

REAL-TIME ESTIMATION OF HUMAN VISUAL ATTENTION WITH DYNAMIC BAYESIAN NETWORK AND MCMC-BASED PARTICLE FILTER

Kouji Miyazato*, Akisato Kimura[†], Shigeru Takagi* and Junji Yamato[†]

[†] NTT Communication Science Laboratories, NTT Corporation, Japan
* Department of Information and Communication Systems Engineering,
Okinawa National College of Technology, Japan

ABSTRACT

Recent studies in signal detection theory suggest that the human responses to the stimuli on a visual display are non-deterministic. People may attend to different locations on the same visual input at the same time. Constructing a stochastic model of human visual attention would be promising to tackle the above problem. This paper proposes a new method to achieve a quick and precise estimation of human visual attention based on our previous stochastic model with a dynamic Bayesian network. A particle filter with Markov chain Monte-Carlo (MCMC) sampling make it possible to achieve a quick and precise estimation through stream processing. Experimental results indicate that the proposed method can estimate human visual attention in real time and more precisely than previous methods.

Index Terms— Saliency-based human visual attention, dynamic Bayesian network, stream processing, Markov chain Monte-Carlo (MCMC), particle filter.

1. INTRODUCTION

Developing a sophisticated object detection and recognition algorithms has been a long distance challenge in computer and robot vision researches. Such algorithms are required in most applications of computational vision, including biometrics, medical imaging, intelligent cars, factory automation, and content-based image retrieval. One of the major challenges in designing object recognition systems is to construct methods that are fast and capable of operating on standard computer platforms. To that end, pre-selection mechanism would be essential to enable subsequent processing to focus only on relevant data.

One promising approach to achieve this mechanism is visual attention: it selects regions in a visual scene that are most likely to contain objects of interest. The field of visual attention is currently the focus of much research for both biological and artificial systems. The first biologically-plausible model for explaining the human attention system was proposed by

Contact address: akisato@ieee.org

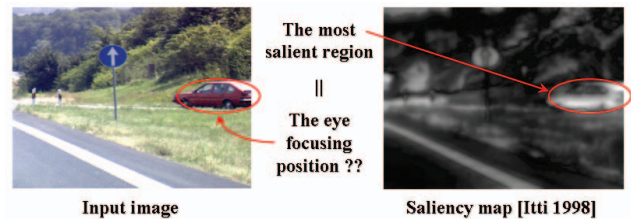


Fig. 1. Example of saliency map through the Koch-Ullman model

Koch and Ullman [1]. This model has been attracting attention of many researchers, especially after the development of an implementation model by Itti, Koch and Niebur [2]. Later, many attempts have been made to improve the Koch-Ullman model [3, 4, 5, 6] and to extend it to video signals [6, 7, 8, 9].

However, all the above methods includes one crucial problem: people may attend to different locations on the same visual input at the same time. Our research group proposed a new stochastic model of visual attention [10, 11] that tackled the problem originated from the deterministic property of the previous attention models). When describing the Bayesian network of visual attention, the principle of the signal detection theory is introduced, namely, the position where the value of the stochastic saliency takes the maximum is the eye focusing positions. The proposed model incorporates another property that eye movements may be affected also by a cognitive state of humans. The introduction of eye movement patterns as hidden states of HMMs enables us to describe the mechanism of eye focusing and eye movement naturally.

Although the method based on the stochastic model well simulated the human visual system, it requires many procedures with high computational costs (about 1 second per frame with a standard workstation, cf. Fig. 5 in Section 7). When considering a human visual attention model as a pre-selection mechanism for subsequent processing to focus only on relevant data, computational cost should be of crucial significance in terms of practical use.

In recent years, there has been strong interest from researchers and developers in exploiting the power of commodity hardware including multiple processor cores for parallel computing. This is because 1) multi-core CPUs and stream processors such as graphics processing units (GPUs) and Cell processors [12] are currently the most powerful and economical computational hardware available, 2) the rise of SDKs and APIs such as OpenMP [13] and NVIDIA CUDA [14] makes it easy to implement desired algorithms for execution on multi-core hardware. This programming paradigm is widely known as *stream processing*. By introducing the idea of stream processing, we expect the model to reduce the execution time greatly. However, stream processing is not versatile for accelerating any kinds of signal processing: Stream processing is only feasible for computations that utilize simple data repeatedly and can compute each sub-process with almost the same calculation cost. The previous stochastic model included several procedures that did not fit the above property.

To this end, we propose a new model and its implementation strategy for estimating human visual attention feasible for multi-core processors through stream processing. MAP estimation through a particle filter with Markov chain Monte-Carlo (MCMC) sampling make it possible to incorporate stream processing into our stochastic attention model. Sampling-based simulation is excellent with stream processing, and the computational cost for each particle does not depend on the characteristics of densities to be estimated. Therefore, the new method can be easily implemented on multi-core hardware without any special artifices.

2. RELATED WORK

Several previous researches focused on modeling of human visual attention by using some kind of probabilistic techniques or concepts. Itti and Baldi [7] investigated a Bayesian approach for detecting surprising events in video signals. Their approach models a surprise by Kullback-Leibler divergence between the prior and posterior distributions of fundamental features. Koike and Saiki [15] introduced a stochastic winner-take-all (WTA) mechanism into the Koch-Ullman model. Avraham and Lindenbaum [16] utilized a graphical model approximation to extend their static saliency model based on self similarities. Boccignone [17] introduced a non-parametric Bayesian framework to achieve object-based visual attention. Gao and Vasconcelos [6] developed a decision-theoretic approach attention model for object detection.

