MATH6380O Project 3

Recurrent Attention Model: Mimicking The Human Vision

DENG, Yizhe, GU Hanlin, HUANG Yifei, SUN Jiaze Department of Mathematics, HKUST

Introduction

As the culmination of hundreds of millions of years of evolution, our optical perception allows us to receive and digest visual information with considerable accuracy and efficiency. Key aspects of our vision include:

- **Focus**: we only focus on one small portion at a time;
- **Sequential processing**: By moving our focus to different parts, we process information one by one.

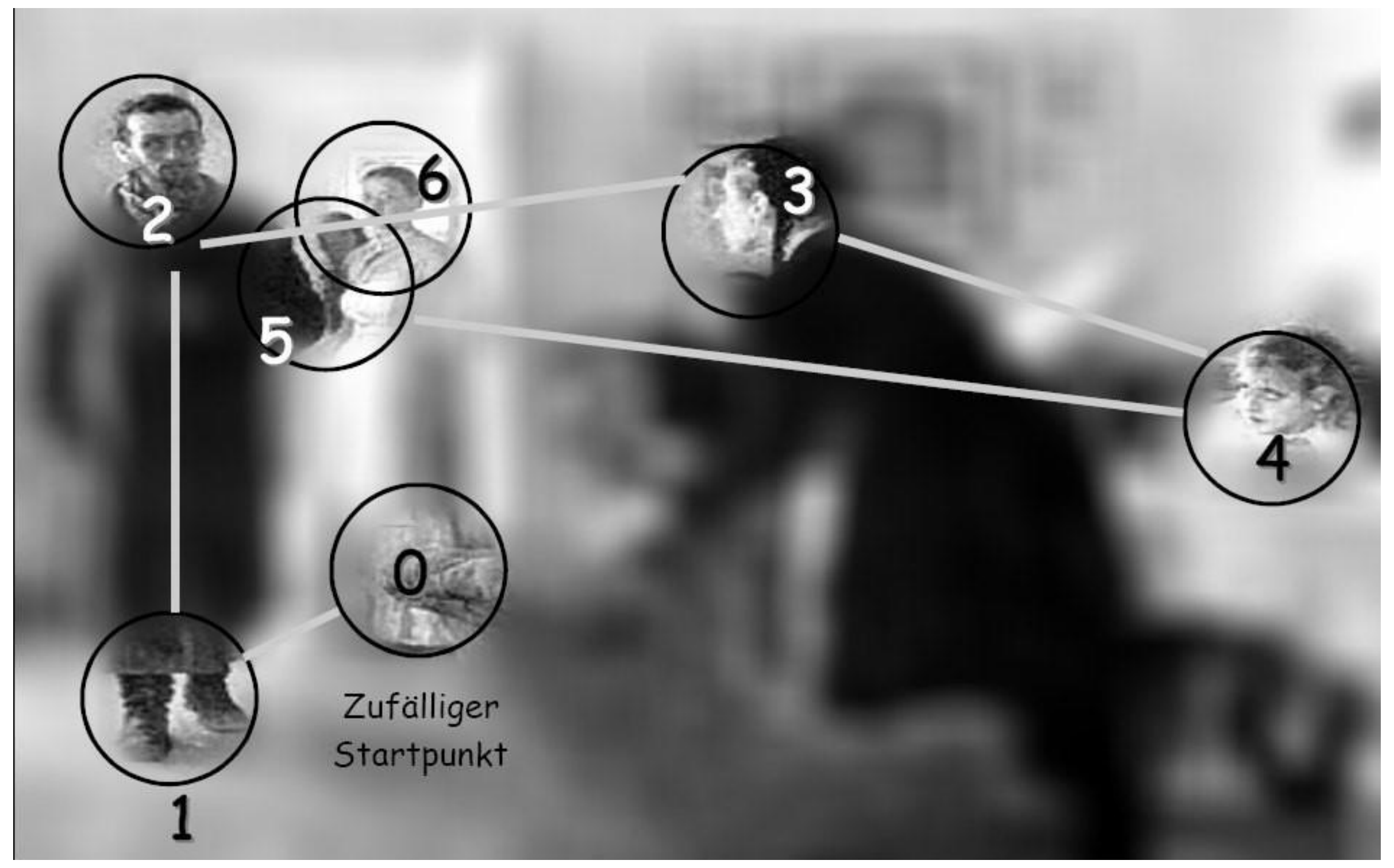

Figure 1. The first 2 seconds of inspecting of photo. The numbers detail the order in which the observer focuses on a patch of the image. The rest of the image is blurred, representing peripheral vision.

