Cardiff University

School of Computer Science and Informatics

PROJECT TITLE

SALIENCY PREDICTION BASED ON DEEP LEARNING

CMT403 – Computing Dissertation

Supervisor: Dr. HANTAO LIU

Author: XIAO LIU

Abstract

With the development of artificial intelligence technology, the application of salient object detection in computer vision has become more and more extensive, and gradually has become a very challenging task. According to the requirements of the task, the algorithm needs to detect the most salient object in an image. It is very helpful to get the interpretability of an image. So it can be used in many image processing tasks. In this task, we summarize the predecessors of fixation prediction scientific researches and combine with state-of-the-art deep artificial neural research and technologies and propose three different distortions on the given datasets. We test the three different neural network algorithm with those distortions on CAT2000 to detect salient objects.

Acknowledgment

In my postgraduate life, I sincerely thank Dr.HANTAO LIU for his concern for my studies, for supporting and helping me when I faced a decision, caring for me like a family member, and guiding me patiently whenever I encounter problems in scientific research, Dr.HANTAO LIU will often discuss with us new ideas in the field of scientific research, which broadens my understanding of my own scientific research field.

Table of Contents

Abstract 2
Acknowledgment 3
List of Figures 5
List of Tables 6
Introduction 7
1.1 Problems 7
1.2 Aims and Objectives 8
1.3 Scope 8
1.4 Contributions 9
2.1.1 Image based salient object detection algorithm 12
2.1.2 Deep learning based salient tracking algorithm 13
2.1.3 Correlation filter-based tracking algorithm 15
2.1.4 Feature representation-based salient tracking algorithms 16
2.2 Current Research Status 16
Methods 18
3.1 Convolutional Neural Network 18
3.1.2 Classical Convolutional Neural Network Architectures 21
3.5 Data Collection 30
3.6 Graphical transformation 31
3.7 Evaluation 34
Results and Analysis 36
4.1 Experiment Setting 36
Conclusion and Discussion 44
5.1 Conclusion 44
5.2 Discussion 44

List of Figures

Fig.1 Conspicuous object detection illustration

Fig 3.1 CNN

Fig 3.1.2 Convolution operation

Fig3.1.3 Maximum Pooling

Fig 3.1.4 VGG Structure

Fig 3.1.5 GoogleNet Structure

Fig 3.1.6 ResNet Structure

Fig 3.2 RNN Structure

Fig 3.5 CAT2000 Datasets

Fig 3.6.1 Image size transformation

Fig 3.6.2 Image Rotate Transformation

Fig 3.6.3 Image color extraction

Figure 4.1 Original Data Training

Figure 4.2 Data Training with rotation.

Figure 4.3 Data Training with Horizontal Flip.

Figure 4.4 Data Training with Vertical Flip.

Figure 4.5 NSS Result in CNN..

Figure 4.5.1 NSS Result in all models.

Figure 4.5.2 MSE Result in all models.

Figure 4.5.3 KL Result in all models.

List of Tables

Table.1 CNN Result with single channel

Table.2 CNN Result with three channels

Table.3 LSTM Result with single channel

Table.4 LSTM Result with three channels

Table.5 LSTM+CNN Result with single channel

Table.6 LSTM+CNN Result with three channels

Chapter I.

Introduction

1.1 Problems

With the rapid development of the Internet since the 21st century, artificial intelligence technologies have been applied to various research and learning fields. Among them, computer vision, as an important research direction in the field of artificial intelligence, aims to make computer systems have a clear perception of the real world like humans by learning human visual perception, so that they can process and analyze image and video information efficiently. Advanced computer vision technologies enable intelligent tasks related to the vision domain, such as detection and recognition of objects or scenes in the objective world, which has led to a wide range of industrial applications such as automatic/assisted driving navigation and human-computer interaction. Nowadays, in the face of images and videos with a large amount of complex information, the hand-based feature design approach is no longer sufficient for effective information processing efficiency, so targeted analysis and extraction of key information from images and videos has become an effective solution, which has attracted the attention of a large number of related technology researchers, brain scientists and cognitive psychologists. According to a large number of relevant technical studies, it has been found that the human visual system actively focuses on the most attractive targets or points of interest in images and videos. Under free conditions, humans undergo the visual system to quickly perceive the scene information and important targets around themselves, such as object line features, object size and other attributes, and in order for computer systems to have this recognition and detection capability at the same time, relevant researchers have modeled it as a salient object detection task.

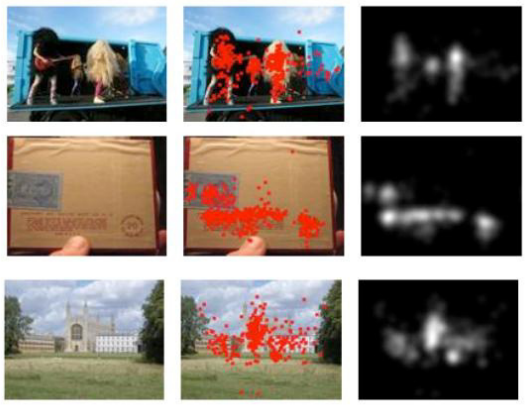

Fig.1 Conspicuous object detection illustration

Salient object detection technique refers to the extraction of the most obvious and meaningful targets from an image or video by an intelligent algorithm, as shown in Figure 1.1, where the white area in the true value map with a binary image shows the salient objects of the original image. Salient object detection has become an effective and mature technique in the field of computer vision and is widely used in tasks such as image recognition [63], image compression [64], and visual tracking [65]. The main work of this paper is to investigate the salient object detection methods based on the problems existing in images and videos, respectively.

1.2 Aims and Objectives

The aim of this project is:

1. Apply different distortions on given datasets.
2. Performing experiments on 3 different saliency prediction models with each distortions.
3. Evaluate each results.

1.3 Scope

The scope of this project includes:

1. Datasets training and evaluation on CAT2000 dataset.

2. Build deep learning tools in python 3.7.5 with pyTorch as the main library.
3. Evaluating the performance using MSE and MAPE.

1.4 Contributions

Saliency prediction tasks are of great research importance, and vision tasks such as Pattern Recognition, Action Recognition, and Face Recognition can be further investigated based on models of saliency prediction. Therefore, saliency prediction has a wide range of applications in advanced computer vision tasks such as Object Recognition [1,2], Video Understanding [3,4], Video Tracking [5], Image Captioning [6], etc. In this task, we are going to contribute on following areas.

(1) Target Recognition

Target recognition is a fundamental research in the field of computer vision, and its core task is to identify which objects are in an image, which class they belong to, and to report the specific location of these objects in the image. Face recognition, license plate recognition, text recognition, etc. are all target recognition techniques. A typical target recognition algorithm consists of several stages: image pre-processing, feature and selection, modeling, matching and localization. In the face of complex background and huge data volume, fast extraction of candidate regions from them is the key to improve the efficiency of target recognition algorithms. Using saliency prediction as a front-end to quickly extract the selection candidate locations before recognition can greatly speed up the recognition of objects in cluttered scenes and reduce the false detection rate.

(2) Image description generation

Image description generation is an integrated problem involving machine learning, computer vision and natural language processing, whose goal is to translate an image into a descriptive text and generate human-readable sentences. The task involves both using computer vision models to understand the content of the images and natural

language algorithms to express the relationships between them. How to accurately capture the semantic information of an image is the basis of image description generation techniques. The use of saliency prediction can help computers to capture the semantic information of images more quickly and accurately, and make image description generation more accurate.

We make the comparisons of performance on three models, CNN, LSTM and CNN+LSTM. We also make our analysis on the performance among those models.

Chapter II.

Background Material

2.1 Related Work

The concept of saliency map was first proposed by C. Koch and S. Ullman in 1985 when they analyzed the visual system of primates [66]. In 1998, Itti et al [47] proposed a saliency computation model by computing the difference between the features of the central region of the saliency map and its edge features. in 2007, Liu et al [68] implemented salient object detection by separating the salient object from the background region.

The purpose of salient object detection is to identify the most obvious objects or regions from a complex scene by simulating the visual nervous system in the biological world, thus enabling computer systems to detect salient objects in images or videos. From the perspective of data processing, salient object detection tasks can be broadly classified into image salient object detection, collaborative salient object detection and video salient object detection. The first one is a data-driven bottom-up salient object detection method, which uses a computer model designed mainly based on the attention-driven operation of low-level features, and it is studied from the perspective of attentional motivation. The second is a task-driven top-down detection approach, which generates attention based on the impressions, experiences, and emotions of the experimental individuals, and it investigates the algorithmic problem from an endogenous perspective of attention.

The research goal of video salient object detection is to identify and detect the most noticeable objects or objects in a video. Only by effectively combining temporal and spatial information in the video can the detection performance of the algorithm be improved. Compared with image data, video data has more complex changing scenes, larger data size and more target interference, thus making the video salient object detection task more difficult.

