from the Internet the experimental results on the simulated music database show that the algorithm based on CDBN significantly exceeds that based on MFCC and CENS feature voice matching algorithm, and the features extracted by CDBN are robust to white Gaussian noise.

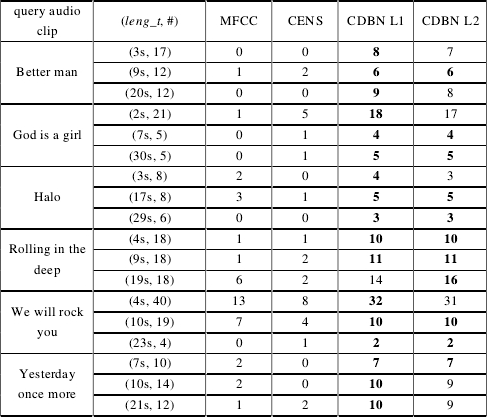


Table 3.2 Number of voice matching hits using different features

**Chapter 4 Multimodal Speech Recognition Based on Recurrent Neural Network**

Visual information plays an important role in improving the accuracy of speech recognition. To include in speech recognition visual information, this chapter proposes a multi-modal recurrent neural network model for combining audio and video modes voice recognition of state information. The multi-modal recurrent neural network consists of three parts: processing audio modal hearing the part deals with the visual part of the video modality and the fusion part that combines the information of the two modalities. The effectiveness of the model is tested in a speech recognition benchmark library based on audio and video modal information

AVletters came up to verify. The experimental results show the effectiveness and robustness of the multi-modal recurrent neural network.

**4.1 Introduction**

Humans recognize the speaker's voice through multi-modal information. In addition to auditory signals, visual information, for example movements such as lips and tongue play an important role in improving speech understanding. Use visual information (mainly recognizing the content of the speaker's speech by observing the movement of the lips is usually called lip reading. People with hearing impairment can use lip-reading techniques to understand the speech of others. Even for people with normal hearing, lip-reading technology can help them we improve speech comprehension, especially in noisy environments. The correlation between auditory information and visual information can be it is demonstrated by the McGurk effect [105], which indicates that conflicting auditory and visual stimuli may cause cognitive confusion.

The importance of visual information motivates researchers to combine visual information and auditory information in computer speech recognition systems. Combine information. Petajan [106] uses dynamic time warping on visual features extracted from open mouth shapes (dynamic time-warping) and found that voice recognition (Audio-Visual Speech Recognition (AVSR) system surpasses the recognition that relies solely on auditory information or visual information alone. Don't system. Goldschen [107] first applied HMMs to speech recognition based on visual information, which greatly improve the accuracy of speech recognition. Since then, researchers have proposed many methods for AVSR, including a representative work was done by Matthews et al. [108]. Matthews and others combined visual information and sound the signals are combined to realize the speech recognition of isolated letters AZ. The combination method they use is to combine each the output probabilities of the recognizer are combined linearly. In addition, they also proposed three lip reading models: Active Shape Model (ASM), Active Appearance Model (AAM), and multiscale spatial analysis (MSA), and use the HMM with continuous density from left to right as the classifier. Their experimental results show that sense information helps to improve the accuracy of speech recognition, especially for low signal-to-noise ratio (Signal-to-Noise rate, the voice signal under SNR) has a more obvious improvement effect.

Recently, the successful application of deep learning in speech processing has led researchers to apply deep learning to AVSR. Ngiam et al. [109] used deep autoencoders in multi-modal unsupervised feature learning, in AVletters the accuracy of lip reading classification on the database has been significantly improved. Huang and Kingsbury [110] constructed DBN in continue to extract auditory and visual features on digital recognition tasks to achieve robust speech recognition against noise. Their the experimental results show that the multi-modal DBN built on the intermediate features extracted by the single-modal DBN is compared with the base

the accurate auditory-visual system reduces the word error rate by 21%. Noda et al. [111] proposed a connectionist HMM is used for AVSR that is robust to noise. They first used a deep denoising autoencoder to obtain robustness against noise. Great voice features, and then use CNN to extract visual features from the original lip region image, and use the right the corresponding characteristics are used to train the auditory HMM and visual HMM, and finally through a multi-stream HMM, the auditory HMM combined with visual HMM. The experimental results on the isolated word recognition task show that when the two modes are combined when combined to express sound models, word recognition accuracy can be improved, especially for numbers with low signal-to-noise ratio. According to data, the lifting effect is more significant. Mroueh et al. [112] proposed a DNN using bilinear softmax layer structure, the network combines class-specific association information between audio modalities and video modalities. Moon et al. [113] a transfer deep learning (TDL) framework is proposed, which uses audio data to tune the network for video recognition it shows that the transferred audio modality improves the classification result of the target video modality.

However, few studies have applied RNN to AVSR, and RNN is suitable for processing sequence data. One of the degree learning models. Considering the sequence characteristics of auditory data and its corresponding visual data, this chapter proposes a multi-modal recurrent neural network model is used to solve the problem of speech recognition combining auditory and visual information. Versus the m-RNN model used to generate image captioning proposed by Mao et al. [114] is similar. This chapter the multi-modal recurrent neural network model consists of three parts: the auditory part, the visual part, and the fusion part. The auditory part uses RNN to learn feature expression from audio modal data; the visual part first uses CNN from extract features from video modal data, and then send these features as input to another RNN; fusion part the auditory part and the visual part are combined through a multi-mode layer.

**4.2 Recurrent neural network model**

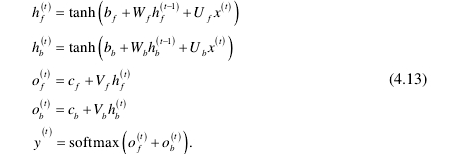
Recurrent neural network is a type of neural network used to process sequence data. Just like CNN is dedicated to processing for grid data like images, RNN is specially used to process sequence data x(1),…,x(r). First of all in this section expansion of the calculation graph to give an intuitive understanding of RNN, and then give the forward propagation and back propagation of a typical RNN derive the formula, and finally give the two most successful RNN structures: bidirectional RNN and LSTM.

**4.2.3 Bidirectional Recurrent Neural Network**

In many applications, researchers may wish to obtain the predicted output at the current time step based on the entire input sequence. Out y(t). For example, in the speech recognition task, the current speech needs to be recognized as the correct phoneme. The current speech recognition may depend on several subsequent phonemes, because the language between adjacent words the recognition of the current speech may depend on the following words: if the two solutions to the current word the explanations are acoustically reasonable, so the researcher needs to look forward (and backward) a few words to distinguish the two explanations. This phenomenon also exists in handwritten font recognition and many other sequence-to-sequence learning tasks.

In order to meet this demand, Schuster et al. proposed a bidirectional recurrent neural network (Bidirectional recurrent neural network, BRNN) [115]. BRNN has been extremely successful in applications where this demand exists, for example, handwritten font recognition [116], speech recognition [77], bioinformatics [117].

BRNN combines an RNN that starts from the start of the sequence and propagates forward in time and starts from the end of the sequence an RNN that propagates backward in time. Figure 4.4 shows a typical BRNN, where h(t) means edge time the state of the child RNN that propagates forward, g(t) represents the state of the sub-RNN propagating backward in time. This knot output unit o(t) rely on both the past and the future. The following formula describes a BRNN:



among them hf(t) and hb(t)  corresponds to the state of propagating the sub-RNN forward along time and propagating the sub-RNN backward along time. bf,Wf,Uf,Vf is the parameter that propagates the sub-RNN forward along time , bb,Wb,Ub,Vb is the backward pass along time broadcast the parameters of RNN.

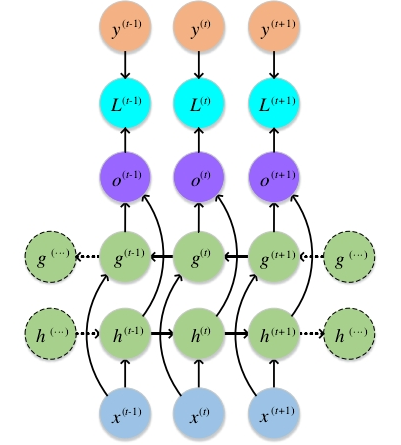


Figure 4.4 Calculation diagram of a typical two-way recurrent neural network

**4.3 Multimodal recurrent neural network model**

This chapter proposes to use multimodal recurrent neural network (multimodal RNN) to solve the problem of combining hearing and vision. Voice recognition problem. The structure of the multimodal RNN mentioned in this chapter is shown in Figure 4.6, including a for the auditory part that processes audio modal data, a visual part that processes video modal data, and a used to combine the fusion part of the two. The visual part of Multimodal RNN is a CNN layer and a double the bidirectional LSTM is the RNN (bidirectional LSTM RNN) layer of the component, and the auditory part is a the bidirectional LSTM RNN layer, both of which contain a weighted state layer, the fusion part is a multimodal layer and a softmax layer.

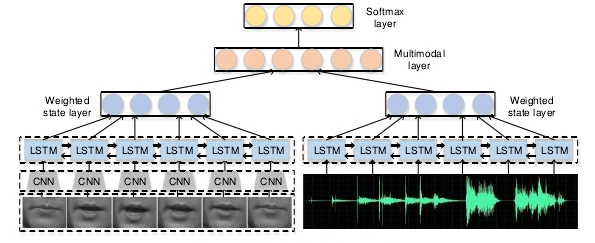


Figure 4.6 Structure diagram of multimodal recurrent neural network

Suppose the number of input video frames is Vn, the number of input audio frames is an multimodal RNN first uses the same CNN extract V the characteristics of n video frames, get Vn one-dimensional feature vectors. These feature vectors are then made is the input of a bidirectional LSTM RNN, the time step of the RNN is Vn. Similarly, these An audio frames are sent as input to another bidirectional LSTM RNN with a time step of An. use OV(t) and OA(t) respectively represent the output and audio of the bidirectional LSTM RNN of the video part at time step t part of the bidirectional LSTM RNN output, this section calculates the video partial weighted state layer OV according to the following formula and audio part weighted state layer OA:





among them WV(t)∈R(1≤t≤nV) and WA(t)∈R(1≤t≤nA). It is a trainable parameter.

After obtaining the output of the weighted state layer, this section calculates the multi-mode layer according to the traditional fully connected layer method and the output of the softmax layer. Hypothesis OV∈Rdv，OA∈RdA, The number of neurons in the multi-mode layer is dM , then this section calculates the output of the multi-mode layer according to the following formula



Among them WV(M)∈RdM×dV，WA(M)∈RdM×dA and b(M)∈RdM. It is the trainable parameter of the multi-mode layer.

The last layer of Multimodal RNN is the softmax layer, which calculates that the input sample belongs to each possible class probability:



among them W(S)∈RC×dM and b(S)∈RC是 softmaxIs the parameter of the softmax layer, and C represents the number of classes.

**4.3.1 Model variants**

In Figure 4.6, the auditory part, the visual part, and the fusion part are "bidirectional LSTM RNN". "CNN plus bidirectional LSTM RNN", and "multimodal layer". In addition to these three parts in addition to configuration, this section explores the variants of each part and replaces a part of the structure in Figure 4.6 with the corresponding variant a variant of the latter multimodal RNN.

