

# Classification Of Electroencephalograph Signals By Convolutional Neural Network

---

**Lin Jiaxin**

**Thesis for Master of Science in Computer Science**

**Submission date: 03 JUN 2017**

**Supervisor: Børre Bang**

**Adviser: Tatiana Kravets , Dazi Li**

**Faculty of Engineering Science and Technology**

**Department of Computer Science and Computational Engineering**

**UiT The Arctic University of Norway**

## Abstract

This thesis presents a brief introduction to epilepsy diagnosis use convolution neural network. The datasets create through Electroencephalograph(EEG) signal. Electroencephalograph(EEG) is one of the most commonly signals recorded from humans like electrocardiogram(ECG). In this thesis give a short introduce about the EEG signal. Because EEG recordings are important in the diagnosis of epilepsy. But there is not a good way to classification these EEG signal. This paper aims to develop epileptic EEG signal classification method.

In this 10 years, deep learning is very popular. Almost everyone heard of AlphaGo. In the AlphaGo also use convolution neural network. This is a very mysterious method. You just put data to train. Then you can get the learning accuracy. I create a CNN network to classification the normal and Epilepsy patients.

The best classification accuracy can be good after on the test tftrecord data.

**Keywords:** EEG, CNN, classification, feature extraction, electroencephalograph signal

## Acknowledgements

I would like to express gratitude and thank to:

- My supervisor - Brre Bang, who has been a great advisor and tell me the dataset is important. I really thank him.
- My adviser -Tatiana Kravetc and Dazi Li.
- My family and classmates and all the people, who were always with me and gave me a good advice during my studies in UIT.

*Lin jiaxin*

## List of Abbreviations and Symbols

- EEG - Electroencephalograph
- CNN - Convolutional Neural Network
- BCI - brain-computer interface
- BPNN - Back-Propagation Neural Networks
- M-ELM - multiplicative extreme learning machine

# Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgements</b>	<b>ii</b>
<b>List of Abbreviations and Symbols</b>	<b>ii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Status . . . . .	2
1.3 Research Method . . . . .	3
1.4 Goals And Work plan . . . . .	4
1.5 Thesis Organization . . . . .	5
<b>2 Background Theory</b>	<b>6</b>
2.1 EEG signal basic knowledge . . . . .	6
2.1.1 EEG signal . . . . .	6
2.1.2 Neural oscillations types . . . . .	7
2.1.3 Brain Wave . . . . .	8

---

2.2	Epilepsy . . . . .	9
2.2.1	Epilepsy syndromes . . . . .	10
2.3	Convolutional Neural Network . . . . .	16
2.3.1	Why CNN for image . . . . .	16
2.3.2	what is Convolution in math . . . . .	19
2.3.3	CNN - Convolution . . . . .	20
2.3.4	CNN - Max pooling . . . . .	21
2.3.5	CNN - Flatten . . . . .	21
2.3.6	CNN - convolution vs fully connected . . . . .	22
<b>3</b>	<b>Epilepsy data</b>	<b>24</b>
3.1	Data Selection . . . . .	24
3.2	Image data . . . . .	25
3.3	DataSets . . . . .	28
<b>4</b>	<b>Experiments and Analysis</b>	<b>31</b>
4.1	Model . . . . .	31
4.2	Experimental Design . . . . .	32
4.3	Experimental Result . . . . .	35
<b>5</b>	<b>Conclusion And Future</b>	<b>37</b>
5.1	Overview . . . . .	37
5.2	Future Work . . . . .	38

<b>A. System Instructions</b>	<b>39</b>
<b>B. Show Project Output</b>	<b>40</b>
<b>Bibliography</b>	<b>40</b>

# Chapter 1

## Introduction

This chapter aims to introduce this thesis. This includes 5 sections. First one give a brief introduction of why I choose this task. In the section 2 showing the status of the EEG signal classification. The research method in section 1.3. Next, the goals of thesis in section 1.4. In the end give an overview of this thesis structure.

### 1.1 Background

Epilepsy is one of the most common neurological disorders. Approximately 1% of the worlds population has epilepsy, and up to 5% of people may have at least one seizure during their lifetime.(Flint, G. 1989)

The Medical Department of Neurology technique reviewed has been developing by leaps and bounds in recent years. Such as CT, MRI, SPECT and DSA, not only can see the shape of the brain structure, but also to understand the function of the brain. Which provides scientific evidence for the cause of epilepsy. But in the diagnosis of epilepsy these can not replace electroencephalography, because the basis of epilepsy is the abnormal discharge of brain cells, only EEG can find this abnormal brain activity. The electroencephalography (EEG) is a recording of electrical activity along the scalp generated by the cerebral cortex nerve cells of



the brain.(Chen, J.,2015) Electrical impulses from the nerves in the head can be recorded by using EEG.

In Chapter 2, provide basic information about EEG. Although EEG can see some of the brain signal, but we can not accurately understand the special significance of these signals. we can only be analyzed according to the relevant test from animal. Although technology is developing very fast, we still require experts to manually check the EEG. Therefore, Develop an automatic device to diagnose brain waves is a good choice.

In Chapter 2, a detailed discussion about EEG is provided. Although EEG signals provide a great deal of information about the brain, research in classification and evaluation of these signals is limited. Even today the EEG is often examined manually by experts. Therefore, there is an ever-increasing need for developing automatic classification techniques to evaluate and diagnose neurological disorders.(Siuly. 2012)

In this project, the task of epilepsy diagnosing in medicine by classification of EEG signals is considered.

## 1.2 Status

In recent years, various computerized analytical methods have been developed to extract relevant information from EEG records and to identify different types of EEG data. From the literature, most of the reported methods have a limited success rate.(Siuly. 2012) On the other hand, ECG automatic diagnostic software has been relatively successful. We can get experience from ECG.

The main steps involved in extracting meaningful information from EEG signal include signal processing, feature extraction and feature classification.(Rahman et al.,2012 and Sun et al.,2014 and Amin et al.,2015)

Any good implementation of such BCI or AI systems requires an efficient feature extraction scheme, capable of extracting relevant information from the EEG signals and resulting in sat-

isfactory classification performance. Although optimal classification algorithms could be used to improve the classification accuracy, the selection of inadequate features could still lead to poor classification performance (according to the general machine learning principle garbage in-garbage out). Therefore, extracting relevant and meaningful features from EEG signals represents a crucial task which needs to be performed before the classification process.

Here presents some methods for extracting features from EEG signals. Each method has specific advantages and disadvantages.(Al-Fahoum A S, Al-Fraihat A A, 2014)

- Fast Fourier Transform (FFT) Method
- Wavelet Transform (WT) Method
- Eigenvectors
- Time-Frequency Distributions
- Autoregressive Method

## 1.3 Research Method

Convolutional neural network is a type of feed-forward artificial neural network. The most popular use these networks is for image processing.

There are many ways to extract features. Here is an informed research for extracting the feature. From the figure 1, Through the process of the sequential method, the features of EEG signals are extracted as the inputs of M-ELM(Li D, Xie Q, Jin Q, et al.,2016).

- Load EEG signal figure
- Prepare training and test image sets
- Daubechies 4 wavelet transform
- phase space construction
- Feature extraction

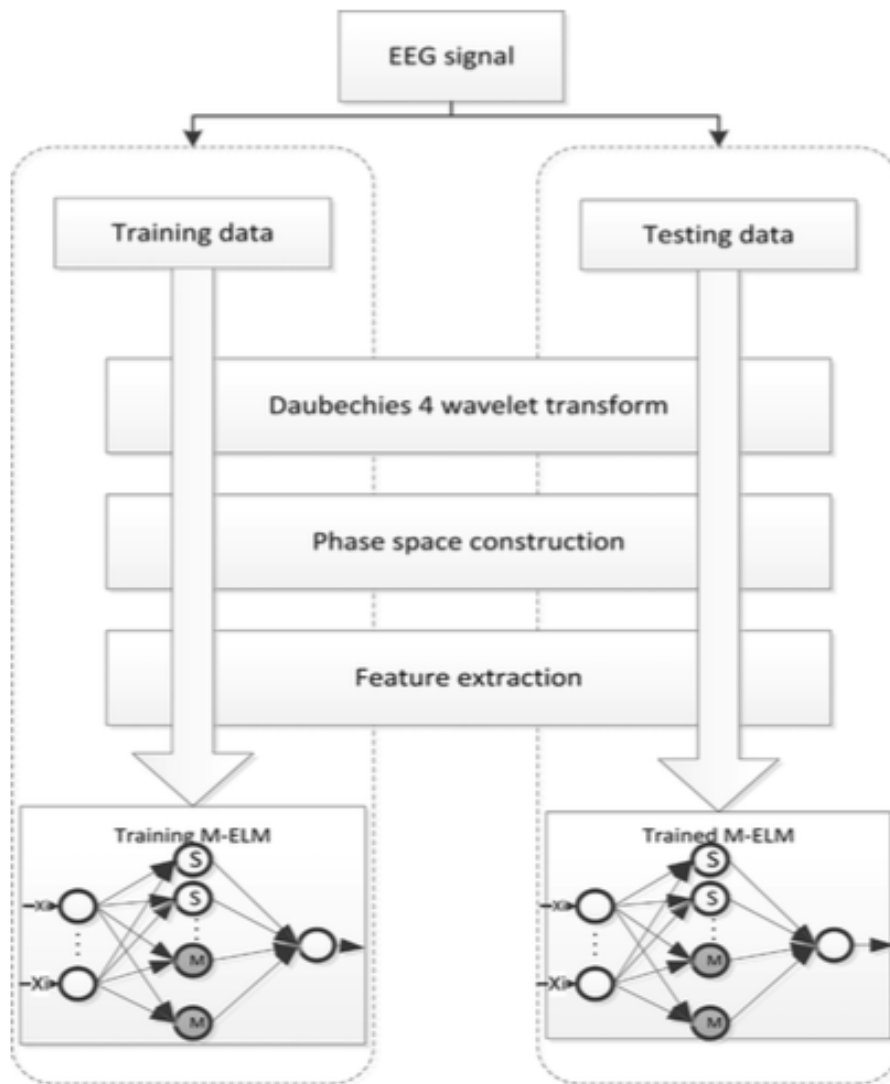


Figure 1.1: Algorithm model for the EEG recognition problem(Li D, Xie Q, Jin Q, et al.,2016)

## 1.4 Goals And Work plan

A combination of the proposed Daubechies 4 wavelet transform, phase space construction, Feature extraction methods can extract the significant features of epileptic seizure signals for classification. Extract training features using CNN instead of using these three methods. That is my goal of this master thesis.

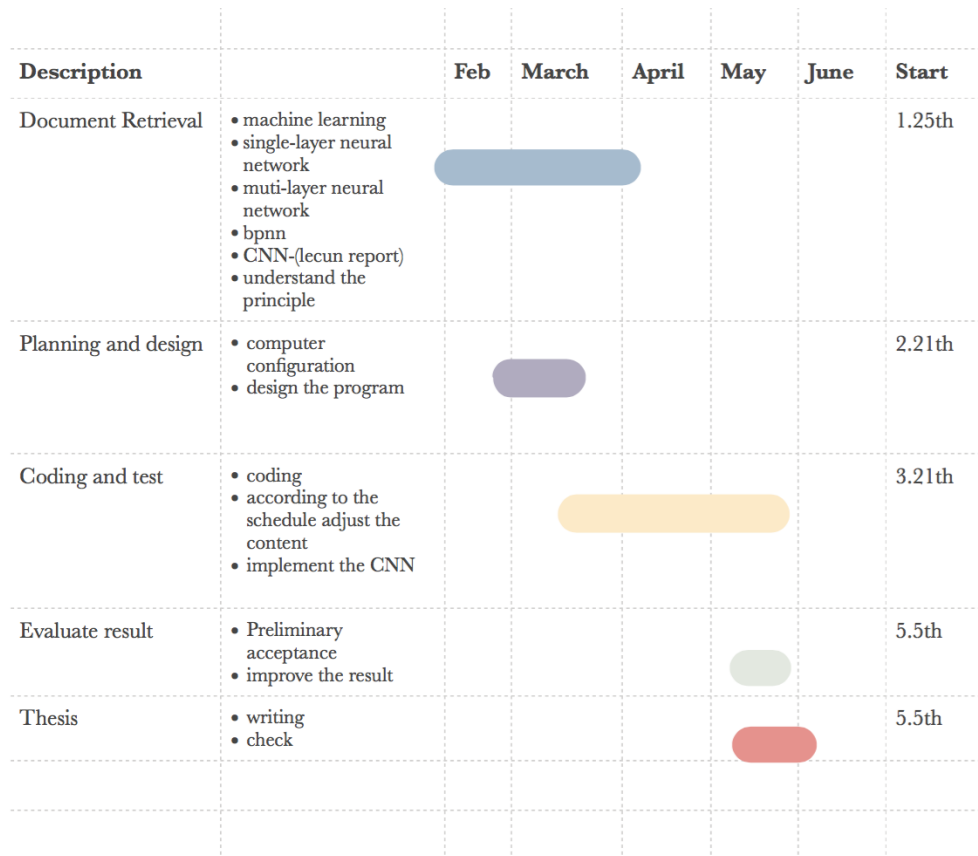


Figure 1.2: Work plan

## 1.5 Thesis Organization

**Chapter 2** Gives us an overview of the background theory. This part contains an introduction on EEG signal, Epilepsy, epileptic seizures and their effects on EEG signals, convolutional neural networks and tensorflow.