2.1.1 Image based salient object detection algorithm

The idea of a salient object detection algorithm based on hand-designed features was introduced in the 1880s by the feature integration theory proposed by Treisman et al [5]. This theory suggests that when the combination of multiple separable features is needed to characterize or distinguish between presented possible objects, attention must be directed to each stimulus in reality in succession so as to achieve the goal of mimicking the image perceived by the animal. Based on this, Itti et al [47] designed the first saliency computational model, whose design idea was derived from the behavioral activity and neuronal cell structure of the animal visual system, using features such as contrast, orientation, and color of the image to extract information from the input model, converting the salient object detection problem into a feature fusion problem to generate a prediction map. As inspired by this approach, a large number of salient object detection algorithms have improved this accordingly from four perspectives based on frequency domain analysis, information theory, sparse representation, and visual a priori.

Among the classical visual target tracking algorithms, they can be divided into generative and discriminative model-based algorithms. Among them, generative model-based tracking algorithms are able to generate suitable appearance models for describing target features based on the extracted features of the target, which has the advantage of finding the region of the image that best matches the model and thus can track the target effectively, but also has the disadvantage of easily ignoring the target background information. The visual target tracking algorithms based on generative models are subspace learning [11], particle filter [12], optical flow algorithm [13], meanshift [14], Kalman filter [15], sparse expression method [16], and kernel density estimation method. Discriminative model-based tracking algorithms can either use information about the tracked object or incorporate relevant information from the scene, and it needs to train a classifier to distinguish the target from the scene. The visual target tracking algorithms in recent years have been based on discriminative

methods, which mainly include Boosting algorithms [17], neural networks, SVM (Support Vector Machine) [18], linear regression and decision trees [19], etc., and have achieved good results in tracking.

2.1.2 Deep learning based salient tracking algorithm

Deep learning-based target tracking algorithms can be divided into the following main categories: feature-based representation, Siamese Network-based, Recurrent Neural Network (RNN)-based, Generative Adversarial Network (GAN)-based, and other specific network-based target tracking algorithms. GAN) and other specific network-based target tracking algorithms. Researchers propose that there is an emerging trend of using convolutional neural networks to automatically learn multi-scale features of images to obtain saliency maps. By feeding different levels of image segmentation to a Convolutional Neural Network (CNN) and finally using fully connected layers to determine the significant part of the image, it provides a new direction for saliency detection. A new direction for saliency detection was provided.In 2015, Wang [18] et al. proposed to combine local search features with global search features for image saliency detection. A two-stage neural network is constructed, the first stage by constructing a neural network (DNN-L), the input image is the original image, learning the underlying image information such as texture, local contrast and other information to learn the features, this stage generates a rough saliency map; the second stage constructs a neural network (DNN-G) the input is the output of the first stage, and the final saliency map is determined locally by learning the global contrast. Zhao [19] et al. proposed to construct a two-stage parallel network, which is different from the two-stage network serially proposed by Wang et al. The local context network of this model extracts features in parallel with the global context network, sending the image superpixel blocks to the local network and the whole image to the global network for image feature learning separately, and at the end of the network the feature vectors output from both networks are fused to obtain the the final salient map.

In 2016, Lu [20] et al. first proposed to combine image saliency prior with convolutional neural network and use recurrent full convolutional neural network for image feature learning. The model proposed to first hyper-segment the image, perform regional contrast analysis to determine the significant regions and form a coarse saliency map, and then input the saliency prior as well as the original map together into the recurrent full convolutional neural network to obtain the final saliency map by adjusting the learning of features through the saliency prior, and the model achieved better results on most of the datasets. In the same year, Li [21] et al. proposed an image saliency detection model based on deep contrast learning features. The model consists of two parts, the first part is to build a full convolutional neural network to learn features to obtain a coarse saliency map at pixel level, and the second part uses a convolutional neural network to perform multi-scale feature learning in local regions of the image to refine the coarse saliency map. Unlike Wang [18] et al. the first part is a multiscale structure that outputs and cascades each layer of the network. In the second part, the convolutional neural network learns hyper-segmentation to obtain three local features and uses spatial pooling for the connection of the network.In 2018, Cheng [22] et al. proposed short connections between layers and multi-level fusion of the side outputs of the network to obtain the final saliency map. This structure can directly cascade the multilevel features learned by the network, and the model achieved better results on multiple datasets.In 2019, Qin [23] et al. pointed out that current image saliency models focus more on the generation of salient subjects and ignore the quality of salient region boundaries. Therefore, a boundary-aware model based on convolutional neural networks was proposed, which consists of a densely supervised encoder decoder and a residual module. The former is used for image saliency prediction, and the latter is applied to the optimization of saliency map region boundaries to make the saliency map boundaries of the network output more clear and coherent.Wang [24] et al. proposed that in image saliency detection, it is most critical to obtain features in the image that

have the ability to represent the salient regions of the image, and the features with multi-scale space usually have stronger representation, so they proposed to apply image pyramids to The image pyramid is therefore proposed to be applied to the image saliency detection model. The model consists of two modules, Pyramid Attention (PA) and Salient Edge Detection (SED), the former can make the model focus more on saliency-related features, and the latter can use the edge information of salient objects to refine the final segmented salient object's Edge information is obtained by predicting the edges of salient objects through supervised learning. Since there are differences between real-world scenes and dataset images Cheng et al. proposed the SOC [25] dataset, which integrates existing dataset images used for image saliency detection and adds some Internet images, and proposed [26], some natural images, which do not contain salient regions. However, the existing model only detects the significant part of the image and does not judge whether the image contains significant or not, so it is proposed that, in the future, the presence of significant subjects should be detected jointly.

2.1.3 Correlation filter-based tracking algorithm

The main advantage of correlation filter-based tracking algorithms is that they can operate at high speed and maintain high performance. Bolme et al [20] proposed the Minimum Output Sum of Squared Error (MOSSE) filter for tracking failure during fast target movement, which can be realized by dynamically readjusting the target as its appearance changes. Henriques et al [21] proposed a minimum output sum of squared error (MOSSE) filter for the case of tracking failure during fast moving targets, which enables dynamic readjustment to track the target as its appearance changes. Henriques et al [21] proposed a kernel detection tracking algorithm using the Circulant Structure of Tracking (CST) to address the situation

where the number of tracking samples is large or small and the tracking is fast or slow, where they used the theory of circular matrices and added fast Fourier transforms for fast learning and detection. Other correlation filtering-based tracking algorithms are improved and optimized based on these, such as context-aware correlation filter tracking [22], correlation filter tracking algorithm by adaptive spatial regularization [23] Correlation Filter Tracking algorithm [23] (Visual Tracking via Adaptive Spatially-Regularized Correlation Filters (ASRCF), Multi-task Correlation Particle Filter tracking algorithm [24] (MCPF), Multi-task Correlation Particle Filter, MCPF), Dynamic Round-Time Algorithm (DRT) and Real-Time Visual Object Tracking Algorithm [26] (Series-Parallel Matching Tracker, SPMT), etc.

2.1.4 Feature representation-based salient tracking algorithms

In feature representation-based target tracking algorithms, optimal training is still mainly achieved by enhancing the ability to represent deep features and shallow feature information to improve tracking results.Sun et al [27] proposed LSART (Learning Spatial-Aware Regressions for Visual Tracking) using a full convolutional algorithm with a spatial regularization kernel, which allows the filter kernel corresponding to each output channel to be forced to focus on a specific region of the target, and then further uses a pool of distance transformations to determine the effectiveness of each output channel of the convolutional layer.Chen et al [28] proposed a simple and effective visual tracking framework, SiamBAN (Siamese Box Adaptive Network) that directly classifies the target and is able to return to its bounding box, avoiding the hyperparameters associated between the target and the candidate box, making the visual detection effect more flexible and general.

2.2 Current Research Status

Since biologists proposed the visual attention mechanism in the 1980s, scholars have devoted themselves to the study of image saliency. After about half a century of technical development, many excellent image saliency detection algorithms have been produced in both image domain, frequency domain or deep learning field, and these algorithms have achieved better results on image saliency detection datasets. However, unlike dataset image scenes that are more homogeneous, in real scenes, the backgrounds of images are more complex. The existing algorithms still face more challenges on images with complex backgrounds, mainly including the following three points.

(1) image saliency detection algorithms in traditional domains have been proven to be effective in a variety of settings, and their saliency maps can be used as significant priors for deep learning algorithms, but convolutional neural network-based algorithms are not sufficiently combined with traditional algorithms.

(2) When the image background is more complex, the convolutional network continuously downsampling will lose image information as the number of network layers deepens, so the output map of the neural network will have the problem of judging non-significant regions as significant regions.

(3) When the foreground of the image, that is, the significant target, has a small contrast with the background, the edge features are fewer, and the features that can be learned by the network are reduced accordingly, so the significant map will not find the exact significant subject and the subject edge incoherence problem. Therefore, this paper takes the above three problems as the starting point to improve the quality of salient maps, and proposes to use the salient maps of traditional algorithms as the image prior of convolutional neural networks to increase the image information of the input network; to splice the deep and shallow layers of the network, so that the shallow maps containing edge textures and the deep maps containing semantic information can be side-by-side fused to increase the learning ability of the network; to increase the foreground background edge using the dataset preprocessing contrast for dataset enhancement.