These traits have certain advantages:

- By focusing on one small portion, we **filter out redundant or irrelevant information**.
- By processing things sequentially, we **obtain more information and can improve understanding**.

In computer vision, a **Recurrent Attention Model (RAM)** is essentially inspired by the above properties. In this project, we aim to test out its effectiveness.

Data

Our experiments were conducted on 3 versions of MNIST: the ***original***, ***translated***, and ***cluttered***. The original is of 28x28, whereas the latter two are of 60x60. Translated and cluttered MNIST are essentially more difficult versions of the original (see Figure 2).

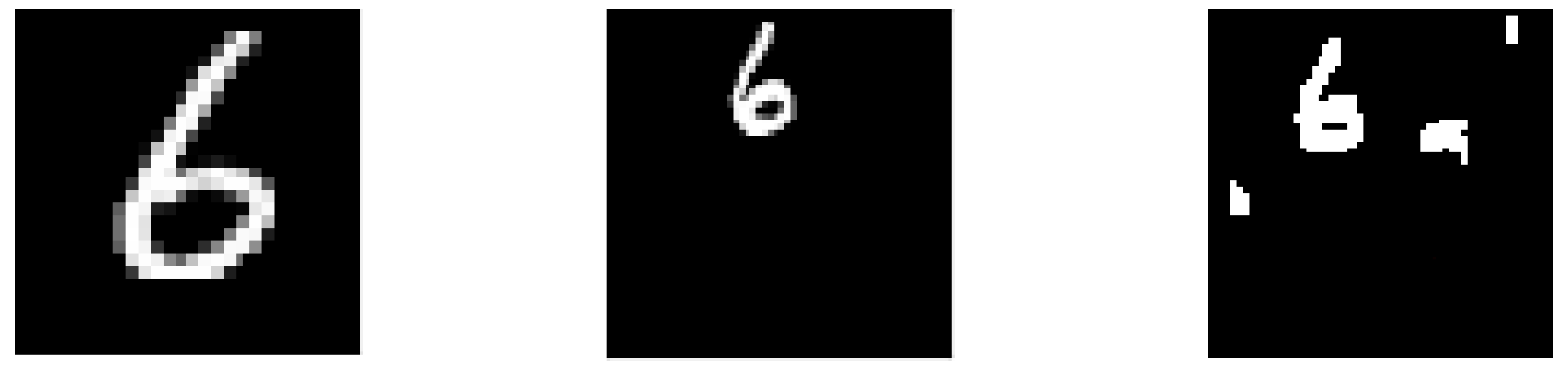

Figure 2. The number 6 in the original (left), translated (middle), and cluttered (right) versions of MNIST.

Recurrent Attention Model (RAM)

RAM combines 3 components in its architecture, a **Glimpse Sensor**, a **Glimpse Network**, and a **Recurrent Neural Network** (RNN). In addition, the model is trained by reinforcement learning. As an overview, at glimpse *t*:

- **Glimpse Sensor**: The glimpse sensor extracts image patch at location l_{t-1}, and generates a representation $\rho(x_t, l_{t-1})$;
- **Glimpse Network**: The Glimpse network computes a feature vector $g_t = f_g(l_{t-1}, \rho(x_t, l_{t-1}))$;
- **Recurrent Neural Network**: To use information from previous glimpses, the model combines previous hidden state h_{t-1} with the feature g_t to generate a new hidden state $h_t = f_h(g_t, h_{t-1})$. From this new hidden state, the model does 2 things:
  1. **Make a prediction:** In a classification task, the model simply computes the class probabilities via a softmax layer f_a.
  2. **Get a new location:** The model computes a new location l_t to be used in the next glimpse ***t+1***.