**Chapter 3** I will description of the project implementation and its functions. The data that was used for training and test. Data selection is also discussed, along with how data are preprocessed.

**Chapter 4** Experiment results and analysis.

**Chapter 5** This chapter give an overview of this thesis and provides the information of the future work.

# Chapter 2

## Background Theory

This chapter describes research relevant to the study in this thesis. Sections 2.1 and 2.2 provide EEG signal and EEG tests and epilepsy. Introduce deep learning methods such as CNN and how they work in Section 2.3.

### 2.1 EEG signal basic knowledge

#### 2.1.1 EEG signal

In 1929, Hans Berger published the first episode on EEG, he found that when the subjects were mentally operated, a brain wave -  $\alpha$  wave (We know that as long as the brain is not dead, the spontaneous activity of neurons will produce a series of EEG today.)

First, what is EEG: Through a monitoring method to the record biological electricity from brain and draw a time curve. Electroencephalogram (EEG) is a method of recording brain activity. Second the scope of application: Epilepsy; poisoning metabolic diseases; coma; brain death diagnosis; encephalitis; intracranial mass; cerebrovascular disease; severe intracranial injury; chronic dural hematoma.

The clinical significance of EEG:

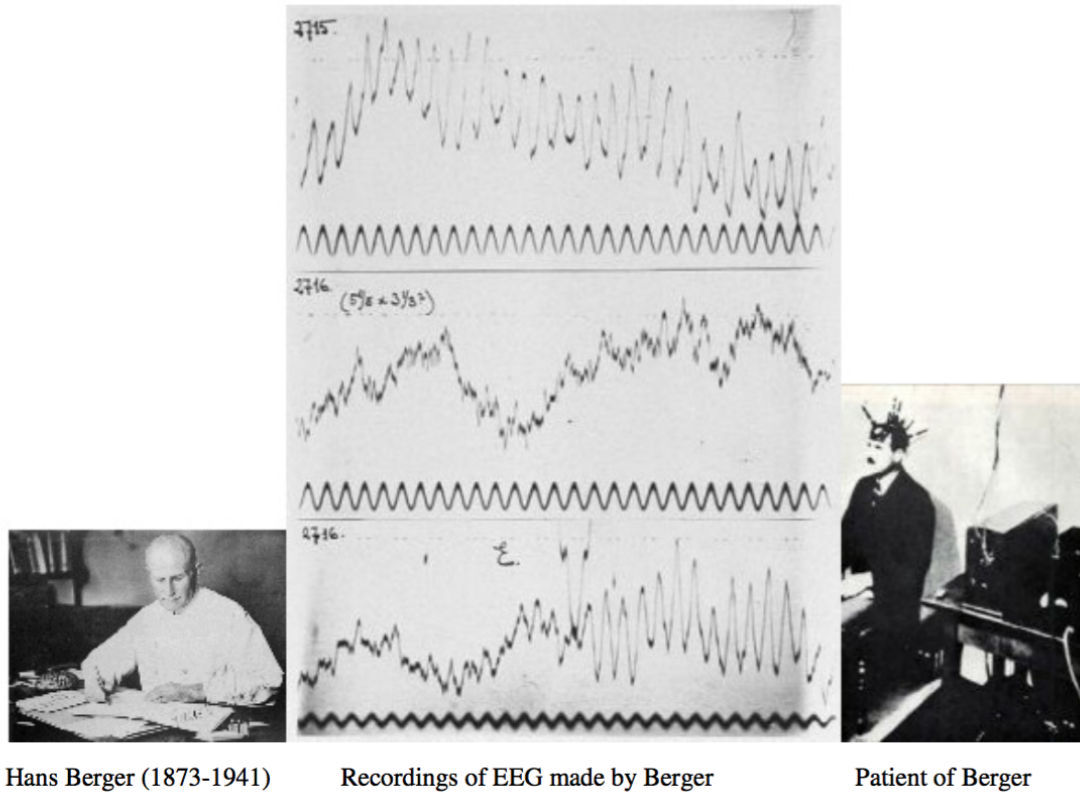


Figure 2.1: First recording of EEG signals made by Hans Berger (Berger, 1929).

Contribute to the diagnosis of the disease

Contribute to the classification of diseases

Contribute to the prognosis of the disease

To serve as an adjunct test of brain death

To prognosticate, in certain instances, in patients with coma

### 2.1.2 Neural oscillations types

During the EEG test, a plurality of electrode devices were placed on the head, with temporary glue. Then connect all the electrodes to the amplifier, and then connect the EEG recorder. After that, the signal in the brain is converted to a curve that can be displayed on the screen.

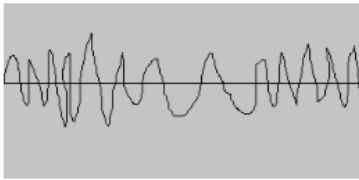
According to the usage, EEG exists in parallel records, with 1 to 256 electrodes, known as multi-

channel EEG records. A pair of electrodes correspond to a channel. Each channel records a signal. There are two ways to get the EEG signal, depending on the signal at the head position: scalp or intracranial. For the scalp EEG, place the small electrode on the scalp, with good mechanical and electrical contact. There is also a craniotomy in the operation, the detection of the inside of the scalp (Siuly 2016) (Siuly 2016)

### 2.1.3 Brain Wave

The Brain waves can be classified using their frequency, amplitude, shape and the position of electrodes on the scalp.

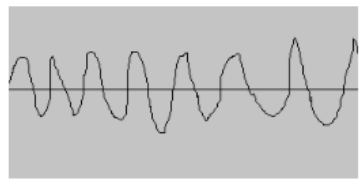
mu (8-13 Hz) have the same frequency range as alpha, but are present over the motor cortex; mu is associated with motor activities, and is also found in the alpha wave frequency range, but where the maximum amplitude is recorded over motor cortex. So it basically triggers when there is an actual movement or there is an intent to move (Bernier, R., Dawson, G., Webb, S., & Murias, M. 2007).



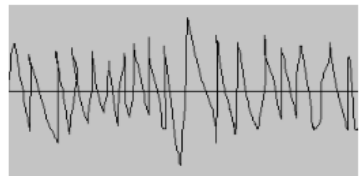
alpha (8-13 Hz) When you just wake up, it will often produce this kind of wave. present over the area strongest when the eyes are closed or when a person is in state of relaxation;



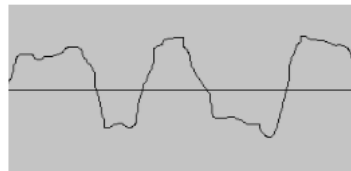
beta (13-30 Hz) are connected to states of alertness and attention;



theta (4-8 Hz) does not occur frequently in adult humans;



gamma (>30Hz) are related to information processing.



delta (0.5-4 Hz) the slowest brain rhythm, during sleep this rhythm is the strongest one;

Table 2.1: EEG signals are classified based on their frequency range.(Jayaraman, V., Sivalingam, S., & Munian, S. 2014)

## 2.2 Epilepsy

Epilepsy is a group of brain neurons caused by transient abnormal discharge of central nervous system dysfunction characterized by chronic brain syndrome, with sudden, recurrent features.

About 50% of patients also have abnormal brain electrical activity during episodes of intermittent seizures, referred to as epileptiform discharges. The main abnormal waves of epileptic discharge usually showing the high amplitude slow wave.



### 2.2.1 Epilepsy syndromes

In this part I have been searching for different kinds of epilepsy syndromes. Most knowledge from: [www.epilepsy.com](http://www.epilepsy.com).

**Generalized tonic seizure** In the period of generalized tonic seizure, 70% to 80% of patient's EEG has a different degree of abnormalities.

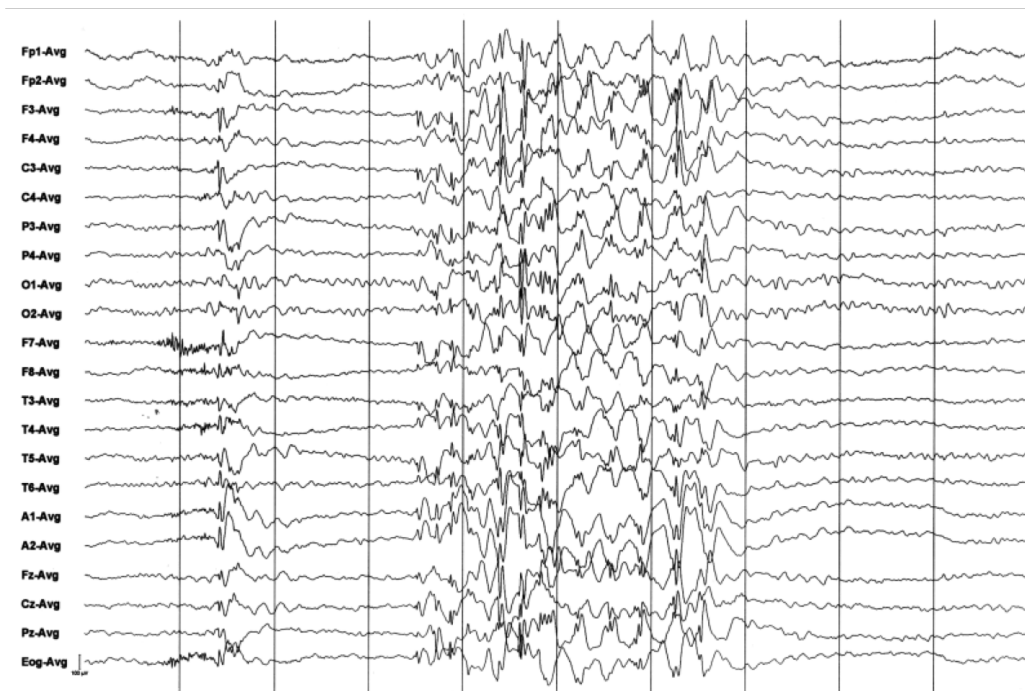


Figure 2.2: Generalize Spike Wave Discharge

#### Interictal EEG:

- Abnormal rhythm with sudden, steep increase in amplitude
- Associated with increase/decrease in frequency content
- May vary considerably from seizure to seizure - hard to detect

**Myoclonic seizures:** Myoclonic seizures are transient, vibrating muscles or a group of muscles. They occur in various epilepsy syndromes with different characteristics. During the episodes of myoclonus, the person is usually awake and able to think clearly.

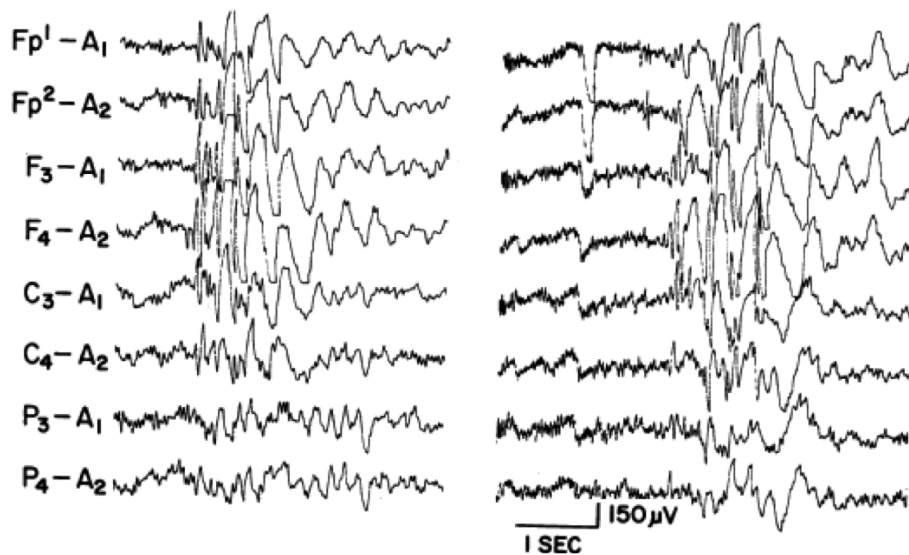


Figure 2.3: Myoclonic Seizure and Polyspike Wave

**Absence seizure:** Epileptic seizures lead to lack of consciousness and sometimes staring. They are a generalized seizure, meaning that they start on both sides of the brain at the same time. Usually for both sides of the hemisphere synchronous discharge, in the top area is more obvious, and the voltage is high, can up to 300  $\mu$ V. They start and suddenly end up for only a few seconds. Absent seizures may be simple and sometimes mistaken for daydreams that may not be detected within a few months. In children more than adults.