Chapter III.

Methods

3.1 Convolutional Neural Network

A convolutional neural network (CNN) consists of a convolutional layer, an activation function, a pooling layer and a fully connected layer, which are superimposed to build a complete network model. The network features local connectivity of neurons and shared weights, and the network architecture is shown in Figure 3.1. In the 2012 ImageNet [27] classification task competition, AlexNet [28] based on deep learning won the first place, beating all the traditional classifiers at that time, demonstrating the power of deep neural networks, so that scholars have turned their attention to deep learning technology, and later used deep neural networks in the field of computer vision, with the neural network The feedforward mechanism of the neural network is to convolve the input image through the convolution kernel to extract the features needed for the task; and the backpropagation mechanism of the network updates the parameters of the convolution kernel to make the kernel respond to specific features to achieve the corresponding visual detection task. In this section, we introduce the basic layers of the convolutional neural network used in this paper.

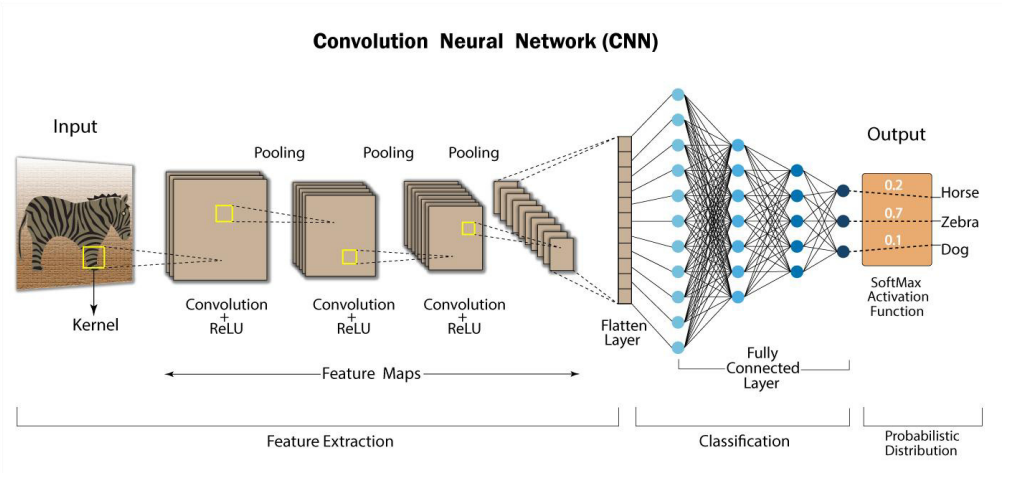

Fig 3.1 CNN

(1) Convolutional layer

The convolution operation has the role of extracting image features, which is implemented by convolution kernels. One or more convolutional kernels together form a convolutional layer. Typically, a convolution kernel includes a kernel size, a step size, and a number of filled pixels. Among them, the convolution kernel size indicates the range of regions where the pixel points of the feature image are mapped in the input image. Usually, the larger the convolution kernel is, the wider the range of regions mapped, and therefore the better the global features obtained. However, it also causes the algorithm and the model to become more computationally intensive, thus reducing the computational performance. The step size indicates the displacement of the convolution kernel. The number of filled pixels is generally done to avoid the padding process when the image size is not the same as the original size after convolution. The two commonly used convolution operations are standard convolution and hole convolution, and their convolution methods are shown in Figure 2.2. The null convolution operation can avoid the information loss problem caused by pooling operation and can increase the mapping area of the pixel points of the feature area to the input image.

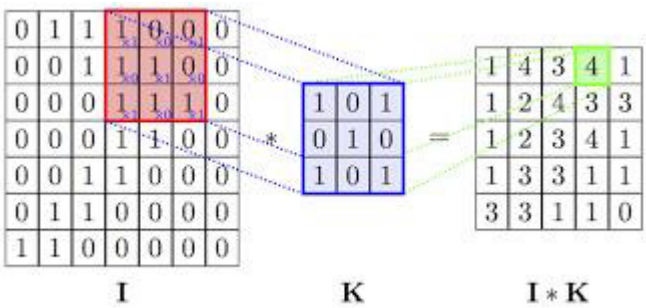

Fig 3.1.2 Convolution operation

(2) Activation Function

The role of the activation function is to be responsible for the correspondence between the input layer features and the output layer features in the neural network, and it can increase the nonlinear function of the model. The most frequently used activation functions in neural networks are sigmoid, relu and tan*h*, which have different usage characteristics and different applicability ranges. sigmoid is characterized by constraining the output size to (0,1) and tan*h* is to constrain the output size in the range of (-1,1), therefore, the activation functions sigmoid and tan*h* are more applicable in the context of probability value processing. Relu is constraining the output size to the range (0, +∞) without the maximum limit, and is therefore suitable for training deep network models. The equation representation is shown in Eqs. (2.1), (2.2), and (2.3), respectively.

$$Sigmoid(x) = \frac{1}{1+e^{-x}} \cdots\cdots (2.1)$$

$$tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \cdots\cdots (2.2)$$

$$ReLU(x) = max\ (0, x) \cdots\cdots (2.3)$$

(3) Pooling layer

The pooling layer has the function of compressing the number of parameters and feature data and avoiding overfitting during the model training process. Pooling is the process of abstracting the information, which can increase the range of regions where the pixel points of the feature image are mapped in the input image and enhance the translation invariance of the pixels to a certain extent. The two most common pooling operations are average pooling and maximum pooling. One of the maximum pooling operations is shown in Figure 3.1.3.

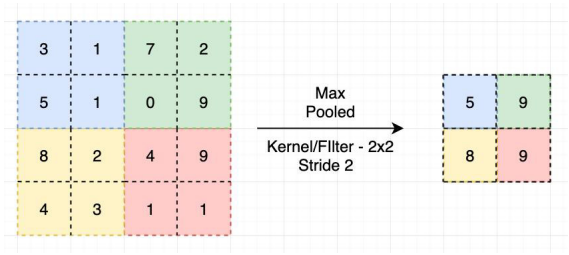

Fig3.1.3 Maximum Pooling

(3) Fully connected layer

The fully-connected operation is usually found at the end of the network model and has the role of a "classifier". The presence of a fully-connected layer allows the model to use the size of the input image with some flexibility, which can be achieved by the convolution operation. It affects the model with three parameters, namely its length, width and activation function.

3.1.2 Classical Convolutional Neural Network Architectures

In the era of artificial intelligence, the development of convolutional neural networks has a pivotal position. Convolutional neural networks are divided into shallow convolutional neural networks and deep convolutional neural networks, but shallow convolutional neural networks have corresponding limitations, both in accuracy and expressiveness are relatively weak, and are very little used in practical development. The most popular neural networks currently are deep convolutional neural networks. The basic CNNs consist of three structures: convolutional layer, activation function and pooling layer. The important work of convolutional neural networks is how to efficiently iteratively update the whole network parameters with the already input training data, a process called backward propagation. The most used deep convolutional neural networks are VGG, GoogleNet, ResNet, etc. are combined from simple CNN tuning [35-37]. In this paper, ResNet is used as the backbone network for

video salient region detection, while the extracted spatial edge features guide the subsequent feature extraction, and the channel attention mechanism is introduced to improve the detection accuracy of the video saliency detection model, and the three mentioned network models are described in detail below.

(1) VGGNet

VGGNet focuses on the correlation between depth and performance of convolutional neural networks, and makes a breakthrough in this direction. The authors cleverly used small 3×3 convolutional kernels instead of large 5×5 and 7×7 convolutional groups, which consisted of 2 to 4 convolutional layers, and this convolutional group structure increased the nonlinear structure of the model, along with the use of 2×2 maximum pooling layers, thus building a deep convolutional neural network with 16 to 19 layers. In 2014, VGGNet won the ILSVR competition for localization, and VGGNet is still one of the most popular deep convolutional neural networks. The deepening of the convolutional neural network will improve the performance of the model, and all of VGGNet will use 3×3 convolutional kernels combined with 2×2 maximum pooling to replace large convolutional kernels. The ingenious design of VGGNet is to use two 3×3 convolutional layers in series as a 5×5 convolutional layer, and three 3×3 convolutional layers as a 7×7 convolutional layer. However, the number of parameters is only half of the previous one while the perceptual field of the model extracts the input signal remains unchanged, which results in a significant reduction in the number of parameters. At the same time, the use of small convolutional kernel sets effectively increases the binning operation of the convolutional neural network, making the model more capable of learning. In addition, the VGG network differs in that the spatial resolution of the feature map tends to be monotonically decreasing, while the number of channels of the feature map tends to be monotonically increasing Figure 2.6 illustrates the structure of VGGNet.

