

國立交通大學

多媒體工程研究所

碩士論文

紋理合成貼圖的注視點與評分預測

A Visual Attention and Perceptual Rating Model for
Synthetic Structural Textures

研究生：黃柏齊

指導教授：林文杰 教授

中華民國 一 百 年 九 月

紋理合成貼圖的注視點與評分預測

A Visual Attention and Perceptual Rating Model for Synthetic Structural
Textures

研 究 生：黃柏齊

Student : Po-Chi Huang

指導教授：林文杰

Advisor : Wen-Chieh Lin

國 立 交 通 大 學

多 媒 體 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Multimedia Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

September 2011

Hsinchu, Taiwan, Republic of China

中華民國一〇一年九月

A Visual Attention and Perceptual Rating Model for Synthetic Structural Textures

Student: Po-chi Huang

Advisor: Dr. Wen-Chieh Lin

Institute of Multimedia Engineering
College of Computer Science
National Chiao Tung University

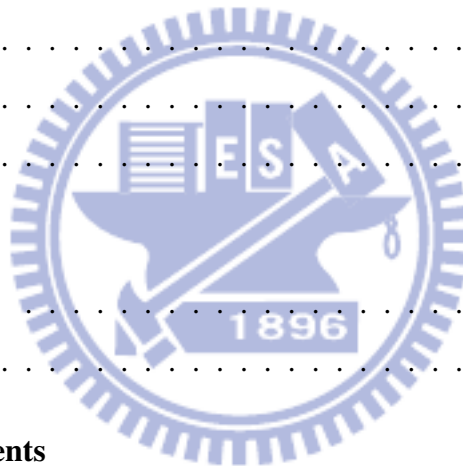


ABSTRACT

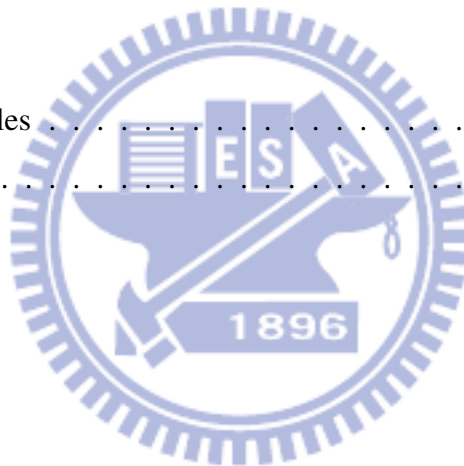
Texture synthesis is a hot topic in computer graphics; however, there is less work on perceptual evaluation of synthetic structural texture. As visual attention is the first stage of visual cognition process, we propose two models, visual attention model and perceptual rating model, to predict visual saliency and human rating on synthetic structural textures. We designed an experiment to gather subjects' eye-tracking data and rating score while evaluating the similarity of an input and its synthesized textures. The visual attention model is developed to associate texture features and fixations. The perceptual rating model is trained to associate the relationship between the fixations and the rating. We compared our visual attention model with the saliency map. Our model correctly predicts 82.7% of fixation positions while the saliency map only achieves 57%. For the perceptual rating, the Chi-square value of our model is 3.98 but non-perceptual metric is 6.95, comparing to human's rating scores. Our model is very helpful for guiding texture synthesis and manipulation algorithms to efficiently allocate computational resources to those regions that humans pay attention to.

Contents

1	Introduction	1
1.1	Background	1
1.2	Overview	2
1.3	Result	3
2	Related Work	5
2.1	Texture Evaluation	5
2.2	Visual Attention	6
3	Eye Tracking Experiments	10
3.1	Experiment Settings	10
3.2	Experiment Procedures	13
4	Our Approach	17
4.1	Feature Extraction	17
4.2	Self-Constructing Neural Fuzzy Inference Network	23
4.3	Visual Attention Model	27
4.4	Perceptual Rating Model	29
5	Results	32
5.1	Evaluation Method	32

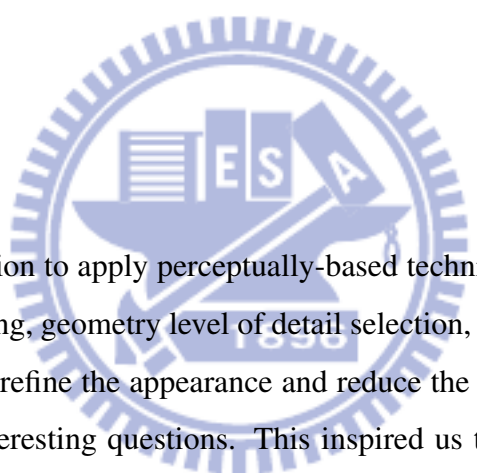


5.2	Prediction Experiments on Structural Textures	33
5.3	Validation of The Visual Attention Model	35
5.3.1	New-Texture-Prediction	35
5.3.2	New-Subject-Prediction	36
5.4	Validation of The Perceptual Rating Model	41
5.5	Rule Analysis	46
5.5.1	Poor-Synthesized Texture	47
5.5.2	Well-Synthesized Texture	47
6	Conclusion and Future Work	53
A	Appendix	55
A.1	Parameters and Rules	55
A.2	NSS Table	55
	Bibliography	59



Introduction

1.1 Background



It becomes a common situation to apply perceptually-based technique to many research fields, such as image-based rendering, geometry level of detail selection, realistic image synthesis and video compression. How to refine the appearance and reduce the computational cost of a synthetic texture are always interesting questions. This inspired us to construct a computational model to perceptually evaluate the quality of synthetic structural textures.

Several previous works presented non-perceptual methods to evaluate synthetic textures. Lin et al.[1] suggested that in evaluating the quality of synthetic near regular textures, geometric structure is a more important feature to viewers than color, intensities, or orientations. Two mathematical metrics, G-score and A-score, are proposed to compare with user data and find the relationship. Nevertheless, the results of non-perceptual methods do not always match those of users. Moreover, it is costly and unrealistic to hire subjects to evaluate synthetic textures all the time. For this reason, we would like to develop a computational model to associate the quality of synthetic texture and human perception. In our survey, less evaluation work is related

to human visual system (HVS). Benard et al.[2] proposed a strategy to analyze the relationship between human rating and textures. In this work, they indicated the average co-occurrence error as a meaningful quality assessment metric for fractalized NPR textures. They validated the relevance of this predictor by showing its strong correlation with the results of a user-based ranking experiment; however, co-occurrence error primarily reflects the results of NPR texture but fails to predict the others.

1.2 Overview

As non-perceptual metrics may not reflect real human rating scores, we want to develop a perceptual evaluation method in this thesis. According to [3], there is a close connection between attention and cognition. To rate a synthetic texture, a subject has to gaze at some regions of the texture. Thus, we assume where people are attracted to would affect the rating. Moreover, an intuitive way to define a perceptual metric to evaluate the quality of synthetic textures is to measure the difference between the input and the synthesized texture. We may collect user rating scores and a model to represent the relationship between image difference and user scores. Nevertheless, the visual rating process is very complicated. It is not easy to directly model this relationship. Therefore, we model the rating process by considering it as two subprocesses, visual attention and perceptual rating, and model them separately.

A popular choice to predict visual attention is the use of saliency map [4], which guides the selection of attended locations based on the spatial distribution of saliency by analyzing the low-level features of an image such as colors, intensities and orientations at every location in the visual field. In contrast to natural images, textures have low variation in color, intensities and orientations. This makes the saliency map not work well on textures, so we claim that the saliency map does not meet our needs.

To model gaze behavior, a convenient way is to learn eye-tracking has been used to understand attractive locations of a scene. We develop our model based on Self-Constructing Neural Fuzzy Inference Network (SONFIN)[5] to learn the relationship between features of textures and subjects' eye-tracking data. Eye movements may even reveal information that viewers are not aware of, because it is not consciously available to the observer. For instance, eye tracking experiments have shown that professional radiologists spend more time gazing at locations where tumors are present, though they had failed to identify and report them [6].

From [1], we have known geometric structure of textures is a more attractive feature to viewers than color, intensities, or orientations, e.g. repeated structures dominate the quality of the synthetic near-regular textures. We extract several new features as the input feature to learn human rating score in perceptual rating model.

1.3 Result

To evaluate the performance of our model, we adopt normalized scan-path saliency (NSS) [7], which measures the similarity of the predicted fixations and the actual fixations recorded by the eye-tracking system. In visual attention model, 82.7% of predicted fixations match with the actual fixations. We also compared our model with saliency map on the same textures. Only 57% of fixations predicted by the saliency map match with the actual fixations. For the perceptual rating, Chi-square value of our model is 3.98 but non-perceptual metric is 6.95, comparing to human's rating scores.

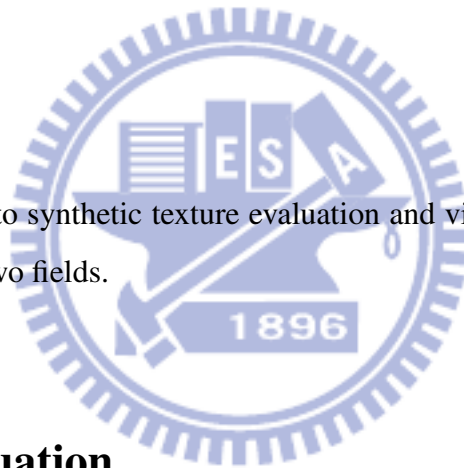
The rest of the thesis is organized as follows. In Chapter 2, we briefly review texture synthesis algorithm and related work in visual attention. In Chapter 3, we will talk about the experiment procedures. In Chapter 4, we describe how to get the feature vectors and develop our model. In Chapter 5, we will show the evaluation method and our results. Finally, conclu-

sions and future work are present in Chapter 6.



Related Work

Our work is closely related to synthetic texture evaluation and visual attention. We will introduce related work in these two fields.



2.1 Texture Evaluation

The evaluation of texture feature is important for several image processing applications. Texture analysis forms the basis of object recognition and classification in several domains. There is a range of non-perceptual texture extraction methods and performance evaluation. They are important parts of understanding the utility of feature extraction tools in image analysis. To compare different non-perceptual evaluation methods, M. Sharma et al. [8] evaluated five popular different feature extraction methods. These were auto-correlation, edge frequency, primitive-length, Laws method, and co-occurrence matrices. According to the result, each of them has their disadvantages to evaluate the quality of textures.

Lin et al. proposed a synthetic way to evaluate the quality of a texture by human perception [1]. The contribution of their work is that they carry out a systematic comparison study on the performance. Geometric regularity(G-score) and appearance regularity(A-score) are two mathematical criteria utilized to compare the quality of textures. They also set up an experiment to record user evaluation data and analyzed the result. According to their result, they suggested that in evaluating the quality of synthetic near regular textures, geometric structure is a more important feature to viewers than color, intensities, or orientations.

Benard et al.[2] designed a rating experiment. They chose twenty gray-scale 2D textures sufficiently representative of the main traditional media used in NPR. To create a sufficient redundancy in the results, they designed two sets of ten texture pairs. For each set, they chose one representative texture per class (pigments on canvas, paint, paper, hatching, cross-hatching, dots, near-regular or irregular patterns, noise and grid). Consequently, they paid special attention to assessing the statistical validity of the resulting data. In this work, they indicated the average co-occurrence error as a meaningful quality assessment metric for fractalized NPR textures. They validated the relevance of this predictor by showing its strong correlation with the results of a user-based ranking experiment; however, co-occurrence error primarily reflects the results of NPR texture but fails to predict the others.

2.2 Visual Attention

Saliency map proposed by Itti et al.[4] is the most popular algorithm used to predict human fixation. In contrast to the early research of visual attention which concentrates on subjective awareness of the world, saliency map (shown as Figure 2.1) divides an image into three separated feature channels: color contrast, luminance contrast, and four orientations. These features detect salient parts in the visual stimulus using the center-surround architecture. The generated feature map will then be normalized to mimic the literal inhibition effect. The sum of the feature

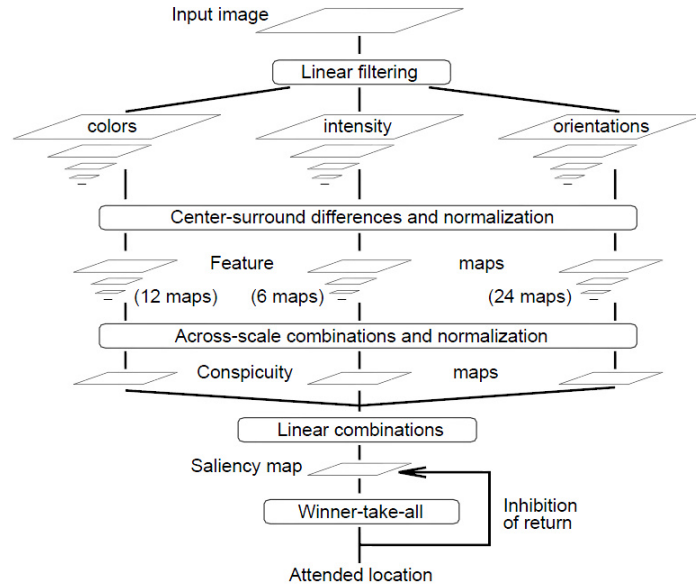


Figure 2.1: Saliency map model.

map for each feature channel results in the conspicuity map, which will also be normalized and then summed up to obtain the saliency map that quantifies visual attention.

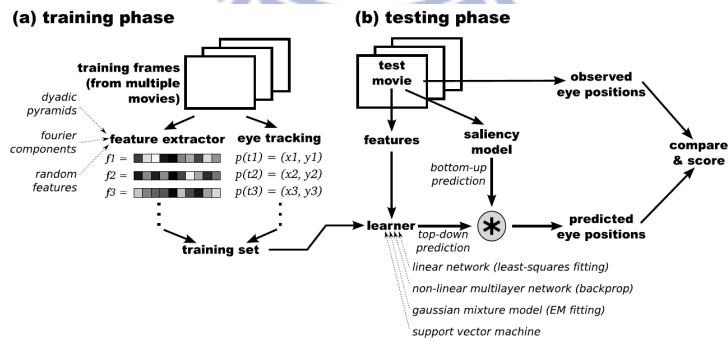


Figure 2.2: Saliency map model.

Peter et al.[9] build up a least-square based model (shown as Figure 2.2) of spatial attention that combines a general computational implementation of both bottom-up saliency and dynamic top-down task relevance. The bottom-up component computes a saliency map from

12 low-level multi-scale visual features. The top-down component computes a low-level signature of the entire image, and learns to associate different classes of signature with the different gaze patterns recorded from human subjects. In a simple statement, the basic idea of this thesis is to train a model associating to the features of saliency map.

Mathew et al.[10] presented a neural network model to simulate saliency map[4]. They introduced a model that expands on Itti and Koch's model by implementing the feature maps and saliency map as a network of neural populations with dynamics based on data from electrophysiological experiments. Their main motivation for this model was to propose a hypothesis for how Itti and Koch abstract model could be implemented by neural networks with biologically realistic dynamics.