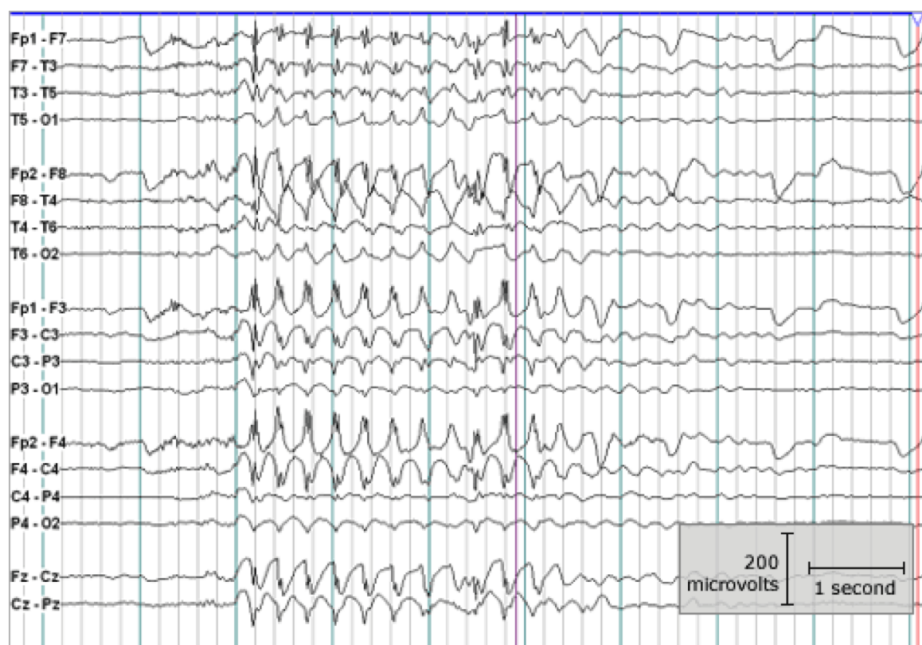


Figure 2.4: This EEG displays an abnormal discharge called a generalized spike and wave. This EEG pattern is typical for absence seizures.

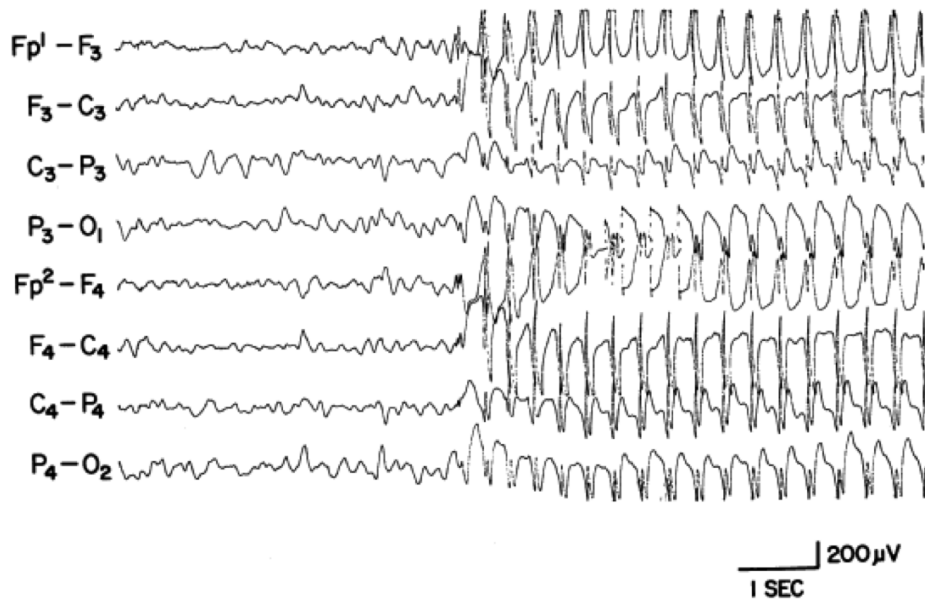


Figure 2.5: Absence: 3 Hz Spike and Wave.

### GTCS:

Tonic clonic seizures are the most common symptoms that most people think of when they think of seizures. Tonic clonic seizures usually start from both sides of the brain, but can start from one side and spread to the entire brain. A person starts to shake and lose consciousness. These types of seizures usually last for 1 to 3 minutes and take longer to recover. If more than 5 minutes of tonic clonic seizures are a life-threatening situation.

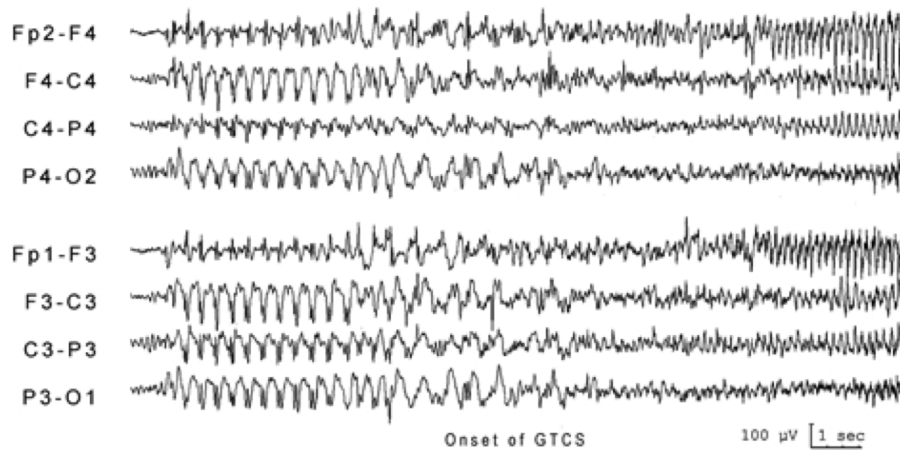


Figure 2.6: GTCS.

**Ohtahara Syndrome:** At least 3 months ago. Usually first seizures, and may even be present when the child is in the mother's womb. Boys and girls will have the same chance of happening.

Often, neonates or infants have ankylosing spasm and some (focal) seizures, with no response to epilepsy. Many cases are caused by brain malformations, metabolic disorders or genetic mutations. EEG is the most important test for diagnose OS.

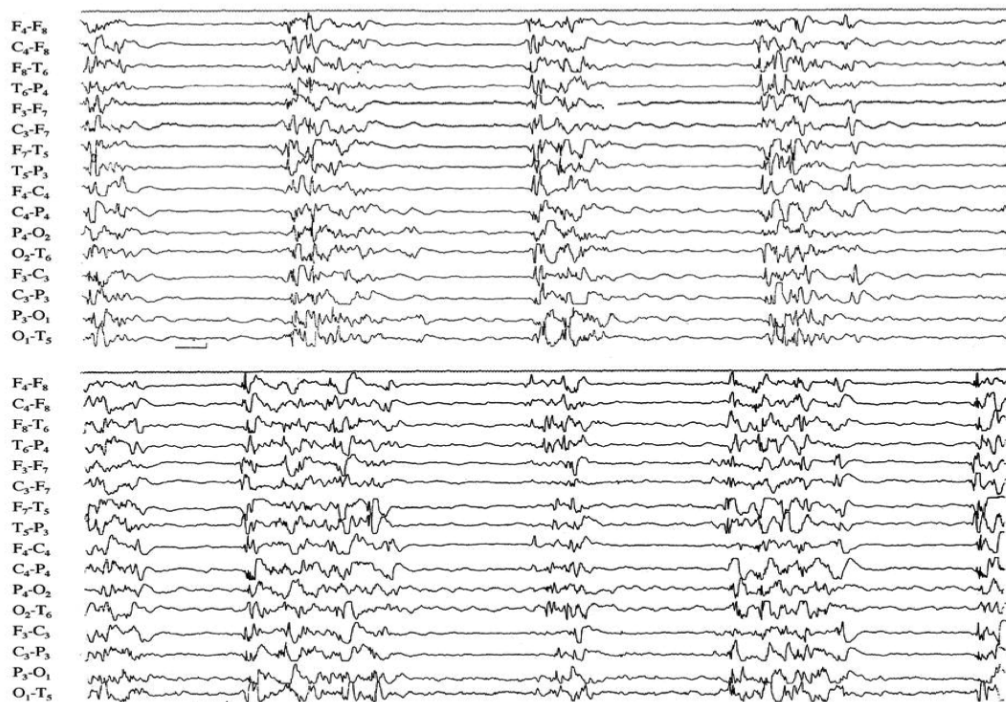


Figure 2.7: Interictal EEG of a 2-month-old boy with Ohtahara syndrome shows bilaterally synchronous suppression-burst pattern both during (above) wakefulness and (below) sleep. Calibrations are 1 second and 50 microvolts.(Ohtahara. 2003)

**West's Syndrome:** Including the sudden stiff head of the limbs Each episode of seizures lasts only a few seconds between 1-2, but usually a series of. When awake is very common, rarely in the sleep epilepsy They usually happen in 3-8 month. Almost all of the baby spasms start from the age of 1 and stop before the age of four. And May be converted into other types of epilepsy

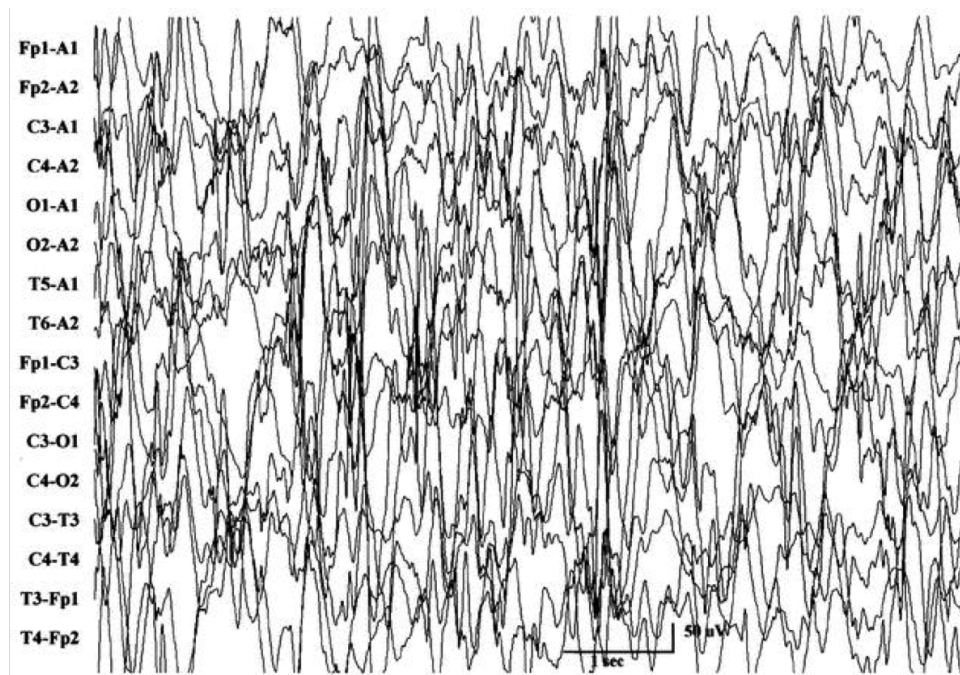


Figure 2.8: Digital recording of a hypersarhythmic pattern in a 23-month-old infant

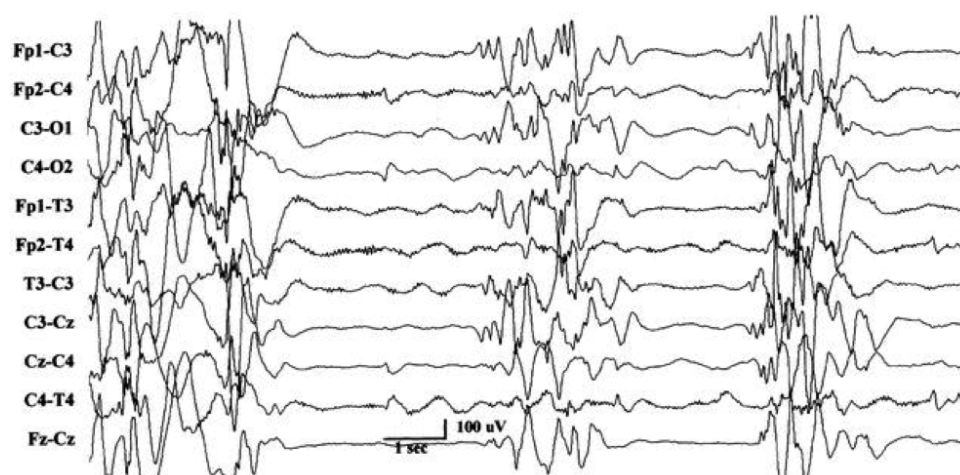


Figure 2.9: Digital recording of suppression-burst variant of hypersarhythmia in a 3-month-old infant. Sample shown at reduced sensitivity

**Lennox-Gastaut Syndrome (LGS):** Generally not the image of children's intellectual development. General 1/4 of the children will happen. Accounting for 2% to 5% of the total number of childhood seizures.