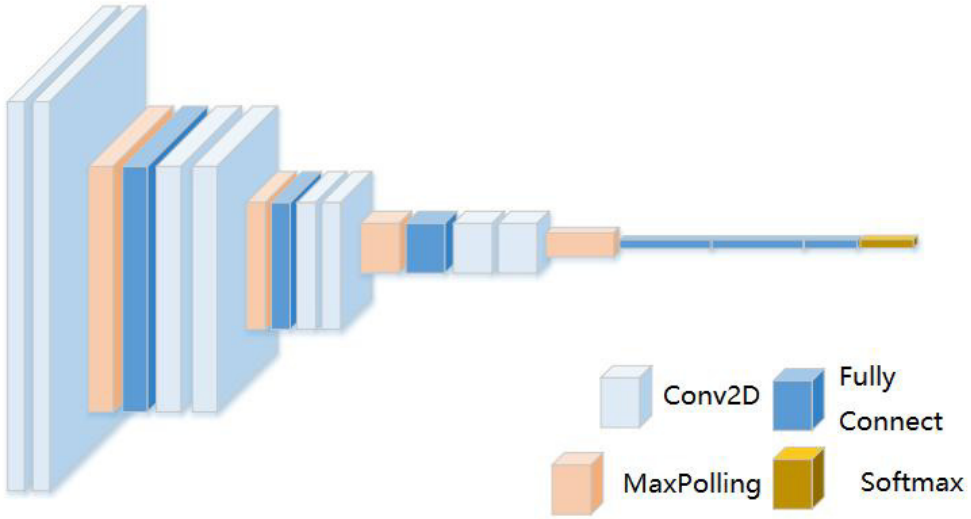

Fig 3.1.4 VGG Structure

(2) GoogleNet

GoogleNet was proposed by Christian Szegedy in 2014, which is a new deep learning network structure, before that most convolutional neural networks increase the depth of the network to improve the performance of the model, but the increase of the number of layers of the network model will have a series of problems, such as gradient explosion, gradient disappearance, etc. GoogleNet solves these problems from another perspective. The base module of GoogleNet is the Inception module, and the structure of Inception module is shown in Figure 3.1.3. The module uses 1×1 and 3×3 convolutional kernels repeatedly to traverse the whole network, three different sizes of convolutional kernels are used by the Inception module, and then pooling operations are performed on three different convolutional layers in turn. The perceptual field of extracted features will cover most of the image area, and the introduction of ReLu at the end of the convolutional layer will increase the nonlinear expression of the model, while the 5×5 convolutional kernels will increase the cost of model computation and make the network training difficult, so the authors add a 1×1 convolutional kernel before the 3×3 convolutional kernels and after the maximum pooling to reduce the feature map dimension respectively.

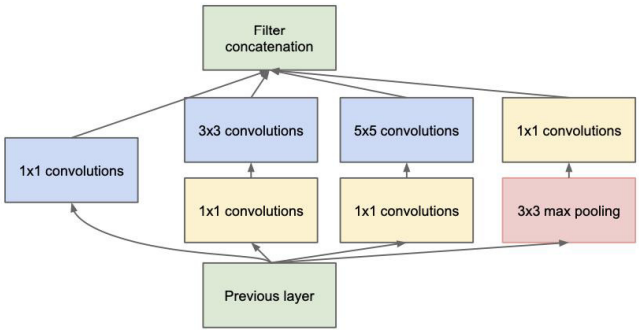

Fig 3.1.5 GoogleNet Structure

(3) ResNet

ResNet was proposed in 2015 and became the most popular convolutional neural network model today. VGGNet tried to explore the boundary of network depth and deepen the network to 19 layers, but it was later shown that adding more layers to the 19-layer network would not improve the performance of the network but would not converge easily during the training process. Experiments show that the reason why the number of layers cannot be deepened is that the deepening of the number of layers will increase the training error of the model, and the gradient will not be assigned to the shallow network when the model is back propagated, so the parameters of the shallow network cannot be updated, and the above phenomenon is the common gradient disappearance, even using random deactivation and other tuning techniques can not solve the above problem. The root cause is that as the number of network layers deepens, the number of network parameters is too large to be optimized, which makes the network model ineffective. In order to solve the problems caused by the deepening of the network, ResNet adopts the idea of residuals and further applies it to the convolutional neural network model. A large number of experiments have shown that learning the residuals with parametric layers is faster than learning the inputs directly, and the learning efficiency is greatly improved.

ResNet is a deep residual network proposed by Kaiming He et al [14]. ResNet is composed of a stack of residual blocks, and each residual block consists of two

aspects, x and f(x), as shown in Figure 2.4. x is called the "shortcut" path and is used for constant mapping. f(x) is called the residual path, which is used to fit the residuals. The residual paths are broadly divided into two types, with and without bottleneck. The bottleneck-free structure is achieved by stacking two 3 × 3 convolutions, as shown in Figure 2. The no-bottleneck structure is achieved by stacking two 3 × 3 convolutions, as shown in Figure 2.4, left. The ResNet authors provide five structures with different depths, namely ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152. The network model in chapter 3 of this paper takes ResNet50 as the backbone The network model in Chapter 3 of this paper takes ResNet50 as the backbone network, and the network model in Chapter 4 takes ResNet34 as the backbone network. At present, the maximum depth of the network has exceeded 1000 layers, which is important for the development of convolutional neural networks.

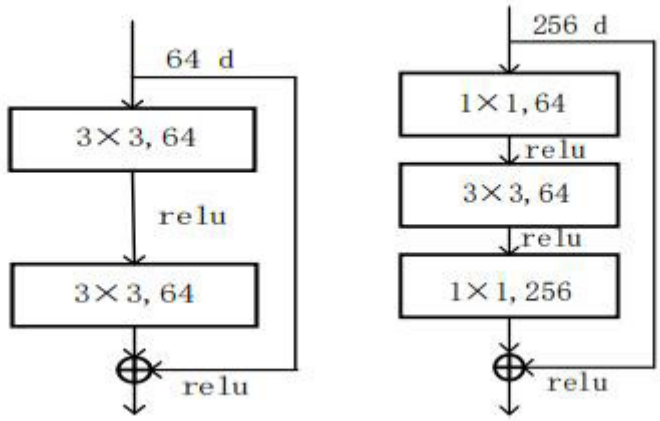

Fig 3.1.6 ResNet Structure

3.2 Recurrent Neural Network

Recurrent Neural Network (RNN) is good at processing variable-length sequence data, which can be Variable length series data are data collected at different points in time, which reflect the change of something over time. The common feature of the data to be processed is that the information at the back of the data has a very strong relationship with the information at the front. The reason why RNNs are so powerful

in processing sequential data is that the structure of RNN networks contains a number of components that are used in the process. This is because the RNN network structure contains hidden layers composed of cyclically connected nodes. These hidden layers have the function of remembering the information already processed These hidden layers have the function of remembering the processed information and filtering and retaining the processed information, so that the correlation between the previous and subsequent information can be effectively improved when processing the later information. These hidden layers are often referred to as stored states. RNNs are trained to correlate sequential information through the output of the previous time step, and RNNs can be previously valued thanks to such recurrent model network. Figure 3.2 illustrates the inner workings of the RNN.

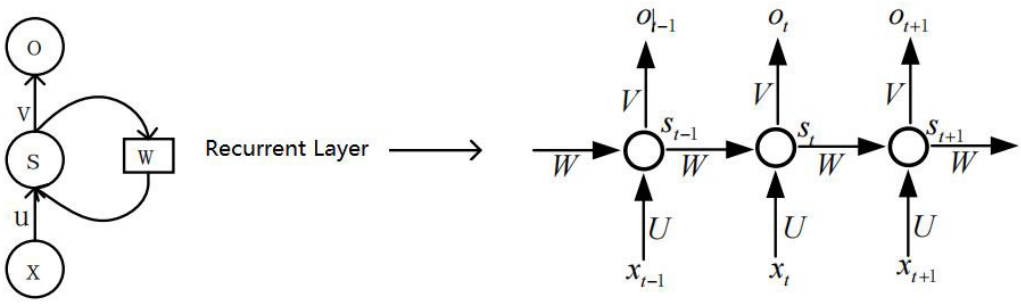

Fig 3.2 RNN Structure

where Xt denotes the input [X0, X1,...,Xn] vector at time t, U denotes the weight matrix from the input layer to the hidden layer, St is the value of the hidden layer at time t, W is the weight matrix when the value of the hidden layer at the previous moment is passed to the hidden layer at the next moment, V is the weight matrix from the hidden layer to the output layer, and Ot is the output of the RNN network at time t. In the single-input feedforward neural network structure, the activation netj in the hidden layer at a certain moment is expressed by following equation

$$net_j = \sum_i^n x_i(t) v_{ji} + \theta_j$$

The RNN can establish relationships between associated samples so that there are connections between neurons, and the hidden layer can store the output information of

the output of the previous layer so that the output of the previous storage will also have an impact on the next data when it is processed. However, due to the constraint of RNN network structure in the gradient learning algorithm, the RNN structure can only capture short-term dependencies in sequential data, but still has no way to capture long-term dependencies in the sequence. The exponential decay of the gradient during propagation weakens the information storage capacity of the RNN, so that the ability of the RNN to capture information is compromised.