For the auditory part, this section explores a simpler variant: one-way long and short-term memory recurrent neural network network (unidirectional LSTM RNN, when bidirectional is not clearly indicated below, all refer to unidirectional), it starts processing from the beginning of the audio sequence until the end of the audio sequence.

Similarly, for the visual part, this section explores the "CNN plus unidirectional LSTM RNN" variant. In order to show the effectiveness of RNN in processing sequence data, this section also explores CNN variants without LSTM RNN. For this CNN variant, this section uses an early fusion strategy: the input nV video frames as the CNN a convolutional layer has nV channel input. The CNN structure contains two convolutional layers, after each convolutional layer all follow a pooling layer, and finally a fully connected layer. However, in the "CNN plus LSTM RNN" variant, this section uses a simpler CNN with only one convolutional layer, one pooling layer, and one full connection layer.

For the fusion part, this section explores three variants: the first variant bases the visual part on the video modality according to the learned features as the initial state of the auditory part LSTM RNN network, the output probability of the auditory part is the output probability of the entire multimodal RNN; the second variant combines the output probability value of the auditory part and the visual part of the output probability values ​​are convexly combined, and the parameters of the convex combination are trainable; the last variant combines the visual part of the learned features are spliced ​​with the original audio modal data, and the spliced ​​data is used as auditory part of the model input.

**4.3.2 Model training**

In order to train multimodal RNN, this chapter uses the cross-entropy loss function and adds a regular term:



Where N represents the number of samples in the training set, y(i) represents the class label of the i- th sample, o(i) indicates that the model corresponds to y(i) the output probability, θ represents the parameters of the network. The goal of training is to minimize the loss function, which is differentiable. This chapter uses the BP algorithm to learn network parameters and implements it with TensorFlow.

**4.4 Summary**

This chapter proposes a multi-modal recurrent neural network model framework for combining audio modal data and video speech recognition of modal data. The visual part of the model used to process video modal data contains a CNN, followed by an LSTM RNN, the auditory part used to process audio modal data is an LSTM RNN, the two parts are fused through a multi-mode layer. The three parts have different variants, so you can the characteristics of the specific task select the best model configuration. The multimodal recurrent neural network model mentioned in this chapter can effectively include visual information in speech recognition tasks, and the recognition accuracy on the AVletters database is over compared with the previous algorithms, the recognition is accurate, especially for data with low signal-to-noise ratio. The rate has been greatly improved.

**Chapter 5 Summary and Outlook**

**5.1 Summary of the work of this article**

Deep learning is one of the most cutting-edge research issues in the field of artificial intelligence, and it has achieved amazing results in many fields. This article uses deep learning to solve two typical application problems in speech processing, namely speech matching and multi-mode state speech recognition. From the application point of view, voice matching and voice recognition are the key technologies of voice search. It is widely used in problems such as analysis and mining. Research on the application of deep learning in these two types of problems is of great importance. Military application value. In theory, speech matching and speech recognition are respectively unsupervised problems in speech processing. The study of deep learning models on these two types of problems has important academic value.

**5.1.1 Voice matching**

Aiming at the shortcomings of poor generalization ability of traditional speech matching algorithms, this paper proposes the use of convolution depth confidence for the first time the network establishes a voice matching model. Convolutional deep belief network combined with convolutional neural network can effectively handle high the advantages of unsupervised learning of dimensional data and deep belief networks can unsupervised extract features. Based on the binary features extracted by the convolutional deep belief network, this paper proposes a faster language tone feature matching algorithm. The experimental results show that compared with the traditional feature-based speech matching algorithm, the proposed voice matching algorithm greatly improves the hit rate of voice matching.

**5.1.2 Multi-modal speech recognition**

The traditional multi-modal speech recognition problem is handled by manually extracting auditory information features, visual perceive information characteristics, and then use a simple combination strategy to synthesize the recognition results of the two. Through this method the feature generalization ability of the design is poor, and the combination of voice mode and visual mode is low. This article focuses on audio modalities the sequence of data and video modal data, a multi-modal recurrent neural network framework is proposed for multi-modal language tone recognition. The framework includes an LSTM RNN for processing audio modal data, a CNN & LSTM RNN is used to process video modal data, and the two are fused through a multi-modal layer. These three parts are a variety of variants, according to the characteristics of different tasks, you can choose the best configuration. The experimental results show that based on the multi-modal speech recognition system of the neural network successfully integrates the two features of video and audio, effectively improving the language the accuracy of voice recognition, especially for data with low signal-to-noise ratio, has been greatly improved.

## 基于深度学习的两类典型语音处理问题研究

1. **绪论**

**1.1研究背景**

**1.1.1深度学习概念**

1981年诺贝尔医学/生理学奖颁发给了David Hubel，Torsten Wiesel，和Roger Sperry。前两位的主要研究成果是发现了视觉系统的信息处理机制，并发现大脑的可视皮层是分级的[1]，如图 1.1 所示。人的视网膜接收到的视觉信号，在 V1 区域是简单的视觉形式，如边、角等；V2、V4 区域将这些边角处理成中级的视觉形式，如特征组等；在 AIT 区域，大脑可以形成高级的对象描述，如脸、物体等。David和 Torsten 认为人的视觉功能是抽象和迭代。抽象就是把具体形象的元素，如原始的像素信息，抽象成有意义的概念。这些有意义的概念又会向上迭代，生成更加抽象、人类可以感知的概念。

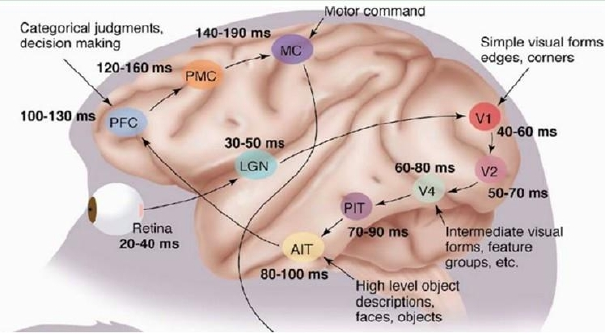


图 1.1 大脑可视皮层的分级性质[1]

深度学习(Deep Learning, DL)起源于人工神经网络的研究，是机器学习领域的一个新课题。深度学习模拟人脑的机制来理解图像、语音、文本等数据，其动机在于建立模拟人脑进行分析学习的神经网络。深度学习通过组合简单的概念表达更加复杂的概念，其识别人脸的过程也是抽象迭代的过程，如图 1.2 所示。输入的像素信息类似人类视网膜接收到的原始信号，第二层的边缘信息类似人脑 V1 区域的边缘检测，第三层的人脸的某一部分信息，类似人脑 V2 区域初始的形状检测，最后的整张人脸模型，类似人脑 V4 区域的高级视觉抽象。

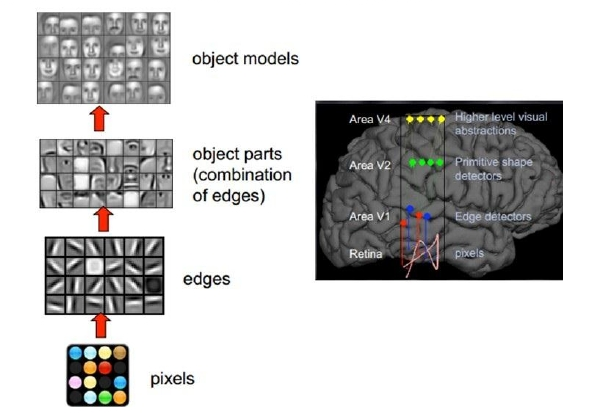


图 1.2 人脸识别过程[122]

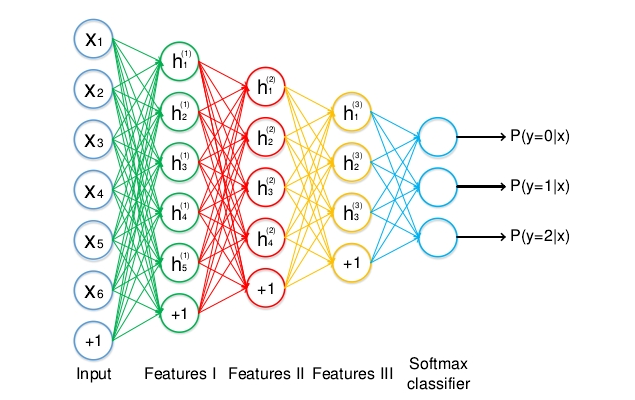


图 1.3 前馈深度神经网络示意图

一个典型的深度学习模型是前馈深度神经网络(Deep Neural Network, DNN)或称多层感知机(Multilayer Perceptron, MLP)，如图 1.3 所示，其中xi表示输入层，hi(1) hi(2) hi(3) 表示隐藏层，最后一层为隐藏层，表示输入属于每个类的概率。MLP是将输入集合映射到输出集合的数学函数。该函数通过组合很多简单的函数来得到。对于输入来说，可以认为使用不同的数学函数，就可以得到不同的输出表达。

**1.2基于深度学习的语音处理现状**

**1.2.1语音识别**

长期以来，语音识别的主导方法是高斯混合模型-隐马尔科夫模型 (Gaussian Mixture Model-Hidden Markov Model, GMM-HMM)，该方法基于上下文相关的GMM 和 HMM 的生成模型。神经网络曾经也是用于语音识别的流行方法，然而效果却不如 GMM-HMM。

2010 年学术界和工业界紧密合作后，深度学习开始在语音识别领域产生影响[70][71]。合作开始于音素识别任务，识别结果展示出 DNN 架构以及后续的卷积网络架构、递归网络架构的能力。他们的工作同时展示出从广泛使用的 MFCC 特征转向更低层的原始语音频谱特征的重要性。他们的合作在大词汇识别任务上也取得了良好效果[72]。DNN 在大词汇语音识别任务上取得成功的主要原因是：类似于GMM-HMM 中的语音单元，DNN 使用了大规模输出层结构。使用这种结构的原因是语音研究者期望利用上下文相关的音素建模技巧，这种技巧在 GMM-HMM 中十分有效；同时这种结构可以尽可能少地改变为 GMM-HMM 开发的高效解码器软件架构。DNN 在大词汇语音识别的工作也表明，如果可以利用大量的有标签数据，类似深度置信网络的预训练过程是不必要的。深度学习在语音识别领域的工业界和学术界均取得了成功，这一成功主要有三个因素：(1)与之前最好的 GMM-HMM系统相比，深度学习显著地降低了语音识别误差率；(2)由于使用音素作为 DNN 的输出，部署基于 DNN 的语音识别器只需要对解码器进行小部分的改变；(3)DNN强大的建模能力降低了语音识别系统的复杂性。

DNN-HMM 系统在语音识别上取得成功之后，研究者们提出了很多新架构和非线性单元用于提升语音识别准确率。

Yu 等人[73]通过使用双投影层和张量层来替换传统 DNN 中的一层或多层，提出了 DNN 的张量版本。双投影层将每个输入向量投影到两个非线性子空间。在张量层，两个子空间投影相互交互并联合地预测整个深度架构的下一层。研究者同时提出一种方法将张量层映射到传统的 sigmoid 层，因此张量层可以与 sigmoid 层以相似的方式进行训练。