Figure 2.10: Symptomatic Generalized Epilepsy Lennox-Gastaut Syndrome

**Understanding of EEG:** Can not only on the EEG report on the written "abnormal" word to diagnosis of epilepsy. Because, if the EEG shows only general non-specific abnormalities, such as slow wave increase, mild asymmetry, poor regulation, can not be used as a basis for diagnosis of epilepsy.

In the normal population, 0.3% to 3% of the EEG had epileptiform discharges, but they do not have epileptic seizures.

Some epilepsy patients EEG is normal. But they still have seizures.

## 2.3 Convolutional Neural Network

Convolution neural network is an artificial intelligence network. CNN has registered a rapid development in recent years. Convolutional Neural Networks has been established as a powerful class of models for image and video.(Karpathy. 2014) CNN method most commonly used in image recognition problems. Because technological innovation CNN also used in video analysis,natural language processing,drug discovery. Today's most famous one-'alpha go',let the computer understand the go. It also used in CNN.

### 2.3.1 Why CNN for image

CNN's is widely used for applications involving images. We can use the general method like neural network. Suppose we can do the image classification problem. We can train neural network. Input is an image. The image can be represented as a pixels. It is a long vector. How many classifications have the number of outputs. If have 1000 categories output will be 1000 categories. In this way we will meet a problem. In fact we train neural network we wish in this neural network each neural represents a classification. And many people get this conclusion based on the test. For example the first layer is the most basic classifiers. The second layer use first layer as module to build classifiers. Then the third base on the second layer to build classifiers. the rest may be deduced by analogy. Like the figure 2.11.

When we use general fully connected network we need more parameters. Considering the properties of image can simplify the network. For example we have a colorful image  $100 \times 100$ . It is not a big picture. When we make it to a vector, the image will have  $100 \times 100 \times 3$  pixels. The colorful image needs to multiply three. Input vector has thirty thousand dimensions. We suppose hidden layer are 1000. So the first have thirty thousand multiply one thousand. CNN can simplify this structure. According to people's knowledge and understanding of image processing we can simplified the network.

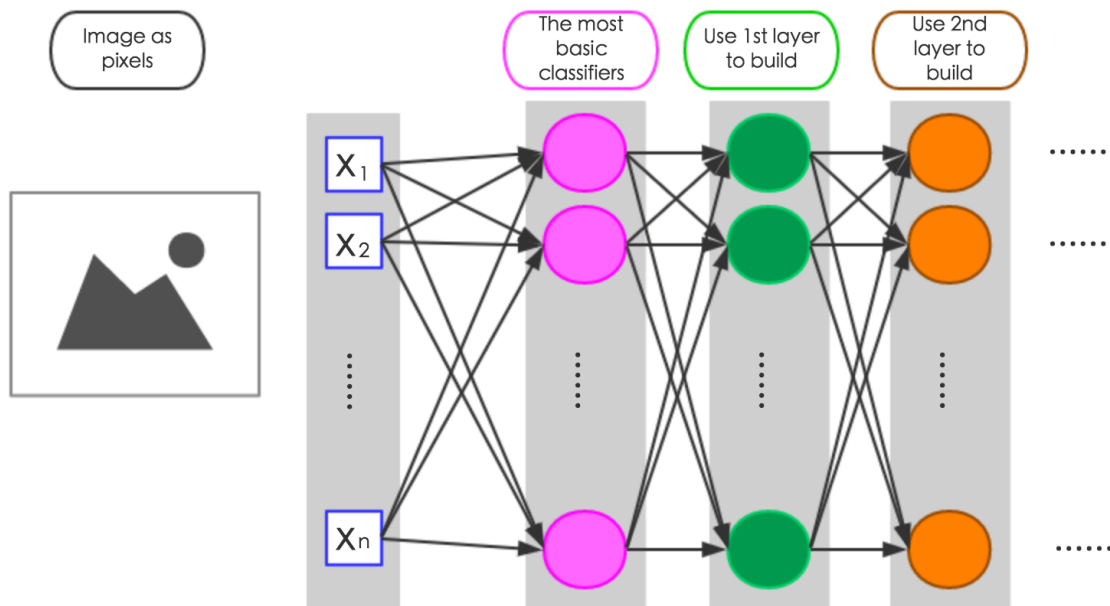


Figure 2.11: Neural network layer structure

**Three property:**

- Some patterns are much smaller than the whole image. A neural does not have to see the whole image to discover the pattern. For example one neural to detect have a head. Another one checks if there is any limbs. Each neural have their own duty. So connecting to small region with less parameters.
- The same patterns appear in different regions. So we can use the same detector.
- Sub-sampling the pixels will not change the object. We can sub-sampling the pixels to make image smaller.

**The whole CNN**

The convolution based on the property 1 and 2. The Max pooling based on the property 3.



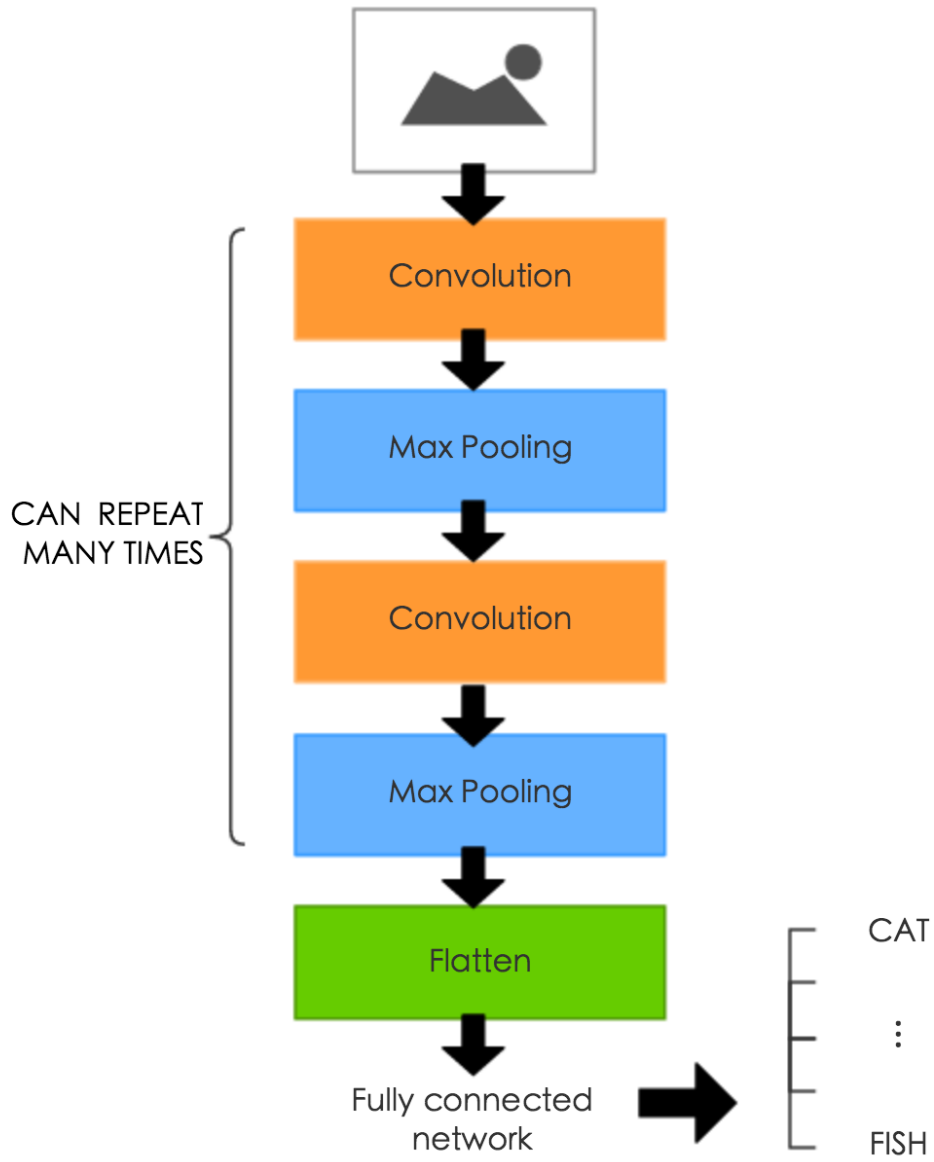


Figure 2.12: The whole CNN structure

### 2.3.2 what is Convolution in math

From the figure 2.11 we will understand it. First we see the left part. From top to bottom, the top one we know  $x[0]=a,x[1]=b,x[2]=c$ . Middle one  $y[0]=i,y[1]=j,y[2]=k$ , we want to know  $x[n]*y[n]$ , it reveals the physical meaning of the convolution. The next show  $x[n]*y[0]$  and translate to position 0. Then  $x[n]*y[1]$  and translate to position 1.  $x[n]*y[2]$  and translate to position 2. Finally, the three figures are superimposed we get  $x[n]*y[n]$ .

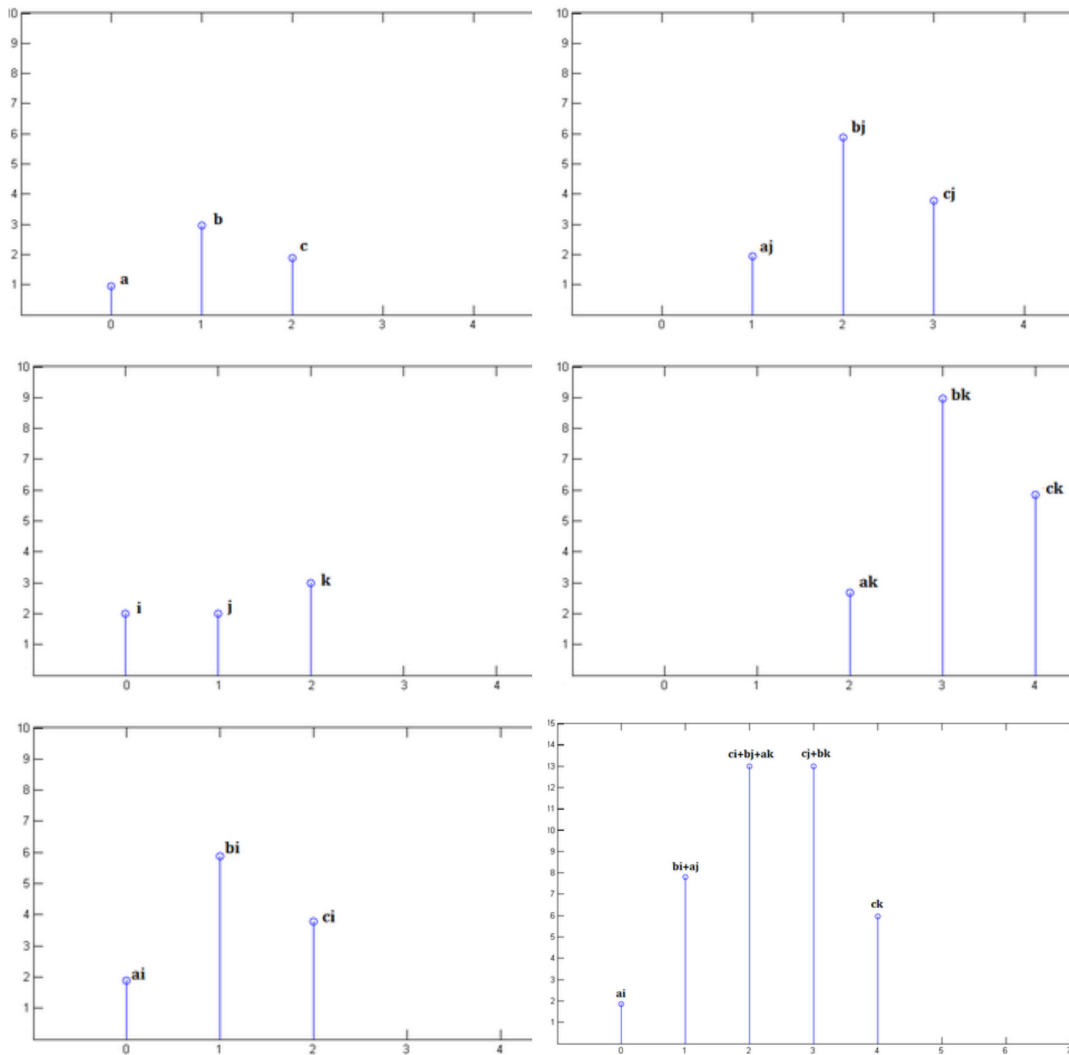


Figure 2.13: Convolution: weighted stack

### 2.3.3 CNN - Convolution

Suppose the image is black and white. In the convolution have a set of filters. Each filter is equal to neural. Each filter is a  $3 \times 3$  matrix. In this matrix each element's value is equal to parameter, weight and bias in the network and also the value is learned not set by people. If the filter is  $3 \times 3$  so the filter only checks the range of  $3 \times 3$ .

The filter has 9 values. The image of the first range of  $3 \times 3$  also has 9 values. The image moves one stride and get next 9 new values. And then with the same filter. These values do inner product and get a  $4 \times 4$  matrix. If have many filters do the same process for each filter. And will get new  $4 \times 4$  matrix. This kind of  $4 \times 4$  matrix called feature map.