3.3 Long Short Term Memory

The rapid development of artificial neural networks has led to the emergence of intelligence in many industries, for example, deep neural networks are used in vision processing, and the longest used model in the field of speech recognition is the recurrent neural network model, Long Short Term Memory (LSTM) neural network is a deformation structure of RNN, LSTM has memory function for time series, and the traditional LSTM is a deformation structure of RNN, LSTM is a kind of temporal recurrent neural network, compared with the traditional RNN network, LSTM can remember the long term information to see the connection.LSTM module has three main components, namely, input gate, output gate and forgetting gate. The key role of the forgetting gate in LSTM is to partially forget the past information using weight control, which is a bridge between the past information and the future information.

The LSTM model is particularly well structured for processing and predicting time series with intervals and delays. The computation of the LSTM is performed by four fully connected layers, and the internal structure of the LSTM is shown in Figure 2.10. As can be seen from the figure, the topmost horizontal horizontal line across the whole cell controls the addition of information and the deletion of information through gates. The input is the computation result of the previous neuron cell ht-1 and the current input vector xt, which are concatenated and passed through the forgetting gate to generate a 0-1 vector, which is dotted with Ct-1 to obtain the information

retained by the previous neuron cell after the computation. The input gate represents the information to be updated as shown in the figure. The input gate connects the ht-1 with the xt information, and after obtaining the amount of information, it passes through the sigmoid function and finally obtains the output result. The output gate determines the hidden vector ht of the current neural cell output. ht is different from Ct, which is a little more complex. ht is composed of the result of the than function of Ct and the result of the dot product of the output gate.

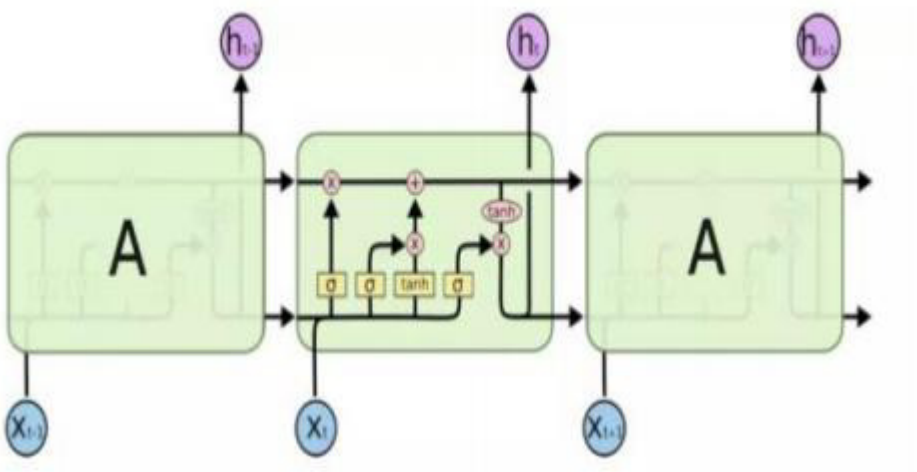

Fig 3.3 LSTM Structure

The role of the input gate is to use the weight to control whether the input information can be input to the current hidden layer, the input gate mainly consists of the Sigmoid function and Tanh function, the information processed by the Sigmoid function is output as it, and the information processed by the Tanh function is output as ct, where it is the gate value of the input gate, the value range is in the [0,1] interval, if the gate value is taken as 1, the input gate accepts all the input data, if the gate value is taken as 0, the input gate does not allow data input, such an operation makes the data in be re-screened again, the input gate function is expressed as

3.4 Convolutional LSTM

The principle of convolutional LSTM is similar to that of LSTM, but the biggest

difference between them is that the multiplication in the convolutional LSTM module is replaced by convolutional operations. operations in the convolutional LSTM module are converted to convolutional operations instead, and the convolutional operations are applied to the sequence model to control the spatial information of each gate input. The convolutional operations are applied to the sequence model to control the spatial information of each gate input. The convolutional LSTM is mainly used for input-to-state and state-to-state transitions. Therefore, the matrix multiplication in the LSTM module The matrix multiplication in the LSTM module is replaced by convolutional operations in the convolutional LSTM module. The convolutional LSTM has been successfully applied to many video detection tasks. The convolutional LSTM also consists of input gates, output gates and forgetting gates, but the difference is that the learning weights of the convolutional LSTM module is a three-dimensional matrix of learning weights.

Channel Attention Module: The channel attention mechanism adaptively calibrates channel feature responses through interdependencies between channels, and uses global information to selectively emphasize informative features and suppress less useful features to achieve better classification results. express the relationship between feature channels and automatically learn the importance of each channel so that the important feature channels get more weight. The channel attention mechanism improves the classification accuracy in all aspects by better describing the relationship between feature map channels, while reducing the effect of background noise and improving the convergence speed.

The advantage of the attention mechanism is that it can be easily accessed in different networks, and unlike changing the width or depth of the network, which can have a huge impact on the network, it does not affect the overall architecture of the network by nature. Inspired by SE Net and Parsenet [58], Hanchao Li et al [59] designed the Pyramid Attention Network (PAN) in which different attentions are introduced in different parts of the pyramid network, which has the effect of not

changing the network width depth and using the attention mechanism alone to improve the pixel-level pixel-level classification accuracy. Combining the attention mechanism with spatial pyramids is used to better represent the higher-level features.

In this task, we are going to use two different type of channels to test our models.

3.5 Data Collection

The training phase of deep learning and the evaluation phase of algorithms require a large number of datasets with annotated information. As an important branch of computer vision, saliency prediction is not used in annotation for other target detection datasets, and its annotation results are not simply annotated with true value boxes, but are divided into two kinds of annotation results: attention points and attention probability. Its annotation method usually uses eye tracking, by which the points that the human eye pays attention to in the first 3-5s when paying attention to an image are automatically recorded, and these points are recorded. In the early years, the saliency dataset had a strong contrast in the foreground background and a single background. In recent years, the background of the saliency dataset has become more and more complex and variable, and the amount of data has increased greatly. The following describes 1 commonly used saliency prediction dataset.

CAT2000 dataset [37]: This dataset contains 4000 images from various categories, consisting of 20 different categories, such as cartoon, art, satellite, low-resolution images, indoor, outdoor, line drawings, etc. Each category has 200 images, regardless of the training set, test set, and validation set. The resolution of the images is 1080×1920 and the ratio is 3:4. The saliency annotation is obtained from the

eye-tracking data of 18 users. Some of the images are shown in Figure 3.5.

Fig 3.5 CAT2000 Datasets

3.6 Graphical transformation

In deep learning based target detection, the target detection form (including localization and classification information) is inferred from the training samples in the training set by the training model under the currently defined network structure (when the type and number of weight parameters in the model are determined), and the difference between the inferred result and the sample label is calculated by the currently defined Loss Function, and then the Weight Parameter in the model is updated by the Back Propagation algorithm. The difference between the inferred result and the sample label is calculated by the currently defined Loss Function, and the Weight Parameter in the model is updated by the Back Propagation algorithm.

However, neural network models in deep learning systems require a large number of parameters, most of which are in the millions or more, and a large amount of data is needed for training to make these parameters work correctly. Therefore, deep

learning models need enough data to support better training. The more the number of training samples, the better the model will fit and the better the generalization ability of the model will be. Therefore, it is necessary and important to get as much high-quality training data as possible.

Image processing is one of the most important application areas of deep learning, and OpenCV is a good helper for deep learning image processing. Given an image, we can transform it in many ways, the most basic ones being color, size, and angle transformation.

(1) Image size transformation

The image size obtained in real environment is often not fixed. However, most deep learning models, there is a requirement for the input size, so it is necessary to unify the input images of different sizes to the same size, and the size transformation method and effect are shown in Figure.