Most saliency approaches are based on bottom-up computation that does not consider top-down image semantics and often does not match actual eye movement. Judd et al.[11], on the other hand, proposed a support vector machine(SVM) based model trained with low, middle and high-level features. These features include subband features, Itti and Koch saliency channels, distance to the center, color features and automatic horizontal, face, person and car detectors. Compare with the former related work, this thesis contains more object-relevant features considered as interesting parts in an image.

Normalized scanpath salience(NSS) proposed by Robert et al.[7] can be used to measure the average normalized salience value across all fixation locations. The normalized scanpath salience indicates that, on average, the model-predicted salience at fixated locations. Since the NSS is scale-free, it can be used to compare the degree of correspondence between observed and predicted behavior for different observers and images.

Stas et al.[12] proposed a new type of saliency, context-aware saliency, which aimed at detecting the image regions that represent the scene. They presented a detection algorithm which

was based on four principles observed in the psychological literature, such as local low-level considerations, global considerations, visual organization rules and highlevel factors.

Yu et al.[13] proposed a computational model of visual attention on structural textures by analyzing human subjects' gaze behavior. We keep the eye-tracking data and user's rating score data. Additionally, we modify her feature extraction and the association model to guarantee a better prediction. Instead of training whole feature map of textures, we sample the training patterns from feature map to reduce the computational cost. Moreover, we replace the training model with SONFIN regarding to her appreciating speed and performance.



Eye Tracking Experiments

Eye-tracking has become much more attractive recently. Why is eye-tracking important? Simply put, we move our eyes to bring a particular portion of the visible field of view into high resolution so that we may see in fine detail whatever is at the central direction of gaze. Most often we also divert our attention to that point so that we can focus our concentration on the object or region of interest. Thus, we may presume that if we can track someone's eye movements, we can follow along the path of attention developed by the observer.

We recorded eye movements from human observers while they are watching, comparing and judging a synthesized texture. The collected eye-tracking data is utilized to train our model. This may give us some insight into what the observer found interesting. In this chapter, we will describe the settings of our experiment and how we process the eye-tracking system.

3.1 Experiment Settings

To record the most natural reaction of viewers, we need to reduce the effects from eye-tracking equipment and provide a relaxing environment for our subjects. For these two reasons, our ex-



Figure 3.1: *This photo shows Tobii T120 Eye-tracker and experiment environment.*

periment was done on the Tobii T120 Eye-tracker, which is a contact free gaze measurement device, as shown in Figure 3.1. The eye tracking system allows for a large degree of head movement, providing a distraction-free test environment that ensures natural behavior, and therefore valid results. The eye tracking technology's high level of accuracy and precision ensures that the research results are reliable. This helps to acquire a more realistic response from human subjects.

The following are the specifications of Tobii T120 eye-tracker:

Data Rate: 120Hz

Accuracy: typical 0.5 degrees

Head Movement Error: typical 0.2 degrees

Head Movement Box: 30*22cm at 70cm

Tracking Distance: 50-80 cm

Max Gaze Angles: 35 degrees

Top Head-motion Speed: 25 cm/second

Screen Size: 17" TFT

Screen Resolution: 1280*1024 pixels

Display Colors: 16.7M

20 undergraduate and graduate students participated in our experiment. After excluding those subjects whose eye movements cannot be successfully tracked, the data of 18 subjects were analyzed. The remaining 18 subjects consist of 14 males and 4 females, aged from 19 to 24 with normal or corrected to normal vision. None of them has relevant knowledge in texture synthesis. No subjects have been exposed to this experiment more than once, so the learning effects can be avoided. They are all naive to the purpose of the whole process.

Several well-known techniques of texture synthesis, such as graph cut [14], image quilting [15], near-regular texture synthesis (NRT) [16], regularized patch-based and patch based [17], are widely applied to many fields. The graph cut approach attempts to handle the global regularity by incorporating a local correlation technique to determine the best pasting location. The main idea of image quilting is to synthesize new texture by taking patches of existing texture and stitching them together in a consistent way. NRT proposed by can depart from regular tiling along different axes of appearance. It is able to produce a regular structural layout and control the color variation. The basic idea of the patch based algorithm is to synthesize textures by directly copying image patches from the input texture. They also propose a modified approximate nearest-neighbor technique to speed up their model.

In our experiment, to make sure the data of textures having no risk of leaving out crucial details, we collect these data from Lin et al.[1]. These data was not produced by reimplementations of Lin et al., but they asked the authors to run their own algorithms or allow them to run their source code on the same set of input textures.

The images used in this experiment are eleven different structural textures as shown in Figure 3.2 and 3.3. Our database includes 10 near-regular and 1 irregular textures. Each texture has four synthesized textures generated by graph-cut[14], image-quilting[15], patch-based texture

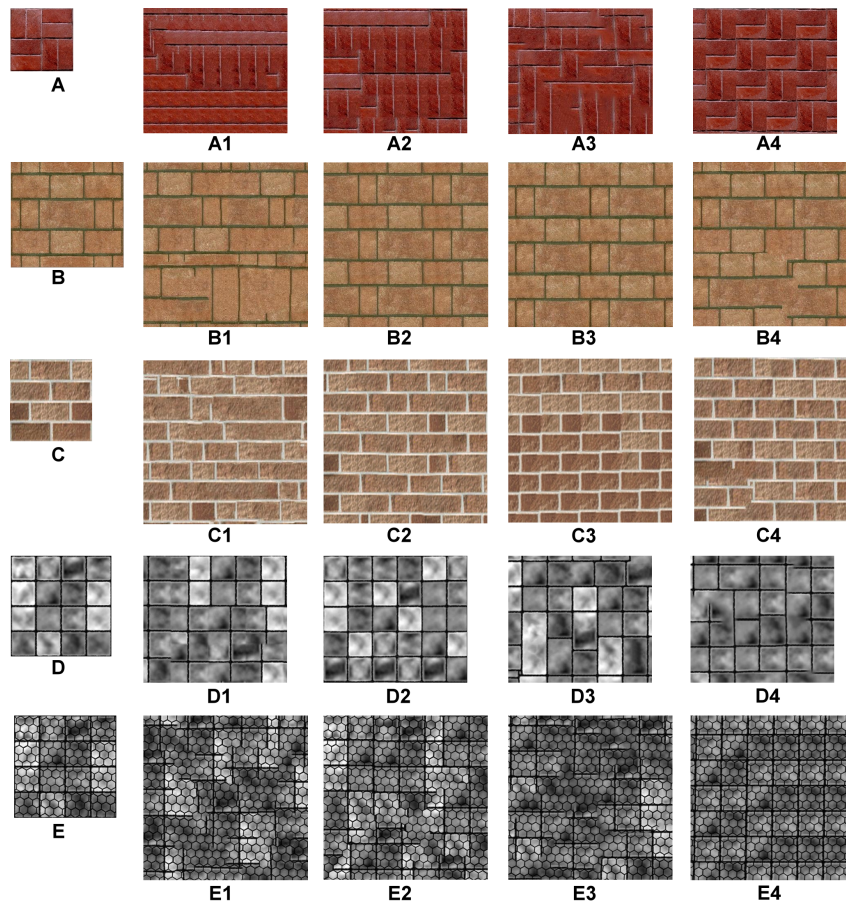


Figure 3.2: This figure shows the texture categories of A, B, C, D, and E and their texture synthesis results. These textures are near-regular textures.

synthesis[17], or near-regular texture synthesis[18]. There are 44 synthesized textures in total. Figure 3.2 and 3.3 show all the input and synthesized textures used in our experiment.

3.2 Experiment Procedures

In the beginning of the experiment, the subjects were asked to sit down with a position that they feel comfortable to look at the screen. The viewing distance from a subject to the screen is controlled within 50-80 cm with the screen, which is acceptable by the eye tracker. All sub-

jects have to do calibration for their eye-positions. During the procedure, a subject was asked to look at specific points on the screen. The resulting information is then integrated in the eye model and the gaze point for each image sample is calculated. If the accuracy of calibration pass the requirement, we can start the following procedure; otherwise the calibration should be performed again or the subject need to be replaced.

After calibration, subjects were told that the following scene will have two images. Left one is the input image for a texture synthesis algorithm; right one is the synthesis result of the left image on the screen. Figure 3.4 is what a subject really saw on the screen during the experiment. Then the subjects will be asked to give a score (between 1 and 5, 1 representing the least satisfactory and 5 representing the most satisfactory) for each right image, according to the quality of the synthetic texture compared to the input texture.

Finally, we began the recording process. Each pair of images appear on the screen for 10 seconds. After images disappear, the text which asks a subject to give a score will be shown on the screen. After the subject gives a score, next pair of images will appear. Each pair of images was shown in a random order. After showing 44 pairs of textures, the whole recording is completed. And we can then get the eye-movement data of subjects while they were rating each synthetic texture. Figure 3.5 illustrates our experiment process.

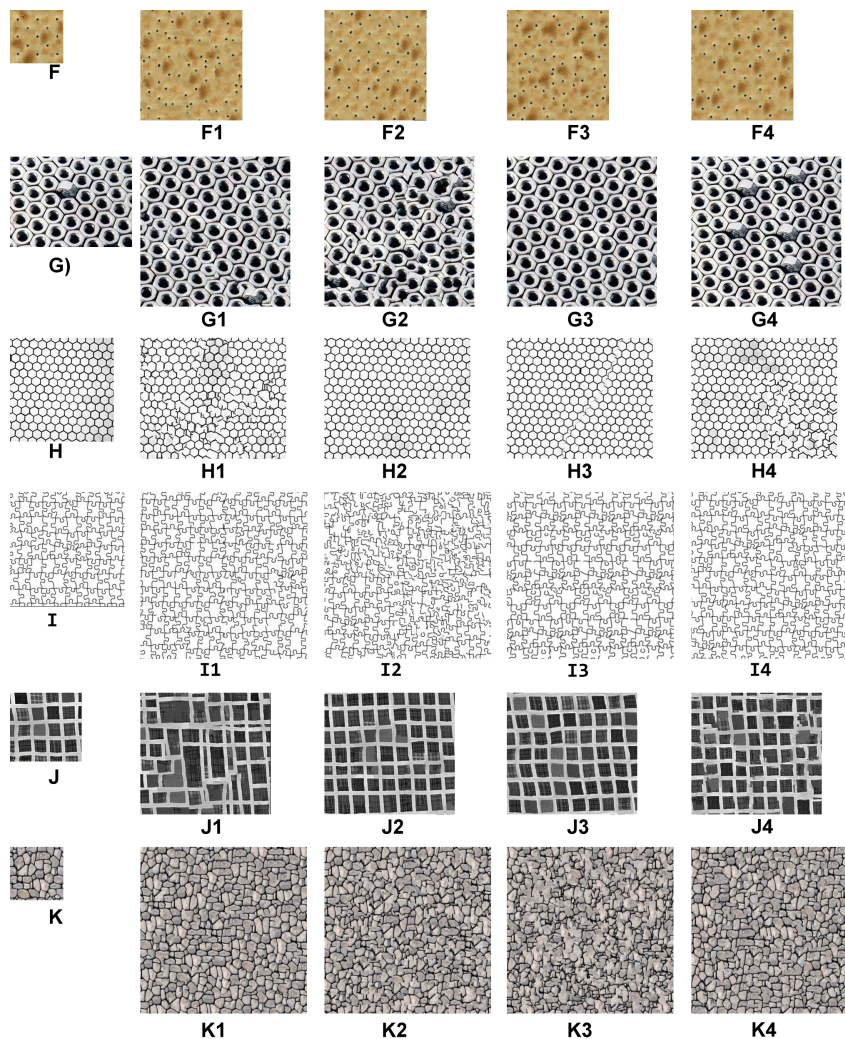


Figure 3.3: This figure shows the texture categories of F, G, H, I, J, and K and their texture synthesis results. Textures G, I, and K are irregular textures. The remaining ones are near-regular textures.

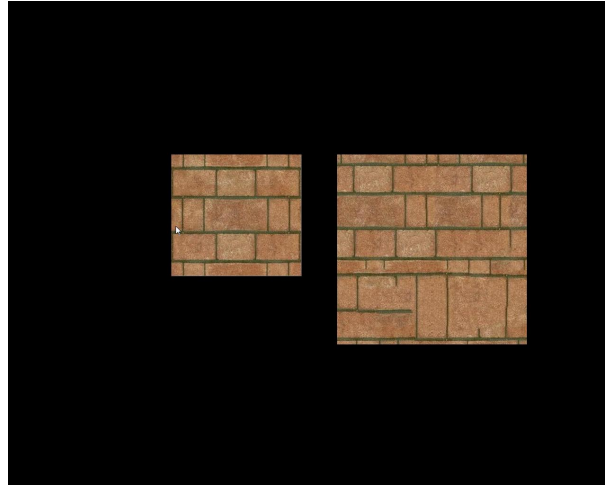


Figure 3.4: *This image shows what subjects really see on the screen.*

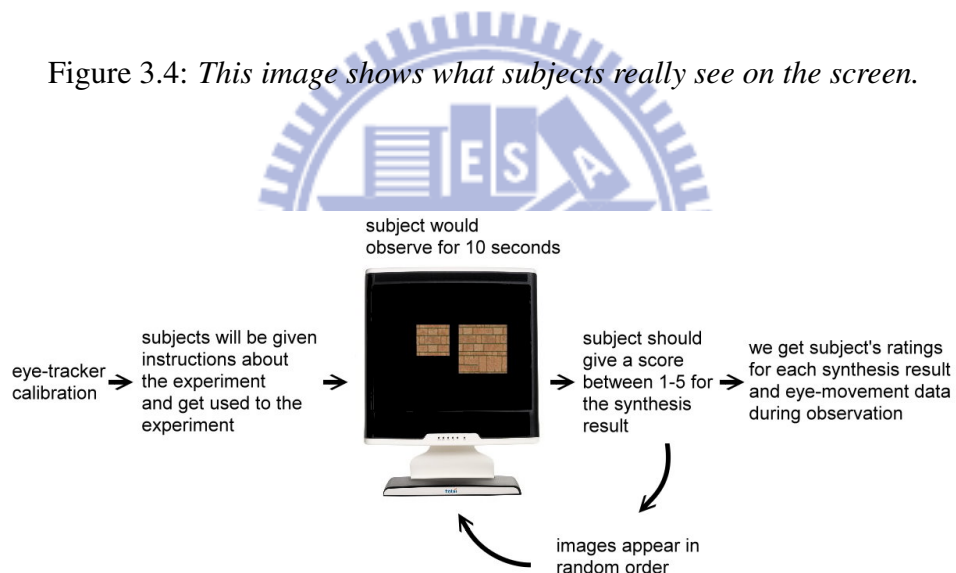
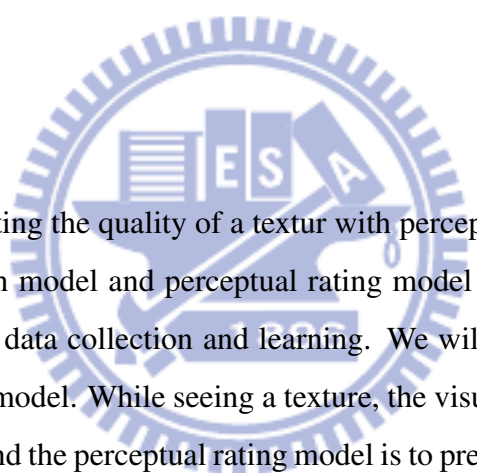


Figure 3.5: *Procedure of the experiment. For each subject, they was asked to sit and have a position which make them feel comfortable ,and then do the calibration for eye-tracker. Next, we will let subjects to see some sample image to let them get familiar with the environment. After these procedures, the main recording will be started: the subject will observe each image for 10 seconds, and then will be ask to give a score for the synthesis result. The images will appear in random order. After all the images have been observed and scored by the subject, we can get the eye-movement data during the observation and scores for each image.*

Our Approach



To satisfy the goal of evaluating the quality of a texture with perception, we present two models consisting of visual attention model and perceptual rating model in this thesis. Both of them include two stages: training data collection and learning. We will explain how to prepare the data and build the SONFIN model. While seeing a texture, the visual attention model is used to simulate human's fixation, and the perceptual rating model is to predict the score. We train these models with the ground truth data recorded from the eye-tracking experiment. In the following sections, we will describe our feature extraction approach, the SONFIN model and proposed visual attention and perceptual rating models..