时域卷积的思想源于延时神经网络(time-delay neural network, TDNN)，作为一种浅层网络应用于早期的语音识别。最近研究者使用深层卷积神经网络用于音素识别任务时，发现频率域的权值共享相较于时域的权值共享更有效[74]。一份研究报告也指出，卷积神经网络有助于大词汇连续语音识别任务，使用大量卷积核或特征图的多层卷积神经网络会有更大的性能提升[75]。Sainath 等人[76]探索了深度卷积神经网络的大量变体，发现在与若干种新方法结合时，深度卷积神经网络在一些大词汇语音识别任务上取得了最好的效果。

在语音识别任务中，最值得关注的深度结构是递归神经网络及其深度版本[77][78]。虽然 RNN 最早在音素识别中取得成功，然而由于训练 RNN 的复杂性，RNN难以扩展到更大的语音识别任务上。此后 RNN 的学习算法得到了提升，使用 RNN在一些任务上取得了更好的结果，尤其是使用双向 LSTM RNN[79]。

除了在用于语音识别的深度学习模型方面进行创新，大量的工作研究如何开发和实现更好的非线性单元。Sigmoid 函数和 tanh 函数是 DNN 中最常用的非线性函数，然而这两者均有局限性。例如，当神经元结点接近饱和时，误差函数相对于参数的梯度值很小，此时网络训练速度慢。为了克服 sigmoid 函数和 tanh 函数的缺点，Jaitly 和 Hinton[80]首次在语音识别中使用 ReLU。ReLU 表示的激励函数为 f(x)-max(0,x)。

另一种对语音识别有效的单元是 maxout 单元，它用于构建深度 maxout 网络[81]。深度 maxout 网络通过在固定数目的带权输入上进行最大值操作或 maxout 操作产生隐层激活值。这种操作与卷积神经网络中的最大值池化操作相同。这些最大值即为前一层的输出。此后，Zhang 等人[82]将 maxout 单元泛化成两种新的类型。第一种为使用 soft-max 函数替换原始 max 函数的 soft-max 单元，第二种是使用y=||x||p的 p 范数单元。实验表明，使用 p=2的 p 范数单元比 maxout 单元、tanh单元、ReLU 单元效果都好。

**1.2.2音频和音乐处理**

在音频和音乐处理领域，涉及深度学习的研究主要为音乐信号处理和音乐信息检索[87][88]。音乐音频信号是以音乐时间，而非真实时间组织的时间序列，其随着韵律和情感而变化。音乐音频信号的影响因素包括音乐传统、风格、作曲以及演绎。音乐音频信号具有复杂性和多变性，深度学习所提供的高级抽象非常适合于音乐音频信号的表征问题。

Lee 等人[89][90]为了处理高维的音频信号，使用结合卷积结构的受限玻尔兹曼机构建了卷积深度置信网络。卷积操作在隐层结点间共享时间维度上的权值，用于检测时间不变性特征。卷积深度置信网络已经应用于音频和语音数据的很多任务上，包括音乐艺术家和音乐流派的分类、说话人识别、说话人性别识别、音素分类等，并取得了不错的效果。

由于 RNN 强大的动态系统建模能力，最近研究者也将 RNN 应用于音乐处理应用中[91]，并使用 ReLU 作为激励函数，而不是 logistic 函数或 tanh 函数。RNN主要应用于从音乐中自动地识别和弦，这是音乐信息检索领域的活跃课题。实验结果表明，基于 RNN 的自动和弦识别系统与现有的最好方法相当。RNN 可以学习到基本的音乐属性，如时间连续性、和音、时间动态等。

深度学习也可以应用于基于内容的音乐推荐系统中[92]。自动音乐推荐已经成为日常生活中十分重要的技术。很多的推荐系统依赖于协同过滤，这种算法存在冷启动问题，即在没有可用数据时，算法就会失败。因此，协同过滤算法在推荐新歌和冷门歌曲方面不是很有效。深度学习使用潜在因素模型进行推荐，若无法从可用数据中获得潜在因素，就从音乐音频中预测潜在因素。实验结果表明，深度学习在基于内容的音乐推荐中可以取得很好的效果。

**1.3论文主要工作**

**1.3.1语音匹配**

语音匹配自动地从给定的语音数据库中检索出与查询语音片段具有相同内容的所有语音片段。语音匹配是一类基于内容的语音检索应用，在音乐检索、歌曲推荐、语音情报分析等方面广泛应用。同时语音匹配是一类无监督学习任务，适用于语音匹配的技术可以应用于机器学习领域其它的无监督学习任务，研究语音匹配算法具有重要的学术价值。因而语音匹配任务是一类典型的语音处理任务。

语音匹配的关键是语音特征的提取。针对传统语音特征提取算法泛化能力差的缺点，本文提出使用卷积深度置信网络进行语音特征的提取，并基于卷积深度置信网络提取的二值特征，提出一种快速的语音特征匹配算法。

**1.3.2多模态语音识别**

用于智能机器，如智能手机、家庭机器人、自动驾驶汽车的人机交互接口在日常生活中变得越来越普遍。对噪音鲁棒的语音识别是实现有效人机交互的关键。多模态语音识别被认为是鲁棒语音识别的有效解决方案之一。在人机交互系统中，机器除了可以接收操作者的语音信号，还可以观测到操作者的行为信息，如身体的移动、口型的变化，这些行为信息可以帮助机器识别操作者的语音信号，研究多模态语音识别在人机交互系统中具有重要的应用价值。此外，多模态语音识别是一种监督学习任务，涉及到多源信息的融合，具有重要的学术研究价值。多模态语音识别也是一种典型的语音处理任务。

本文基于递归神经网络，提出一种结合音频和视频的语音识别框架：多模态递归神经网络。其包含处理音频的听觉部分，处理视频的视觉部分，以及将听觉部分和视觉部分进行融合的部分。每个部分存在多种变体，可以根据具体任务的特性适当的配置网络架构。

**第二章 背景知识**

**2.1深度置信网络**

深度置信网络由深度学习著名学者 Geoffrey Hinton 于 2006 年提出[27]，为深度学习领域的重大进展。

DBN 为概率生成模型，包含若干层隐藏变量。隐藏变量通常为二值的，而可见单元可以为二值，也可以为实值。层内节点间没有连接。通常，每层的每个节点与相邻层的所有节点之间都是连接的，尽管可以构造更加稀疏连接的深度置信网络。 DBN 通过堆叠受限玻尔兹曼机构成。DBN 首先通过逐层贪婪训练策略初始化网络，然后利用期望的输出联合地调优网络的所有参数。DBN 的逐层贪婪训练策略有两个优点：一是可以为网络生成一个恰当的初值，因为不恰当的参数选择可能使得网络收敛到比较差的局部最优解，因此这在一定程度上解决了参数选择的困难；二是该训练过程是无监督的，因而不需要训练数据的类标。

**2.1.1受限玻尔兹曼机**

RBM 为两层网络，分别称为可见层V 和隐藏层 H 。可见层单元 v 和隐藏层单元 h 的联合概率 p 由能量函数 E 定义：

(2.1)

其中分割函数 Z 通过将所有可能的可见层与隐藏层对累加起来得到：

(2.2)

若可见层单元为二值的，定义能量函数 E ：

(2.3)

其中 ,vi ,hj 为可见层单元 i 和隐藏层单元 j 的状态， ,ai, bj为其偏置值，wij 为连接它们的权重值。若可见层单元为实值的，定义能量函数 E ：

(2.4)

RBM 通常使用对比散度[93]来训练。给定可见层单元的状态，隐藏层单元状态hj置为 1 的概率 p 为：

(2.5)

其中 为 sigmoid 函数1/(1+exp(-x)) 。隐藏层单元的二值状态确定以后，可以“虚构”可见层。该“虚构”可见层单元状态vi 置为 1 的概率 p 为：

(2.6)

隐藏层单元的状态会再次更新，以表示“虚构”可见层的特征。权值的更新w通过下面的公式：

(2.7)

其中 为学习率， 为原始数据可见层单元与隐藏层单元的乘积，为“虚构”数据可见层单元与隐藏层单元的乘积。偏置值的更新使用更加简化的学习规则，即不使用成对单元状态的乘积，而是使用单独单元的状态。

**2.2卷积神经网络**

卷积神经网络是最著名的深度学习模型之一，其在多种计算机视觉任务上取得了最好效果。一般的 CNN 结构如图 2.1 所示，图中展示了将 CNN 应用于图像识别，其包含一个输入层，两个卷积层 C1 和 C3，一个池化层 S2，一个全连接层，以及一个输出层。

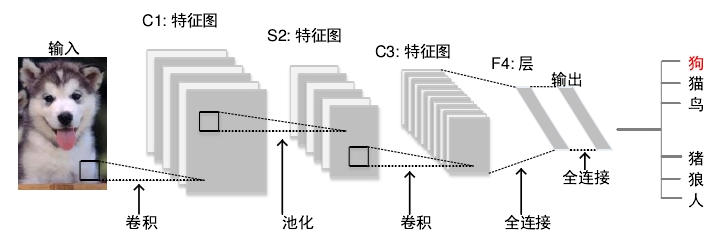


图 2.1 卷积神经网络结构

通常 CNN 包含三种类型的层，即卷积层、池化层、全连接层。不同类型的层扮演着不同的角色。接下来，本节首先介绍各种层的功能以及每种类型层的最近研究进展，然后给出 CNN 的训练算法，最后给出典型的 CNN 模型。

**2.2.1层的类型**

CNN 是分层的神经网络，其中卷积层和池化层交替连接，最后连接一些全连接层。本小节给出三种层的功能，并简要回顾对这些层的最近研究进展。

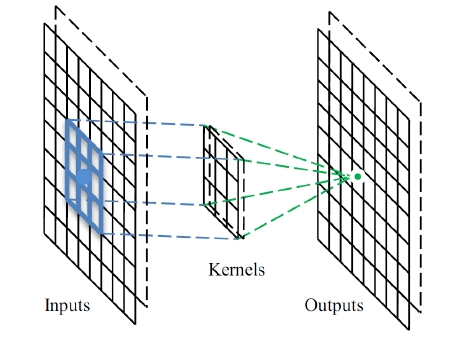


图 2.2 卷积层操作[123]

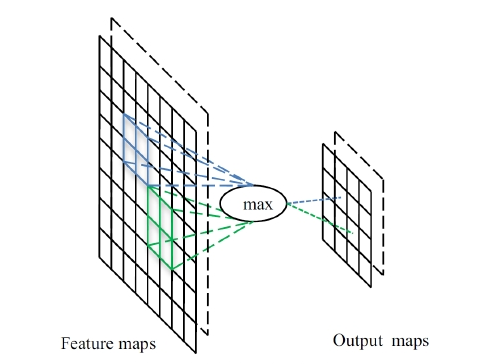


图 2.3 最大值池化层操作[123]

