Learning to Attend Relevant Regions in Videos from Eye Fixations

Advanced Machine Learning (ECE54401)
Advisor: Prof. Sung Ju Hwang
Students (ID): Thanh Nguyen (20165328) , Dung Nguyen (20165326)
Affiliation: UNIST
Email: {sjhwang, thanhnt, nguyendzung92}@unist.ac.kr

Abstract

Attentively important objects in videos account for a majority part of semantics in a current frame. Information about human attention might be useful not only for entertainment (such as auto generating commentary and tourist guide) but also for robotic control which holds a larascope supported for laparoscopic surgery. In this work, we address the problem of attending relevant objects in videos conditioned on eye fixations using RNN-based visual attention model. To the best of our knowledge, this is the first work to approach the problem from RNNs.

1 Introduction

When viewing a scene, human visual system does not see the whole image at once but selectively fixate on some informative regions. These informative regions are referred to as 'salient' which they are simply spatial regions in the visual field that attracts attention [5]. Salient regions which are usually obtained in a form of eye fixations are correlated to salient objects to which observers are paying attention at a particular time. The eye tracking data in individual frames typically lies on high level semantic objects; therefore, it coarsely localizes it. Moreover, the eye fixations can be cheaply obtained by using eye tracking equipment such as Eyelink 2000 eye tracker [3]. Based on these observations, in this project, we pose a problem of localizing attentive objects in video from eye fixation data and address it from deep learning perspectives.

Attending salient objects in a video is a very interesting problem in computer vision that have many potential applications. For example, consider a recorded video of two people playing UNO. Given a frame at some point from the video that contains multiple objects such as each players' hand holding UNO cards, can we tell whose turn is this in the current frame? While the task may be obvious to human observers, it requires a continuously considerable focus on semantics occurred previously . By automatically localizing attentive objects in current frames, we can help guide observer's attention to important objects in current frames based on one's own eye fixations on previous frames.

With the recent success of deep learning, especially Convolutional Neural Network (CNN), on a variety of visual recognition and classification tasks [8], [16], most recent works adapted CNN to the aforementioned problem [11], [12], [2]. Despite their great success, CNNs have fixed kernel sizes to learn context and do not scale well to large images. Therefore, recurrent neural networks (RNNs) were developed to extend neural networks to sequential data. One of the very successful application of RNNs to the related context is the work in [1] in which the authors used RNNs to model visual attention for multiple object recognition. In this work, we propose to adapt the ideas of RNN-based visual attention from [1] and [15] to localize attentively semantic objects in a video conditioned on eye fixations from the previous frames.

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

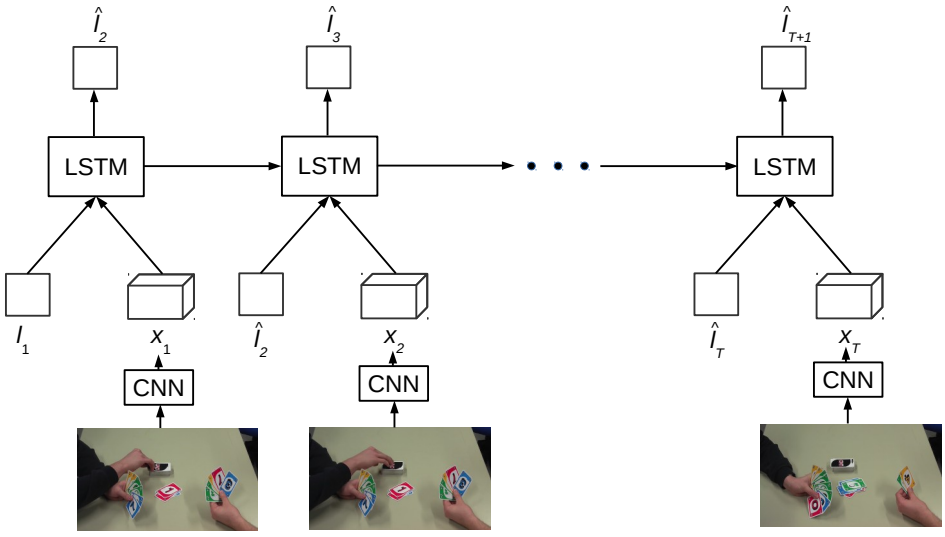

Figure 1: Our proposed attention recurrent model predicts attention maps of relevant regions for the next frame based on the attention map and convolutional features of the previous frames