4.1 Feature Extraction

Inspired by Peters et al.[9], we develop our model with bottom-up stage and top-down stage. The bottom-up stage defines features of synthesized textures and the top-down stage contains subjects' fixation data. Judd et al.[11] shows that we may sample positive-fixation parts and

negative-fixation parts as an input to train our model.

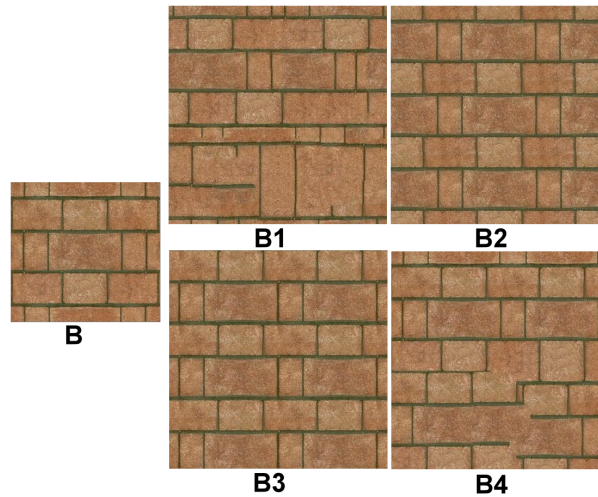


Figure 4.1: Example of structural textures. The left one is input texture and the right four are synthesized textures.

The goal of our experiment is to develop a model to evaluate textures. The result of Lin et al.[1] shows that that most regularity-preserved textures have higher user scores than regularity-broken textures. This implies that the regularity of structure has more effects than the color/intensity of textures while human judge synthetic structural textures as shown in Figure 4.1.

Example-based texture synthesis algorithm requires an input texture to synthesize a larger size of texture, so here we extract the feature by comparing synthesized texture and input texture. Figure 4.2 illustrates our feature extraction process. Since structure is the most important feature we need, we implement a Gaussian filter to remove details and noise, which would affect the extraction of structural features. We use the Canny edge detection algorithm to find edges of the textures. To obtain the best extraction result, we also change the sigma and threshold values of the Canny edge detection algorithm for different textures. After detection, the edge regions will be denoted as 1 and the other regions as 0. Then we dilate the detected edges, so a little

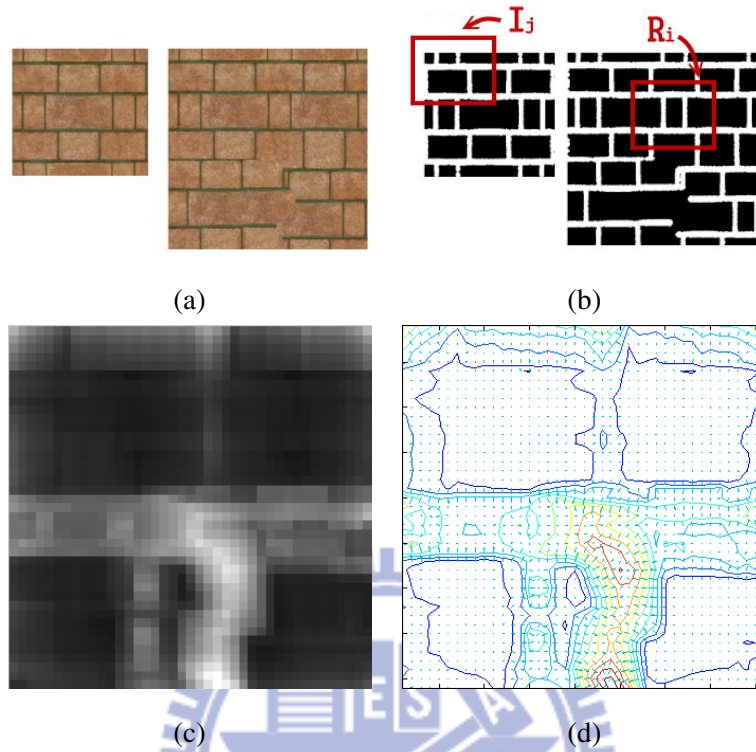


Figure 4.2: (a) is the original image of the two textures. Left one is the input texture, right one is the synthesis result from the input texture. (b) shows the edge images of (a). In (b), we calculate the difference of R_i with I_j which is the most similar patch to R_i in the edge image of the input texture. We visualize these error values as a structural error map as shown in (c). Brighter region means larger error. (d) is the gradient of structure error gained from computing the gradient of (c).

shift of the edges would not affect the structural error too much.

We measure the structural error of synthesized textures by comparing the differences between the edges detected in the synthesized texture and the input texture. We acquire many image patches from a synthesized texture by uniformly sampling the synthesized texture. For a near-regular texture, the size of the patches is equal to the size of the tile [18]; For an irregular texture, the size of patch is manually set to be the average size of its texture elements. We denote $R_1, R_2, \dots, R_i, \dots, R_n$ as the patches of synthesized texture, respectively. These patches are

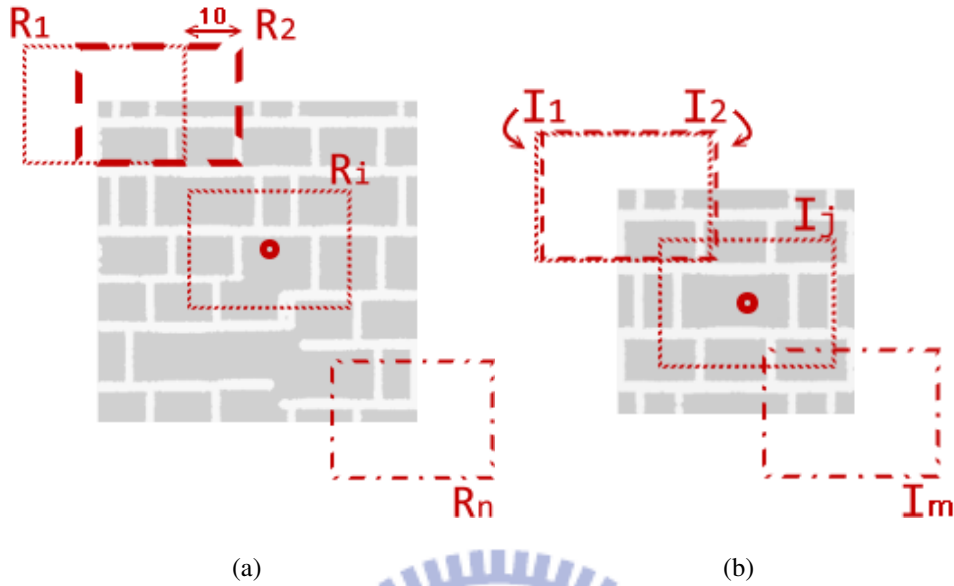


Figure 4.3: (a) shows some patches on the edge image of synthesized texture. (b) shows some patches on the edge image of input texture.

10 pixels apart along the horizontal and vertical directions. The center of R_i is larger than 0 and less than W_R and H_R . Here, W_R and H_R denote the width and height of the synthesized texture, respectively. Figure 4.3 (b) shows this sampling process.

The center of R_1 is on the top-left corner of the image, and the next patch R_2 is 10 pixels right to it. After we sampled all the patches in the first row, we will move 10 pixels downward to the next row, and continue the same process until we have sampled all the patches, R_n , in the texture.

To compute the structural error of each R_i , we compare it with the edge image of its input texture. For each R_i , we find the best matched patch in the edge image of the input texture. We denote these patches in the edge image as $I_1, I_2, \dots, I_j, \dots, I_m$ (with the same patch size of R_i). W_I and H_I denote the width and height of the input texture. The center of I_j should be in the range larger than 0 and less than W_I and H_I . Figure 4.3 (a) illustrates the patches in the edge image of the input texture. Note that the center of I_j is moved pixel by pixel in finding the best match process.

The center of I_i is on the top-left corner of the input texture, and we move 1 pixel to the right to get the next patch I_2 . After we sampled all the patches in the first row, we move 1 pixel downward to continue the same process until all the patches are sampled in the input texture. See Figure 4.3 (a)

For each R_i , the comparison is to calculate $\|R_i - I_j\|$, $\forall I_j \in \{I_1, I_2, \dots, I_m\}$, which simply calculated the difference of the value of R_i and I_j , and the norm of $(R_i - I_j)$.

We have to let R_i compare with every possible patch in the input texture to make sure that each R_i does find the smallest structural error value, even if the patches are not entirely included in the input texture.

As the portion of patch I_j that is not inside the input texture will bias the computation of the structure difference $\|R_i - I_j\|$, we multiply $\|R_i - I_j\|$ by a mask to mark the valid region. Another purpose of the mask is to calculate the area of the valid region. We divide the structural difference by the area of the valid region of a patch. This avoids the area bias the computation of structural difference,

$$\min_j \frac{Sum(\|(R_i - I_j)\| * Mask_j)}{area(Mask_j)}, j \in 1, 2, \dots, m, \quad (4.1)$$

where $Sum(\cdot)$ denotes the sum of all elements of a matrix, $\|\cdot\|$ is the absolute value, $*$ is the element-wise product of two matrices, and $Mask$ mark the valid region. We record this value in a matrix as structural error map.

After we get the structural error map of each synthetic textures, we normalize them among all textures synthesized from the same input texture to form normalized matrix, E , shown as Figure 4.2(c).

Besides the structural error, gradient error along x-axis and y-axis dominates the viewer's judgements. We get this feature by computing the gradient of E .

$$\nabla E = \left(\frac{\delta E}{\delta x}, \frac{\delta E}{\delta y} \right) = (G^x, G^y) \quad (4.2)$$

where

$$\begin{cases} G_{i,j}^x = E_{i,j+1} - E_{i,j} \\ G_{i,j}^y = E_{i+1,j} - E_{i,j} \end{cases} \forall i, j \quad (4.3)$$

To visualize G^x and G^y , we show them in Figure 4.2(d).

In addition, human visual habits affect the result. Hence, the information of the gaze position are considered as important features. We set two feature vectors, the first collects the vector of normalized structural error E and the position X, Y ,

$$\Phi = \begin{bmatrix} E_{X_1, Y_1}, X_1, Y_1 \\ E_{X_2, Y_2}, X_2, Y_2 \\ \cdot \\ \cdot \\ \cdot \\ E_{X_n, Y_n}, X_n, Y_n \end{bmatrix}, \quad (4.4)$$

The second feature vector consists of the gradient of structural error map G^x, G^y and the

information of positions X, Y ,

$$\Psi = \begin{bmatrix} G_{X_1, Y_1}^x, G_{X_1, Y_1}^y, X_1, Y_1 \\ G_{X_2, Y_2}^x, G_{X_2, Y_2}^y, X_2, Y_2 \\ \cdot \\ \cdot \\ \cdot \\ G_{X_n, Y_n}^x, G_{X_n, Y_n}^y, X_n, Y_n \end{bmatrix}, \quad (4.5)$$

for the synthesized texture. We also visualize it as which can be seen in Figure 4.2(c). In the chapter of Result, the results of these two feature vector will be compared.

4.2 Self-Constructing Neural Fuzzy Inference Network

There are many machine learning algorithm to find the mapping function. These models include multi-layer neural networks, support-vector machines, and the Expectation/Maximization algorithm. The feature vectors in our experiment, however, are too large and too complicated to train by a simple learning approach. Fuzzy system would be a good choice. Obviously, it is difficult for human experts to examine all the input-output data from a complex system to find a number of proper rules for the fuzzy system. Though several approaches, such as consisting of two learning phases, the structure learning phase and the parameter learning phase, are presented to solve this difficulty, they have to be done sequentially. This fact makes the traditional fuzzy system suitable only for off-line operation.

Chia-Feng Juang and Chin-Teng Lin[5] proposed a novel machine learning model called Self-Constructing Neural Fuzzy Inference Network(SONFIN) with on-line learning ability. The SONFIN is inherently a modified Takagi-Sugeno-Kang(TSK)-type fuzzy rule-based model possessing neural network's learning ability.

Finding the number of proper rules is always a difficulty in fuzzy system. Owing to these problems, SONFIN is developed to simultaneously handle structure as well as the parameter learning phases. So, here we choose SONFIN as our model for two reasons. First, SONFIN can find its structure and parameters to model an economic network size automatically. Second, the learning speed as well as the modeling ability of the SONFIN are all appreciated.

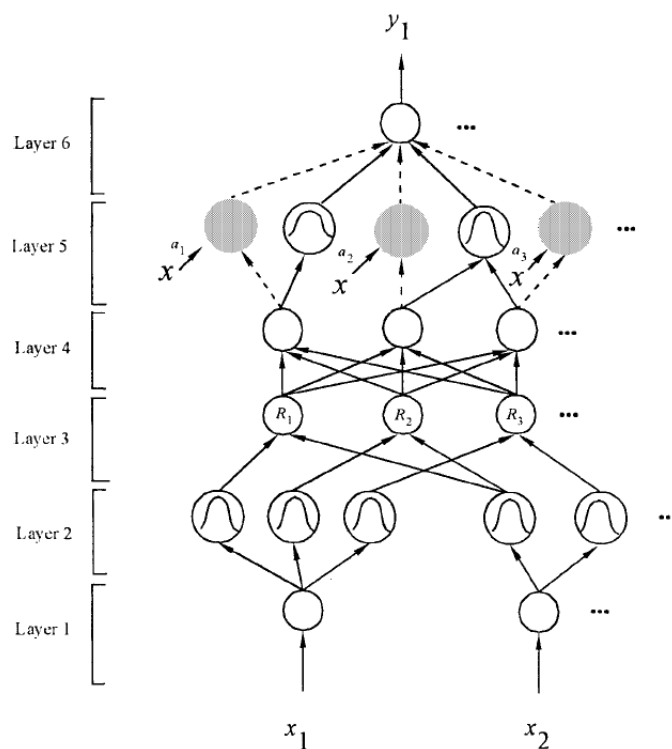


Figure 4.4: Graphical illustration of Self-Constructing Neural Fuzzy Inference Network proposed by Chia-Feng Juang and Chin-Teng Lin in 1998. The goal of this model is to learn the mapping from the input to the output using historical data so the model can then be used to produce an output while the desired output is unknown.