**2.2.1.1卷积层**

在卷积层，CNN 利用不同的卷积核与整幅输入图像和中间产生的特征图做卷积操作，并生成不同的特征图，如图 2.2 所示，其中两个卷积核与输入特征图做卷积操作生成两个输出特征图。 卷积操作有三个主要的优点：1)在同一个特征图中的权值共享机制减少了网络中参数的数量；2)局部连接学习到相邻像素的关联信息；3)对物体在图像中的位置具有不变性。

处理卷积层的一种方法是 Network in Network[94]，其主要思想是用一个小规模的多层感知机代替传统的卷积层。该多层感知机具有多个全连接层，并使用非线性激励函数，通过这种方式，用非线性神经网络替代了线性卷积核。该方法在图像分类任务上取得了较好的效果。

**2.2.1.2池化层**

通常，池化层跟在卷积层之后，其可以减少特征图的维度以及网络的参数数目。因为池化层的计算将相邻的像素考虑在内，所以与卷积层类似，池化层也具有偏移不变性。均值池化和最大值池化是最常用的策略。图 2.3 给出了最大值池化操作的例子。对于8\*8 的特征图，使用大小为 2\*2的池化操作子并以 2 为步长，则输出图减小到 4\*4。 对于最大值池化层和均值池化层，Boureau 等人[95]对它们的性能进行了详细的理论分析。Scherer 等人[96]对这两种操作进行了进一步的比较，并发现最大值池化层具有更快的收敛速率，可以选择更好的具有不变性的特征，并提高模型的泛化能力。最近几年 CNN 的实现中，大多数使用最大值池化策略。 池化层是三种层里研究的最多的。关于池化层，针对不同的目的有三种不同的策略。

**随机池化**

最大值池化的一个缺点是容易对训练集过拟合，使得其难以很好地泛化到测试集。为了解决这个问题，Zeiler 等人[97]提出了随机池化方法。通过在每个池化区域根据多项式分布函数随机地选取激活值，随机池化用一个随机过程替代传统的确定性的池化操作。随机池化操作等价于在标准的最大值池化操作中，输入图像有多个拷贝，每个拷贝具有局部的小形变。随机池化操作的随机特性有助于防止过拟合。

**空间金字塔池化**

CNN 通常要求输入图像为固定大小。这种限制可能降低任意尺寸大小的图像的识别准确率。为了消除这种限制，He 等人[98]使用空间金字塔层(Spatial pyramid pooling, SPP)替换了传统 CNN 的最后一个池化层。SPP 层可以从任意图像中提取固定长度的特征，为处理不同尺寸、高宽比的图像提供了灵活的解决方案。将 SPP 应用于 CNN，可以提升 CNN 的性能。

**形变约束池化**

在计算机视觉，尤其是物体检测任务中，处理形变是一个基础的挑战。最大值池化和均值池化可以用于处理形变，但无法学习局部物体的形变约束和几何模型。为了更有效地处理形变，Ouyang 等人[99]提出了形变约束池化层，其通过学习视觉模式的形变来丰富深度模型。形变约束池化层可以用来替换传统 CNN 的任一池化层。

**2.2.1.3全连接层**

CNN 中，最后一个池化层后通常跟着若干个全连接层。全连接层将二维特征图转换为一维特征向量，以便于后续的特征表示。全连接层类似于多层感知机，如第一章图 1.3 所示。 全连接层执行与传统 DNN 相似的计算，其参数占据了 CNN 全部参数的 90%。通常不会改变全连接层的结构。然而在迁移学习中，一种常见的做法是保留在ImageNet 数据集上学习到的网络参数，并使用两个新的全连接层替代已训练好网络的最后一个全连接层，使得新的网络适合于新的视觉识别任务。全连接层的缺点是其包含大量参数，给 CNN 的训练带来巨大的计算开销。

**2.2.3 经典模型**

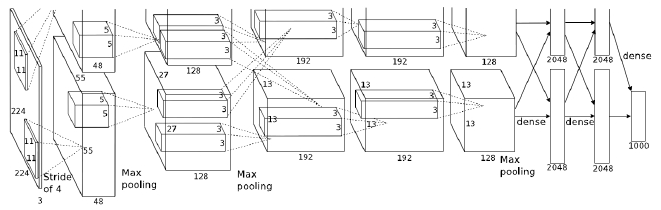


图 2.4 AlexNet[48]架构图

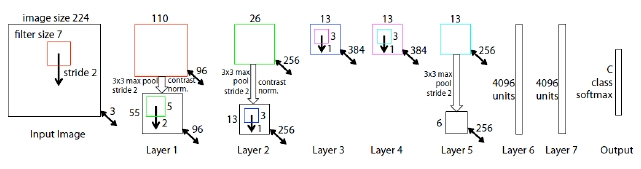


图 2.5 Clarifai[100]架构图

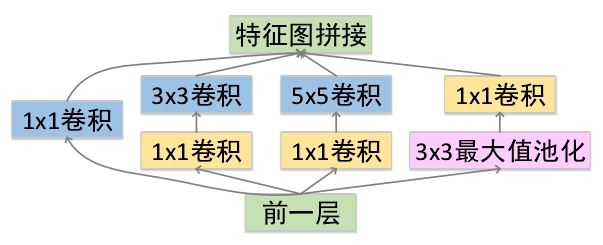


图 2.6 GoogLeNet[50]的 Inception 模块

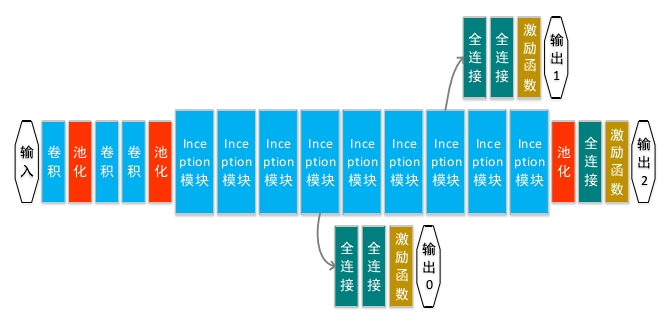


图 2.7 GoogLeNet[50]架构图

AlexNet[48]是一个著名的 CNN 架构，其包含 5 个卷积层(第一、二、五个卷积层后有池化层)和 3 个全连接层，65 万个神经元，6000 万个参数，输入为固定大小( 224224 )的图像，如图 2.5 所示。AlexNet 在 ImageNet 数据集上进行训练，将一百二十万张高分辨率训练图像分到 1000 个不同的类中。AlexNet 使用了多种泛化技巧，包括数据增加，dropout 等。该网络赢得了 ILSVRC2012 比赛，图像分类的Top-5 错误率为 15.3%。 在 2013 年，Zeiler 等人[100]提出了一种新颖的可视化技术，使得他们可以分析AlexNet 中间特征层的内部工作原理，并基于此提出了一种新的 CNN 架构，称为Clarifai。Clarifai 与 AlexNet 类似，包含 5 个卷积层和 3 个全连接层，但两个网络中卷积核的大小与步长有所区别，Clarifai 的架构如图 2.6 所示。Clarifai 在 Image Net分类任务上超越了 AlexNet，并赢得了 ILSVRC2013 比赛，在 ImageNet 分类任务上的 Top-5 错误率降到了 11.2%。 不同于 AlexNet，GoogLeNet[50]通过添加更多的卷积层增加了网络的深度，并在所有层中使用很小的卷积核。GoogLeNet 减少了全连接层的层数，因为全连接层占据了 CNN 的大部分参数，这使得 GoogLeNet 的网络参数只有五百万。GoogLeNet 引入了 Inception 模块，如图 2.6 所示，以使用不同大小的卷积核。GoogLeNet 包含 9 个 Inception 模块，整个网络共有 21 个卷积层和 1 个全连接层，其架构如图 2.7 所示。GoogLeNet 赢得了 ILSVRC2014 比赛，在 ImageNet 图像分类任务上的 Top-5 错误率降到了 6.67%。

**第三章 基于卷积深度置信网络的语音匹配**

**3.1 引言**

随着多媒体和网络技术的发展，人们可以自由地播放和下载感兴趣的音乐文件。因而，有效简便地搜索方式对获取音乐文件及相关信息显得尤为重要。目前，大多音乐搜索引擎机制依赖音乐文件的标注信息。这些标注信息通常为演作者、歌曲名、专辑名或其他信息。最为典型的信息就是使用作曲家名字为关键词，如“Beethoven”(如图 3.1(a)所示)。这些音乐搜索引擎往往要求用户要掌握基本的音乐信息，即传统文本标注检索系统只能搜索用户已经知道的内容。为克服基于标注的检索系统的局限性，研究者提出了基于音乐标签的检索系统。音乐标签通常包含音乐类型或者流派，以及歌曲中所传达的情绪、节拍等信息。图 3.1(b)展示了对“Beethoven”的标签云，其最显著的标签是“古典”。当用户关注特定音乐属性的音乐时，标签信息就变得十分重要。但是，获取这些标签信息是件既耗力又耗时的大工程。特别是，获得可信、一致、有音乐意义的标签，还需音乐方面的专业知识指导。最近，基于音乐本身内容的检索策略(如图 3.1(c)所示)展示出巨大的潜能。基于音乐内容的检索策略不依赖任何文本标注或手工生成的标签，而是仅仅基于语音本身。一个典型的场景是用户在商场里录下一段自己喜欢的歌曲，却不知歌曲名和演唱者等信息，该用户便可以使用录下的歌曲片段作为查询语音检索出该首歌曲。 基于内容的语音检索和分析是信号处理领域十分具有挑战性的工作。在基于内容的语音检索任务中，语音匹配旨在从给定的语音数据库中检索出与查询语音片段具有相同内容的所有语音片段。语音匹配的一个典型场景是同一首歌曲有多个演唱版本，例如给定 Queen 乐队演唱的“We will rock you”的一个 10 秒时长的歌曲片段，语音匹配的目标是找出数据库中与这 10 秒片段具有相同内容的所有片段，包括 Queen 乐队演唱的“We will rock you”中重复这 10 秒片段的歌曲片段，也包括其他歌手如 Britney Spears, Russian Red, The Park 等演唱的“We will rock you”中与这 10 秒片段具有相同内容的片段。