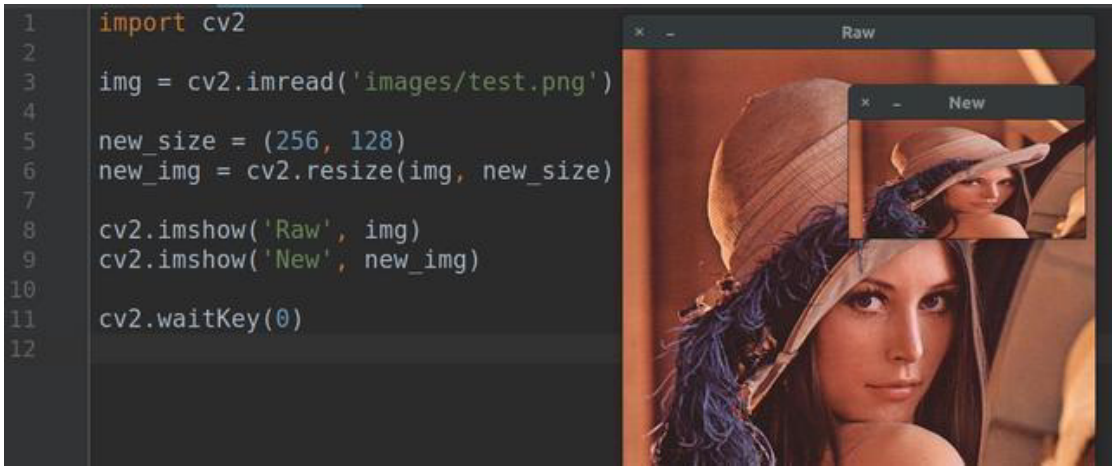

Fig 3.6.1 Image size transformation

（2）Rotation of images

You can rotate an image at any angle in OpenCV, but it involves parameters such as rotation matrix and rotation center, which will be discussed in depth in subsequent articles. In OpenCV, there is a function cv2.flip(img, flipMode) that implements flip, and its flopMode can specify three different modes, the effect of flip is shown in Figure 3, the parameter is specified as follows.

- Greater than 0: flip along the Y-axis, that is, horizontally
- Equal to 0: flip along the X-axis, that is, vertical
- Less than 0: flip both along the X and Y axis.

Fig 3.6.2 Image Rotate Transformation

（3）Image color extraction and combination

An image is composed of RGB (red, green, blue) primary colors, and an image is also represented by an array of three channels in OpenCV. The transformation method and effect are shown in Figure 3.6.3.

Fig 3.6.3 Image color extraction

（4）Image color extraction and combination

3.7 Evaluation

Conventional target detection assessment criteria are not applicable to significance prediction due to the difference in output results. In the course of long-term research, researchers have summarized and proposed specialized evaluation criteria applicable to the use of significance prediction for assessing the goodness of models. The significance prediction evaluation criteria are based on the following four parameters.

(1) Normalized Scanpath Saliency, NSS [38]: the NSS is a metric designed for significance map evaluation, which quantifies the significance mapping values of the gaze points and normalizes them with the significance mapping variance, and is mainly used to evaluate the overall difference between the significance gaze map obtained by the algorithm and the significant true gaze map.

The difference is calculated as shown in the following equation.

$$L_1\left(\tilde{y}, y^{fix}\right) = \frac{1}{N}\sum_i \frac{\tilde{y}_i - \mu(\tilde{y})}{\sigma(\tilde{y})} \cdot y_i{}^{fix}$$

where y-hat denotes the significance prediction map obtained from the test model, y-fix is the labeled true value map of the points of interest, yi is the ith pixel point in the original image, N is the number of all pixel points in the binarized true value map, and u is the point in the significance map normalized to a mean of 0 and a variance of 1. NSS is mainly due to the overall difference between the significance map obtained by the evaluation algorithm and the true map of significance, which is less than or equal to 0 means that the significance model does not work better than the random point model.

(2) Linear Correlation Coefficient (CC) [39]: CC is a statistical method that called

Pearson correlation coefficient, which usually measures the correlation or dependence between two variables.CC can be used to evaluate the direct linear correlation between the predicted eye focus saliency plot and the labeled plot. It is calculated as shown in equation.

$$L_2(\tilde{y}, y^{den}) = \frac{\sigma\ (\tilde{y}, y^{den})}{\sigma\ (\tilde{y}) \cdot \sigma\ (y^{den})}$$

where y-hat denotes the significance prediction plot obtained from the test model, and y-den is the labeled density truth plot for covariance. the maximum value of CC value is 1 and the minimum value is -1. The closer the value is to either the maximum or minimum value, the better the algorithm works. A CC value of 0 indicates that the two significant plots are not correlated at all.

(3) AUC (Area Under the Curve) [40]: among these evaluation methods, AUC is the most widely used evaluation criterion. Since the significance problem is defined as a binary segmentation problem, different thresholds will affect the segmentation results, and in order to better obtain the optimal threshold, the evaluation of the significance problem requires the use of a specialized evaluation method, namely AUC. curve (ROC) [41], the ROC curve refers to the significance prediction graph output by the model as a classifier in the computation process, with True Positive Rate (TPR) and False Positive Rate (FPR) as the horizontal and vertical axes, by varying the significance graph classification threshold, shifting the threshold slowly from 0 to 1 The AUC is the area of the graph enclosed by the ROC curve and the horizontal and vertical coordinates. The use of AUC allows to fully take into account the effect of threshold changes on the results of the segmentation problem, so the evaluation is better. When the value of AUC is 0, it means that the significant graph prediction is completely wrong and the accuracy is 0%; when the value of AUC is 1, it means that the output result is exactly the same as the true value graph and the accuracy is 100%.

Chapter IV.

Results and Analysis

4.1 Experiment Setting

This experiment requires the use of a GPU, in this chapter the GPU is NVDIA GTX 1070ti, the whole experiment is done under Windows Jupyter Notebook operating system, the required software environment includes: Cuda 8.0, Python3.6. Since the use of attention mechanism does not affect the network as a whole, its parameters are set the same as the original network. The initial learning rate is 10-4 when the base network is CNN, and the learning rate is reduced by a factor of 10 every two time points.

The data transformation process with different distortions will listed in the following figure:

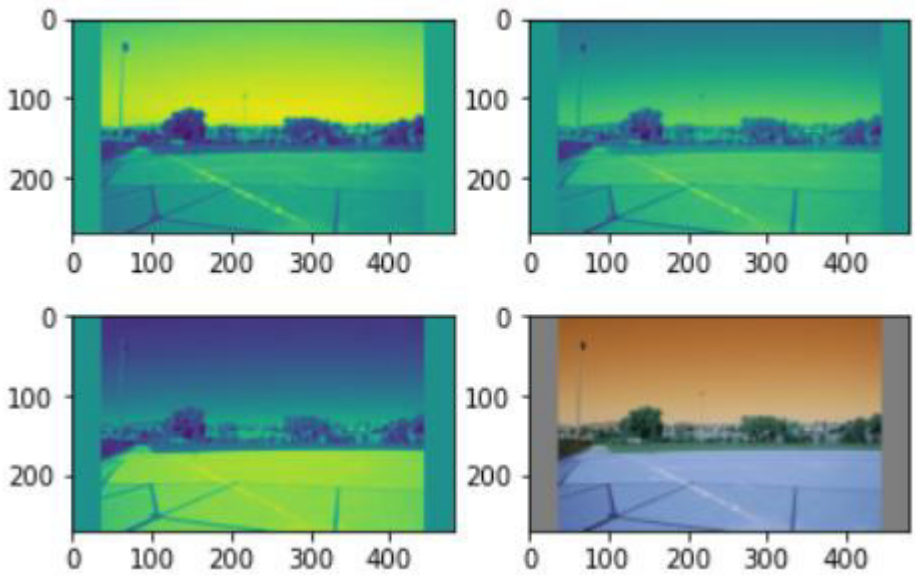

Figure 4.1 Original Data Training

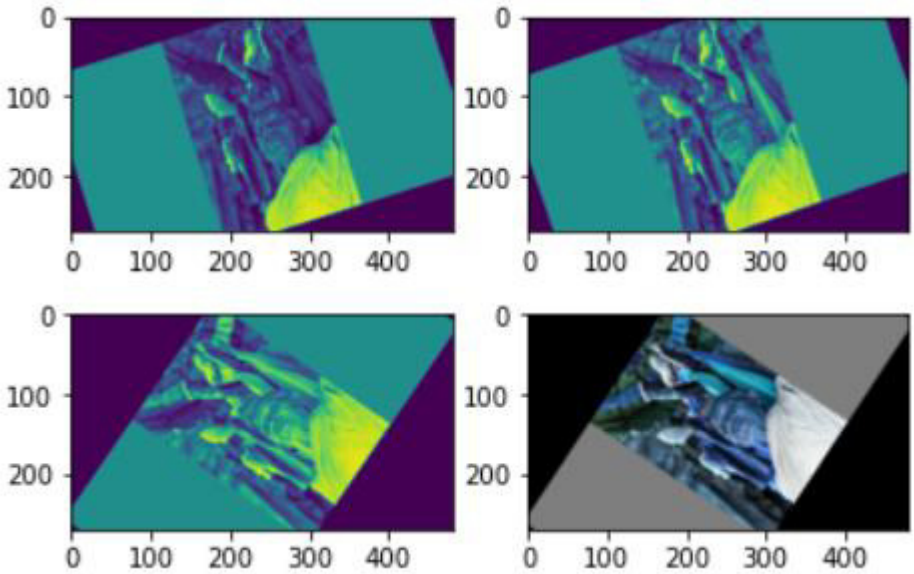

Figure 4.2 Data Training with rotation.