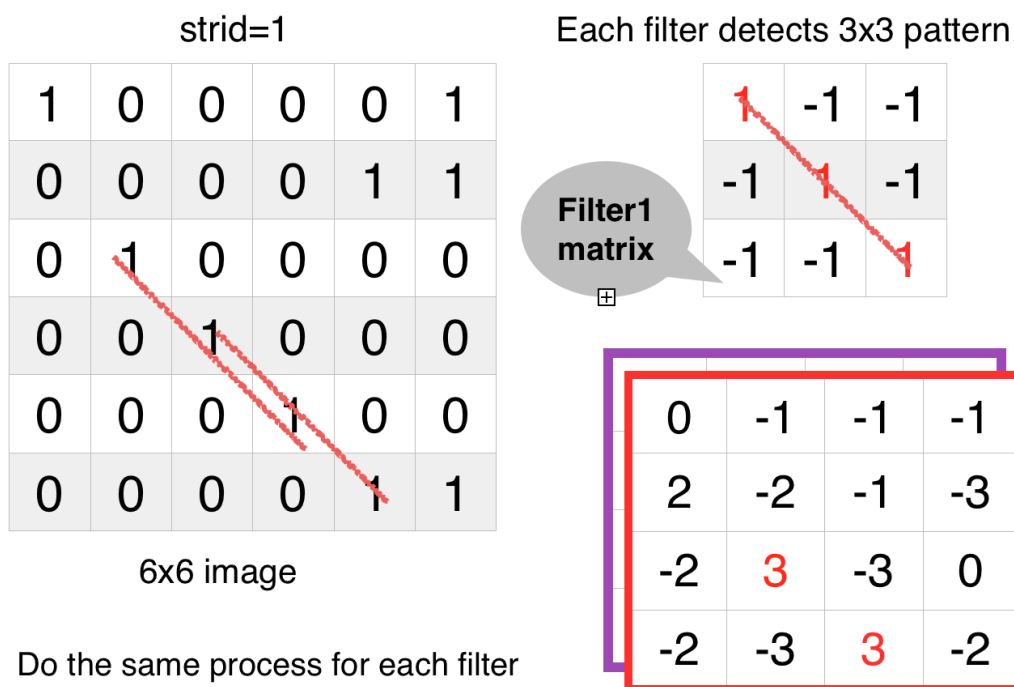


Figure 2.14: Filter and image do operate

If the image is colorful. The image represented three matrices stacked together. Different feature considering three color:red,green and blue.

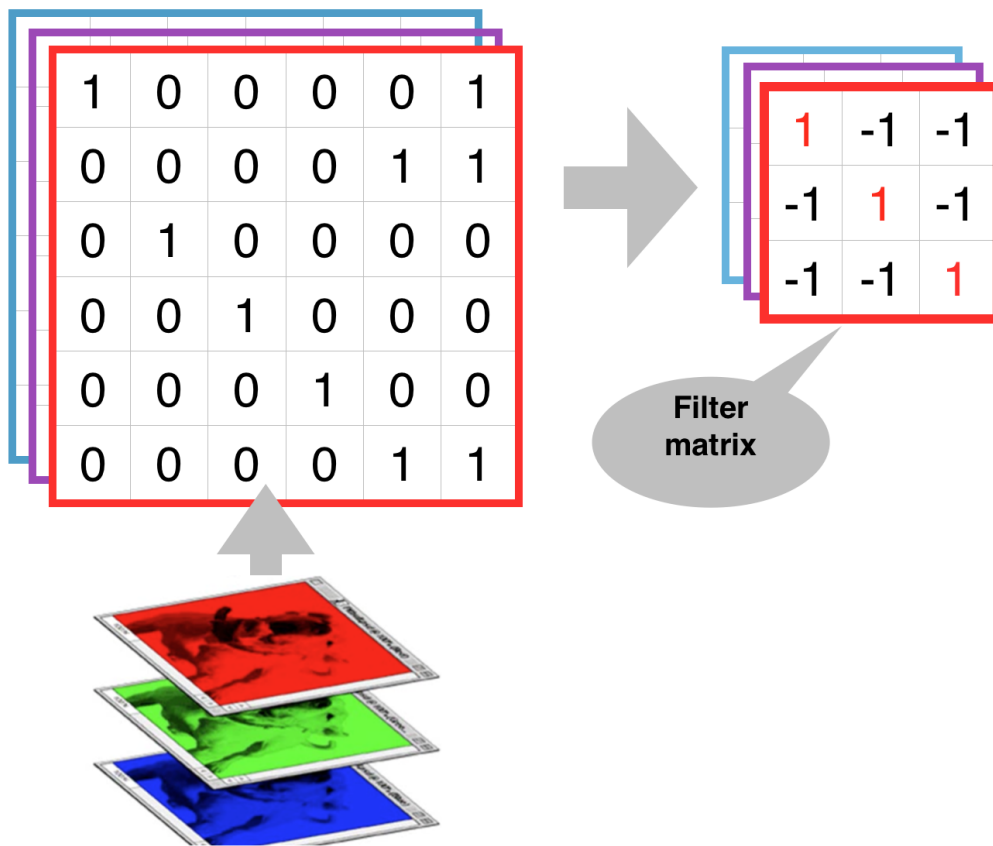


Figure 2.15: The colorful image structure

### 2.3.4 CNN - Max pooling

According to the filter 1 get a  $4 \times 4$  matrix1, Do the same process for the filter 2. Get the matrix2. Four elements( $2 \times 2$ ) becomes a group and then pick the max or average one. In this way will get a smaller matrix.

When we finish one time convolution and one time max pooling we can make  $6 \times 6$  turn to  $2 \times 2$  pixels. The number of the channel is the number of filters.

### 2.3.5 CNN - Flatten

In the flatten pull straight the feature map. And then put it into fully connected network.

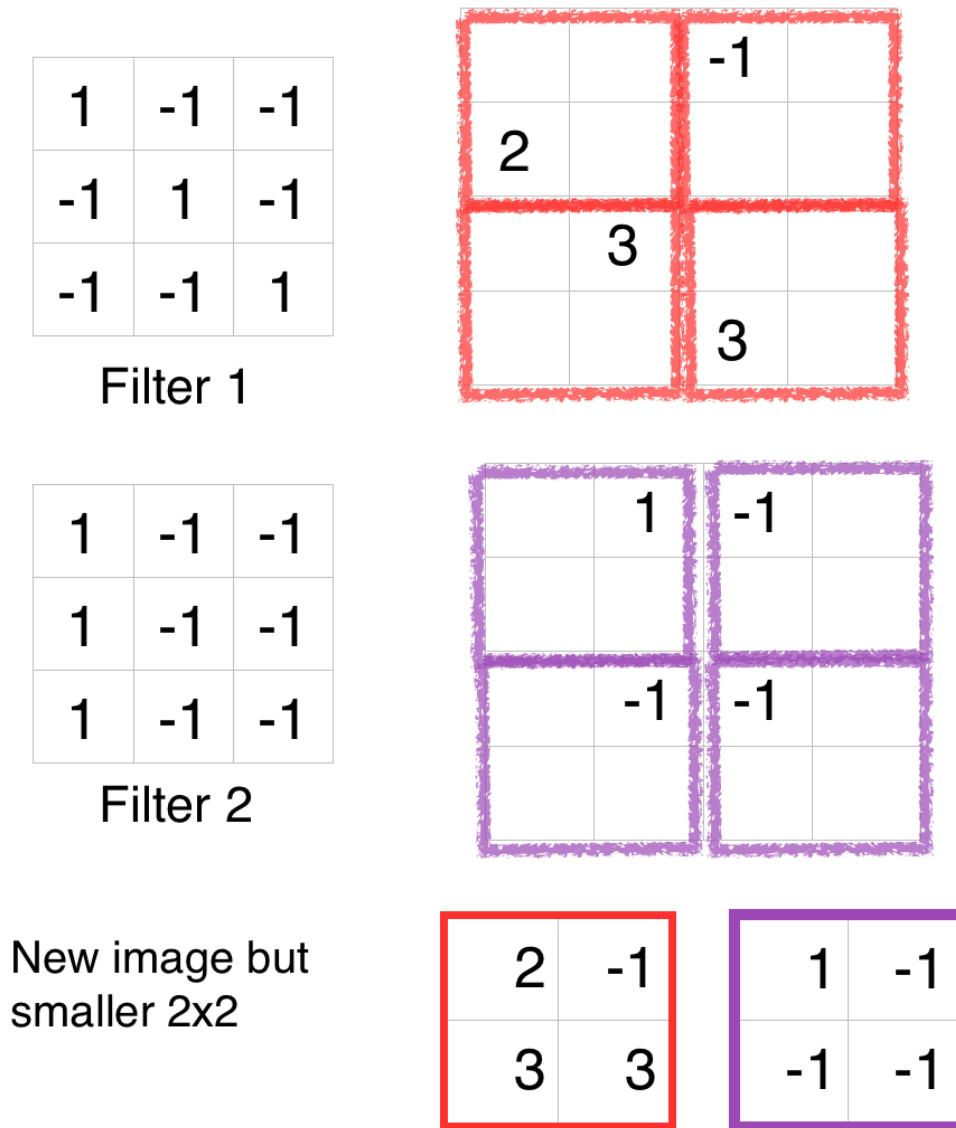


Figure 2.16: Max pooling

### 2.3.6 CNN - convolution vs fully connected

We suppose the output of feature map equal to one of hidden layer's output in fully connected. After two link convolutions is the fully connected remove some weight. If we consider the filter 1 and we pull straight the  $6 \times 6$  image. The output after do inner product in the filter 1 equal to one of neural's output.

Fully connected have different weights, but in the convolution need to share same weight(filter). Like the figure 2.17.

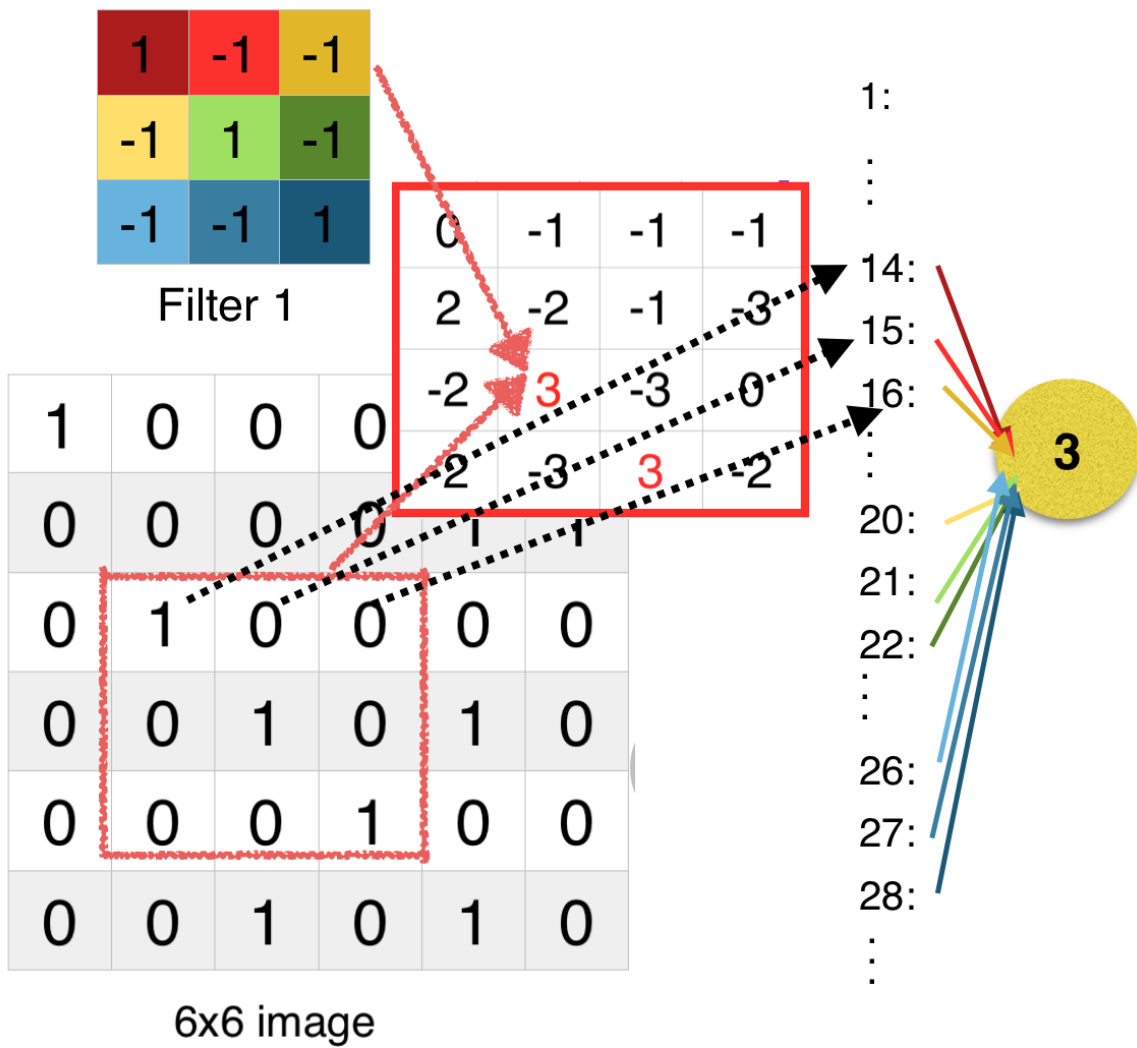


Figure 2.17: convolution use fully connected represent

In this Chapter I read some papers. Most of them have lots of mathematics hard to understand. I am trying to explain without math. This structure is the first successful CNN. The simple one. It is good for beginners understand CNN structure. (Lecun, Y. 1998)(Jake Bouvrie. 2016)(Krizhevsky, A. 2012)

# Chapter 3

## Epilepsy data

This chapter describes the how to get the data. The data that was used for training and validation described. Further, this kind of data from the patient. Section 3.1 describes where get this epilepsy data. How to read this data in the Section 3.2. Section 3.3 Training and Validation Sets.