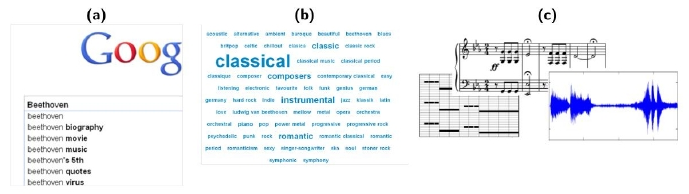


图 3.1 不同的检索方式

传统的语音匹配依赖于语音鉴定的结果。语音鉴定从语音数据库中鉴定出包含查询语音片段的完整语音。目前，语音鉴定算法取得了很大的进展，即使查询语音存在噪音、压缩痕迹、或是轻微的时间扭曲。然而这些算法不能处理强烈的非线性时间扭曲或波谱偏差。 为了克服基于语音鉴定的语音匹配算法的缺陷，Kurth 和 Muller 等人[101][102]提出了色度能量归一化统计特征 CENS 用于语音匹配。基于 CENS 特征的语音匹配算法流程为：首先将查询语音片段和数据库中的所有语音文件转换成 CNES 特征序列，然后计算查询语音 CENS 特征序列与数据库中语音文件 CENS 特征序列的任意子序列的距离函数，最后通过距离函数，确定出数据库中与查询语音最匹配的语音片段。该算法的关键是 CNES 特征的提取。CENS 特征首先从语音信号中提取色度特征(chroma feature)，然后时间上模糊色度特征以获得对局部节拍差异的鲁棒性，最后标准化上一步得到的特征以实现对动态偏差的不变性。CENS 特征对动态、音质等的偏差以及局部节拍差异具有一定程度的鲁棒性。然而，CNES 特征为手工设计的特征，其泛化能力较差。实验表明，基于 CENS 特征的语音匹配算法在很多情况下难以取得令人满意的效果。 为了克服 CENS 特征的缺陷，本章使用 CDBN 无监督地从语音数据中提取特征以用于语音匹配。CDBN 结合了 CNN 和 DBN 两者的优点，可以无监督地从高维数据中提取特征，与语音匹配任务的特点相契合。进一步，基于 Kurth 和 Muller等人提出的语音匹配算法流程，本章提出了更加高效的语音特征匹配算法，该算法针对二值特征而设计。本章在 TIMIT 数据库和一个模拟的音乐数据库上评估基于 CDBN 的语音匹配算法，实验结果表明了所提算法的有效性。

**3.2 卷积深度置信网络模型**

当将 DBN 应用于计算机视觉任务时，其缺点是没有考虑输入图像的二维结构信息。为了克服这个问题，Lee 等人提出了 CDBN[103]。由于引入了卷积操作和池化操作，CDBN 可以有效地扩展到高维图像数据。下面本节首先分析 CDBN 的基本构造块：卷积受限玻尔兹曼机(Convolutional Restricted Boltzmann Machine, CRBM)，然后分析类似于传统 CNN 中池化操作的概率最大池化。

**3.2.1 卷积受限玻尔兹曼机**

为了写法方便，本节做出一些简化假设。假设 CDBN 的输入为Nv × Nv的图像，并假设所有的单元均为二值的(可以直接扩展到实值可见层单元)。本节用\*表示卷积操作，用·表示对应元素相乘然后相加，即 A·B=trATB，并用 A表示将数组 A 水平并垂直翻转。

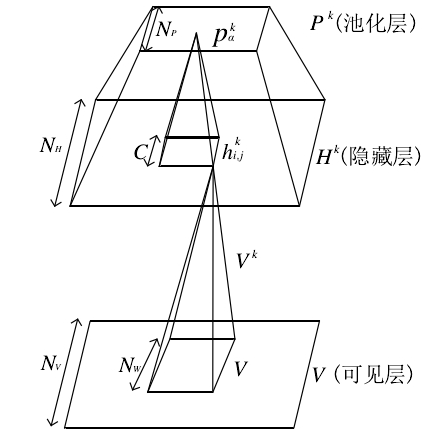


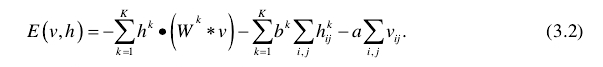
图 3.2 带有概率最大池化层的卷积受限玻尔兹曼机

直观上，CRBM 与 RBM 类似，但是 CRBM 中，可见层和隐藏层之间的权值在输入图像的所有位置是共享的。基本的 CRBM 包含两层：输入层V 和隐藏层H (对应于图 3.2 的低两层)。输入层包含Nv × Nv个二元单元，隐藏层包含 K 组，每一组包含Nh × Nh个二元单元，共有 NH2K 个隐藏层单元。 K 组隐藏层的每一组关联一个Nw × Nw 的过滤器(Nw=Nv –NH +1)；同一组内所有的隐藏层单元共享过滤器权值。此外，每一组隐藏层单元hk 对应一个偏置值bk，而所有的可见层单元v共享一个偏置值 a 。

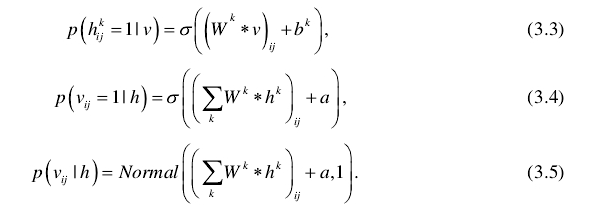
定义能量函数 E ：



使用先前定义的操作符，



类似 RBM，可以得到下面的条件概率分布 p ：

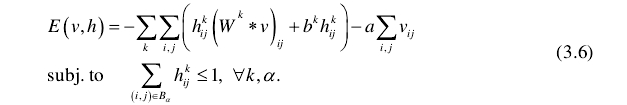


其中 为sigmoid 函数，Normal 为正态分布函数，式 3.4 对应于二元可见层单元，式 3.5 对应于实值可见层单元。

**3.2.2 概率最大池化**

通常，越高层的特征检测子，越需要更大的输入区域的信息。具有平移不变性的表征，如 CNN，经常交替地包含两种类型的层：特征检测层和池化层。检测层的响应通过将特征检测子和前一层进行卷积计算得到，而池化层通过一个常量因子减少检测层的表征信息，更具体地，池化层的每个单元计算检测层一个小区域内的最大值。通过最大池化操作减少表征信息，一方面可以使得高层表征对输入的局部小偏移具有不变性；另一方面可以大幅减少计算压力。最大池化操作只是为前馈结构设计的，因此为了在生成模型(既包含自上向下的推断，又包含自底到顶的推断)中包含类似于最大池化的操作，Lee 等人[103]提出了概率最大池化。 考虑如图 3.2 中包含一个可见层V ，一个特征检测层(隐藏层) H ，一个池化层P 的模型。检测层和池化层均有 K 组单元，池化层的每一组有Np × Np 个二元单元。对于每一个 k，池化层 Pk 以因子C 沿着每一维减少检测层 Hk 的表征，这里C 是一个小整数，例如 2 或 3。也就是说，检测层HK 被划分成大小为 C × C 的小块，每个小块被连接到池化层的唯一一个二元单元pαk，因此NP=NH/C。正式地，可以定义Bα=｛( I , j ) | hij∈ block α｝。

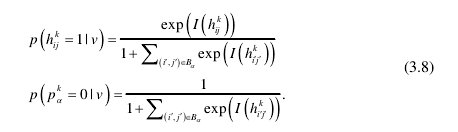
块 Bα中的检测层单元和池化层单元 pα以一种单一的可能性相连接，并满足如下的限制：检测层至多一个单元是打开状态，并且池化层状态是打开的当且仅当有一个检测层单元是打开的。等价地，可以认为这些C2+1个单元作为一个单一的随机变量，该随机变量可以取C2+1个个可能的值： C2 个取值对应于每个检测层单元处于打开状态，另外一个取值对应于所有单元均处于关闭状态。 Lee 等人按照下式定义简化的概率最大池化 CRBM 的能量函数 E ：



据此能量函数讨论给定可见层V 时，如何采样特征检测层 H 和池化层 P 。第 k 组从可见层V 接收到的自底向上的信号为：



Lee 等人以每个块输入的多项式函数独立地为每个块进行采样。假设 hijk 是包含在块 中的隐藏层单元(即（i，j）属于Bα)，打开单元 hijk 导致的能量增加为 *-I（hijk）*，条件概率由下式给出：



给定隐藏层 H 后，对可见层V 的采样与 3.2.1 节相同。

类似于 DBN 通过堆叠 RBM 构成，CDBN 通过交替堆叠 CRBM 和概率最大池化层构成，得到具有偏移不变性的生成模型。CDBN 也是通过贪婪地逐层训练方式来优化：即当前给定层训练完毕后，其权值不再改变，该层的激活值将作为输入送到下一层。除了图像数据，CDBN 在语音处理任务上也取得了成功[90]。

**3.3 算法流程**

基于 CDBN 的语音匹配算法流程图如图 3.3 所示。首先，对语音数据进行预处理，主要是对语音数据进行降维，然后使用 CDBN 从预处理后的语音数据中提取特征，最后基于 CDBN 提取的语音特征，执行语音特征匹配算法。

**3.3.1 预处理**

在预处理阶段，首先将时域信号转换为频谱数据，频谱具有 20ms 的窗口大小，并具有 10ms 的重叠。然而该频谱的维度比较高，因此本节使用 PCA 白化降低频谱的维度。最终输入到 CDBN 的语音数据包含nc 通道的一维向量，其中nc 是 PCA成分的数目(在本章的实验中设置为 80)。

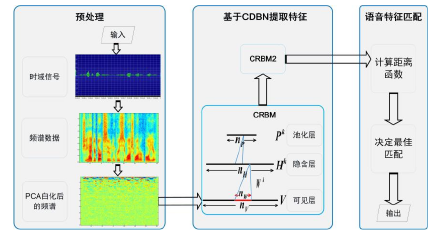


图 3.3 基于卷积深度置信网络的语音匹配算法流程图

**3.3.2 语音特征提取**

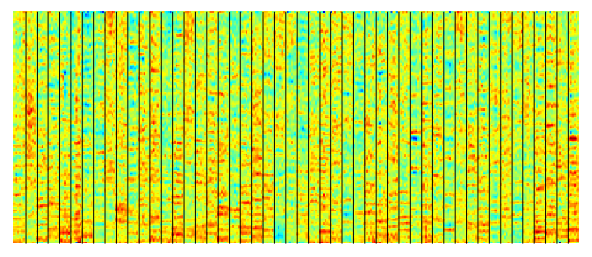


图 3.4 卷积深度置信网络第一层随机选取的特征检测子的可视化

本章使用 CDBN 提取语音特征。不过要注意的是，常见的 CDBN 用于处理二维图像数据，本章使用 CDBN 处理一维的语音向量。 用于提取语音特征的 CDBN 由两个 CRBM 构成。两个 CRBM 的配置相同：均包含 300 个卷积核，卷积核的大小均为 6，概率最大池化比(局部邻居大小)均为3。 CDBN 学习到的特征可以通过可视化来展示。本节通过将每个特征检测子乘以 PCA 白化的逆来可视化第一层特征检测子，如图 3.4 所示。可以看到，不同的特征检测子学习到了不同的语音特征。一些特征检测子在频谱的高频区域能量比较集中，一些特征检测子在低频区域比较集中，还有一些特征检测子比较均匀地分布在整个频谱带。

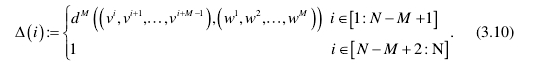
**3.3.3 语音特征匹配**

使用 CDBN 提取的语音特征，根据 Kurth 和 Muller 等人所提的语音匹配框架，本节提出了一种更加高效的语音特征匹配算法。语音数据库包含语音记录集，这些语音可能是不同人说的同一句话或是不同歌手演唱的同一首歌曲。通过将所有单独的语音记录拼接起来，本节使用一个大文件 D 来表示语音数据库，并使用Q 来表示短的查询语音片段。所有的语音记录文件和查询语音文件经过预处理和特征提取后，均具有C 通道。定义ΩC:=｛x=(x1,…,xc)T|xi∈｛0,1｝｝。在特征提取阶段，将拼接而成的语音文件D和查询片段Q转换成二值特征序列。本节用F[D]=(v1,…,vN)和F[Q]=(w1,…,wM)表示这些特征序列，其中对所有的n ∈[1:N] 有vn∈ΩC ，对所有的m∈[1:M] 有wm属于ΩC 。