Shown as Figure 4.4, there are totally 6-layers in the SONFIN model.

Rule i : IF x_i is A_{i1} and x_n is A_{in}
 THEN y is $a_{0i} + a_{ji}x_j + \dots$

where A_{ij} is the fuzzy set of the j th linguistic term of input variable x_j , and a_{ji} 's are the consequent parameters. a_{0i} is the center of a symmetric membership function on y . Let $u_i^{(k)}$ and $o_i^{(k)}$ denote the input and output of the i th node in layer k , respectively. The followings describe the functions of each layer of the SONFIN.

Layer 1:

$$o^{(1)} = u_i^{(1)}. \quad (4.6)$$

We only transmit input values to the next layer directly in this layer. Note that we may apply a linear transformation in this layer proposed in enhanced SONFIN[5]. Though no computation performs in this layer, we still keep it as one layer of six-layered structural SONFIN.

Layer 2:

$$o^{(2)} = e^{-\frac{(u_i^{(2)} - m_{ij})^2}{\sigma_{ij}^2}}, \quad (4.7)$$

where m_{ij} and σ_{ij} are the center and width of the Gaussian membership function of the j th partition for the i th input variable x_i , respectively. In this layer, each node corresponds to a fuzzy set of the input variables in Layer 1. Here Gaussian membership is performed as the function of Layer 2 shown in Equation 4.7.

Layer 3:

$$\begin{aligned} o^{(3)} &= \prod_{i=1}^q u_i^{(3)} \\ &= e^{-[D_i(\mathbf{x}-\mathbf{m}_i)]^T [D_i(\mathbf{x}-\mathbf{m}_i)]}, \end{aligned} \quad (4.8)$$

where q is the number of Layer 2 nodes participating in the IF part of the rule. Each node in this layer stands for a fuzzy logic rule, AND operation.

Layer 4:

$$o^{(4)} = \frac{u_i^{(4)}}{\sum_{j=1}^r u_j^{(4)}}, \quad (4.9)$$

where r is the number of rule nodes in Layer 3. The firing strength calculated in Layer 3 in this layer is normalized.

Layer 5:

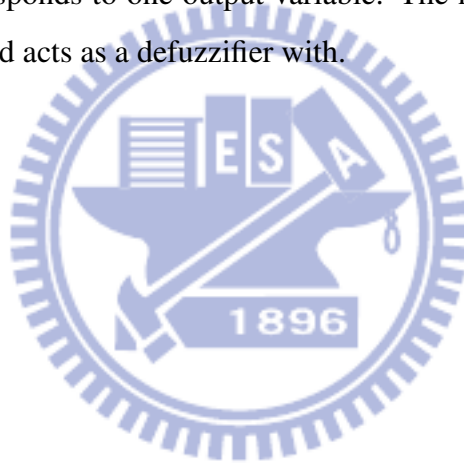
$$o^{(5)} = \left(\sum_{j=1}^n a_{ji}x_j + a_{0i} \right) u_i^{(5)}. \quad (4.10)$$

This layer is called the consequent layer. The blank and shaded circles represent two types of nodes used in this layer. The blank circle (blank node) is the essential node representing a Gaussian fuzzy set of the output variable. As to the shaded circle (shaded node), it represents a linear combination of input variable.

Layer 6:

$$o^{(6)} = \sum o_i^{(5)}. \quad (4.11)$$

In Layer 6, each node corresponds to one output variable. The node integrates all the actions recommended by Layer 5 and acts as a defuzzifier with.



4.3 Visual Attention Model

In the eyetracking experiment, viewers examine scenes in a series of fixations (periods which the eye movements is stable, viewing a single point) and saccades (quick eye movements between points). Fixations are the points that viewers find meaningful. Since viewing is inhibited during saccade, all samples recorded during saccades can be discarded.

Here, we aggregate fixations by Clear View fixation filter, which is a built-in function of tobii studio. The Clear View fixation filter will first check if two gaze points are within a pre-defined minimum distance from each other. If yes, they should be considered as belonging to the same fixation. It also need a minimum time limit for which gaze need to be within the pre-defined minimum distance to be considered a fixation. Here, as recommend for pictures, the minimum distance is 50 pixels and the time limit is 200ms. After all data have been processed, we get each subject's fixation points on each texture. The score of each texture is also recorded.

We prepared the training data by processing the fixation data as follows. The fixation data from an eye-tracker contain the x and y coordinates and duration of each fixation. We have to encode these fixation data into a fixation intensity map. The image being observed is divided into square regions of 20×20 pixels. We define the intensity of a fixation as $\lfloor \frac{t}{100} \rfloor$, where t is the duration of the fixation in millisecond. Furthermore, we define the fixation intensity within a square region as the sum of the intensities of all fixations located in the square, as shown in Figure 4.5.

Following above definitions, we build the fixation intensity map of a texture by combing the fixation data from all subjects viewing the texture shown in Figure 4.6(a). The fixation intensity map is treated as ground truth data for training SONFIN. By setting the most salient 50 percent of these data as threshold, we have two sets of fixation information, positive and negative fixation as Figure 4.6(b), We randomly gather 50 samples from each of these two sets as the training

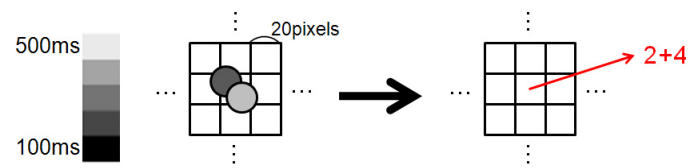


Figure 4.5: The figure shows how we encode fixations in x and y coordinates into fixation intensity map. In the left image, darker dot in the square region means fewer fixations, lighter dot means more fixations. The right image shows the intensities values in the square region, which will be added together.

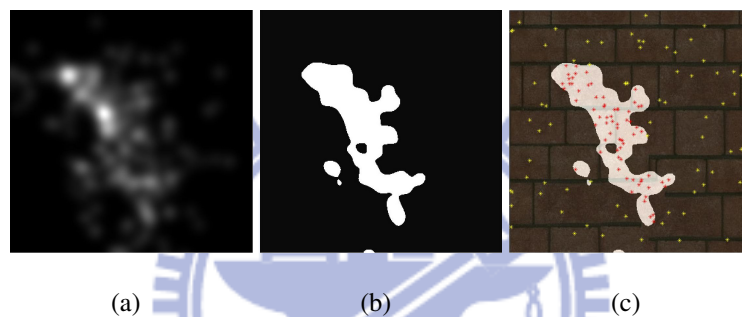


Figure 4.6: (a) The eye-tracking data of a synthesized texture collected from all testers. (b) set salient 50% as threshold. (c) 100 samples from positive and negative fixations (50 for positive and 50 for negative).

data, see Figure 4.6(c). There are totally 11 kinds of input texture. For each input texture, we pick 2 from 4 kinds of synthesized texture. For each synthesized texture, we gather 50 positive samples and 50 negative samples. For each sample, we construct a feature vector consisting of structural error and sampled position. Thus, we obtain a training set of 2200 items. Each item includes a feature vector and a scalar fixation intensity value.

4.4 Perceptual Rating Model

In eye-tracking experiment, we have recorded rating data of each subject for every texture, respectively. That is, each texture has 18 subjects' rating. Before setting the data as desire output in SONFIN model, we recursively compute the mean and remove outliers, which are beyond 3 times variance from mean. If there are no more outliers, the mean is kept as rating of the texture. In the following paragraphs, we call it as user score.

There are definitely some connections between structure error and user score. The assumption is that the larger structure error human focuses the less user score will be and we will prove this assumption in the next chapter. In this model, we compare three different fixation models (e.g. user-fixation-map, predict-fixation-map, random-fixation-map shown as Figure 4.7). We calculate error with fixation map as the first pattern. Fixation map becomes a weighting matrix used to multiply structure error map as one of the input training patterns shown as equation 4.12.

$$F = \frac{\sum_{i=1}^N \sum_{j=1}^M (\Gamma_{i,j} \cdot E_{i,j})}{N \cdot M}, \quad (4.12)$$

where F denotes fixation error, Γ may be a fixation map of user-fixation-map, predict-fixation-map, random-fixation-map or uniform-fixation-map, and E denotes the structural error map. N and M stand for the height and width of E .

According to Judd et al.[11] and Goldberg et al.[19], human eyes usually search an object of interest from the center of the image. The second training pattern is defined as distribution ratio (D). While seeing a well-synthesized texture, subject would focus around the center most. So we consider distribution ratio as an important feature. To compute D , every fixation point on the fixation map must be examined whether it is closed to the center. Shown as Figure 4.8, we define P as fixations within the circle and Q as fixations outside the circle. Equation 4.13 illustrates the formula of D .

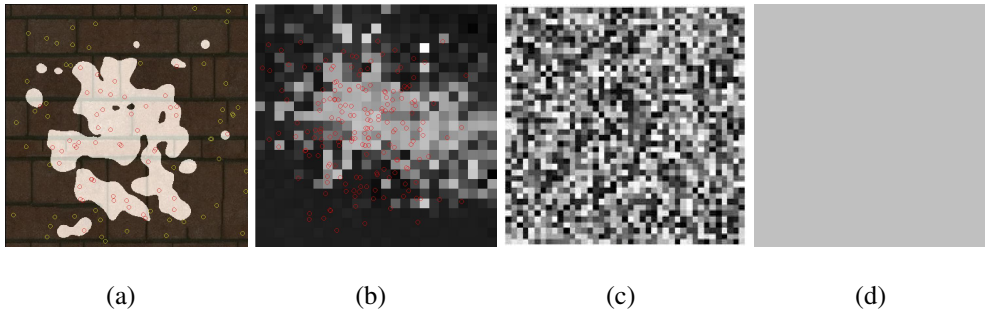


Figure 4.7: (a) is the user-fixation-map gathered from ground truth data. (b) is the prediction-fixation-map gathered from visual attention model. (c) is the random-fixation-map. (d) is the uniform-fixation-map

$$D = \frac{P}{P+Q}$$

(4.13)

The third and the fourth training patterns are defined together. Considering the basic evaluation method, the structure error of texture cannot be ignored. Observing the ground truth data and structure error, for well-synthesized texture, we would find low mean value and low variance, but for poor-synthesized texture, we would find unstable mean value and high variance. So we compute mean and variance of the structural error as the third and fourth training patterns, A and V (shown as figure 4.9).

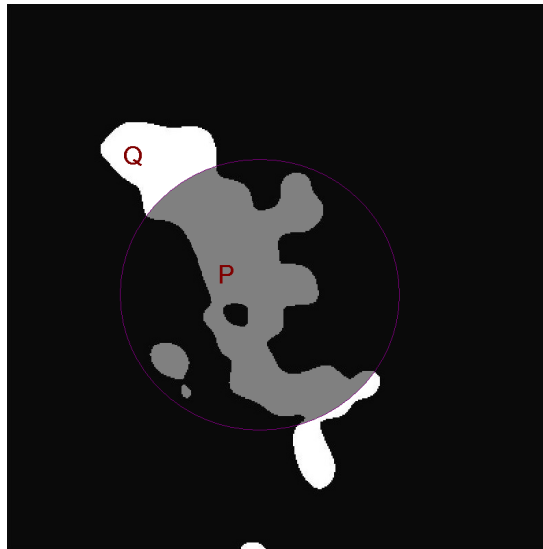


Figure 4.8: P denotes fixations within the circle and Q denotes fixations outside the circle.

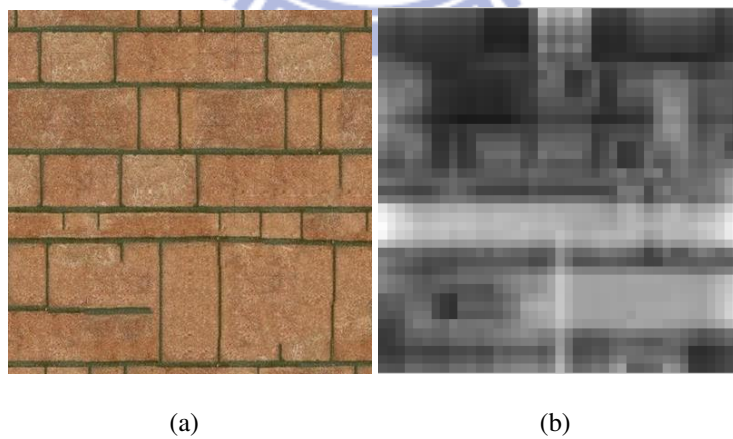


Figure 4.9: (a) is an example of synthesized textures, B1. (b) The structural error map of B1. A and V standing for the third and fourth feature are the mean and variance of structural error map.

CHAPTER 5

Results

In this chapter, we will first introduce our evaluation method and then the validation results of visual attention model and perceptual rating model. We then compare our model with saliency map. A modified saliency map is also compared with our model. Additionally, we compare the results of our model with different feature inputs of Φ and Ψ defined in chapter 4. Note that Φ are typically the feature vector used by Yu [13] consisting of structure error and position. Ψ proposed in this thesis contains gradient of structure error and position.

5.1 Evaluation Method

To quantitatively evaluate how accurate our models prediction matches observers actual fixation positions, we use two metrics, normalized scan-path saliency (NSS) [9] and receiver operating characteristic (ROC). The NSS method is defined as the response value at the current fixations on a model's predicted gaze density map that has been normalized to have zero mean and unit variance. Here, we put all observers' fixations on the predicted gaze density map and compute the average of all response values of the current fixations. This average value is called average NSS. We will compute the prediction rate $P\%$, which means average NSS is above $P\%$ of the

distribution of the response value across the entire gaze density map. Higher prediction rate means better prediction.

Another method is ROC, which is a well-known and useful technique for organizing classifiers and visualizing their performance. ROC plots are commonly used in medical decision making. Here we set a threshold between a range from 5% to 100%. For each threshold, two values are calculated; the True Positive Ratio (the number of outputs greater or equal to the threshold divided by the number of one targets) and the False Positive Ratio (the number of outputs less than the threshold divided by the number of zero targets). Note that the more each curve hugs the left and top edges of the plot, the better the classifier is. Figure 5.1 shows an example result of NSS and ROC of the prediction rate on a synthesized texture.

5.2 Prediction Experiments on Structural Textures

We designed two sets of experiments. In the first set, two synthesized textures were picked as training data from each kind of synthesized textures, and then the remaining two synthesized textures used to test our prediction model. That is, we test how a trained model performs on unseen textures. We call the first set of experiment visual attention experiment. In the second set, we build an association model to learn the relationship between subjects' gaze behavior and rating. This experiment is called perceptual rating experiment.