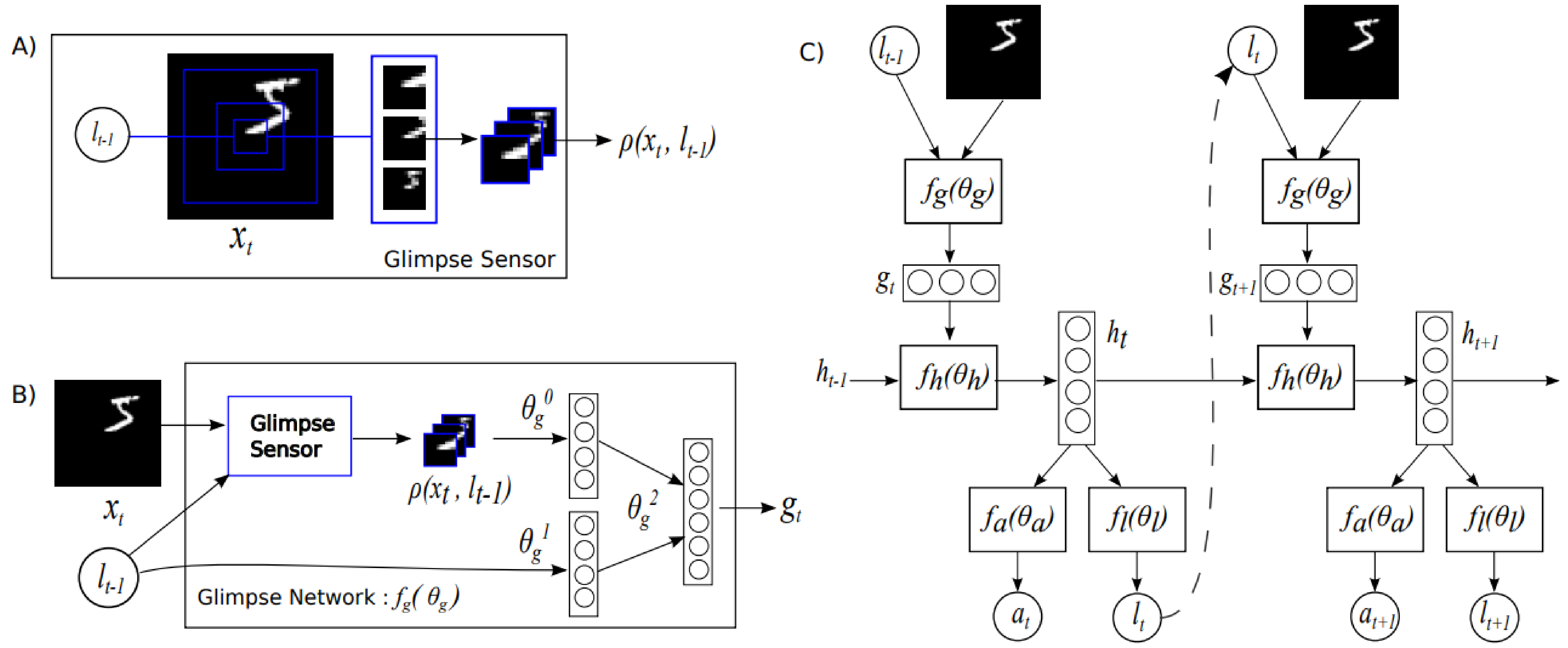

Figure 3. The 3 components of RAM: a Glimpse Sensor (A), a Glimpse Network (B), and an RNN (C).

Training & Prediction

Training

RAM is trained via reinforcement learning. At each glimpse ***t***, the agent is given a reward r_t. In a classification task, $r_t = 1$ if a correct classification is made, and ***0*** otherwise. The purpose of the agent is to maximize the total reward $\sum_{t=1}^{T} r_t$.

Prediction

RAM makes predictions by sequentially processing the information obtained at each glimpse, where the judgement at each step depends on that in the previous step. Figure 4 displays the predictions made by a very well-trained RAM. As one can see, the predictions gets more accurate in later glimpses.