语音匹配的目标是在拼接成的语音文件D中识别出所有的与查询语音Q具有相同内容的片段。本节把特征序列F[Q]和F[D]中包含M个连续向量的子序列进行比较。具体地，令X=(x1,…,xM)∈ΩCM ，Y=(y1,…,yM)∈ΩCM并定义 ,其中x∈ΩC，y∈ΩC，的定义为：



然后，本节设定dM (x，y):=1-,其中X∈, Y∈。注意dM的取值范围为 [0,1] ∈ℜ ，并且当X和Y重合时取值为0。接下来，针对F[D] 和F[Q]，本节定义距离函数△：[1:N]→[0,1]:



△(i)描述了F[Q]和F[D]中一个子序列的距离，该子序列起于位置i并包含M各连续向量。

△的计算如图 3.5 所示，以步长为 1 滑动F[D] 中大小为 M 的窗口，并将该窗口与查询语音片段F[Q] 相比较。

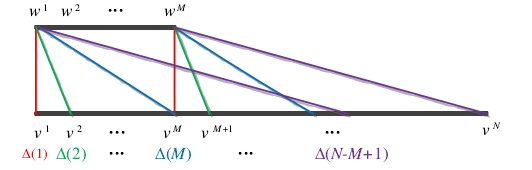


图 3.5 距离函数计算图示

Q 在 D 中的最佳匹配可以通过连续地选择距离函数△的最小值来决定：在第一步，确定最小化△的索引i∈[1: N]，指明对应于特征序列(vi,…,vi+M-1)的语音片段是最佳匹配。然后最佳匹配附近的长度为 M 的序列被排除在进一步考虑范围之内，以避免后续的匹配与当前的最佳匹配具有较大的重叠。在第二步，确定最小化修改后的距离函数的特征索引，得到第二个最佳匹配结果。该过程会一直进行下去，直到检索到预定义数目的匹配语音片段或者检索到的匹配语音与查询语音的距离超过特定的阈值。

特征匹配算法的主要开销为 dM 的计算。由于 CDBN 提取的语音特征为二值的，利用此特性，本节重写dM 的计算公式如下：



该公式中的异或计算相较于 Kurth 和 Muller 等人所提特征匹配算法的向量内积运算速度更快。

**3.4 实验**

本节使用 MATLAB 实现基于 CDBN 的语音匹配算法，并在 TIMIT 数据库和一个模拟音乐数据库上评估其性能，该音乐数据库包含从互联网上收集的歌曲。为了训练 CDBN，本节使用无标签语音数据库 TIMIT[104]。

**3.4.1 TIMIT 数据库上语音匹配结果**

TIMIT 数据库共包含 6300 个句子，有来自美国 8 个主要方言区(dr1—dr8)的630 个说话人，每个说话人 10 句话。在每个方言区，存在 20 多个说话人讲述的 2个句子(sa1，sa2)，仅有一个说话人讲述的句子(si--)，和 1 到 5 个说话人讲述的句子(sx--)。在每个方言区，本节把所有的语音文件拼接成一个大的语音文件，得到8 个拼接起来的语音文件，每个方言区域一个。在每个方言区域，本节选择 8 个短的语音文件(sa1, sa2, 3 个 si--, 3 个 sx--)做为查询语音，并在对应的拼接起来的语音文件中检索这些查询语音。每个方言区域选择的查询语音文件以及这些查询语音文件在对应的拼接语音文件中重复的次数如表 3.1 所示。

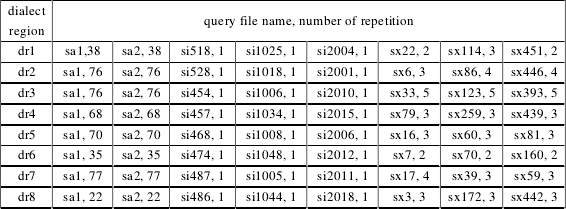


表 3.1 查询语音文件以及每个查询语音重复的次数

接下来讨论所提算法的语音匹配结果。对于一个在数据库中有nq个重复出现的查询语音文件，语音特征匹配过程会返回nq个指明查询片段出现位置的索引。如果算法预测的位置正好是查询片段出现的位置，称其为一个“命中”。不同语音文件的“命中”数目如图 3.5 所示，其中每个子图展示一个方言区域：左上角为 dr1，右上角为 dr2，右下角为 dr8。除了基于 CENS 特征的结果，本节同时将基于 CDBN的算法与基于 MFCC 特征的语音匹配算法相比较。这里，“CDBN L1”为第一层的特征，“CDBN L2”为第二层的特征。

从图 3.6 可以看出基于 CDBN 的语音匹配算法远远超过基于 MFCC 和 CENS特征的语音匹配算法，不论是对在数据库中具有 20 个多重复的查询语音，还是对只有有限个重复的查询语音。具体地，对于“si--”查询语音文件，基于 CDBN 的匹配算法可以成功地在对应的语音文件中定位到所有的查询文件，但基于 MFCC 和CENS 特征的算法无法成功地定位到这些查询文件；对于“sx--”查询语音文件，基于 CDBN 的方法可以定位到这些查询语音文件出现的部分位置，然而基于 MFCC和 CENS 特征的算法却一个位置也定位不到；对于 sa1 和 sa2，基于 CDBN 的算法对大多数查询语音文件可以定位到更多的查询语音出现的位置。 为了验证基于 CDBN 的语音匹配算法对噪音的鲁棒性，本节对查询语音文件添加白高斯噪声，针对每个查询语音文件得到 3 个额外的查询语音文件，这些额外的查询语音文件的信噪比分别为 10db, 20db, 30db。原始的干净的查询语音文件以及 3 个添加了白高斯噪声的查询语音文件所取得的“命中”数目分别如图 3.7 和图3.8 所示。在图 3.7 中，使用的是 CDBN 第一层的特征，在图 3.8 中，使用的是 CDBN第二层的特征。

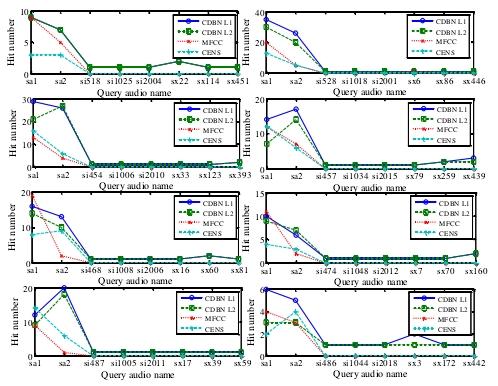


图 3.6 不同查询语音文件在不同特征下得到的命中数

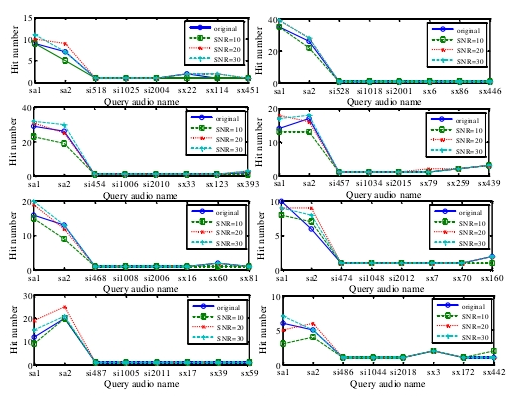


图 3.7 不同 SNR 查询语音在 CDBN L1 上得到的命中数

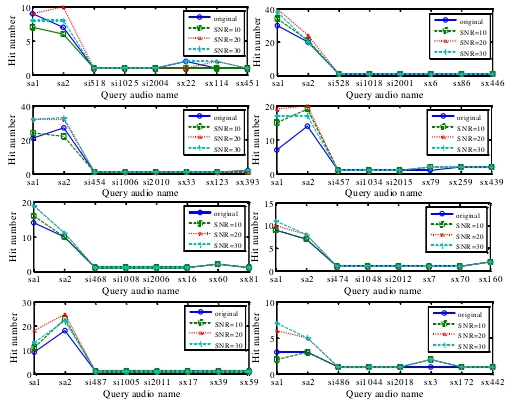


图 3.8 不同 SNR 查询语音在 CDBN L2 上得到的命中数

从图 3.7 和图 3.8 中可以发现，添加了白高斯噪音的查询语音的“命中”数目不少于对应的原始干净的查询语音文件，不论是使用 CDBN 第一层特征还是使用CDBN 第二层特征，表明了基于 CDBN 的语音匹配算法对噪音的鲁棒性。

**3.4.2 音乐数据库上的语音匹配结果**

本节同时在一个音乐数据库上验证所提算法的有效性。该音乐数据库，收集于互联网，包含下面的歌曲：“Better man”, “God is a girl”, “Halo”, “Rolling in the deep”, “We will rock you”, 和 “Yesterday once more”，每首歌曲由若干个歌手所演唱。对每首歌，存在 3 个不同长度的查询语音文件。所有歌曲中，3 个查询文件中最长的为 20 秒到 30 秒，最短的为 2 秒到 7 秒，中间的为 7 秒到 17 秒。针对每一个查询语音文件的“命中”数目如表 3.2 所示。这里，每个查询文件的时间长度(leng\_t)以及在数据库中出现的次数(#)在第二列给出。该查询语音片段所摘自的歌曲名字在第一列给出。表 3.2 同样表明基于 CDBN 的语音匹配算法性能远远超出基于MFCC 和 CENS 特征的语音匹配算法。

**3.5 小结**

本章基于 CDBN 提取的特征提出了一个新的语音匹配算法。具体地，给定一个查询语音片段，语音匹配需要从数据库中自动地、有效地识别出与查询语音具有相同内容的所有的语音片段。本章使用 CDBN 来提取语音特征，并基于 CDBN提取的特征提出一种有效的匹配算法。在 TIMIT 数据库和一个从互联网上收集的模拟音乐数据库上的实验结果表明：基于 CDBN 的算法显著地超过基于 MFCC 和CENS 特征的语音匹配算法，并且 CDBN 提取的特征对白高斯噪声具有鲁棒性。

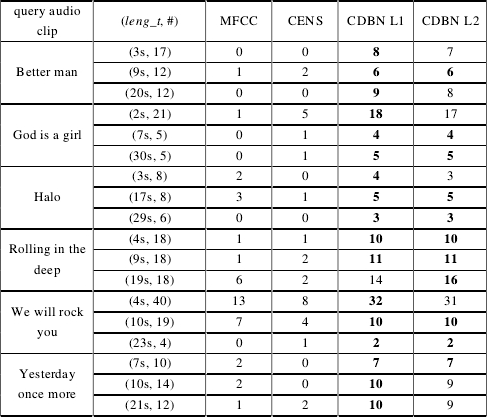


表 3.2 使用不同特征的语音匹配命中数

**第四章 基于递归神经网络的多模态语音识别**

视觉信息在提升语音识别准确率上扮演着重要角色。为了在语音识别中包含视觉信息，本章提出了一种多模态递归神经网络模型用于结合音频和视频两种模态信息的语音识别。该多模态递归神经网络包含三个部分：处理音频模态的听觉部分，处理视频模态的视觉部分，以及将两种模态的信息进行结合的融合部分。模型的有效性通过在一个基于音频和视频两种模态信息的语音识别基准测试库AVletters 上来验证。实验结果表明了多模态递归神经网络的有效性和鲁棒性。