We tuned the parameters for the SONFIN to obtain the best results. We found that the value of membership threshold and the distribution of sampling play an important role in our experiments. Smaller membership threshold would cause more rules and larger threshold leads to fewer rules. Too many rules or too few rules makes the training result overtraining or non-convergent. The membership threshold is set as 0.003 and the learning rate is set as 0.004. To avoid overtraining, the minimal error is set to 0.01 and the maximal number of iteration is 5000.

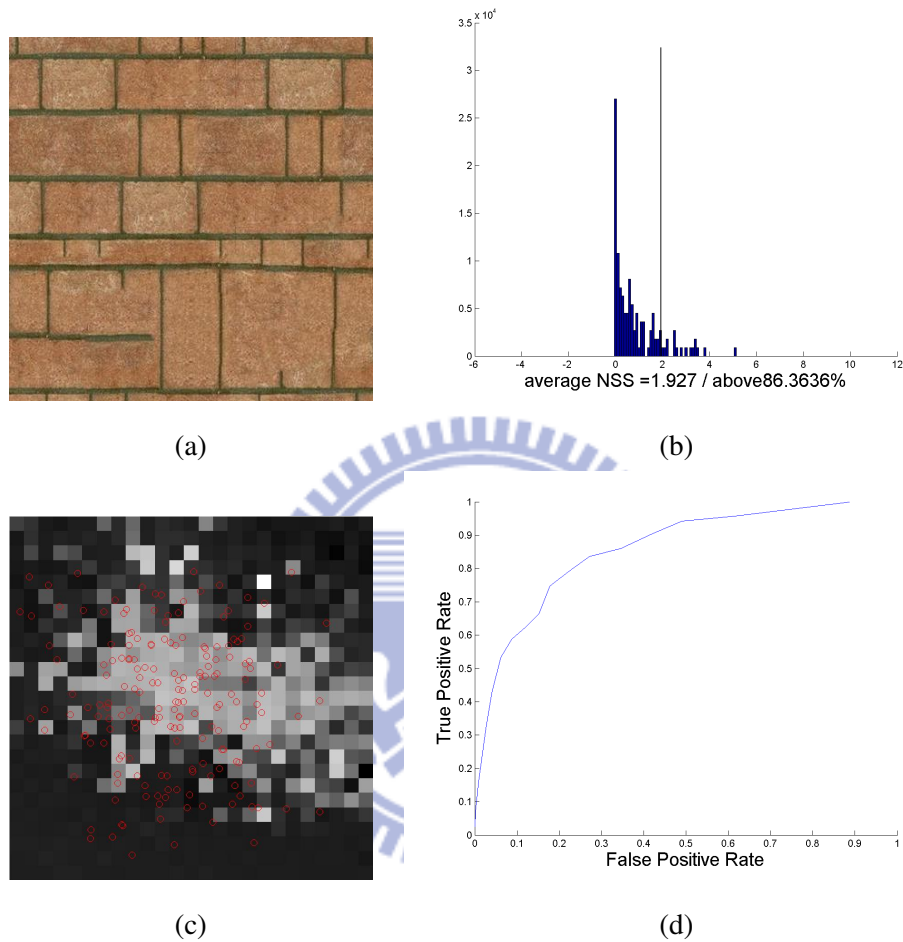


Figure 5.1: (a) The original image B1 shown to the subjects. (b) The average normalized response value across all fixations is taken as the average normalized scan-path saliency (NSS). The dashed line shows average NSS, which is compared against the distribution of response value across the entire gaze density map (gray histogram). (c) The normalized gaze density map of (a), where red circles represent a series of fixations of a subject. (d) The ROC curve defined by true positive ratio and false positive ratio. The more each curve hugs the left and top edges of the plot, the better the classification.

One problem is that we might gather biased samples which would cause bad training result, so we repeated sampling and training for three times to guarantee a better result.

5.3 Validation of The Visual Attention Model

We ran two sets of prediction experiments to validate the effectiveness of our model. One is new-texture-prediction and the other is new-subject-prediction, respectively. The setup of new-textures-prediction is to divide 44 textures into 22 training textures and 22 testing textures. For new-subjects-prediction, we separate 18 subjects into 9 training subjects and 9 testing subjects.

5.3.1 New-Texture-Prediction

In this result, we compare our results with the results generated by the saliency map. One can observe that Saliency map usually cannot predict fixations well. Its predicted fixations scatter in the entire synthesized textures. According to Yu's thesis[13], we also adopt multi-layer perceptron (MLP) as the learning model and compare with our results. The average NSS of MLP is 74.95%. Our model has NSS average predictive rate of 80.47% and saliency map only has 56.00% (Figures 5.2). Some of our results do not have a higher predictive rate than saliency map since those results are irregular textures, on which our structural error feature does perform well. Figure 5.3 shows the comparison between our model and saliency map on ROC curve. One can find our model more accurate than saliency map since our ROC curve hugs the left and top edges more, which means a better classifier.

Besides, we replace feature vectors of our model from Φ to Ψ . Shown as Figures 5.2, the average NSS of Ψ is 82.73%. The result of Φ only exceeds Ψ by 2.26%, but from the angle of view that the lowest NSS of Ψ is not lower than 75.0% and the lowest NSS of Φ is 57.61%, Ψ performs much more stable than Φ does. Notice that Ψ climbs slower than Φ only at the

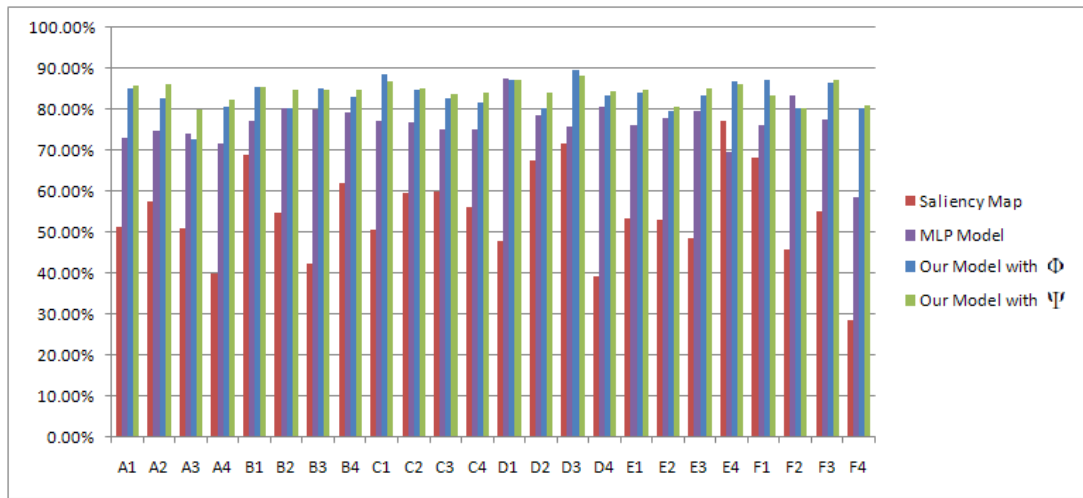
beginning in Figure 5.3. After false positive rate exceeds 0.2, Ψ has better performance than Φ .

One may argue that saliency map does not acquire or learn from human subjects' eye tracking data. To fairly compare with saliency map, we train SONFIN using saliency map value as the input feature vector. We call it modified saliency map. The comparison result is shown in Figure 5.4. The NSS average value of modified saliency map is about 57%, not higher than the result of original saliency map. Also, compare to our model, the results of modified saliency map are very unstable. Although the highest prediction rate is more than 90%, the lowest prediction rate is only around 10%. From this comparison, one can find that structural error is a better feature than saliency map for predicting the fixations on structural textures. By the way, modified saliency map has higher average NSS than our model in texture J because texture J has higher diversity on color and intensity, shown as Figure 5.6. Figure 5.5 shows the ROC curves of modified saliency map and our models with two feature sets.

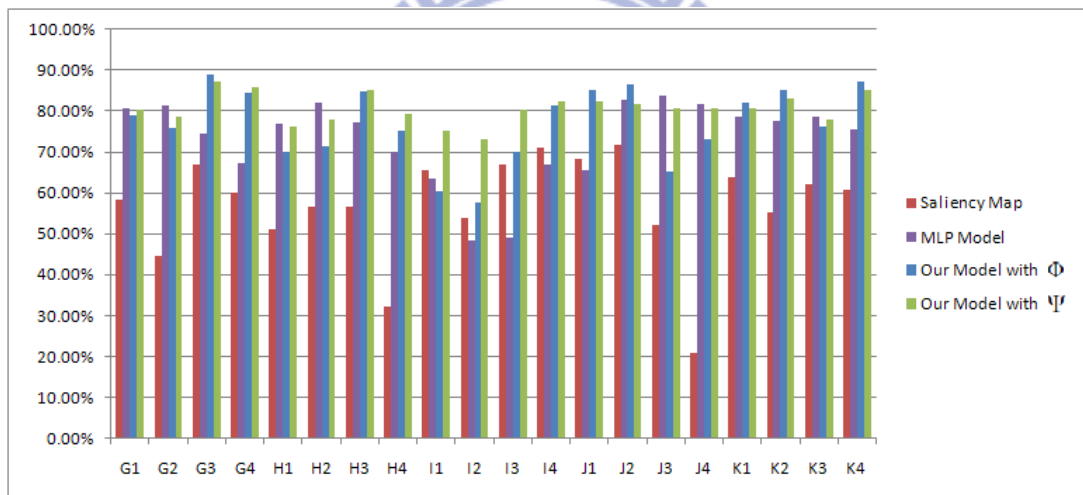
Some predictions of the results are rather lower than the others. Take Figure 5.7 as an example, NSS of H1 and H2 are 76.22% and 77.85%. One may observe that poor-synthesized parts occupy the most region of H1 and well-synthesized parts occupy H2. For those extremely good or bad textures, we have to predict the fixations only with position information, and thus lack of feature of gradient makes the result not so well. This makes our model to predict subjects' fixations worse than H3 and H4.

5.3.2 New-Subject-Prediction

All subjects are divided into two groups, group 1 and group 2. We set one of the two groups to be the training set and the other to be the testing set, and vice versa, so that we are able to predict new subjects' fixations. In this model, Φ and Ψ are used as the training patterns, respectively. Figure 5.8 and 5.9 illustrate the comparison. Figure 5.10 is the ROC curve. In this graph, while



(a)



(b)

Figure 5.2: This figure shows the NSS average rate of our model and saliency map for each synthesized texture. Our model has two results generated by two feature input sets, Φ and Ψ . (a) shows the results for texture A, B, C, D, E, F. (b) shows the results for texture G, H, I, J, and K. Most of our results have higher predictive rate than saliency map except I3. This is because textures I is an irregular texture. Larger variations makes it harder to extract valid structural features. The result of Φ and Ψ are similar, but Ψ performs slightly better in I, J, K, which Φ does not predict well.

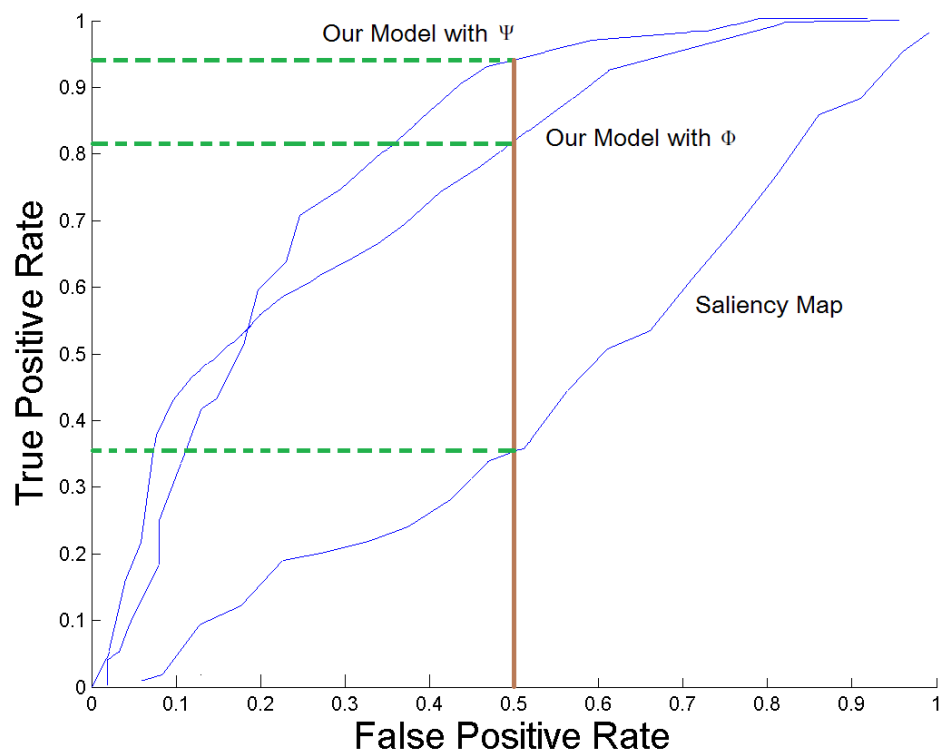
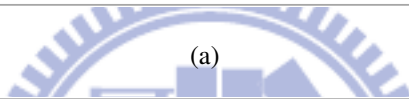
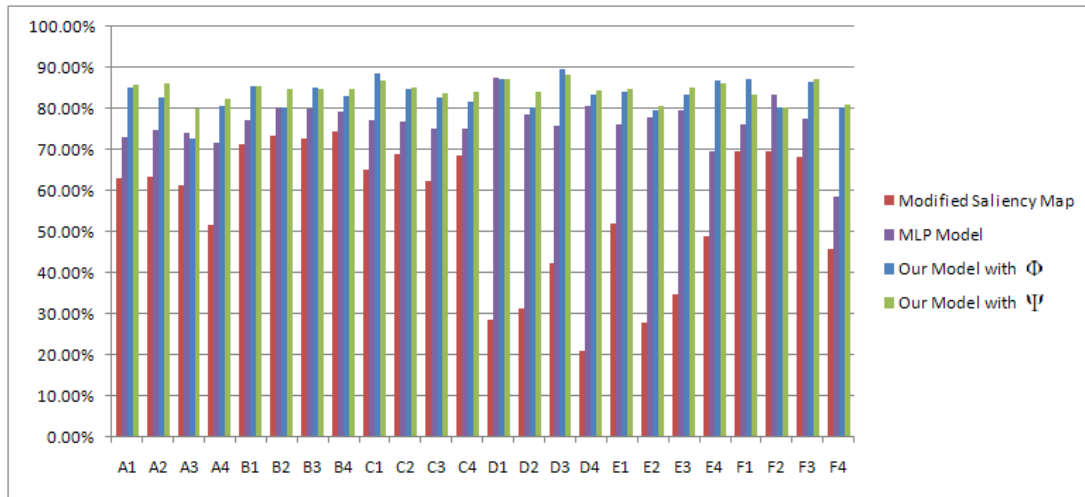
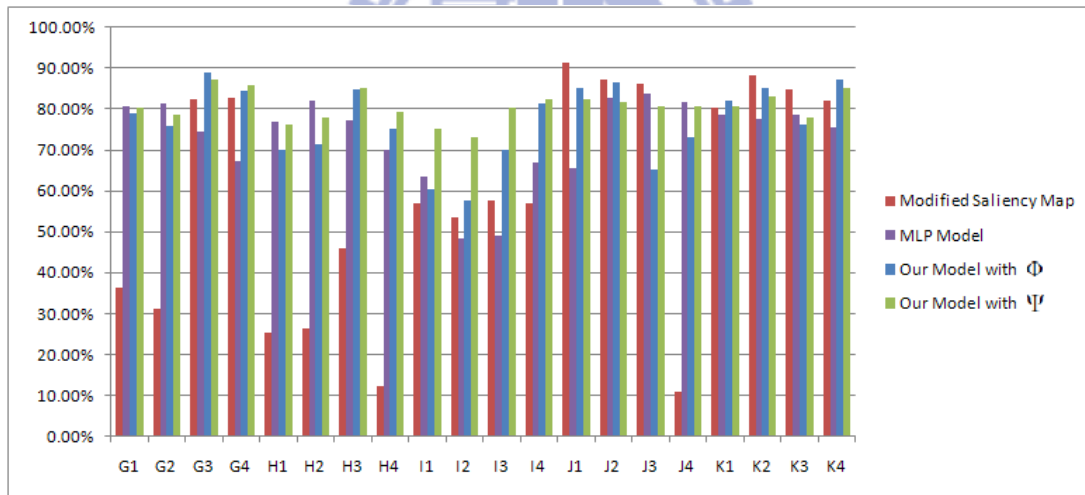


Figure 5.3: This graph shows the comparison of our model and saliency map in ROC curve. Since that the more each curve hugs the left and top edges the better the classification, we find our model more accurate than saliency map.