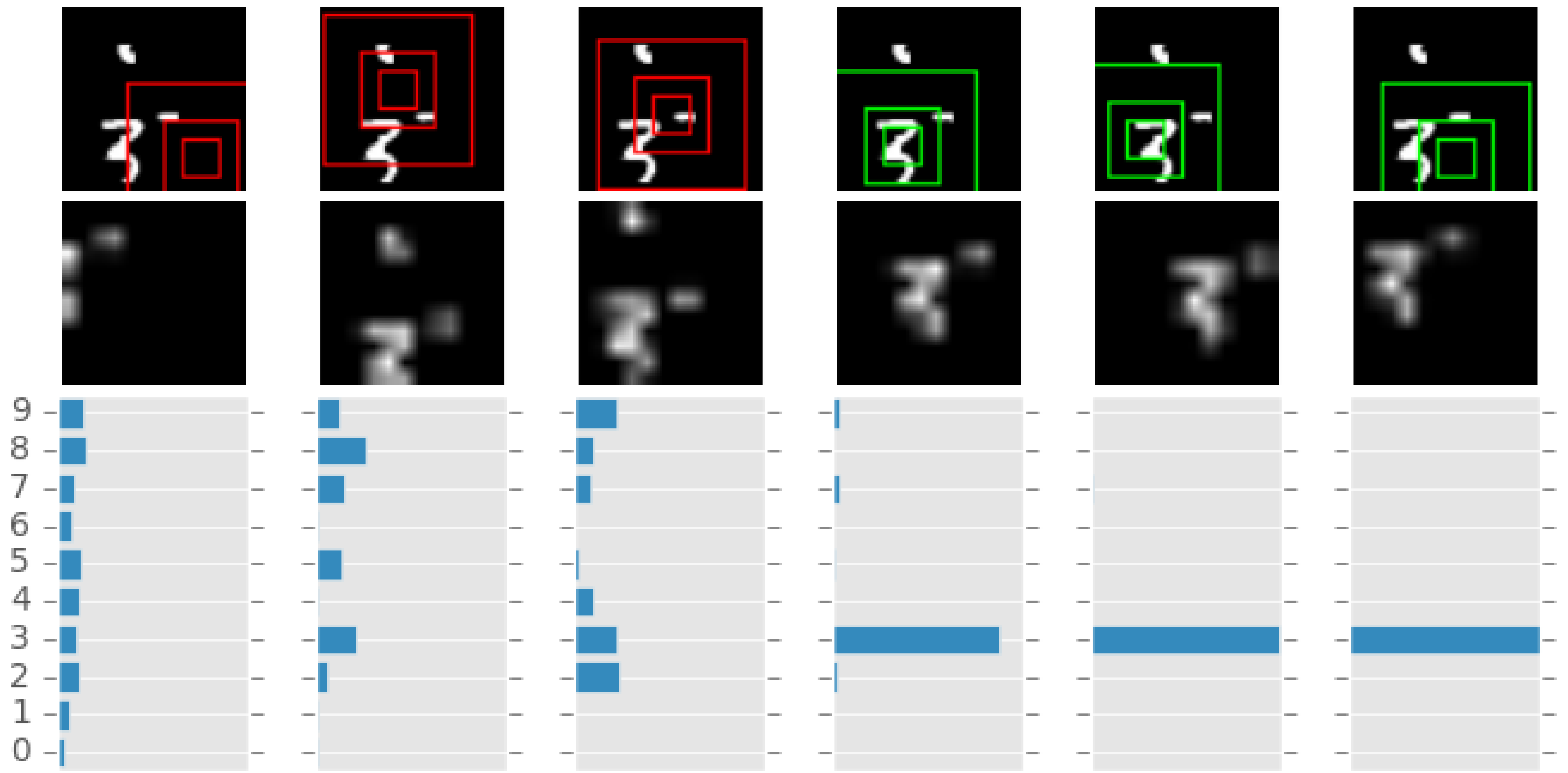

Figure 4. RAM's prediction for "3" during each glimpse. Each columns is a glimpse, in sequential order from left to right. The first row are the original images, within each square is the image patch extracted by the Glimpse Sensor. The second row shows the image patch from the largest square. The third row are the predicted class probabilities, where the class names are specified on the left. This image was retrieved from https://github.com/amasky/ram.

Results

Due to limited computational resources, we were only able to run the model for a maximum of 50 epochs. The training process takes an extremely long time to complete, making parameter tuning a very time-consuming task.