**4.1 引言**

人类通过多模态的信息来识别说话人的语音。除了听觉信号，视觉信息，例如嘴唇和舌头的运动，在提升语音理解方面扮演着重要角色。使用视觉信息(主要是通过观察嘴唇的移动)来识别说话人的讲话内容通常称为读唇。听力受损的人可以用读唇技术来理解他人的讲话。即使是听力正常的人，读唇技术也可以帮助他们提升语音理解力，尤其是在嘈杂的环境下。听觉信息和视觉信息的关联可以通过 McGurk 效应[105]来展示，该效应表明具有冲突的听觉刺激和视觉刺激可能导致认知混乱。

视觉信息的重要性激励研究者在计算机语音识别系统中将视觉信息和听觉信息结合起来。Petajan[106]对从张开的口型中提取的视觉特征使用动态时间规整(dynamic time-warping) 并发现结合听觉和视觉信息的语音识别 (Audio-Visual Speech Recognition, AVSR)系统超过单独依赖听觉信息或单独依赖视觉信息的识别系统。Goldschen[107]首先将 HMMs 应用于基于视觉信息的语音识别中，极大地提升了语音识别的准确率。此后，研究人员提出了很多方法用于 AVSR，其中具有代表性的一个工作是 Matthews 等人[108]完成的。Matthews 等人将视觉信息和声音信号结合起来以实现对孤立字母 A-Z 的语音识别，他们使用的结合方法是将每个识别器的输出概率进行线性组合。此外，他们还提出了三种读唇模型：Active Shape Model (ASM), Active Appearance Model (AAM),和 multiscale spatial analysis (MSA)，并使用从左至右的、连续密度的 HMM 作为分类器。他们的实验结果表明增加视觉信息有助于提升语音识别的准确率，尤其是对于低信噪比(Signal-to-Noise rate, SNR)下的语音信号，提升效果更加明显。

最近，深度学习在语音处理上的成功应用使得研究人员将深度学习应用于AVSR。Ngiam 等人[109]在多模态无监督特征学习中使用深度自编码机，在 AVletters数据库上读唇分类准确率有了显著提升。Huang 和 Kingsbury[110]构建了 DBN 在连续数字识别任务上提取听觉和视觉特征，以实现对噪音鲁棒的语音识别。他们的实验结果表明构建于单模态 DBN 提取的中级特征之上的多模态 DBN，相较于基准的听觉-视觉系统，字误差率减少了 21%。Noda 等人[111]提出了一个连接主义的HMM 用于对噪音鲁棒的 AVSR。他们首先使用深度降噪自编码机来获得对噪音鲁棒的语音特征，然后使用 CNN 从原始的嘴唇区域图像中提取视觉特征，并使用对应的特征分别训练听觉 HMM 和视觉 HMM，最后通过一个多流 HMM 将听觉 HMM和视觉 HMM 结合起来。在孤立字识别任务上的实验结果表明，当将两种模态结合起来表达声音模型时，字识别准确率可以得到提升，尤其是对于低信噪比的数据，提升效果更加显著。Mroueh 等人[112]提出了一种使用双线性 softmax 层的 DNN结构，该网络结合了音频模态和视频模态之间特定于类的关联信息。Moon 等人[113]提出了一种迁移深度学习(TDL)框架，其使用音频数据来调优用于视频识别的网络，并表明迁移的音频模态提升了目标视频模态的分类结果。

然而，很少有研究将 RNN 应用于 AVSR，而 RNN 为适于处理序列数据的深度学习模型之一。考虑到听觉数据及其对应的视觉数据的序列特性，本章提出了一种多模态递归神经网络模型用于解决结合听觉和视觉信息的语音识别问题。与Mao 等人[114]提出的用于生成图像标题(image captioning)的 m-RNN 模型类似，本章的多模态递归神经网络模型包含三个部分：听觉部分，视觉部分，以及融合部分。听觉部分使用 RNN 从音频模态数据中学习特征表达；视觉部分首先利用 CNN 从视频模态数据中提取特征，然后将这些特征作为输入送到另一个 RNN；融合部分通过一个多模层将听觉部分和视觉部分结合起来。

**4.2 递归神经网络模型**

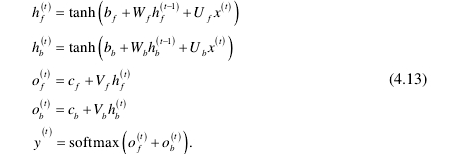
递归神经网络是用于处理序列数据的一类神经网络。就像 CNN 专门用于处理类似图像这样的网格型数据，RNN 专门用于处理序列数据x(1),…,x(r)。本节首先通过展开计算图给出 RNN 的直观理解，然后给出典型 RNN 的前向传播和反向传播推导公式，最后给出目前最成功的两种 RNN 结构：双向 RNN 和 LSTM。

**4.2.1 双向递归神经网络**

在很多应用中，研究者可能希望根据整个输入序列得到当前时间步的预测输出y(t)。例如，语音识别任务中需要将当前的语音识别为正确的音素，由于共同发音的影响当前语音的识别可能依赖于后续的若干个音素，由于相邻单词之间的语言学关联当前语音的识别可能依赖于后续的几个单词：如果对当前单词的两种解释从声学上都合理，那么研究者需要向前(并向后)看几个单词以区分这两种解释。这种现象，在手写字体识别和其他的很多序列-到-序列学习任务中也存在。

为了满足这种需求，Schuster 等人提出了双向递归神经网络(Bidirectional recurrent neural network, BRNN)[115]。BRNN 在存在这种需求的应用上极为成功，例如手写字体识别[116]，语音识别[77]，生物信息学[117]。

BRNN 结合了从序列起点出发沿时间向前传播的一个 RNN 和从序列终点出发沿时间向后传播的一个 RNN。图 4.4 展示了一个典型的 BRNN，其中h(t)表示沿时间向前传播的子 RNN 的状态，g(t)表示沿时间向后传播的子 RNN 的状态。这种结构使得输出单元o(t)既依赖于过去，又依赖于未来。下面的公式描述了一个 BRNN：



其中hf(t)和hb(t)分别对应于沿时间向前传播子 RNN 和沿时间向后传播子 RNN 的状态，bf,Wf,Uf,Vf为沿时间向前传播子 RNN 的参数，bb,Wb,Ub,Vb为沿时间向后传播子 RNN 的参数。

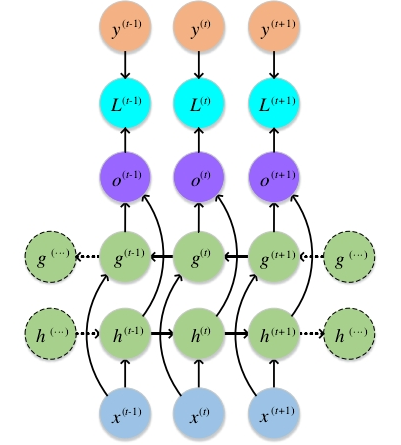


图 4.4 典型双向递归神经网络的计算图

**4.3 多模态递归神经网络模型**

本章提出使用多模态递归神经网络(multimodal RNN)来解决结合听觉和视觉的语音识别问题。本章所提的 multimodal RNN 的结构如图 4.6 所示，包含一个用于处理音频模态数据的听觉部分，一个处理视频模态数据的视觉部分，以及一个用于结合两者的融合部分。Multimodal RNN 的视觉部分为一个 CNN 层和一个双向的 LSTM 为构件的 RNN (bidirectional LSTM RNN)层，听觉部分为一个bidirectional LSTM RNN 层，两者后面均包含一个加权状态层(weighted state layer)，融合部分为一个多模层(multimodal layer)和一个 softmax 层。

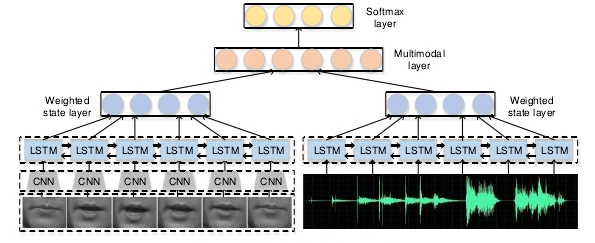


图 4.6 多模态递归神经网络结构图

假设输入视频帧数为Vn ，输入音频帧数为An 。Multimodal RNN 首先使用同一个 CNN 提取Vn 个视频帧的特征，得到nV 个一维的特征向量。这些特征向量然后作为一个 bidirectional LSTM RNN 的输入，该 RNN 的时间步长为nV 。类似地，这些An 个音频帧作为输入送入另一个 bidirectional LSTM RNN， 其时间步长为An 。用OV(t) 和OA(t) 分别表示在时间步 t ，视频部分的 bidirectional LSTM RNN 的输出和音频部分的 bidirectional LSTM RNN 的输出，本节按照下式计算视频部分加权状态层OV和音频部分加权状态层OA：





其中WV(t)∈R(1≤t≤nV) 和WA(t)∈R(1≤t≤nA)为可训练的参数。

在获得加权状态层的输出之后，本节按照传统的全连接层的方式计算多模层和 softmax 层的输出。假设OV∈Rdv，OA∈RdA，多模层神经元的个数为dM ，则本节按照下式计算多模层的输出OM ：



其中WV(M)∈RdM×dV，WA(M)∈RdM×dA和b(M)∈RdM是多模层可训练的参数。

Multimodal RNN 的最后一层为 softmax 层，该层计算输入样本属于每个可能类的概率：



其中W(S)∈RC×dM和b(S)∈RC是 softmax 层的参数， C 表示类的数目。

**4.3.1 模型变体**

在图 4.6 中，听觉部分、视觉部分、融合部分分别是“bidirectional LSTM RNN”， “CNN plus bidirectional LSTM RNN”，和“multimodal layer”。这三个部分除了这种配置外，本节探索每个部分的变体以及将图 4.6 结构的某一部分使用相应变体替换后的 multimodal RNN 的变体。

对于听觉部分，本节探索一个更加简单的变体：单向长短时记忆递归神经网络(unidirectional LSTM RNN，下面未明确指出 bidirectional 时，均指 unidirectional)，其从音频序列的起点开始处理，直到音频序列的终点。

类似地，对于视觉部分，本节探索“CNN plus unidirectional LSTM RNN”变体。为了表明 RNN 处理序列数据的有效性，本节也探索没有 LSTM RNN 的 CNN 变体。对于该 CNN 变体，本节使用一种早融合策略：将输入的nV 个视频帧作为 CNN 第一个卷积层具有nV 通道的输入。该 CNN 结构中包含两个卷积层，每个卷积层后面均跟着一个池化层，最后为一个全连接层。然而在“CNN plus LSTM RNN”变体中，本节使用一个结构更加简单的 CNN，其只有一个卷积层、一个池化层、一个全连接层。