(a)



(b)

Figure 5.4: The NSS average value of Modified Saliency Map is only 57%, not higher than the result of original saliency map. Also, compare to our model, the results of modified saliency map are very unstable. The lowest accuracy is around 10% but the highest is more than 90%. This comparison shows that saliency map has lower correlation to the fixations on synthesized textures than our structural error feature.

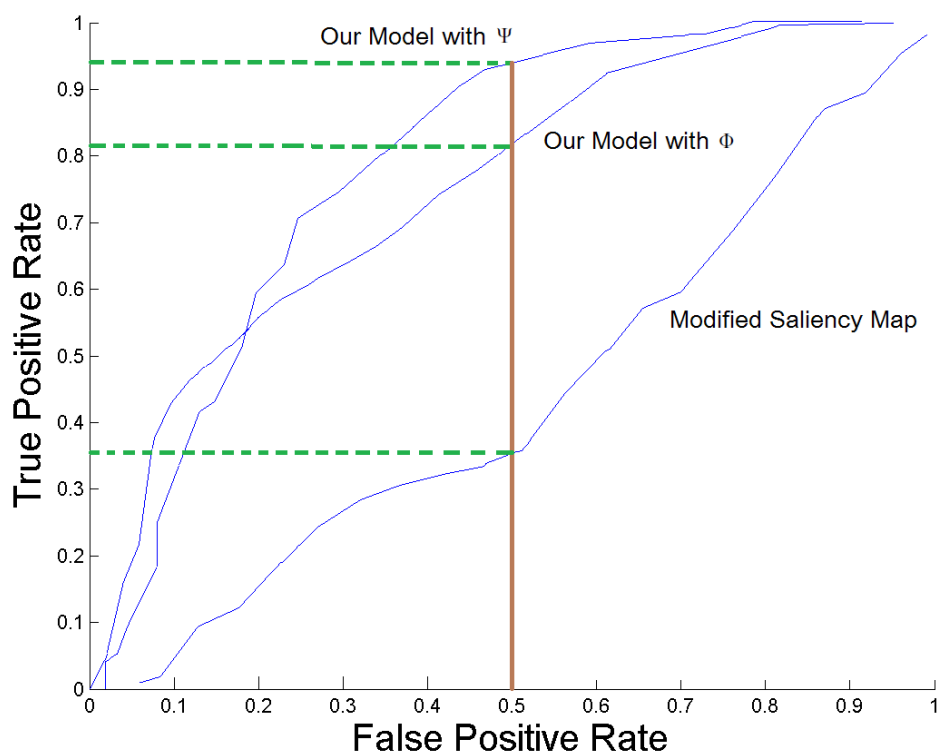


Figure 5.5: This graph shows the comparison of our model and modified saliency map in ROC curve. Our model still performs better than modified saliency map. In this result, we compare the ROC curve of Φ and Ψ . We may find that Ψ has higher precision than Φ .

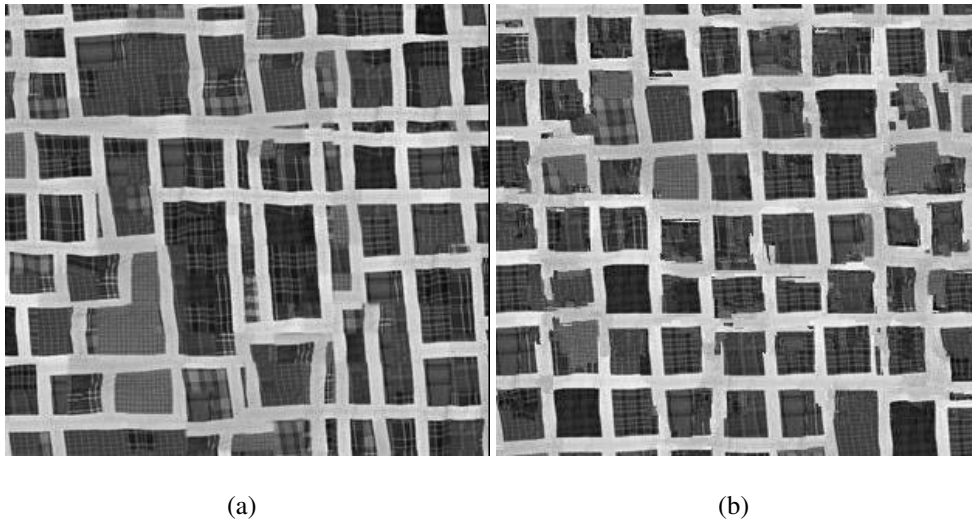


Figure 5.6: (a),(b) are two textures of texture J. One may find that these textures contain more diversity on color intensity. It makes modified saliency map performs better than our model.

increasing the threshold of ROC, Ψ has better result than Φ at the beginning. After the threshold exceeds around 0.35, curve of Φ hugs the top left faster than Ψ . Due to the diversity on peoples sensitivity to illumination, color and aesthetics, we do not always get high consistence for our subjects fixation. In our result, the average NSS of Φ is 75% and Ψ is 73%. One may wonder why Ψ does not work better than Φ just like he does in New-Textures-Prediction. We observe that the characteristics of Ψ is more likely to memorize human habits while deciding the scan-path. Because of the diversity on people's habits, we claim that Φ is a more general feature for predicting human behavior.

5.4 Validation of The Perceptual Rating Model

One may wonder that what is the difference to adopt fixations and fixation-free model to predict rating. In order to validate the effectiveness of the features we proposed, we run The perceptual rating model with four different fixation maps, e.g. user fixation map, predicted fixation map, random fixation map and uniform fixation map. The user fixation map is typically the

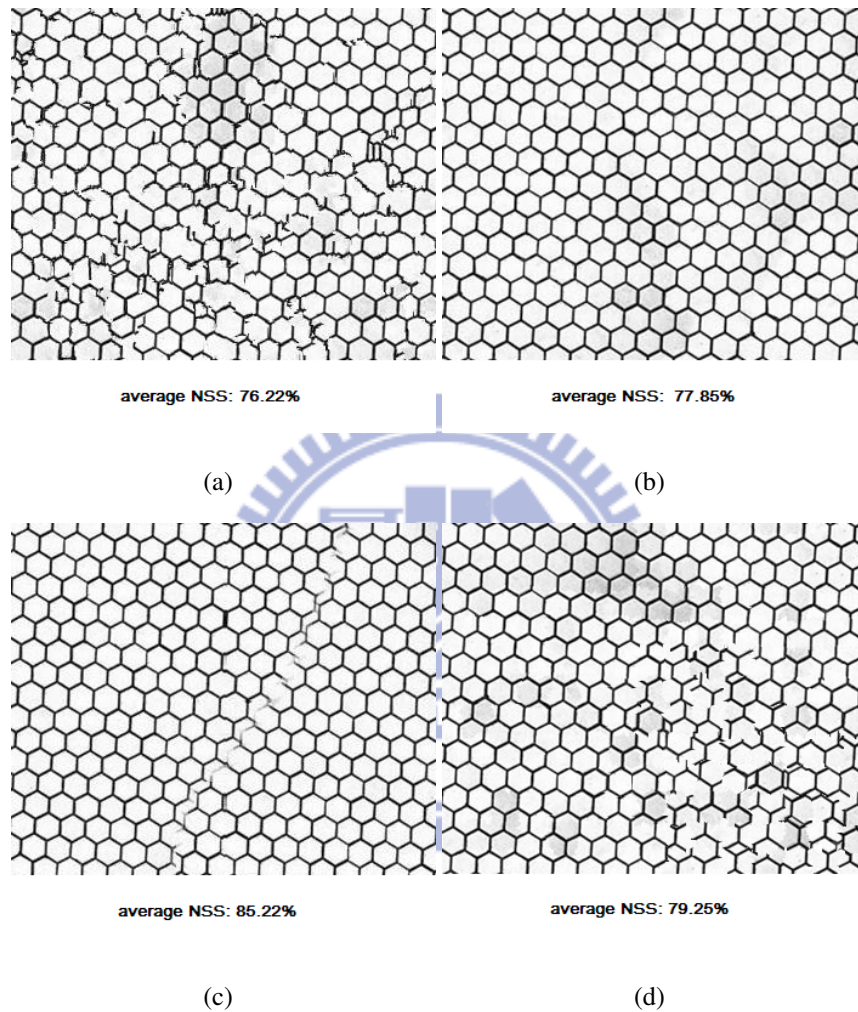
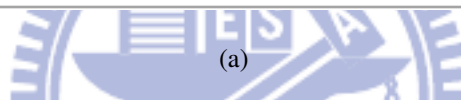
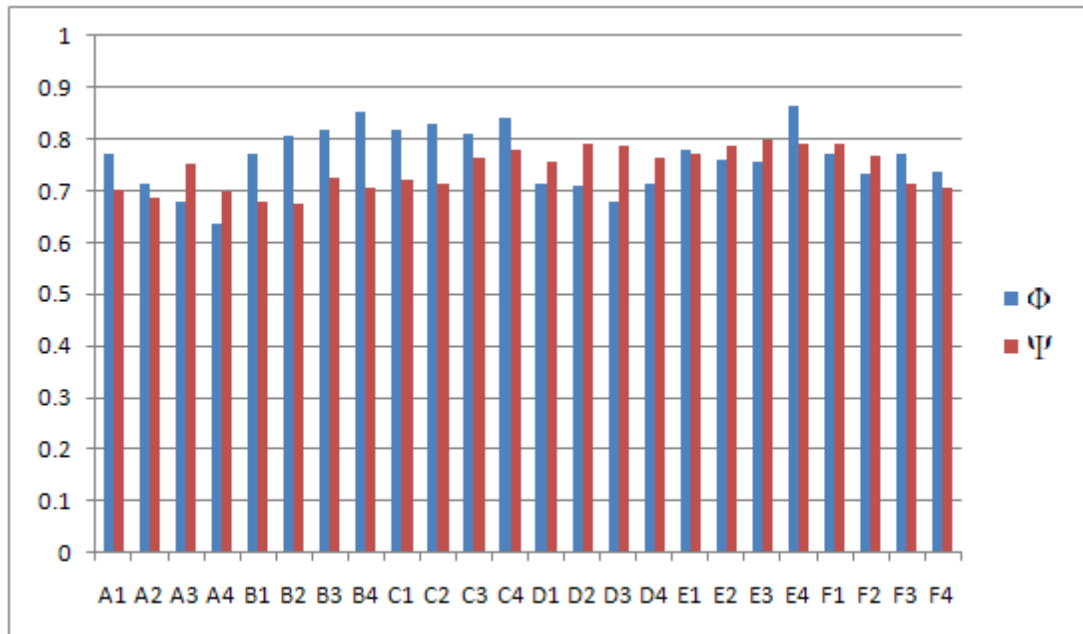
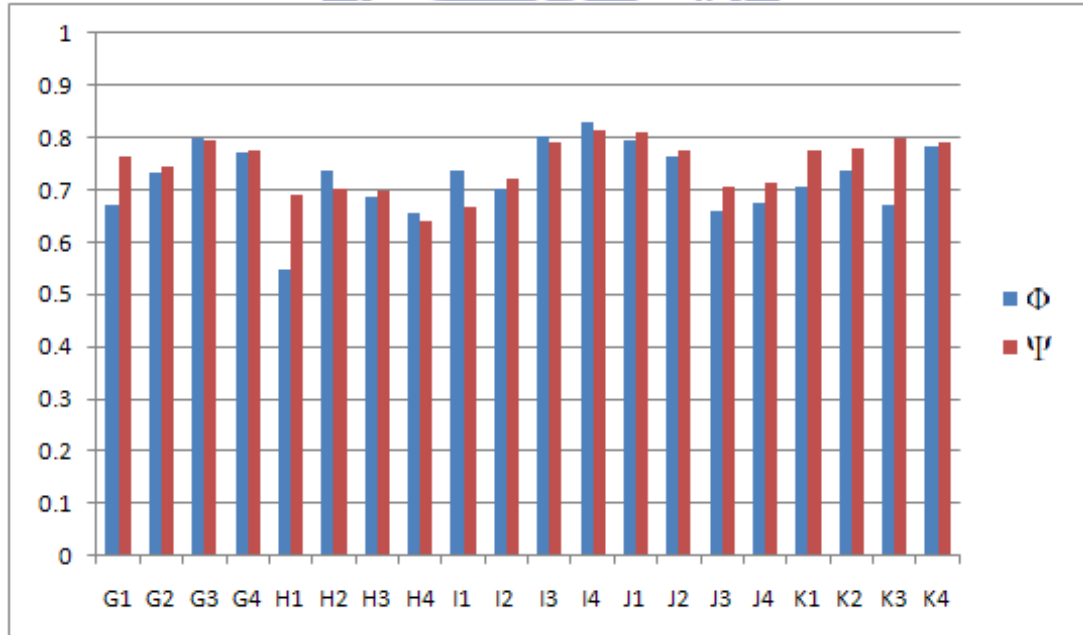


Figure 5.7: (a) is H1. (b) is H2. (c) is H3 and (d) is H4. They are textures of H. The predictions of this set are rather lower than the others. To understand why, we put them together to figure out the cause. Poor-synthesized parts occupy the most region of H1 and well-synthesized parts occupy the most region of H2 so that makes our model to predict subjects' behavior worse than H3 and H4.