### 3.1 Data Selection

Since there is a no public database for EEG data to our knowledge. I have been search on the internet to get more data for some days. Until I find some data from ucsd. EEG / ERP data available for free public download. I download the epilepsy data. There are all raw data. I need to show this data and get image data. In this data have two groups :patient and normal people.

I also connect to a hospital to get more image data. But need to get patient signs to agree before they test EEG. So it not easy to get enough data.

## 3.2 Image data

How to get the image data:

First I need to install some software. First I have to read this data so I need EEG-LAB.

EEGLAB is an interactive matlab toolbox for processing continuous and event-related EEG, MEG and other electro physiological data incorporating independent component analysis (ICA), time/frequency analysis, artifact rejection, event-related statistics, and several useful modes of visualization of the averaged and single-trial data. EEGLAB runs under Linux, Unix, Windows, and Mac OS X. (Poolman, P., 2010)

I use the window 10 and matlab 2017a. Next I will give an example showing how I get data image.

First start matlab. Then go to Start, just type eeglab and run it.

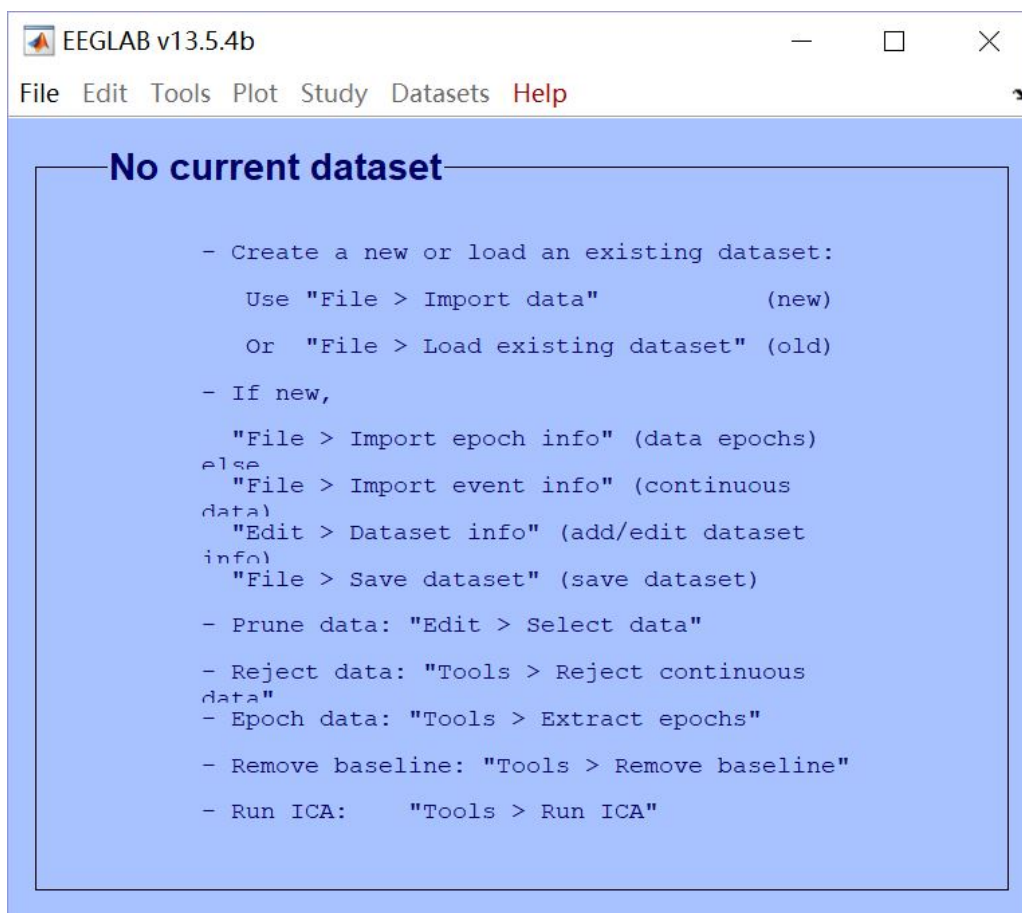


Figure 3.1: The blue main EEGLAB interface



Because our data not regular format. So we use biosig to import. The Biosig toolbox (biosig.sf.net) contains links to functions to read other EEG data formats in Matlab. Next we can choose the range we wanted. Like below I pick the 100 seconds to 200 seconds.

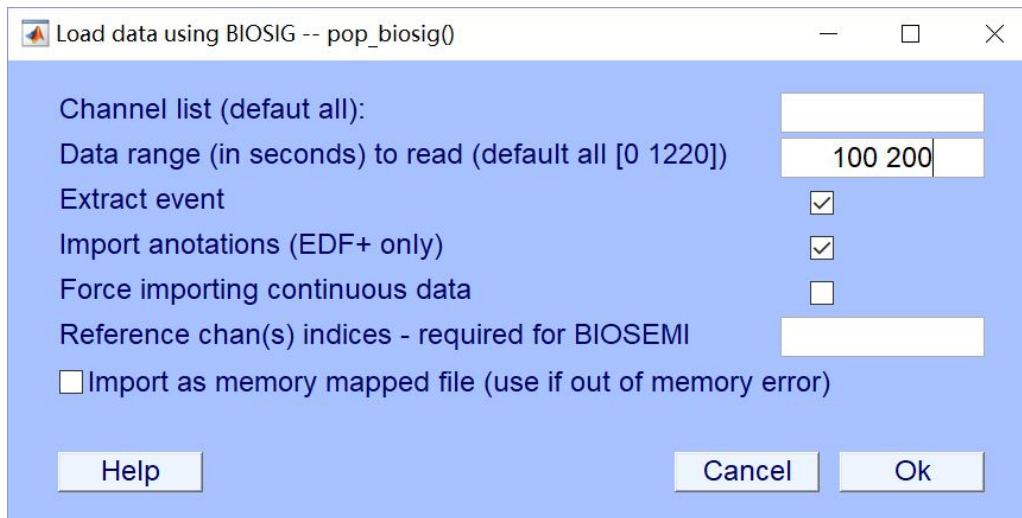


Figure 3.2: Importing data

Then we will get the data show on the interface. From this interface I can see some information about the dataset. Channels per perframe is 32, that is channel number is 32. Frames per epoch are 3000, indicating the number of samples of an epoch.

Before I vizualise data I can use preprocessing tools. The filtering option in EEGLAB, `eegfilt()`, uses linear finite impulse response (FIR) filtering. (Vecchio, F., 2014) Scrolling through the data. Press `selectPlot - Channel data(scroll)`. We can select time range to display. In the plot window also allows you to reject. You can delete the range you do not want it.



Figure 3.3: Data information interface

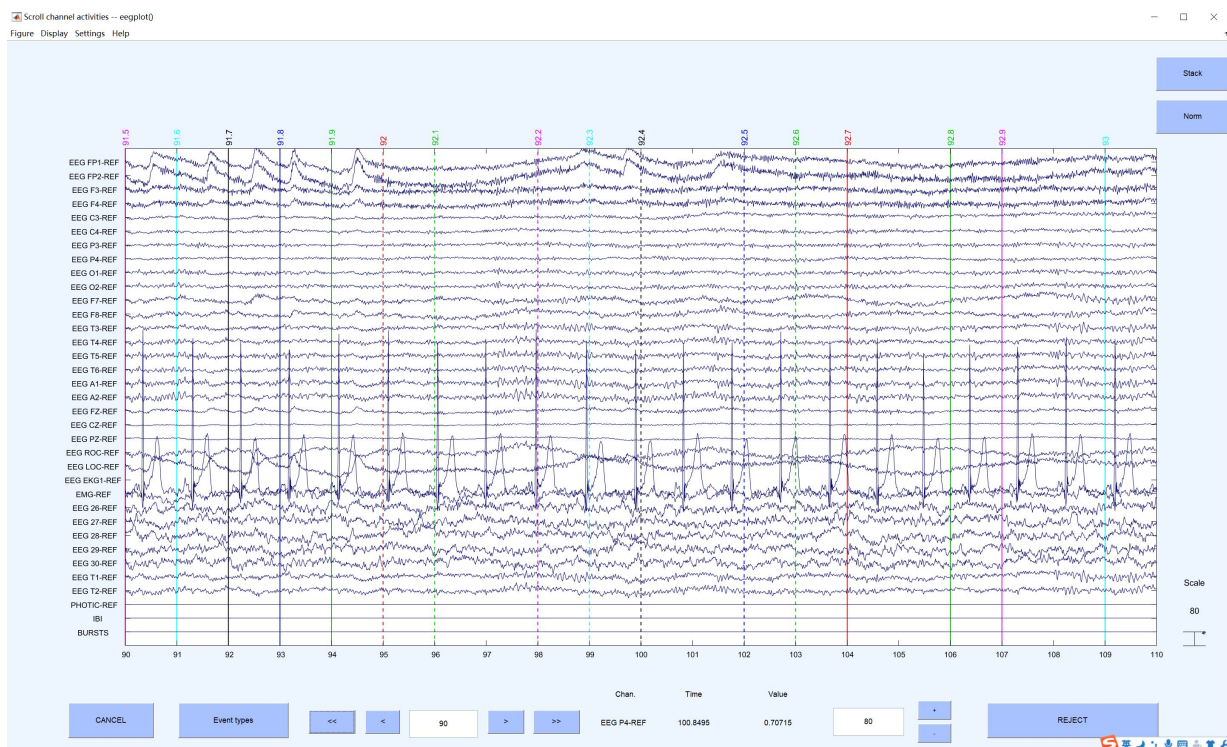


Figure 3.4: Visualise data

## 3.3 DataSets

In this part describe how to create dataset. Tfrecored, which is a binary file that puts image data and labels together, can make better use of memory, quickly copy, move, read, store in tensorflow.

### Prepare work-Requirement

Pycharm - Python IDE

PyCharms smart code editor provides first-class support for Python.

Tensorflow Package - I will use this package to create dataset.

TensorFlow is an open source software library for numerical computation using data flow graphs.

In the website have the detail about how to install for different computer system.

Tfrecored will automatically type the same tag for each class based on the class you choose to enter the file. In this case, only 0,1 two categories. I get an error in this part because my mac system. The Finder automatically places a .DS\_Store files into every folder you have opened. .DS\_Store files are created by the Finder during its normal course of operation. Have to delete .DS\_Store file.

```

/System/Library/Frameworks/Python.framework/Versions/2.7/bin/python2.7 /Users/Jessie/PycharmProjects/CNN/dataset.py
Traceback (most recent call last):
  File "/Users/Jessie/PycharmProjects/CNN/dataset.py", line 32, in <module>
    data = create_record()
  File "/Users/Jessie/PycharmProjects/CNN/dataset.py", line 20, in create_record
    img = Image.open(img_path)
  File "/Users/Jessie/Library/Python/2.7/Lib/python/site-packages/PIL/Image.py", line 2286, in open
    % (filename if filename else fp))
IOError: cannot identify image file '/Users/Jessie/PycharmProjects/CNN/imagdata/patient/.DS_Store'

Process finished with exit code 1

```

```

Jessie-MBPro:~ Jessie$ ~ Jessie$ sudo find / -name ".DS_Store" -depth -exec rm {
} \;

```

Table 3.1: Fix error command

In the figure 3.6 showing the train dataset.

testdata.tfrecords	445.1 MB
traindata.tfrecords	3.79 GB

Figure 3.5: Tfrecord file

Here are the all raw data for creating Tfrecord file. Have 2 kinds of pictures. One are the normal data. Another are patient data.