对于融合部分，本节探索三种变体：第一种变体将视觉部分基于视频模态数据学习到的特征作为听觉部分 LSTM RNN 网络的初始状态，听觉部分的输出概率为整个 multimodal RNN 的输出概率；第二种变体将听觉部分的输出概率值和视觉部分的输出概率值进行凸组合，凸组合的参数是可训练的；最后一种变体将视觉部分学习到的特征与原始音频模态数据拼接起来，并将拼接起来的数据作为听觉部分模型的输入。

**4.3.2 模型的训练**

为了训练 multimodal RNN，本章采用交叉熵损失函数，并加上一个正则项：



其中 N 表示训练集中样本的数目，y(i)表示第 i 个样本的类标，o(i)表示模型对应于y(i)的输出概率，θ表示网络的参数。训练的目标是最小化损失函数，该损失函数是可微的。本章使用 BP 算法来学习网络参数，并用 TensorFlow 来实现。

**4.4 小结**

本章提出了一个多模态递归神经网络模型框架用于结合音频模态数据和视频模态数据的语音识别。该模型用于处理视频模态数据的视觉部分包含一个 CNN，其后跟着一个 LSTM RNN，用于处理音频模态数据的听觉部分为一个 LSTM RNN，这两个部分通过一个多模层进行融合。三个部分均有不同的变体，因而可以根据具体任务的特性选择最佳的模型配置。本章所提的多模态递归神经网络模型可以有效地在语音识别任务中包含进视觉信息，在 AVletters 数据库上的识别准确率超过了以往的算法，尤其是对于信噪比较低的数据，相较于以往的算法，识别准确率得到较大提升。

**第五章 总结**

**5.1 本文工作总结**

深度学习是人工智能领域最前沿的研究问题之一，在很多领域取得了惊人效果。本文使用深度学习解决语音处理中的两种典型应用问题，即语音匹配和多模态语音识别。从应用上讲，语音匹配与语音识别是语音搜索的关键技术，在情报分析与挖掘等问题中广泛应用，研究深度学习在这两类问题上的应用具有重大的军事应用价值。从理论上讲，语音匹配和语音识别分别是语音处理中的无监督问题和监督问题，研究这两类问题上的深度学习模型具有重要的学术价值。

**5.1.1 语音匹配**

针对传统语音匹配算法泛化能力差的缺点，本文首次提出使用卷积深度置信网络建立语音匹配模型。卷积深度置信网络结合了卷积神经网络可以有效处理高维数据以及深度置信网络可以无监督学习的优点，能够无监督地从高维语音数据中提取特征。基于卷积深度置信网络提取的二值特征，本文提出一种更快速的语音特征匹配算法。实验结果表明，相较于基于传统特征的语音匹配算法，本文所提的语音匹配算法大大提高了语音匹配命中率。

**5.1.2 多模态语音识别**

传统的多模态语音识别问题的处理思路是，首先手工提取听觉信息特征、视觉信息特征，然后以一种简单的结合策略综合两者的识别结果。通过这样的方法设计的特征泛化能力差，语音模态与视觉模态的结合程度低。本文针对音频模态数据与视频模态数据的序列性，提出一种多模态递归神经网络框架用于多模态语音识别。该框架包含一个 LSTM RNN 用于处理音频模态数据，一个 CNN & LSTM RNN 用于处理视频模态数据，两者通过一个多模层进行融合。这三个部分分别有多种变体，针对不同任务的特性，可以选择最佳的配置。实验结果表明，基于递归神经网络的多模态语音识别系统，成功融合视频和音频两种特征，有效提高语音识别准确率，尤其是对于信噪比较低的数据，语音识别准确率得到较大提升。

参考文献：

[1] Hubel D H, Wiesel T N. Receptive fields of single neurons in the cat's striate cortex [J]. The Journal of Physiology, 1959, 148(3): 574-591.

[27] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets [J]. Neural computation, 2006, 18(7): 1527-1554.

[48] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [C]//Advances in Neural Information Processing Systems. 2012: 1097-1105.

[50] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions [C]//IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[70] Deng L, Hinton G, Kingsbury B. New types of deep neural network learning for speech recognition and related applications: An overview [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2013: 8599-8603.

[71] Hinton G, Deng L, Yu D, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups [J]. IEEE Signal Processing Magazine, 2012, 29(6): 82-97.

[72] Dahl G E, Yu D, Deng L, et al. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2012, 20(1): 30-42.

[73] Yu D, Deng L, Seide F. The deep tensor neural network with applications to large vocabulary speech recognition [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2013, 21(2): 388-396.

[74] Abdel-Hamid O, Mohamed A, Jiang H, et al. Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2012: 4277-4280.

[75] Sainath T N, Mohamed A, Kingsbury B, et al. Deep convolutional neural networks for LVCSR [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2013: 8614-8618.

[76] Sainath T N, Kingsbury B, Mohamed A, et al. Improvements to deep convolutional neural networks for LVCSR [C]//IEEE Workshop on Automatic Speech Recognition and Understanding. 2013: 315-320.

[77] Graves A, Mohamed A, Hinton G. Speech recognition with deep recurrent neural networks [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2013: 6645-6649.

[78] Pascanu R, Gulcehre C, Cho K, et al. How to construct deep recurrent neural networks [J]. ArXiv preprint arXiv: 1312.6026, 2013.

[79] Graves A, Jaitly N, Mohamed A. Hybrid speech recognition with deep bidirectional LSTM [C]//IEEE Workshop on Automatic Speech Recognition and Understanding. 2013: 273-278.

[80] Jaitly N, Hinton G. Learning a better representation of speech soundwaves using restricted boltzmann machines [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2011: 5884-5887.

[81] Miao Y, Metze F, Rawat S. Deep maxout networks for low-resource speech recognition [C]//IEEE Workshop on Automatic Speech Recognition and Understanding. 2013: 398-403.

[82] Zhang X, Trmal J, Povey D, et al. Improving deep neural network acoustic models using generalized maxout networks [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2014: 215-219.

[87] Humphrey E J, Bello J P. Rethinking automatic chord recognition with convolutional neural networks [C]//IEEE International Conference on Machine Learning and Applications. 2012, 2: 357-362.

[88] Humphrey E J, Bello J P, LeCun Y. Moving beyond feature design: deep architectures and automatic feature learning in music informatics [C]//The International Society for Music Information Retrieval. 2012: 403-408.

[89] Lee H, Grosse R, Ranganath R, et al. Unsupervised learning of hierarchical representations with convolutional deep belief networks [J]. Communications of the ACM, 2011, 54(10): 95-103.

[90] Lee H, Pham P, Largman Y, et al. Unsupervised feature learning for audio classification using convolutional deep belief networks [C]//Advances in Neural Information Processing Systems. 2009: 1096-1104.

[90] Lee H, Pham P, Largman Y, et al. Unsupervised feature learning for audio classification using convolutional deep belief networks [C]//Advances in Neural Information Processing Systems. 2009: 1096-1104.

[91] Boulanger-Lewandowski N, Bengio Y, Vincent P. Audio chord recognition with recurrent neural networks [C]//The International Society for Music Information Retrieval. 2013: 335-340.

[92] Van den Oord A, Dieleman S, Schrauwen B. Deep content-based music recommendation [C]//Advances in Neural Information Processing Systems. 2013: 2643-2651.

[93] Hinton G E. Training products of experts by minimizing contrastive divergence [J]. Neural Computation, 2002, 14(8): 1771-1800.

[95] Boureau Y L, Ponce J, LeCun Y. A theoretical analysis of feature pooling in visual recognition [C]//The 27th International Conference on Machine Learning. 2010: 111-118.

[96] Scherer D, Müller A, Behnke S. Evaluation of pooling operations in convolutional architectures for object recognition [C]//International Conference on Artificial Neural Networks. 2010: 92-101.

[97] Zeiler M D, Fergus R. Stochastic pooling for regularization of deep convolutional neural networks [J]. ArXiv preprint arXiv: 1301.3557, 2013.

[98] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[99] Ouyang W, Luo P, Zeng X, et al. DeepID-Net: multi-stage and deformable deep convolutional neural networks for object detection [J]. ArXiv preprint arXiv: 1409.3505, 2014.

[100] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks [C]//European Conference on Computer Vision. 2014: 818-833.

[101] Müller M, Kurth F, Clausen M. Audio matching via chroma-based statistical features [C]//The International Society for Music Information Retrieval. 2005: 288-295.

[102] Kurth F, Muller M. Efficient index-based audio matching [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2008, 16(2): 382-395.

[103] Lee H, Grosse R, Ranganath R, et al. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations [C]//The 26th International Conference on Machine Learning. 2009: 609-616.

[104] Fisher W M, Doddington G R, Goudie-Marshall K M. The DARPA speech recognition research database: specifications and status [C]//DARPA Workshop on Speech Recognition. 1986: 93-99.

[105] McGurk H, MacDonald J. Hearing lips and seeing voices [J]. Nature, 1976, 264: 746-748.

[106] Petajan E D. Automatic lipreading to enhance speech recognition (speech reading) [D]. University of Illinois at Urbana-Champaign, 1984.

[107] Goldschen A J, Garcia O N, Petajan E. Continuous optical automatic speech recognition by lipreading [M]//Motion-Based Recognition. Springer Netherlands, 1997: 321-343.

[108] Matthews I, Cootes T F, Bangham J A, et al. Extraction of visual features for lipreading [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(2): 198-213.

[109] Ngiam J, Khosla A, Kim M, et al. Multimodal deep learning [C]//The 28th International Conference on Machine Learning. 2011: 689-696.

[110] Huang J, Kingsbury B. Audio-visual deep learning for noise robust speech recognition [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2013: 7596-7599.

[111] Noda K, Yamaguchi Y, Nakadai K, et al. Audio-visual speech recognition using deep learning [J]. Applied Intelligence, 2015, 42(4): 722-737.

[112] Mroueh Y, Marcheret E, Goel V. Deep multimodal learning for audio-visual speech recognition [C]//IEEE International Conference on Acoustics, Speech and Signal Processing. 2015: 2130-2134.

[113] Moon S, Kim S, Wang H. Multimodal transfer deep learning for audio visual recognition [J]. ArXiv preprint arXiv: 1412.3121, 2014.

[114] Mao J, Xu W, Yang Y, et al. Deep captioning with multimodal recurrent neural networks (m-rnn) [C]//International Conference on Learning Representation. 2015.

[115] Schuster M, Paliwal K K. Bidirectional recurrent neural networks [J]. IEEE Transactions on Signal Processing, 1997, 45(11): 2673-2681.

[116] Graves A, Liwicki M, Fernández S, et al. A novel connectionist system for unconstrained handwriting recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(5): 855-868.

[117] Baldi P, Brunak S, Frasconi P, et al. Exploiting the past and the future in protein secondary structure prediction [J]. Bioinformatics, 1999, 15(11): 937-946.

[122] Andrew Ng. Machine Learning and AI via Brain Simulations [R]. <http://forum.stanford.edu/events/2011/2011slides/plenary/2011plenaryNg.pdf>.