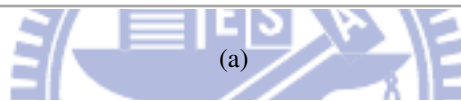
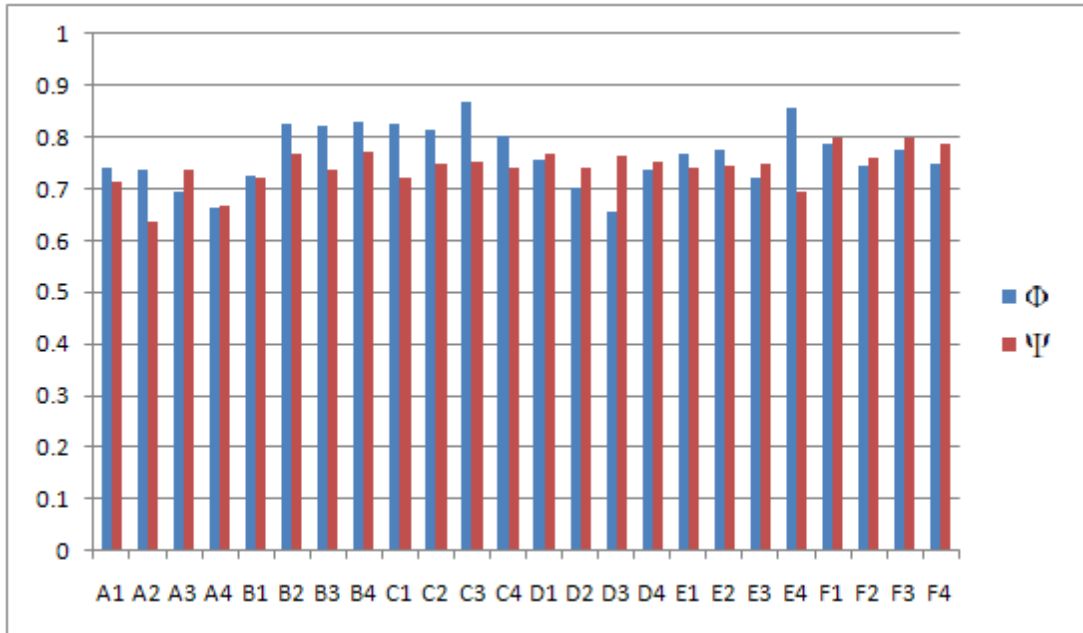


(a)

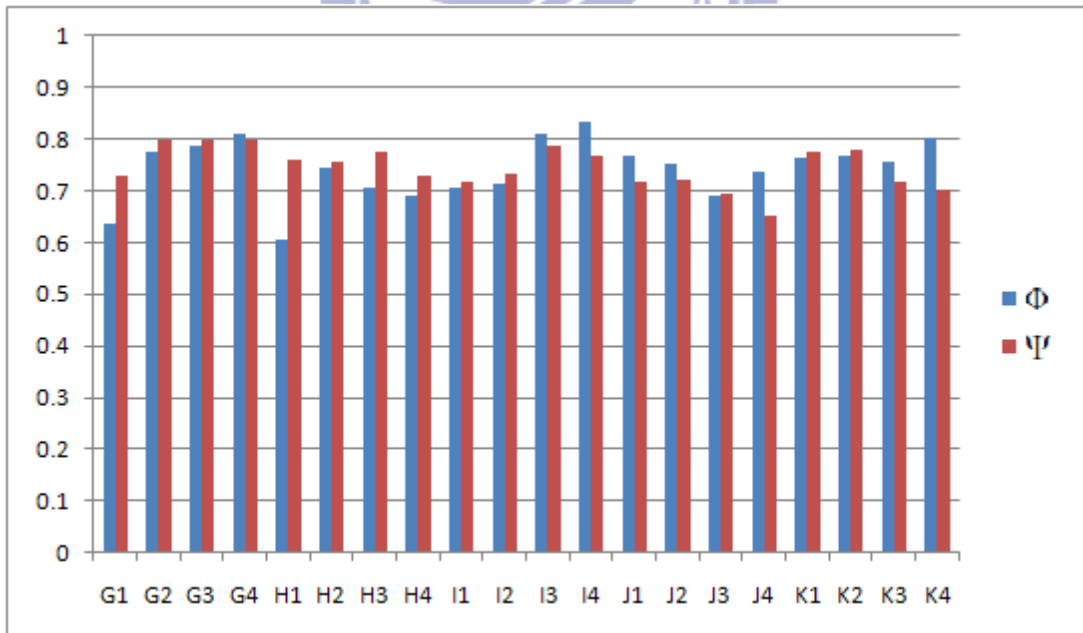


(b)

Figure 5.8: Result of group 1. Here group 2 is training pattern and group 1 is testing pattern.



(a)



(b)

Figure 5.9: Result of group 2. Here group 1 is training pattern and group 2 is testing pattern.

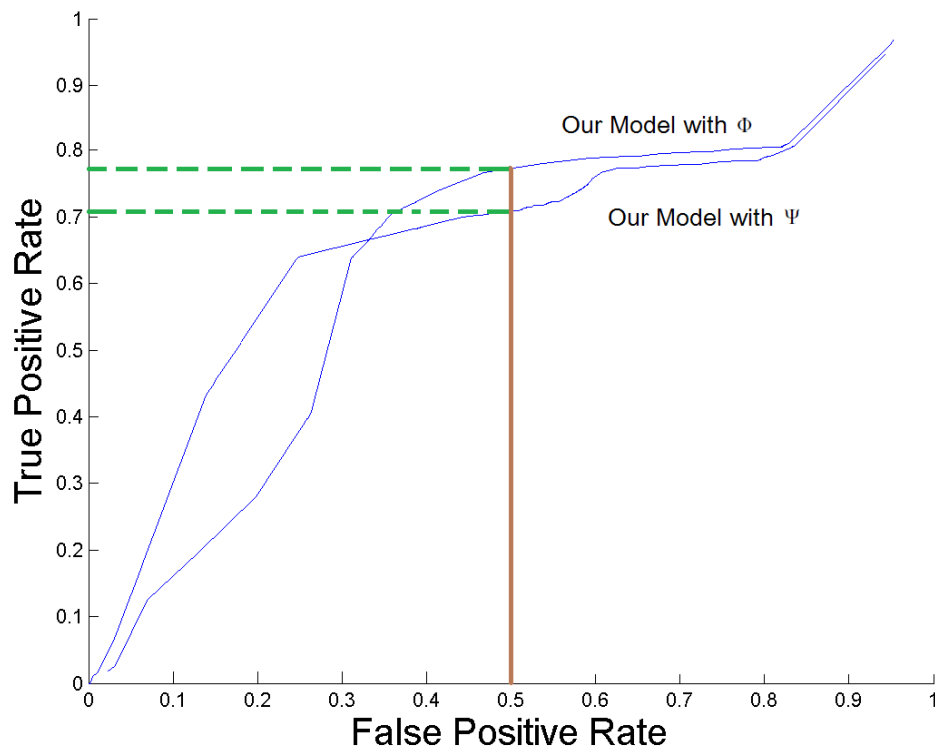


Figure 5.10: ROC of new-subjects-prediction. We merge the results of group1 and group2 to draw the ROC curve. While we increase the threshold of ROC, Ψ has better result than Φ at the beginning. After the threshold exceeds around 0.35, curve of Φ hugs the top left faster than Ψ . We claim that Φ is a more general feature for predicting human behavior.

eyetracking-data. The predicted fixation map is the fixation map predicted by virtual attention model, while the random fixation map and the uniform fixation map are almost irrelevant to eye-tracking data.

We round off the predicted rating and compare this result with the ground truth data shown as Table 5.1 and 5.2. Here we use 2-fold to train our model and test the result. Each set of the textures are picked two as the training set and the other two considered as testing sets, and vice versa.

Shown as Table 5.3, the results derived from four fixation maps prove the fact that fixation points affects the rating. The user-fixation-model uses subjects' fixation data to train perceptual rating model, and as a result, the precision of this model is the highest to the user score. The input fixation map of preict-fixation-model is the predicted fixation map of visual attention model. This result is slightly worse than user-fixation-model. So far, one may see that the more accurate visual saliency is, the more precise to predict the rating. The last two models are random-fixation-model and uniform-fixation-model which we adopt as a fixation free model. The results of random-fixation-model and uniform-fixation-model are the worst that proves the fixation data affecting the rating to us. Furthermore, from the view of P-value, both random-fixation-model and uniform-fixation-model are less than the significance level (0.05), the results are said to be statistically observed significant.

5.5 Rule Analysis

The value of membership threshold plays an important role in our model. Smaller membership threshold produces more rules and larger threshold leads to fewer rules. Too many rules or too few rules makes the training result overtraining or non-convergent. In this section, we discuss how to explain the rules of SONFIN in our model. SONFIN model generates rules automatically

during the training procedure and the number of rule is around ten to twenty. The distribution of rules is able to represent the training patterns and simulate the testing patterns. In the following paragraph, we will discuss how the rules of SONFIN affect the results. We observed, in the most cases, that the rules could be divided into two different situations, poor-synthesized and well-synthesized.

5.5.1 Poor-Synthesized Texture

Shown as Figure 5.11, we take B4 as an example of poor-synthesized textures. Figure 5.11 (c) and (d) show the rules and the distribution of testing patterns. By importing the testing patterns in Fixation-Predict-Model, we may find the coefficients for each rules. These coefficients let us define the significance of these rules. The numbers for the ellipses in Figure 5.11 (c) and (d) indicate the significance. Notice that B4 contains not only the poorly-synthesized parts but also the well-synthesized parts. This characteristic leads the subjects to browse between the two parts and compare their differences. In (d), through the distribution of rules, we can confirm that the rules definitely simulate the fixation data.

5.5.2 Well-Synthesized Texture

Shown as Figure 5.12, we take B2 as an example of well-synthesized textures. In this case, gradients of the structure error are almost the same, moreover they are all low. So it can not efficiently predict fixation points only using gradient information. Position becomes an additional information to memorize human habits. In our assumption, we think that human gets used to focus on the center while seeing a well-synthesized texture. Shown as Figure 5.12 (b), the most important rules spread at the center of a texture. We may declare that the rules still take a quite important place in position even if gradients don't work well.

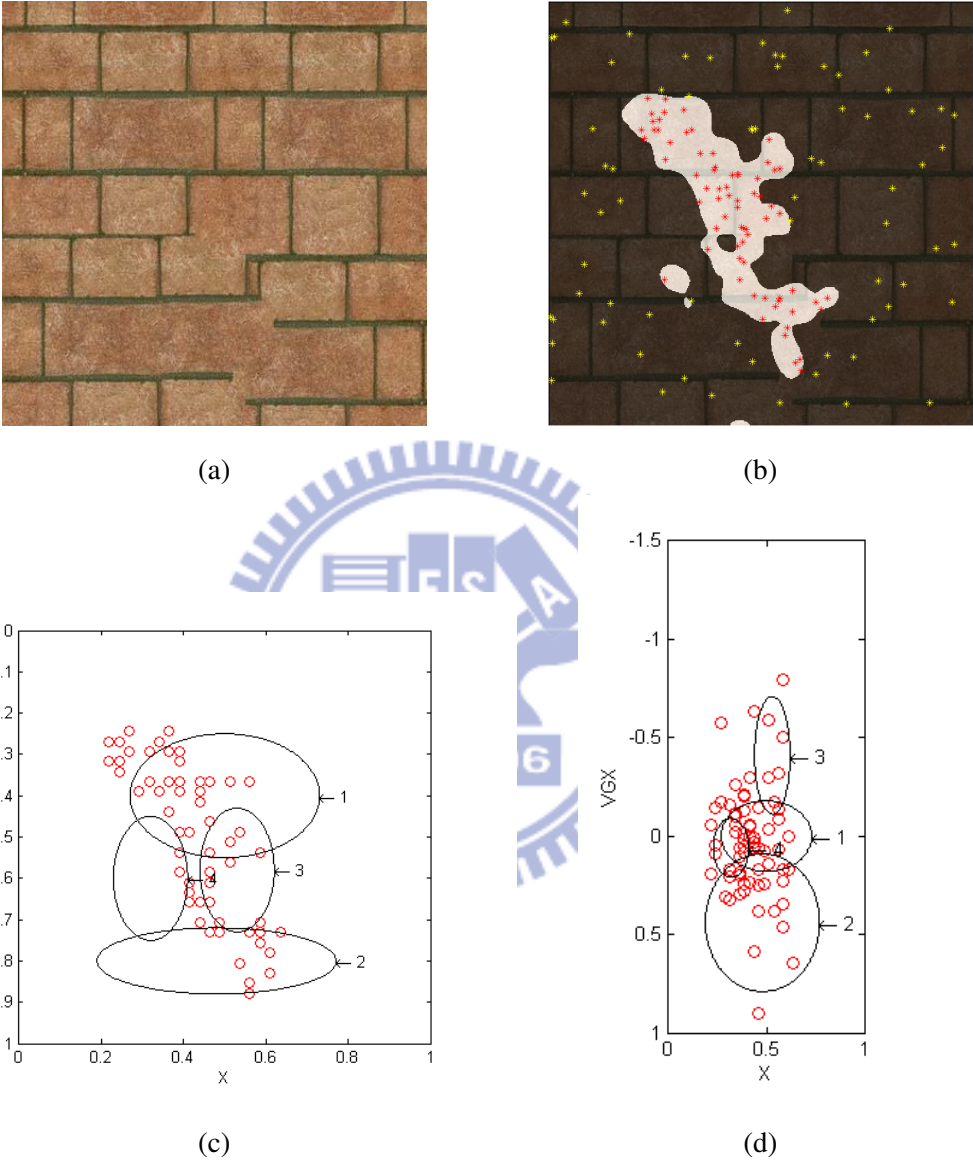


Figure 5.11: (a) is a synthesized texture B4. (b) is the fixation data blending with B4. (c) contains rules and position information of testing patterns. Each ellipse stands for a rule. The numbers for the ellipses indicate the significance. (d) shows the rules and the gradient of structure error of B4.