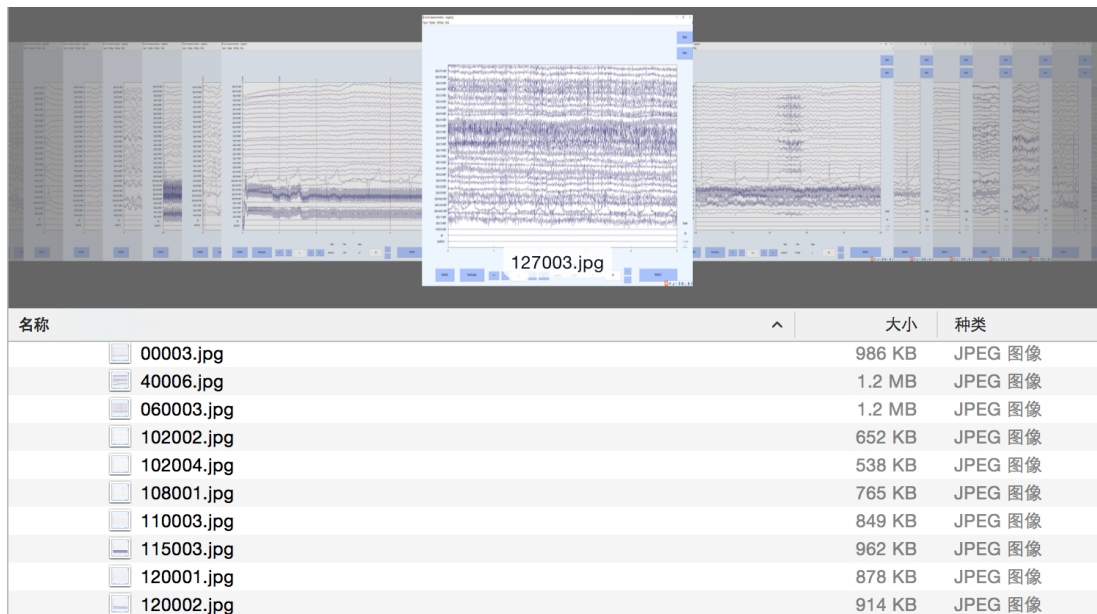


Figure 3.6: Image data file

Sometimes we want to check whether the classification is wrong, then we can call the session, the Tfrecord picture from the stream read out, and then save it. The below pictures read from the Tfrecord.

Do the same step for test dataset.

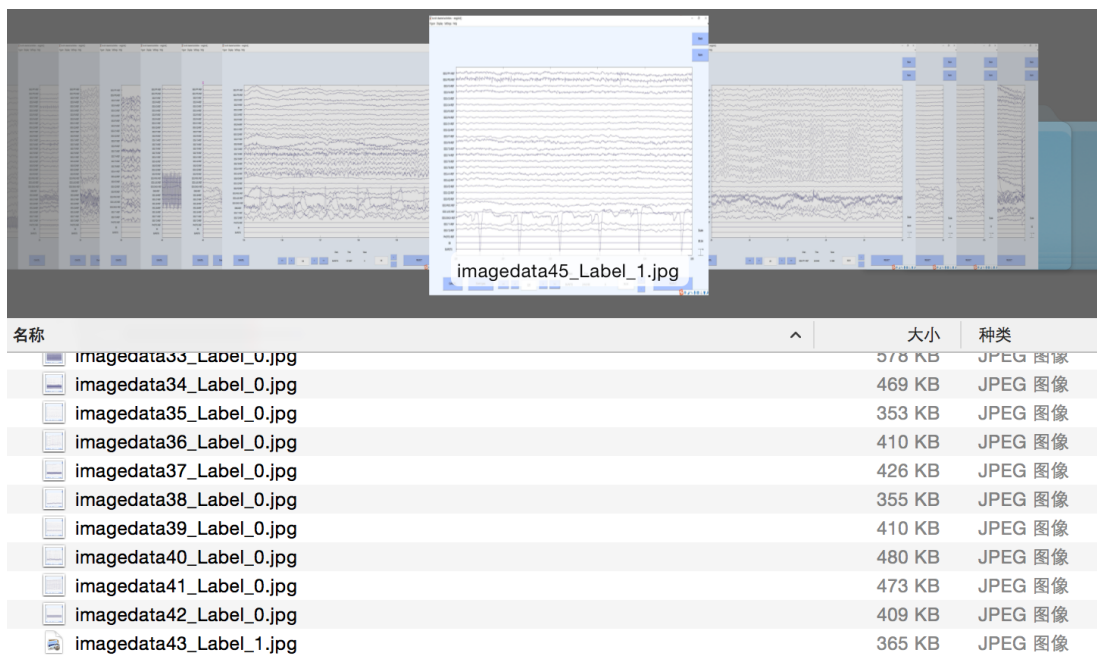


Figure 3.7: Show read from tfrecords format.

# Chapter 4

## Experiments and Analysis

In the following sections, will show how the CNN works in my project. Section 4.1 The experimental model chooses. In Section 4.2 outlines the experimental design, and Section. In Section 4.3 the results are presented. The results are then further discussed in Section 4.4.

### 4.1 Model

LeNet 1986. The first successful applications of Convolutional Networks were developed by Yann LeCun in 1990s. This code used to read code.

However, SVM and other features instead of LeNet. With the introduction of ReLU and dropout, as well as the historical opportunities brought by GPU and large data. CNN ushered in a historical breakthrough in 2012 - AlexNet.

AlexNet. The first work that popularized Convolutional Networks in Computer Vision was the AlexNet, developed by Alex Krizhevsky, Ilya Sutskever and Geoff Hinton.

In my project I used CNN base on AlexNet.

Here also have other model(karpathy,2016). see the table below.

ZF Net.	ILSVRC 2013 winner and improvement on AlexNet.
GoogLeNet.	ILSVRC 2014 winner and main contribution was reduced the number of parameters in the network.
VGGNet.	runner-up in ILSVRC 2014 and main contribution was in showing that the depth of the network is a critical component for good performance.
ResNet.	winner of ILSVRC 2015. It features special skip connections and a heavy use of batch normalization.

Table 4.1: Other CNN models from [cs231n.github.io](https://github.com/cs231n)

## 4.2 Experimental Design

### Data augmentation:

- a. No augmentation → Random Crop 225x225(get 1 image)
- b. Flip augmentation → Flip augmentation (get 2 images)
- c. Crop + Flip augmentation (get more images)

If the original image is 256x256 . Smaller patch is 125x125 . This increase the size of training by 20000  $((256-125) \times (256-125)) \times 2$ .

Because my input data, random brightness,color jittering not include in the augmentation. If put the raw data can under substantial over-fitting.

### Activation functions:

Sigmoid is a commonly used non-linear activation function that compresses the input continuous

real value between 0 and 1. It has two major drawbacks: Sigmoid saturates and kill gradients. Sigmoid outputs are not zero-centered.

ReLU. The Rectified Linear Unit has become very popular in the last few years. It computes the function  $f(x)=\max(0,x)$ . In other words, the activation is simply thresholded at zero. (<http://cs231n.github.io/neural-networks-1/>)(Krizhevsky, A. 2012)

## Create Network

The first thing to mention is that multiple GPUs require `tf.get_variable()` to share data, and a single GPU requires only `tf.Variable()`. In my project I only use the default GPU.

The first layer is conv1, the field view is  $5 * 5$ , each image from 3 channels (rgb) to 64 channel. So give the give an input tensor of shape=`[5, 5, 3, 64]`=[*filter\_height, filter\_width, in\_channel, out\_channel*].

The next layer is pool1: The Stride=`[1, 2, 2, 1]` The first one and last one must be 1. If the stride size equals the filter size, there will be no overlapping.

After the pooling layer is Local Response Normalization. In short, the output is normalized a bit. Good for the training.(Krizhevsky, A. (2012).)

The first echelon is followed by a convolution layer conv2. Similar to the first convolution layer is only 64 channels to 64 channels, the offset is 0.1, no change, but after that is normalized layer norm2, and then is the same structure Pooling layer2.

After two standard convolutions are fully connected layers:

The local3 layer The local3 layer expands each dimension of the sample after the two times repeat convolutions. From dim mapping to 384 neurons. The next are same from 384 to 192. Then from 192 to 10.

Because this network without softmax layer. So I use `softmax_cross_entropy_with_logits` function.





Figure 4.1: Project structure

## 4.3 Experimental Result

Deep neural networks usually have very high capacity and are therefore particularly prone to over-fitting. Several methods, collectively known as regularization methods, can be used to prevent over-fitting when training high-capacity models. In my project I use the *l2\_regularizer* to avoid over-fitting.

In the figure 4.2 the result showing the 20 epochs and 50 epochs. Because the dataset so large and I only use my default my compute gradients. So I set epochs only 20 and 50. From the left figure we can see the The train accuracy in the 8 epoch can reach to 1. The network was trained on data set with *l2\_regularizer* , which not causes it to over-fit the training set. In the left figure can see the test accuracy can become smooth after 30 epochs.

It looks can show the good result for recognition. I think because my input data can easy to get feature.

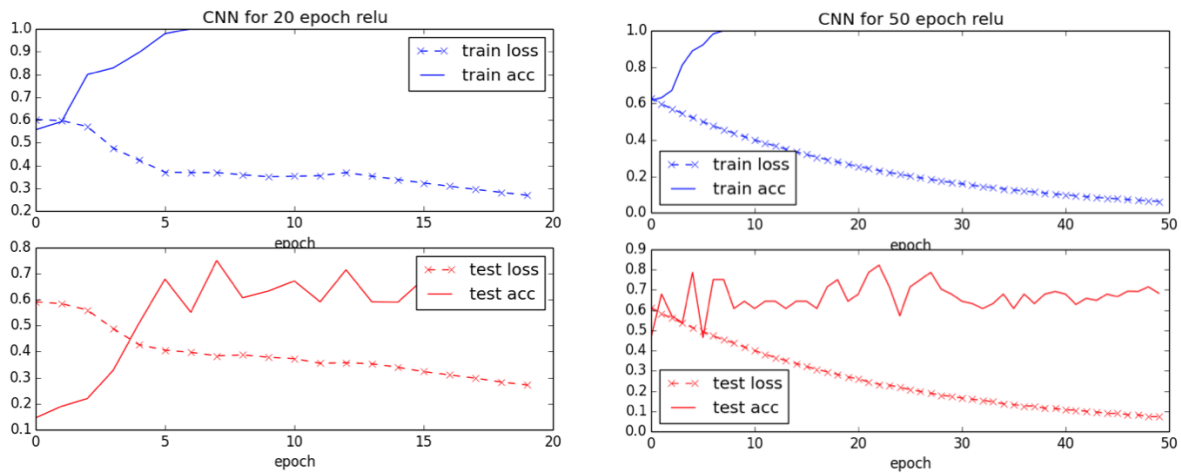


Figure 4.2: Image data file

For my data after convolutions and get convolved feature. Each map gets two feature maps. In the below figure convolution 1 and convolution 2. The convolution operation is performed by theano `conv.conv2d`.

The most commonly used Maxpooling. Solved two questions: 1. reduce the amount of calculation 2. Rotation invariance Maxpooling's downsampling process halves the length of the

feature map. (The following results are not reflected in the figure, python automatically to pull as big as, but in fact the number of pixels is halved)