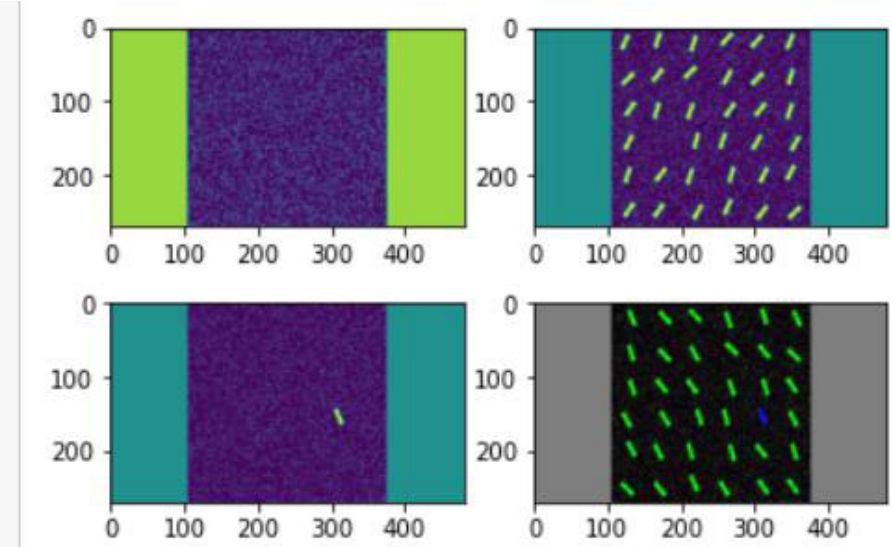

Figure 4.3 Data Training with Horizontal Flip.

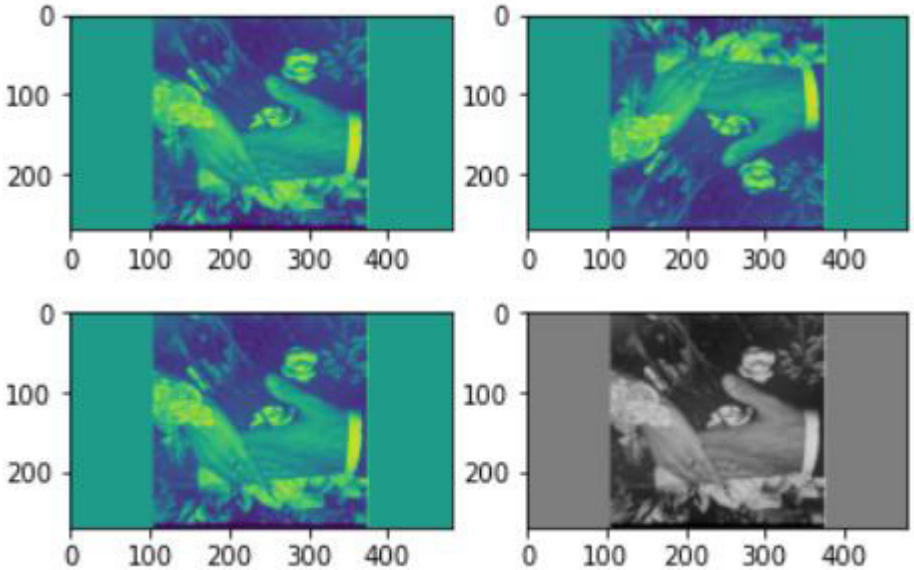

Figure 4.4 Data Training with Vertical Flip.

As we can see, the three distortion methods perform different changing through the datasets. Particularly, the horizontal flip only shows the little 'lines' since it shows the mirror of the images.

4.2 CNN Model Result

The result of CNN model of single channel with different distortion will be

shown in table 1:

	Original	Rotation	Horizontal Flip	Vertical Flip
NSS	0.3270	0.2952	0.3891	0.3811
MSE	0.0680	0.0555	0.0465	0.0474
KL-Divergence	7.4281	8.4843	7.1727	7.0168

Table.1 CNN Result with single channel

The result of CNN model of three channels with different distortion will be shown in table 1:

	Original	Rotation	Horizontal Flip	Vertical Flip
NSS	0.5005	0.3021	0.3991	0.3943
MSE	0.0393	0.0548	0.0459	0.0456
KL-Divergence	5.9411	13.8	6.9878	6.6939

Table.2 CNN Result with three channels

We also plot the result for two different channels based on NSS.

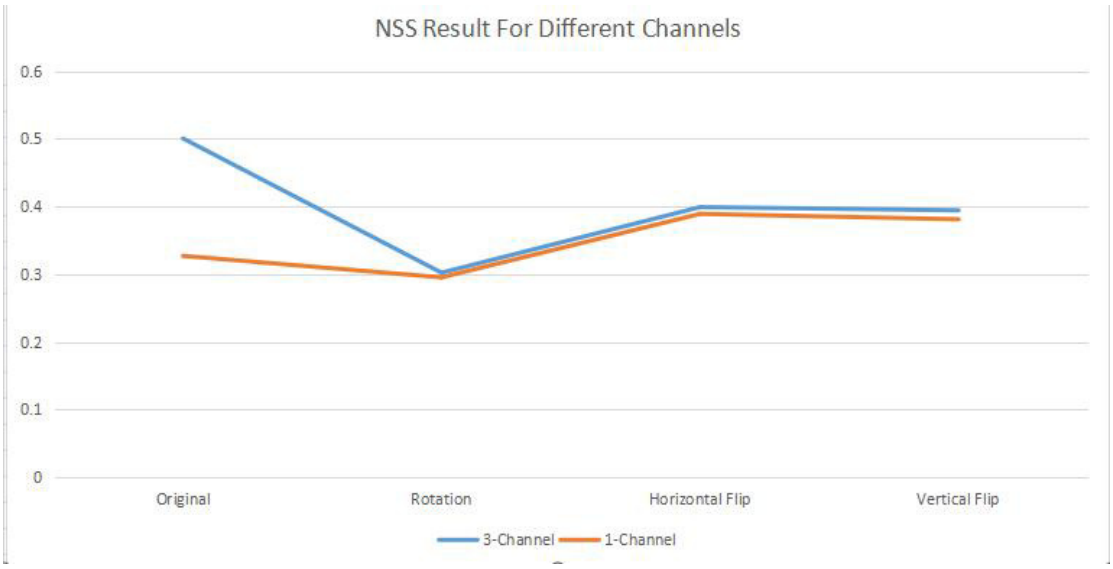

Figure 4.5 NSS Result in CNN..

As we can see in the table.1 and table.2, there are no significant difference between table 1 and table 2. The most deduction for MSE is rotation distortion. The horizontal flip and vertical flip have no obvious difference in MSE. However, the horizontal flip have larger KL-Divergence compared to vertical flip.

4.3 LSTM Model Result

The result of LSTM model of single channel with different distortion will be shown in table 3:

	Original	Rotation	Horizontal Flip	Vertical Flip
NSS	0.2859	0.2873	02858	0.2836
MSE	0.0657	0.0657	0.0657	0.0657
KL-Divergence	8.8377	8.8507	8.8583	8.8568

Table.3 LSTM Result with single channel

The result of CNN model of three channel with different distortion will be shown in table 4:

	Original	Rotation	Horizontal Flip	Vertical Flip
NSS	0.2880	0.2953	0.2865	0.2838
MSE	0.0656	0.0653	0.0656	0.0656
KL-Divergence	8.8105	8.48189	8.8851	8.8014

Table.4 LSTM Result with three channels

4.4 LSTM +CNN Model Result

The result of LSTM model of single channel with different distortion will be

shown in table 3:

	Original	Rotation	Horizontal Flip	Vertical Flip
NSS	0.2859	0.33	0.3797	0.3672
MSE	0.0657	0.0529	0.0486	0.0474
KL-Divergence	8.8377	7.5594	7.1022	6.7695

Table.5 LSTM+CNN Result with single channel

The result of CNN model of three channel with different distortion will be shown in table 4:

	Original	Rotation	Horizontal Flip	Vertical Flip
NSS	0.4332	0.3317	0.41	0.3969
MSE	0.0467	0.0529	0.0463	0.0472
KL-Divergence	6.5844	7.5530	6.8435	6.6948

Table.6 LSTM+CNN Result with three channels

On the CAT2000 database, the combined performance of the test results of the LSTM+CNN model proposed was the highest, achieving the lowest MSE on the highest parameter. In contrast, on the CAT2000 database, CNN and LSTM models proposed in this paper achieves the second and third place results for all metrics. Moreover, the LSTM+CNN model also has a significant improvement over other methods on NSS. NSS was identified as the main criterion for the evaluation of gaze point detection at the ECCV 2016.