2 Related work

To generate saliency map of video frames, several of well-understand methods were presented. Laptev [10] proposed Harris corner detector which works well in action classification. For applying to real-world application including less-corner, periodic detector was introduced by Dollar [4]. However, it is not adequate to deal with the problem in which not only video data but also fixation point set was given; therefore, Kienzle et al show a new method that outperform two previous methods by training a small neural network model to predict where people look [6]. After that, including some meaningful information to neural network become a trend. Multiresolution convolutional neural network (Mr-CNN) [12] which combining both top-down visual features and bottom-up visual saliency cues was applied to solve similar problem using image and eye movement data in it as inputs. More recently, depth information was included to the novel model Depth-Aware Video Saliency approach to predict saliency map for each frame in video. In additional, Deep CNN was used to ensures the learning of salient areas in order to predict the saliency maps in videos.

RNN is a well-known neural network structure which is suitable for processing sequential information such as language, numbers and especially video. Because of classical RNN models cannot remember long sequence, long short term memory was introduced [7]. Related to our work, a RNN combined with glimpse and three other networks was introduced to cope with multi object localization and recognition problem [1]. Simulating the visual attention, this model was shown to perform more accurate and less computational than ConvNets in the reading house numbers task.

3 Proposed Method

3.1 Architecture

Figure 1 presents our proposed approach to localizing attentive objects in videos. At each step t, our proposed model takes as inputs the convolutional feature $\mathbf{X}_t$ extracted from the current frame and the location vector $\mathbf{l}_t$. The RNN uses $\mathbf{X}_t$ and $\mathbf{l}_t$ as its inputs to predict the location probabilities $\hat{\mathbf{l}}_{t+1}$ for the next frame and regress bounding boxes of interest objects in the current frame. After learning the dependencies over a duration of T, the model learns to localize attentive objects in the remaining frames in the video.

While CNNs can extract powerful feature representation from images, RNNs are enable to encode long-term dependency into the network and naturally handle sequential data in videos. It is important to note that the model in Figure 1 represents an unrolling version of RNN over time in which the same RNN is applied at all steps. This architecture enables parameter sharing that forces the network to learning the dependencies over time.

3.2 Features extraction

We use a pre-trained GoogLeNet to extract features X_t for each frame. X_t is a $K \times K \times D$ tensor in which each frame is evenly divided into K^2 regions and our attention network predicts which of these regions to attend based on observations of the previous frames of the same video.

$$X_t = [X_{t,1}, X_{t,2}, ..., X_{t,K^2}]$$

The extracted CNN features and corresponding fixation points are then fed into an attentive LSTM to learn to attend relevant regions.