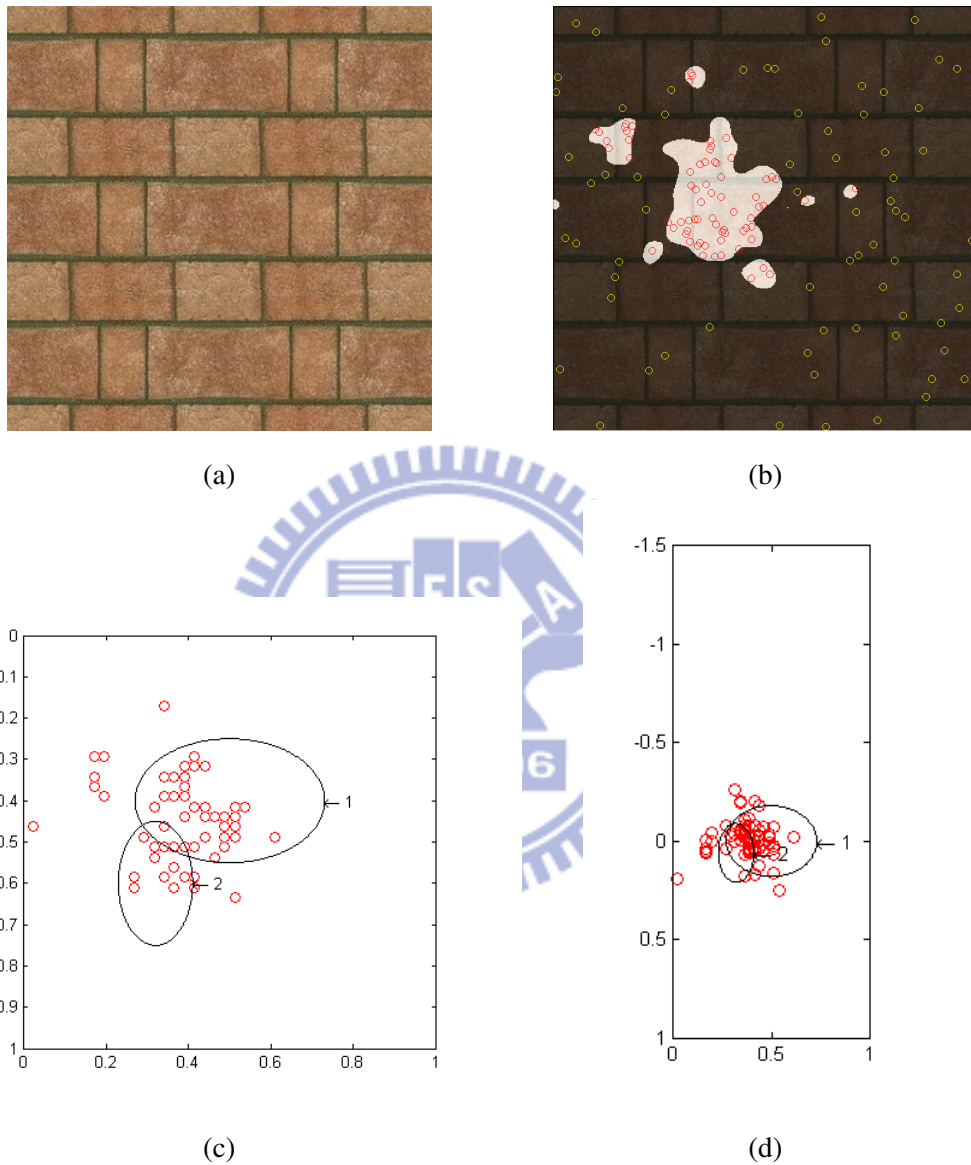


Figure 5.12: (a) is a synthesized texture B2. (b) is the fixation data blending with B2. (c) contains rules and position information of testing patterns. Each ellipse stands for a rule. The numbers for the ellipses indicate the significance. (d) shows the rules and the gradient of structure error of B2.

Table 5.1: The results from A to F. Results which are derived from user-fixation-model, predict-fixation-model and random-fixation-model are compared with user score, the ground truth data.

Texture	User Score	User Fixation Model	Predict Fixation Model	Random Fixation Model	Uniform Fixation Model
A1	1	1	1	2	2
A2	2	1	1	2	2
A3	1	1	1	2	2
A4	4	4	4	3	4
B1	2	2	1	2	1
B2	5	5	5	4	4
B3	1	3	2	2	1
B4	3	4	2	3	2
C1	2	2	2	2	3
C2	3	3	3	2	3
C3	4	4	4	2	3
C4	2	2	2	2	2
D1	2	2	1	2	2
D2	4	4	3	3	3
D3	3	4	2	2	3
D4	2	1	2	2	2
E1	2	2	1	2	1
E2	3	3	3	3	2
E3	2	2	1	2	1
E4	4	5	5	3	3
F1	2	3	2	2	1
F2	3	4	4	2	2
F3	2	1	2	2	2
F4	3	3	2	3	2

Table 5.2: The results from G to K. Results which are derived from user-fixation-model, predict-fixation-model and random-fixation-model are compared with user score, the ground truth data.

Texture	User Score	User Fixation Model	Predict Fixation Model	Random Fixation Model	Uniform Fixation Model
G1	4	4	3	3	2
G2	2	2	2	2	2
G3	4	4	4	3	3
G4	5	5	5	3	3
H1	1	1	1	2	1
H2	5	5	5	3	4
H3	3	3	2	3	2
H4	2	2	2	2	2
I1	4	3	4	3	2
I2	1	1	1	2	2
I3	3	3	3	2	2
I4	3	3	2	2	3
J1	1	1	1	2	2
J2	4	3	4	3	3
J3	5	4	4	3	4
J4	3	3	3	3	2
K1	4	4	4	3	3
K2	3	3	3	2	3
K3	1	1	2	2	2
K4	2	2	2	2	2

Table 5.3: The precision of user-fixation-model is the highest to the user score. The result of predict-fixation-model is slightly worse than user-fixation-model. The result of random-fixation-model is the worst. This table tells us that fixation points would affect human behavior on rating.

	User Score	User Fixation Model	Predict Fixation Model	Random Fixation Model	Uniform Fixation Model
Mean	2.77	2.79	2.56	2.41	2.32
Standard Deviation	1.22	1.27	1.28	0.54	0.83
Average Error	-	0.30	0.38	0.61	0.73
Chi Square	-	3.14	3.98	6.95	7.62
T-test	-	3.85	5.20	7.08	8.24
P-value	-	0.47	0.22	0.03	0.02

Conclusion and Future Work

The main contribution of this work is that we present a proper feature and a computational model to predict human fixation and rating on structural textures. Our approach can generate features that genetically reflect the textures characteristics and the model can find the association between human eye-tracking data and their textures characteristics. Previous work rarely addressed evaluating the quality of structural textures and modeling the human evaluating process. In our study, we achieve the goal to predict human fixation and rating. Once our model learned the association, we can use it to evaluate new synthesized structural textures.

Second, we propose a credible evaluation approach. Since there is no agreement on evaluating synthetic textures, for different texture synthesis algorithms, they may have their own beneficial non-perceptual evaluation methods. For example, those of near regular texture synthesis having rather low structure error than the others means to be better results with respect to the structure error; however, from the view of co-occurrence error, they are not likely to be the best. Our model overcomes such problem that we adopt perceptual and non-perceptual features to predict the rating. This truly reflects the quality of textures no matter what texture synthesis algorithm it is.


According to Elhelw et al. who used Markov transition matrix to investigate the underlying viewing strategy of the subjects in[20]. They labeled different features in the stimuli as the states for Markov model, and try to find subjects viewing transitions from one feature to the others. [19] designed an experiment to analyze scanpaths of subjects while they view well-organized and poorer interfaces. Moreover, they analyzed human scanpath and found that the transition of fixations reflecting the quality of interfaces. This result inspired us that there are some relationship between rating and transitions. As a future work, we might also label our features from textures as regularity-preserved and regularity-violated, and find the relation between the transition of fixations and user's rating.



CHAPTER A

Appendix

A.1 Parameters and Rules



The SONFIN model generates rules automatically in the process of training. The value of membership threshold plays an important role in our experiments. Smaller membership threshold would cause more rules and larger threshold leads to fewer rules. Too many rules or too few rules makes the training result overtraining or non-convergent. The membership threshold is set as 0.003 and the learning rate is set as 0.004. To avoid overtraining, the minimal error is set to 0.01 and the maximal number of iteration is 5000. Here we list the number of rules for all models mentioned in this thesis, shown as Table A.1

A.2 NSS Table

Shown as Table A.2 and A.3, we list all average NSS value in these two tables. The table of NSS corresponds to the histogram, the result of visual attention model, in chapter 5.

Table A.1: The number of rules for each model. For the visual attention model, there are two prediction models, new-texture-prediction and new-subject-prediction, and then we adopt 2-fold to train the model. For the perceptual rating model, 2-fold is also used as our validation strategy. Thus, there are totally 6 models.

Visual Attention Model 1896				Perceptual Rating Model	
New-Texture-Prediction		New-Subject-Prediction		-	
16	24	28	20	34	31

Table A.2: The results of NSS from texture A to F.

Texture	Saliency Map	Modified Saliency Map	MLP Model	Our Model with Φ	Our Model with Ψ
A1	51.25%	62.91%	72.86%	85.18%	85.88%
A2	57.56%	63.41%	74.88%	82.79%	86.21%
A3	51.10%	61.15%	74.03%	72.68%	79.84%
A4	40.00%	51.63%	71.65%	80.47%	82.35%
B1	68.83%	71.20%	77.22%	85.35%	85.37%
B2	54.65%	73.00%	80.12%	80.21%	84.82%
B3	42.36%	72.80%	80.04%	84.98%	84.87%
B4	61.98%	74.24%	79.34%	83.16%	84.88%
C1	50.50%	65.10%	77.14%	88.69%	86.85%
C2	59.65%	68.92%	76.78%	84.68%	85.21%
C3	59.80%	63.00%	75.26%	82.69%	83.58%
C4	56.24%	69.00%	74.95%	81.55%	84.23%
D1	47.86%	28.39%	87.36%	87.21%	87.20%
D2	67.42%	31.33%	78.67%	80.19%	84.22%
D3	71.66%	42.00%	75.64%	89.57%	88.35%
D4	39.23%	21.00%	80.77%	83.26%	84.32%
E1	53.24%	51.84%	76.11%	84.20%	84.59%
E2	53.12%	27.88%	77.85%	79.58%	80.48%
E3	48.44%	35.00%	79.67%	83.42%	84.99%
E4	77.03%	49.00%	69.65%	86.98%	86.20%
F1	68.32%	69.58%	76.30%	87.12%	83.21%
F2	45.67%	69.58%	83.25%	80.43%	80.21%
F3	55.22%	68.00%	77.62%	86.48%	87.10%
F4	28.68%	46.00%	58.57%	80.18%	81.11%

Table A.3: The results of NSS from texture G to K.

Texture	Saliency Map	Modified Saliency Map	MLP Model	Our Model with Φ	Our Model with Ψ
G1	58.36%	36.44%	80.68%	78.79%	80.22%
G2	44.56%	31.26%	81.33%	76.01%	78.51%
G3	66.98%	82.00%	74.65%	88.91%	87.36%
G4	60.03%	83.00%	67.21%	84.57%	85.99%
H1	51.01%	25.40%	76.96%	70.13%	76.22%
H2	56.68%	26.46%	82.02%	71.55%	77.85%
H3	56.51%	46.00%	77.33%	84.89%	85.22%
H4	32.35%	12.00%	70.10%	75.16%	79.25%
I1	65.65%	57.12%	63.34%	60.34%	75.22%
I2	53.98%	53.60%	48.31%	57.61%	73.22%
I3	67.00%	58.00%	49.00%	70.00%	80.00%
I4	71.18%	57.00%	66.84%	81.52%	82.36%
J1	68.19%	91.47%	65.51%	85.01%	82.29%
J2	71.63%	87.06%	48.31%	86.45%	81.55%
J3	52.31%	86.00%	83.64%	65.12%	80.69%
J4	21.00%	11.00%	82.00%	73.00%	81.00%
K1	63.77%	80.29%	78.58%	82.12%	80.58%
K2	55.15%	88.32%	77.67%	85.11%	83.25%
K3	61.98%	85.00%	78.67%	76.16%	77.89%
K4	60.78%	82.00%	75.37%	87.20%	85.25%
Average	56.00%	57.00%	74.95%	80.48%	82.73%

Bibliography

- [1] W. Lin, J. Hays, C. Wu, Y. Liu, and V. Kwatra. Quantitative evaluation of near regular texture synthesis algorithms. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 427–434, 2006.
- [2] P. Benard, J. Thollot, and F. Sillion. Quality assessment of fractalized npr textures: a perceptual objective metric. In *APGV '09*, October 2009.
- [3] J. M. Henderson and A. Hollingworth. High-level scene perception. In *Annual Review of Psychology*, volume 50, pages 243–271, 1999.
- [4] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. In *IEEE Trans. Pattern Anal. Mach. Intell.*, volume 20, pages 1254–1259, 1998.
- [5] C. Juang and C. Lin. An on-line self-constructing neural fuzzy inference network and its applications. In *IEEE Transactions on Fuzzy Systems*, volume 6, pages 12–32, February 1998.
- [6] C. Mello-Thoms, C. F. Nodine, and H. L. Kundel. What attracts the eye to the location of missed and reported breast cancers? In *ETRA '02: Proceedings of the 2002 symposium on Eye tracking research & applications*, pages 111–117, 2002.
- [7] R. J. Peters, A. Iyer, L. Itti, and C. Koch. Components of bottom-up gaze allocation in natural images. In *Vision Research*, volume 45, pages 2397–2416, August 2005.

- [8] M. Sharma and S. Singh. Evaluation of texture methods for image analysis. In *Proceedings of the 7th Australian and New Zealand Intelligent Information Systems Conference*, pages 117–121. ARCME, 2000.
- [9] R. J. Peters and L. Itti. Beyond bottom-up: Incorporating task-dependent influences into a computational model of spatial attention. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2007.
- [10] M. Brecht and J. Saiki. A neural network implementation of a saliency map model. In 19, editor, *Neural Networks*, pages 1467–1474, 10 2006.
- [11] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *IEEE 12th International Conference on Computer Vision*, pages 2106–2113, September 2009.
- [12] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [13] H. Yu. Visual attention modeling on structural textures. In *Master Thesis in Institute of Computer Science and Engineering, College of Computer Science, National Chiao Tung University*, 2010.
- [14] V. Kwatra, A. Schodl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: Image and video synthesis using graph cuts. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 277–286, 2003.
- [15] A. Efros and W. Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346, 2001.
- [16] Y. Liu, W. Lin, and J. Hays. Near-regular texture analysis and manipulation. In *ACM Transactions on Graphics*, volume 23, August 2004.

- [17] L. Liang, C. Liu, Y. Xu, B. Guo, and H. Shum. Real-time texture synthesis by patch-based sampling. In *ACM Transactions on Graphics*, volume 20, pages 127–150, 2001.
- [18] Y. Liu, Y. Tsin, and W. Lin. The promise and perils of near-regular texture. In *International Journal of Computer Vision*, volume 62(1-2), pages 145–159, 2005.
- [19] X. P. Kotval J. H. Goldberg. Computer interface evaluation using eye movements: methods and constructs. In *International Journal of Industrial Ergonomics*, volume 24, pages 631–645, October 1999.
- [20] M. Elhelw, M. Nicolaou, A. Chung, G. Yang, and M. S. Atkins. A gaze-based study for investigating the perception of visual realism in simulated scenes. In *ACM Trans. Appl. Percept.*, volume 5, pages 1–20, 2008.