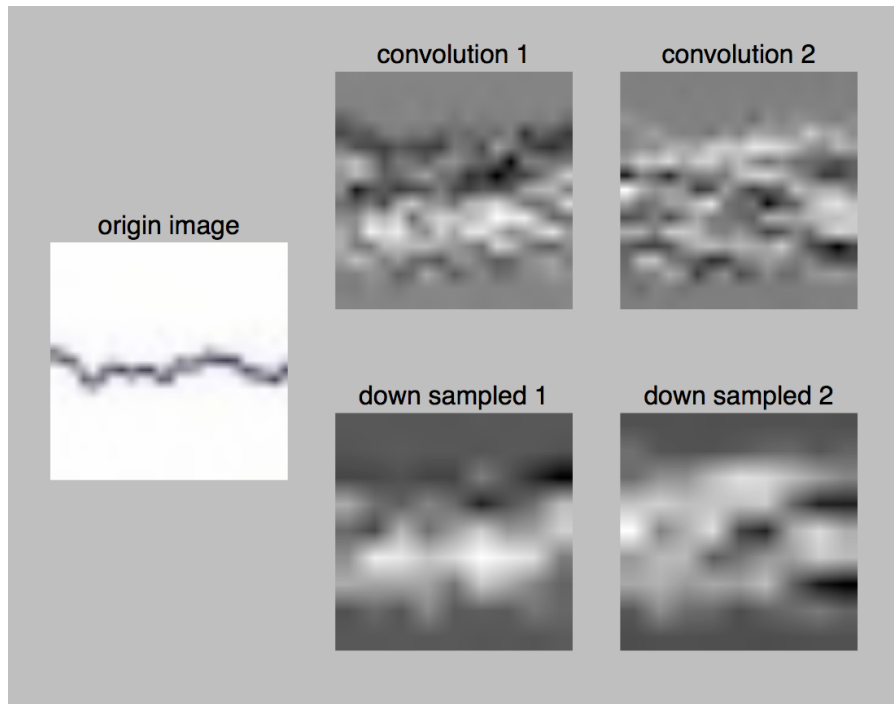


Figure 4.3: Image data file

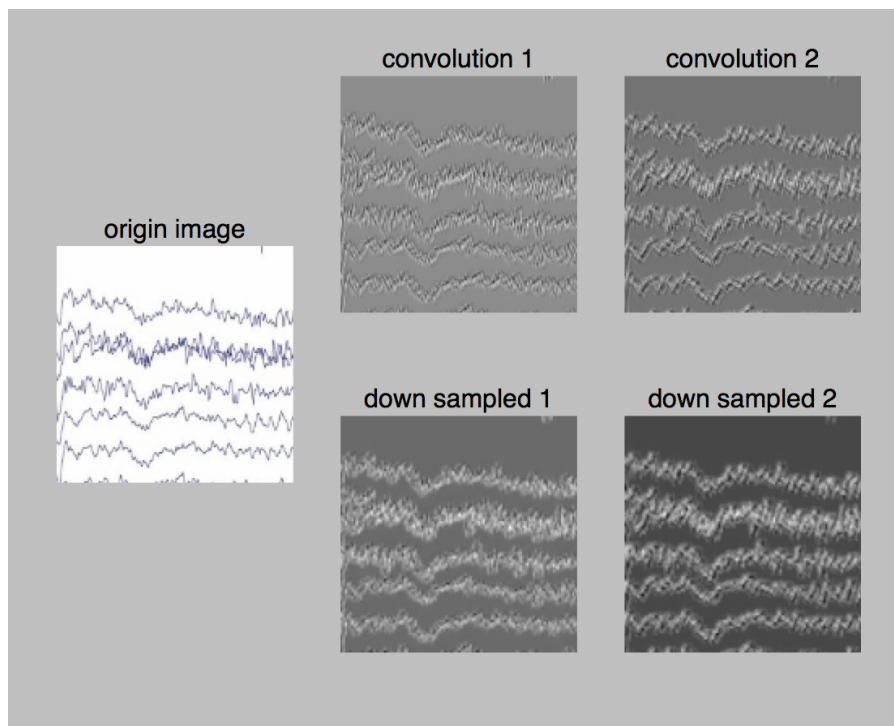


Figure 4.4: Image data file

# Chapter 5

## Conclusion And Future

In this thesis I showed that the usage of CNNs for diagnosis of epilepsy. To classification the normal people and epilepsy. This was achieved by utilizing a convolutional neural network and training it on dataset created from real EEG-signal data.

### 5.1 Overview

The Status and background were introduced in Chapter 1. In Chapter 2, the thesis presented background theory and how the CNN works. The components of a CNN was briefly introduced. In Chapter 3 showing how to create the dataset. In the Chapter 4 give the results.

The goal of this thesis was to create a CNN to diagnosis of epilepsy. Actuary most people used CNN to detect real image. Like a dog or some real thing. Even in the diagnosis some kinds of skin disease. I want to use CNN to detect lines from data. In the beginning I am not sure is a good idea. Because transfer all of this data take long time. To manually label the data can be expensive and very time-consuming in many domains.(Ghahreman, N., 2011)

The first things I should do is to get input data. Fortunately, the next things going well.

To create the CNN network looks good. Use the tensorflow. It is a package of python. After read some tutorial to create the CNN network.

From my project I get some tips: For the large input data should do the batchnorm or Relu. The clear data is more important. In the begin I get not good result because of the raw data. You can reduce the stride can get the same result for put more image.

## 5.2 Future Work

This section presents future work, such as how to update the project and bootstrapping loss function.

First I want to add dropout layer.

Then try to add more GPU to train. To save time for getting output.

Trying to use average-pooling. Use the model MAX+AVG+AVG. To compare result.

In my project without try to compared with another model. The next step I will try to add VGG-Net model.

In the end I want the result will be better to present. So I want to try use tensorboard.

For the input data I should learn more knowledge about epilepsy to give more detail classification. Not only two types. Give different labels for different kinds of epilepsy. Then create the dataset. Let the CNN model train this dataset. That is my final task.

# Appendix A.

## System Instructions

Hardware capacities:

- CPU: Intel(R) Core(TM) i7-4870HQ CPU @ 2.50GHz
- GPU: AMD Radeon R9 M370X

In order to run the code the following dependencies are required:

- Python 2.7
- Tensorflow
- Tensorlayer
- matplotlib

The URL of these repositories are listed below:

- <https://github.com/jiaxinLin92/CNNClassification>

## Appendix B.

### Show Project Output

```

'System/Library/Frameworks/Python.framework/Versions/2.7/bin/python2.7 /Users/Jessie/PycharmProjects/CNN/cal0.py
[TL] InputLayer model/input_layer: (28, 125, 125, 3)
[TL] Conv2dLayer model/cnn_layer1: shape:[5, 5, 3, 64] strides:[1, 1, 1, 1] pad:SAME act:identity
[TL] BatchNormLayer model/batch_norm1: decay:0.900000 epsilon:0.000010 act:identity is_train:True
[TL] PoolLayer model/pool_layer1: ksize:[1, 3, 3, 1] strides:[1, 2, 2, 1] padding:SAME pool:max_pool
[TL] Conv2dLayer model/cnn_layer2: shape:[5, 5, 64, 64] strides:[1, 1, 1, 1] pad:SAME act:identity
[TL] BatchNormLayer model/batch_norm2: decay:0.900000 epsilon:0.000010 act:identity is_train:True
[TL] PoolLayer model/pool_layer2: ksize:[1, 3, 3, 1] strides:[1, 2, 2, 1] padding:SAME pool:max_pool
[TL] FlattenLayer model/flatten_layer: 65536
[TL] DenseLayer model/relu1: 384 relu
[TL] DenseLayer model/relu2: 192 relu
[TL] DenseLayer model/output_layer: 10 identity
[TL] InputLayer model/input_layer: (28, 125, 125, 3)
[TL] Conv2dLayer model/cnn_layer1: shape:[5, 5, 3, 64] strides:[1, 1, 1, 1] pad:SAME act:identity
[TL] BatchNormLayer model/batch_norm1: decay:0.900000 epsilon:0.000010 act:identity is_train:False
[TL] PoolLayer model/pool_layer1: ksize:[1, 3, 3, 1] strides:[1, 2, 2, 1] padding:SAME pool:max_pool
[TL] Conv2dLayer model/cnn_layer2: shape:[5, 5, 64, 64] strides:[1, 1, 1, 1] pad:SAME act:identity
[TL] BatchNormLayer model/batch_norm2: decay:0.900000 epsilon:0.000010 act:identity is_train:False
[TL] PoolLayer model/pool_layer2: ksize:[1, 3, 3, 1] strides:[1, 2, 2, 1] padding:SAME pool:max_pool
[TL] FlattenLayer model/flatten_layer: 65536
[TL] DenseLayer model/relu1: 384 relu
[TL] DenseLayer model/relu2: 192 relu
[TL] DenseLayer model/output_layer: 10 identity
param 0: (5, 5, 3, 64) model/cnn_layer1/W_conv2d:0
param 1: (64,) model/batch_norm1/beta:0
param 2: (64,) model/batch_norm1/gamma:0
param 3: (64,) model/batch_norm1/moving_mean:0
param 4: (64,) model/batch_norm1/moving_variance:0
param 5: (5, 5, 64, 64) model/cnn_layer2/W_conv2d:0
param 6: (64,) model/batch_norm2/beta:0
param 7: (64,) model/batch_norm2/gamma:0
param 8: (64,) model/batch_norm2/moving_mean:0
param 9: (64,) model/batch_norm2/moving_variance:0
param 10: (65536, 384) model/relu1/W:0
param 11: (384,) model/relu1/b:0
param 12: (384, 192) model/relu2/W:0
param 13: (192,) model/relu2/b:0
param 14: (192, 10) model/output_layer/W:0
param 15: (10,) model/output_layer/b:0
num of params: 25349770
layer 0: Tensor("model/cnn_layer1/Identity:0", shape=(28, 125, 125, 64), dtype=float32, device=/device:CPU:0)
layer 1: Tensor("model/batch_norm1/Identity_2:0", shape=(28, 125, 125, 64), dtype=float32, device=/device:CPU:0)
layer 2: Tensor("model/pool_layer1:0", shape=(28, 63, 63, 64), dtype=float32, device=/device:CPU:0)
layer 3: Tensor("model/cnn_layer2/Identity:0", shape=(28, 63, 63, 64), dtype=float32, device=/device:CPU:0)
layer 4: Tensor("model/batch_norm2/Identity_2:0", shape=(28, 63, 63, 64), dtype=float32, device=/device:CPU:0)
layer 5: Tensor("model/pool_layer2:0", shape=(28, 32, 32, 64), dtype=float32, device=/device:CPU:0)
layer 6: Tensor("model/flatten_layer:0", shape=(28, 65536), dtype=float32, device=/device:CPU:0)
layer 7: Tensor("model/relu1/Relu:0", shape=(28, 384), dtype=float32, device=/device:CPU:0)
layer 8: Tensor("model/relu2_1/Relu:0", shape=(28, 192), dtype=float32, device=/device:CPU:0)
layer 9: Tensor("model/output_layer/Identity:0", shape=(28, 10), dtype=float32, device=/device:CPU:0)
learning_rate: 0.000100
batch_size: 28
n_epoch: 50, step in an epoch: 0, total n_step: 0
Epoch 0 : Step 10-10
train loss: 0.963498
train acc: 0.682143
test loss: 0.954065
test acc: 0.678571
Epoch 1 : Step 20-20
train loss: 0.091543
train acc: 1.000000
test loss: 0.635732
test acc: 0.785714
Epoch 2 : Step 30-30
train loss: 0.011198
train acc: 1.000000
test loss: 0.714634
test acc: 0.714286

```

Figure B.1: showing running output



# Bibliography

Flint, G. (1989). Seizures and epilepsy.. Seizures and epilepsy /. F.A. Davis Co. .

Chen, J., Li, X., Mi, X., & Pan, S. (2015). A high precision EEG acquisition system based on the CompactPCI platform. International Conference on Biomedical Engineering and Informatics (pp.511-516). IEEE.

Siuly. (2012). Analysis and classification of eeg signals.

Smith, E.J.INTRODUCTION TO EEG. <http://www.ebme.co.uk/articles/clinical-engineering/56-introduction-to-eeg/>

Siuly Siuly, Yan Li, Yanchun Zhang. "EEG Signal Analysis and Classification", Springer Nature, 2016

Niedermeyer E. and Lopes da Silva F. (2005) Electroencephalography: basic principles, clinical applications, and related fields, Lippincott Williams Wilkins, ISBN 0781751268, 5th edition, 2005.

Sun et al.(2014): A Review of Adaptive Feature Extraction and Classification Methods for EEG-Based Brain-Computer Interfaces, , IJCNN, 2014

Rahman et al.(2012): A comprehensive survey of the feature extraction methods

in the EEG research, ICA3PP, 2012

Amin et al.,(2015): Feature extraction and classification for EEG signals using wavelet transform and machine learning techniques, Volume 38, Issue 1, pp 139149, 2015

Al-Fahoum A S, Al-Fraihat A A(2014): Methods of EEG signal features extraction using linear analysis in frequency and time-frequency domains[J]. ISRN neuroscience,2014.

Li D, Xie Q, Jin Q, et al.,(2016): A sequential method using multiplicative extreme learning machine for epileptic seizure detection[J]. Neurocomputing, 214: 692-707.

Siuly(2012): Analysis and classification of EEG signals[J].

Berger, H. (1929): Uber das Elektroenkephalogram des Menschen, Archives of Psychiatry, Vol. 87, pp. 527-70.

Bernier, R., Dawson, G., Webb, S., & Murias, M. (2007, August): EEG mu rhythm and imitation impairments in individuals with autism spectrum disorder. Brain and cognition, 64(3), 22837.

Jayaraman, V., Sivalingam, S., & Munian, S. (2014). Analysis of real time eeg signals. Department of Physics & Electrical Engineering.

Ohtahara. (2003): J Clin Neurophysiol, Volume 20(6).November/December.398-407

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Li, F. F. (2014). Large-Scale Video Classification with Convolutional Neural Networks. IEEE Conference on Computer Vision and Pattern Recognition (pp.1725-1732). IEEE Computer Society.

Poolman, P., Muller, G. J., & Keller, B. M. (2010). Observation device and method. US, US20100280372.

Vecchio, F., Miraglia, F., Bramanti, P., & Rossini, P. M. (2014). Human brain networks in physiological aging: a graph theoretical analysis of cortical connectivity from eeg data. *Journal of Alzheimers Disease* *Jad*, 41(4), 1239.

Lecun, Y., Haffner, P., Bottou, L., on, & Bengio, Y. (1998). Gradient-based learning applied to document recognition.

Jake Bouvrie. (2016). Notes on Convolutional Neural Networks. Massachusetts Institute of Technology Cambridge, MA 02139.

karpathy. (2016) <http://cs231n.github.io/convolutional-networks/>

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *International Conference on Neural Information Processing Systems (Vol.25, pp.1097-1105)*. Curran Associates Inc.

Ghahreman, N., & Dastjerdi, A. B. (2011). Semi-automatic labeling of training data sets in text classification. *Computer & Information Science*, 4(6), 48-56.