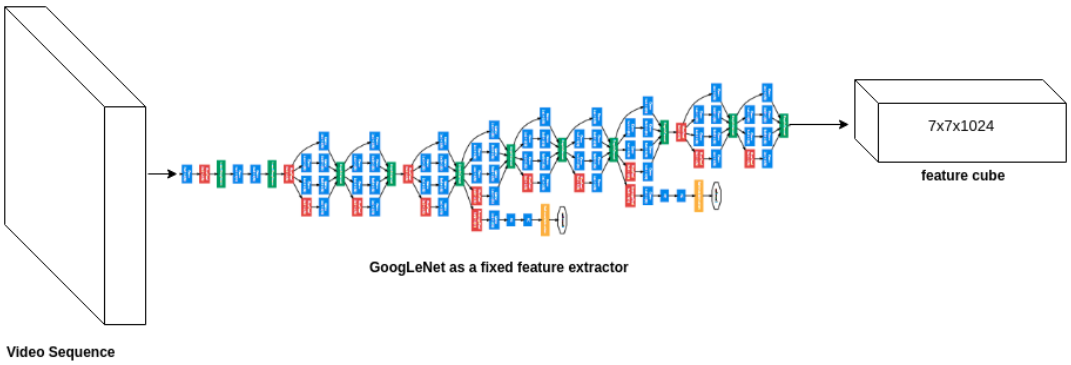

Figure 2: A pre-trained CNN network is used to extract feature maps for each frames before an attentive LSTM is applied

3.3 Attentive LSTM for predicting fixation regions

For implementation, we mainly adopt the LSTM network and attention mechanism from [15]. Particularly,

$$\begin{bmatrix} i_t \\ f_t \\ o_t \\ g_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} M(h_{t-1}, x_t)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t$$
$$h_t = o_t \odot \tanh(c_t)$$

where i_t, f_t, o_t are input gate, forget gate, and output gate, respectively, h_t is state, c_t is memory, and M is an affine transformation with learnable weights.

At each time step t, the attentive LSTM predicts the next attention location $\hat{l}_{t+1}$ based on the current attention location $\hat{l}_t$ and feature x_{t+1}.

$$\hat{l}_{t+1,i} = p(L_t = i | h_t, x_{t+1}) = \frac{\exp(W_i^T h_t + (W_i^{(c)})^T x_{t+1})}{\sum_i \exp(W_i^T h_t + (W_i^{(c)})^T x_{t+1})}$$

The feature x_t is calculated as weighted sum of feature cubes X_t:

$$x_t = \sum_{i=1}^{K^2} \hat{l}_{t,i} X_{t,i}$$

The feature x_t represents the input feature which encodes soft attention $\hat{l}_t$ over the CNN feature cube X_t.

The initial state and memory of the LSTM network is initialized via multilayer neuron networks for fast convergence

$$h_0 = f_{h,\text{init}} \left(\frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{K^2} \sum_{i=1}^{K^2} X_{t,i} \right) \right)$$
$$c_0 = f_{c,\text{init}} \left(\frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{K^2} \sum_{i=1}^{K^2} X_{t,i} \right) \right)$$

We simply use cross-entropy loss to compute the mismatch between the groundtruth fixations l_t and the predicted softmax attention locations $\hat{l}_t$:

$$L = -\sum_{t=1}^{T} \sum_{i=1}^{K^2} l_{t,i} \log \hat{l}_{t,i} + \gamma \sum_{i} \theta_i^2$$

where l_t is the one-hot groundtruth location vector at time step t.

4 Experiments

4.1 Datasets

At the beginning, we proposed to design a learning model which watches a video for first few frames and learns to attend relevant regions in the following frames, we constraint to a single video or multiple videos if their activities are consistent. However, a single video or multiple videos of consistent activities with eye fixation annotations are scarce and short. Therefore, in this work, we evaluate our proposed model only on the UNO data [9] and the Car data (including the Car Pursuit and Turning Car data) [9]. These datasets are the best one that meet our constraint that we could find.

The UNO data provides video stimuli with eye tracking data acquired from 25 participants. The clip has a frame rate of 25 fps and is extracted to 3025 frames. Each frame has one label which is the position of the fixation point over the features map. Since the UNO data size is small, we use the first 80% of its frames for training and the rest for testing.

With the similar components, Car Pursuit and Turning Car datasets (considered as Car dataset in this paper) have less number of frames than the previous data (700 for the former and 625 for the latter) but they used the same car in both videos. Because of these characteristics, we designed the experiment using Turning Car as training set and Car Pursuit as the test set.