4.5 Overall Analysis

The NSS value between all models for 1-channel and 3-channel will be shown below:

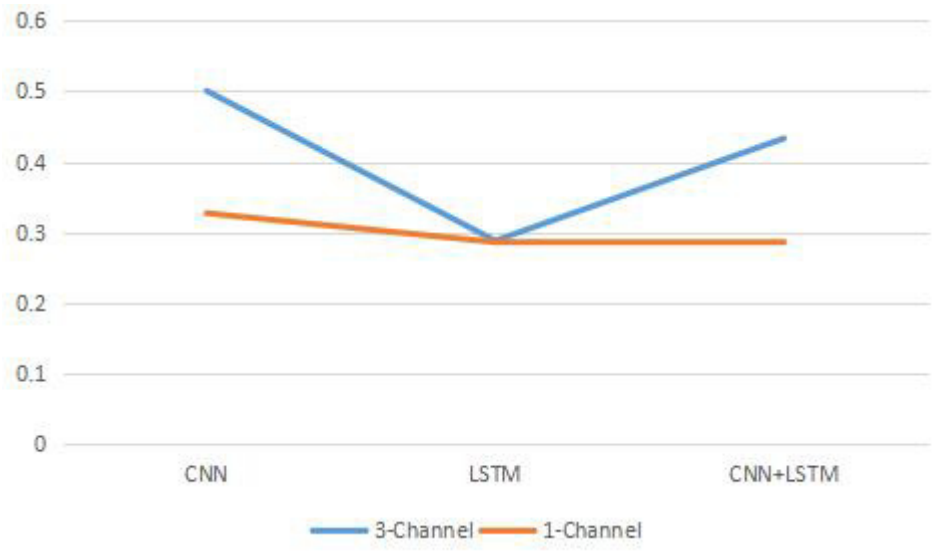

Figure 4.5.1 NSS Result in all models.

The MSE value between all models for 1-channel and 3-channel will be shown below:

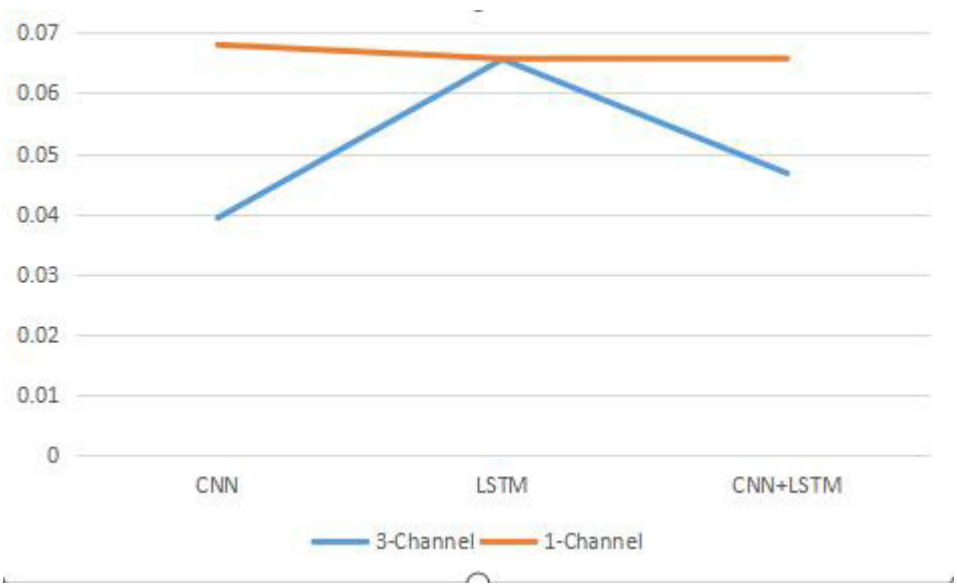

Figure 4.5.2 MSE Result in all models.

The KL-divergence value between all models for 1-channel and 3-channelwill be shown below:

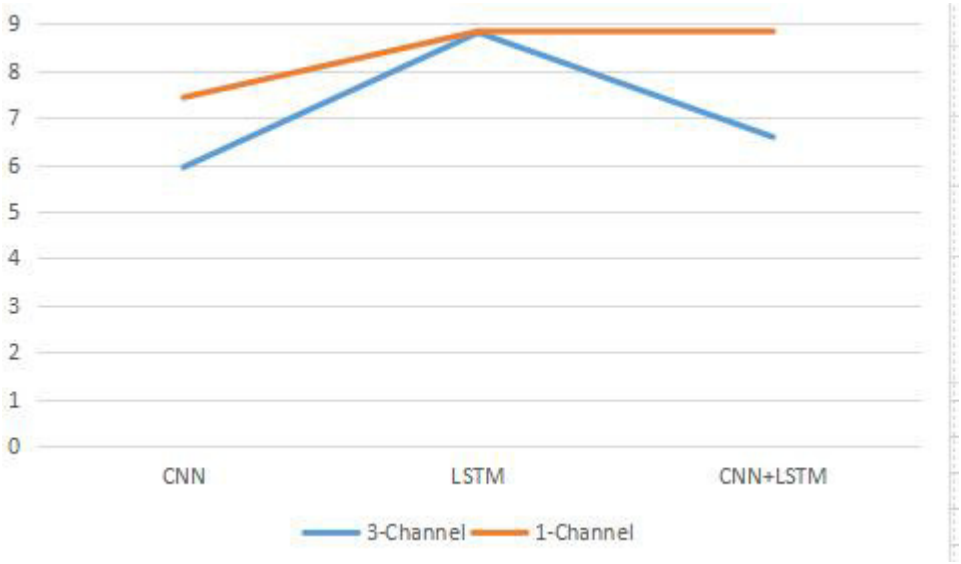

Figure 4.5.3 KL Result in all models.

As we can see, there are no obvious different between 3-channel or 1-channel, If channel selection is not used, the MSE decreases significantly, and after adding channel selection, the information is retained for classification to the maximum extent, and the KL decreases little. Due to the relatively smaller feature dimension and convolutional kernel size, it is easier to express the high-dimensional feature map, and the network obtains more diverse contextual information, and the NSS is improved compared with the single-channel network, which contains enough information and the effect of channel selection is minimal. Similarly, the AUC and sAUC also change with the richness of contextual information.

Chapter V.

Conclusion and Discussion

5.1 Conclusion

As a popular research topic in the field of computer vision, saliency prediction has been significantly improved in recent years with the advent of deep neural networks due to specific architectures and large annotated datasets, although these algorithms Although these algorithms go beyond the limitations of manual models, convolutional neural network-based models are still limited by the perceptual wilderness caused by network downsampling, and the use of full convolutional neural networks to improve accuracy results in a significant increase in computational resource consumption and model size. The existing deep learning based saliency prediction models are difficult to achieve the detection speed and accuracy are guaranteed. To solve these problems, the following work is done in this paper.

(1) Evaluating different distortions on OpenCV.

(2) Discuss various deep learning methods.

(3) Testing the effect of different distortion on different algorithms.

The difference between splitting a picture into three channels and inputting a model and inputting a picture directly is not very big, the direct input model will perform slightly better, the CNN will perform better than the LSTM, but the LSTM can extract some sequence information between some of their pixels, so the CNN plus the LSTM will be slightly better than the CNN alone.

5.2 Discussion

Moreover, as we mentioned above, the reason why HARHN performs best is

because RHN extends the structure of the LSTM, allowing progressive transformation depths greater than 1, leading to the capture of complex temporal properties at different semantic levels.

For the proposed module based on feature comparison, the module uses a simple convolutional kernel template, which takes the same weight for the surrounding environment and can be adjusted later to accomplish the significance detection task under the background diversity. The method of feature comparison can also be extended to use other comparison methods, not limited to distance and similarity comparison, but also adaptive comparison functions, to realize the diversity of modules.

The proposed network based on perceptual logic and feature matching can also be extended and improved. The network framework proposed in this paper is a general framework, and this composition is not limited to the saliency detection task. The design of different special feature extraction modules to obtain features corresponding to the task requirements allows this framework to be applied to different visual detection tasks, and in subsequent experiments, we will experiment with this framework in other detection tasks to prove the feasibility of this design.

In this paper, the structure of the feature preprocessing module only uses simple convolutional layers, activation layers and BN layers superimposed on the visual perception mechanism-based saliency network training method; the structure of the preprocessing can also be improved, and the backbone network structure can be adjusted later to make the preprocessing module fit better with the backbone model in order to improve the effectiveness of this combined training method. The structure of preprocessing can also be improved to make the preprocessing module fit better with the backbone model to improve the effect of this combined training method.

5.3 Limitation

Significance prediction is a challenging task. In this paper, we study the significance prediction algorithm based on attention mechanism and lightweight convolutional neural network, although it improves the accuracy and testing speed to some extent, there are still some problems and shortcomings, the main problems include.

(1) Due to the depth of the network is too deep, the parameters are difficult to adjust, this paper does not improve the lightweight of the underlying network, in order to better reduce the computational cost.

(2) As the change of the base network will have a large impact on the network as a whole, the loss function will be relatively adjusted in the subsequent work in order to adapt to the change of the base network

(3) The algorithm in this paper is only at the experimental stage, which has not reached the application requirements of practical scenarios, and the future work is to optimize the algorithm to achieve the effect of real-time.

(4) Advanced vision tasks based on saliency prediction, such as action recognition, human-computer interaction, behavior recognition, etc., are also of important research significance, and future work can be based on the study of saliency prediction, and in-depth research for advanced vision tasks.