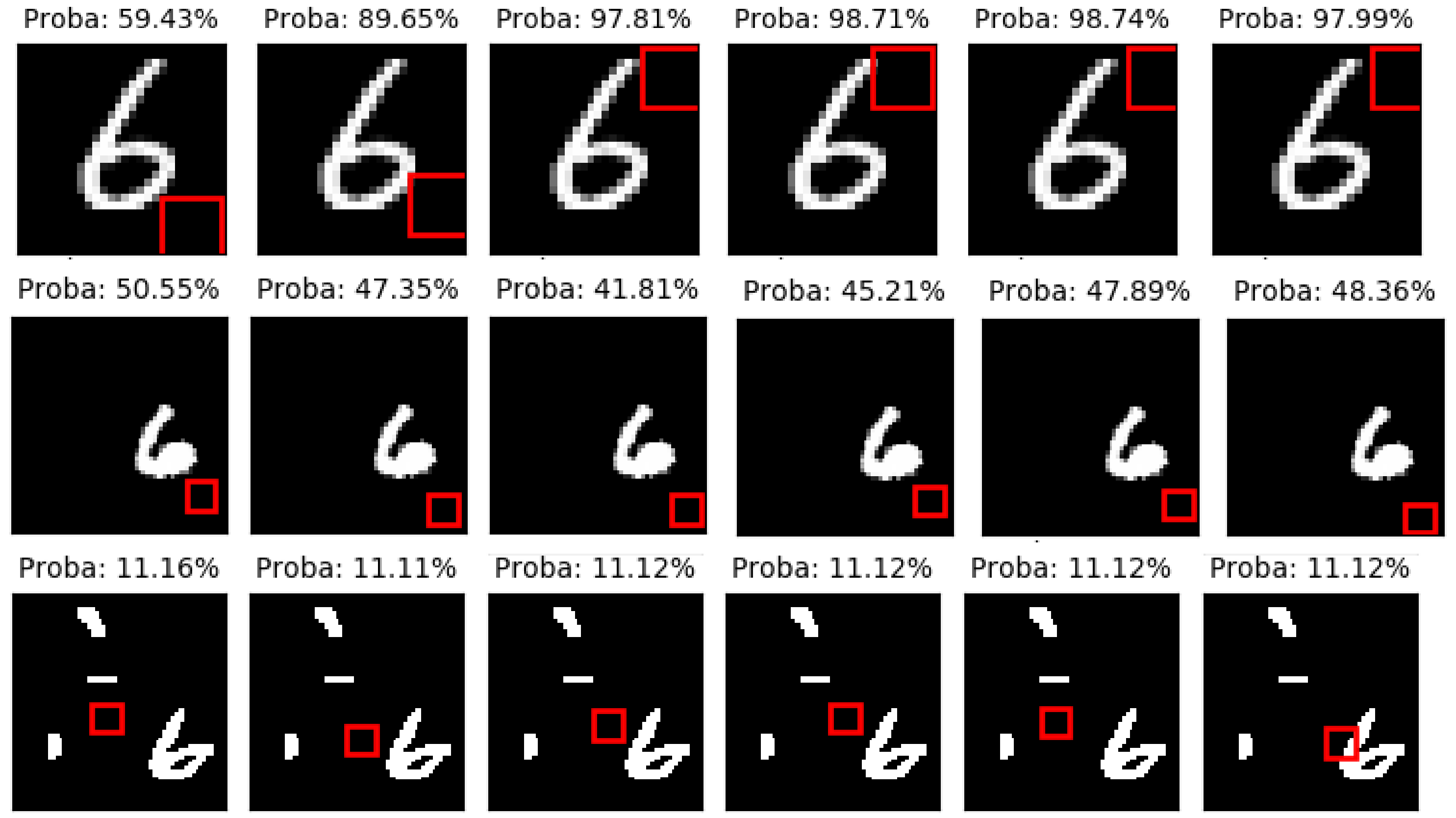

Figure 5. Our RAM predictions for the number 6 in the original (1st row), translated (2nd row), and cluttered (3rd row) versions of MNIST. Each column is a glimpse, in temporal order from left to right. Above each image is the corresponding predicted probability for the correct class.

In Figure 5, one can see that the model works effectively for the original MNIST data set. The predicted probability shows a nice increase towards the later glimpses. However, it performs rather poorly for the translated or cluttered images, as not only is the predicted probability low, but also it does not show any increase in the later glimpses. A possible explanation for this poor performance is the lack of parameter tuning, as well as insufficient training.

Data	Accuracy
Original	92.4%
Translated	71.1%
Cluttered	62.3%

Table 1. The classification accuracies for different data sets.

Acknowledgement

- **DENG Yizhe**: Reading papers & discussion
- **GU Hanlin**: Reading papers & discussion
- **HUANG Yifei**: Code modification & running
- **SUN Jiaze**: Compiling & interpreting results, writing the poster

References

- Mnih V, Heess N, Graves A, and Kavukcuoglu K. *Recurrent Models of Visual Attention.* 24 Jun, 2014.
- Amasky. RAM. Retrieved from https://github.com/amasky/ram.
- Yarbus A L. *Eye movements and vision*. 1967. Retrieved from https://en.wikipedia.org/wiki/Eye_movement.