Before learning, the data was processed by three steps. Firstly, because of some stop-frames in the data which does not have any fixation points, we assigned the fixation point of the nearest frame labeled to these frames. Secondly, based on the number of location considered in features maps (7×7 grid), the number in range $(0, 49]$ was labeled to each frame. Thirdly, to combine the fixation point information of 25 participants, we used the voting approach.

4.2 Results

For qualitative evaluation, we use the Kullback–Leibler divergence between the groundtruth fixations and our prediction maps as in [13] and evaluate our model on the UNO data and Car data. In both datasets, we use $\gamma = 0.01$, set dimensionality of LSTM hidden state to 64, use Dropout for avoiding

Figure 3: Training curve in the UNO dataset. We stop the training at iteration 1000 before overfitting

Figure 4: Training curve in the Car dataset. We stop the training at iteration 200 before overfitting

overfitting and use Adam optimization for better convergence. In addition, we train our model in 1 epoch since the datasets are very small and that our model's learning is almost saturated after the first epoch. We use one-layered LSTM for the Car data and two-layered LSTM for the UNO dataset since the one-layered LSTM does not empirically works well to capture variations in the UNO dataset. The learning performance is presented in Fig. 3 and Fig. 4. The experimental results have shown that our proposed model is indeed capable of learning to attend relevant regions in videos conditioned only on eye fixations. In easy contexts as in the Car data, our model can learn and generalize quite well after 200 iterations despite that the Car data size is small. A probable reason is that contexts in the Car data is consistent and do not have high variations. In the UNO data, however, the error is higher that that in the Car data which indicates that our proposed model has difficulty capturing variations in this UNO dataset. This is probably due to the fact that UNO data exposes very high variations. The UNO players take turn to play cards and the playable cards at a particular moment depends on which card has been played and which uncovered cards are available. Combining such main factors already leads to very large variations which are almost impossible to be captured by watching only a single short clips.

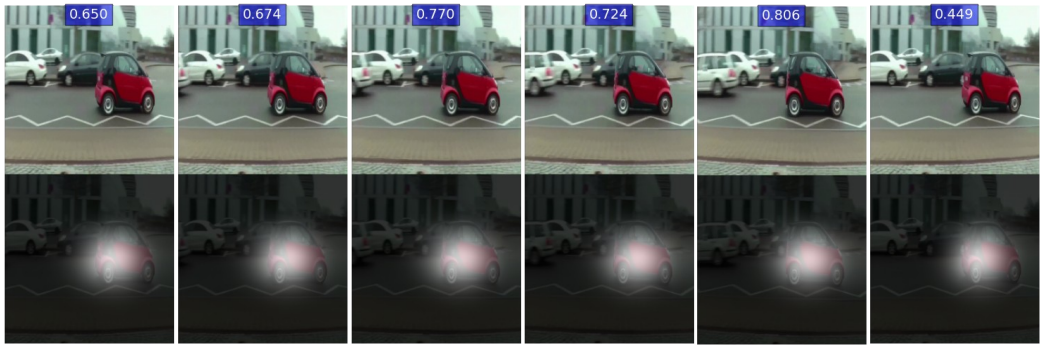

Figure 5: Visualization of our model on the Pursuit car test data. It learns to correctly attend regions of the red car. The blue numbers represent error between our predicted attention map and the groundtruth fixations. For the full demo video, check out `https://youtu.be/HGex1CpUins`

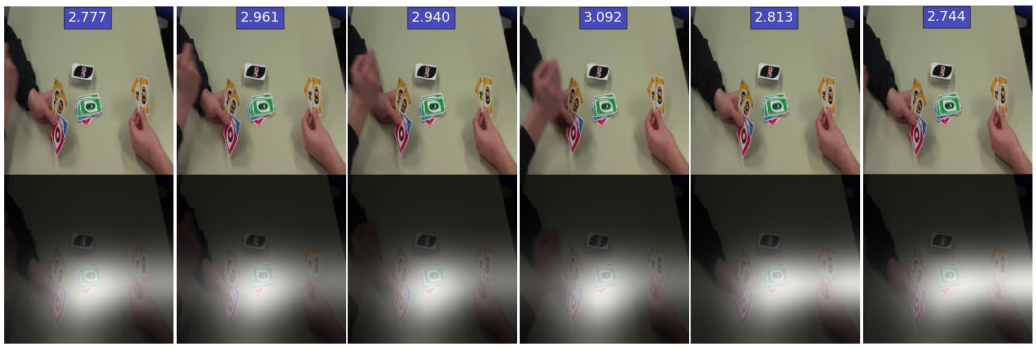

Figure 6: Visualization of our model on the UNO test data. In this complicated context, our model fails to attend to playable cards because the UNO data is small for our model to capture variations in UNO-play context.

5 Conclusion and Future works

To learn relevant attention regions in a video from fixation points, we present a RNN model which is more sufficient for sequential data than CNN. The experimental results has shown that our model can learn to attend relevant regions in videos with simple setting as in the Car dataset but fail to learn in a more complicated context as in the UNO dataset. An apparent reason for this failure is the lack of data. Eye fixations at relevant objects in a complicated context follows a complicated pattern which requires more data to capture such pattern. The second possible reason is that fixation data is usually noisy and subjective because human attention on specific objects in videos for a period of time might not be consistent due to distractions. This sort of noise introduces more nuisance factors to fixation points which makes learning such fixation points more difficult.

Our constraint about consistency of single video makes data collection hard and limited. In contrast, we believe that our proposed model can learn from a large dataset of multiple contexts and generalizes to new videos of any of such contexts. From this perspective, we will do the experiment in dataset Holiwood [13] [14] for the future works to get better results.

References

[1] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *CoRR*, abs/1412.7755, 2014.

[2] Souad Chaabouni, Jenny Benois-Pineau, Ofer Hadar, and Chokri Ben Amar. Deep learning for saliency prediction in natural video. *CoRR*, abs/1604.08010, 2016.

[3] Frank Keller Vittorio Ferrari Dim P. Papadopoulos, Alasdair D. F. Clarke. Training object class detectors from eye tracking data. *European Conference on Computer Vision*, pages 361–376, 2014.

[4] Piotr Dollár, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. Behavior recognition via sparse spatio-temporal features. In *2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72. IEEE, 2005.

[5] K. Duncan and S. Sarkar. Saliency in images and video: a brief survey. *ET Computer Vision*, 6(1):514–523, 2012.

[6] Fred A. Hamprecht, Christoph Schnörr, and Bernd Jähne, editors. *Pattern Recognition, 29th DAGM Symposium, Heidelberg, Germany, September 12-14, 2007, Proceedings*, volume 4713 of *Lecture Notes in Computer Science*. Springer, 2007.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[9] Kuno Kurzhals, Cyrill Fabian Bopp, Jochen Bässler, Felix Ebinger, and Daniel Weiskopf. Benchmark data for evaluating visualization and analysis techniques for eye tracking for video stimuli. In *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization, BELIV 2014, Paris, France, November 10, 2014*, pages 54–60, 2014.

[10] Ivan Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[11] George Leifman, D. Rudoy, Tristan Swedish, Eduardo Bayro-Corrochano, and Ramesh Raskar. Learning gaze transitions from depth to improve video saliency estimation. *CoRR*, abs/1603.03669, 2016.

[12] Nian Liu, Junwei Han, Dingwen Zhang, Shifeng Wen, and Tianming Liu. Predicting eye fixations using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 362–370, 2015.

[13] Stefan Mathe and Cristian Sminchisescu. Dynamic eye movement datasets and learnt saliency models for visual action recognition. In *Computer Vision–ECCV 2012*, pages 842–856. Springer, 2012.

[14] Stefan Mathe and Cristian Sminchisescu. Actions in the eye: dynamic gaze datasets and learnt saliency models for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(7):1408–1424, 2015.

[15] Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. Action recognition using visual attention. *CoRR*, abs/1511.04119, 2015.

[16] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.