The main contribution of our stochastic model against the previous researches is the introduction of a unified stochastic model that integrates “covert shifts of attention” (shifts of attentions without saccadic eye movements) driven by bottom-up saliency with “overt shifts of attention” (shifts of attention with saccadic eye movements) driven by eye movement patterns by using a dynamic Bayesian network. Our proposed model also provides a framework that simulates and

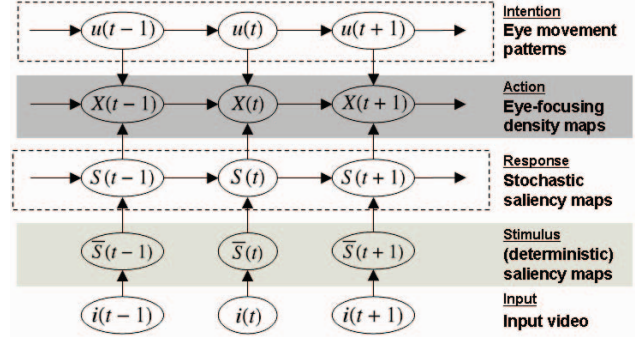


Fig. 2. Graphical representation of the proposed stochastic model of human visual attention, where arrows stands for stochastic dependencies.

combines the bottom-up visual saliency response and the top-down cognitive state of a person to estimate probable attended regions, if eye movement patterns can deal with more sophisticated top-down information. How to integrate such kinds of top-down information is one of the most important future researches.

3. MODEL OVERVIEW

Fig. 2 illustrates the graphical representation of the proposed visual attention model. The proposed model consists of four layers: (deterministic) saliency maps, stochastic saliency maps, eye focusing positions and eye movement patterns. Before describing the model of the proposed visual attention model, let us introduce several notations and definitions.

$I = i(1 : T) = \{i(t)\}_{t=1}^T$ denotes an input video, where $i(t)$ is the t -th frame of the video I and T is the duration (i.e. the total number of frames) of the video I . The symbol I also denotes a set of coordinates in the frame. For example, a position \mathbf{y} in a frame is represented as $\mathbf{y} \in I$.

$\bar{S} = \bar{S}(1 : T) = \{\bar{S}(t)\}_{t=1}^T$ denotes a *saliency video* which comprises a sequence of *saliency maps* $\bar{S}(t)$ obtained from the input video I . Each saliency map is denoted as $\bar{S}(t) = \{\bar{s}(t, \mathbf{y})\}_{\mathbf{y} \in I}$, where $\bar{s}(t, \mathbf{y})$ is called *saliency* which is the pixel value at the position $\mathbf{y} \in I$. Each saliency represents the strength of visual stimulus on a position of a frame with the value between 0 and 1.

$S = S(1 : T) = \{S(t)\}_{t=1}^T$ denotes a *stochastic saliency video* which comprises a sequence of *stochastic saliency maps* $S(t)$ obtained from the input video I . Each stochastic saliency map is denoted as $S(t) = \{s(t, \mathbf{y})\}_{\mathbf{y} \in I}$, where $s(t, \mathbf{y})$ is called *stochastic saliency* which is the pixel value at the position $\mathbf{y} \in I$. Each stochastic saliency corresponds to saliency response perceived through a random processes.

$U = u(1 : T) = \{u(t)\}_{t=1}^T$ denotes a sequence of *eye movement patterns* each of which represents a pattern of eye

movements. Eye movement patterns reflect purposes or intentions of human eye movements.

$X = X(1 : T) = \{\mathbf{x}(t)\}_{t=1}^T$ denotes a sequence of eye focusing positions. The proposed model estimates the eye focusing position by integrating the bottom-up information (stochastic saliency maps) and the top-down information (eye movement patterns). A map that represents a density of eye focusing positions is called an *eye focusing density map*.

In what follows, we denote a probability density function (PDF) of an argument x as $p(x)$, a conditional PDF of an argument x given y as $p(x|y)$, and a PDF of x with a parameter θ as $p(x; \theta)$.

4. SALIENCY MAPS

Consider an input video $I = i(1 : T) = \{i(t)\}_{t=1}^T$ of duration T , where $i(t)$ is the t -th frame of the video I . Then, a sequence $\bar{S} = \bar{S}(1 : T) = \{\bar{S}(t)\}_{t=1}^T$ of *saliency maps* $\bar{S}(t)$ is obtained from the video I . We used a standard model proposed by Itti et al. [2] to extract the saliency maps. Our implementation includes 9 fundamental features sensitive to luminance, color opponents (red/green and blue/yellow), orientations (0° , 45° , 90° and 135°), and oriented motion energies (horizontal and vertical). Each pixel value $\bar{s}(t, \mathbf{y})$ of the saliency map $\bar{S}(t)$ at a position \mathbf{y} is called the *saliency*. We have to note that I also represents a set of coordinates in the frame. Each saliency represents the strength of the visual stimulus at a given position.

We employ several minor modifications to accelerate the original implementation. However, we have yet to introduce any GPU implementations for extracting saliency maps. Several previous researches have focused on GPU implementation for extracting saliency maps [18, 19]. We expect to extract saliency maps within a few milliseconds by incorporating such algorithms in our method in the near future.

5. STOCHASTIC SALIENCY MAPS

When estimating a *stochastic saliency map* $S(t) = \{s(t, \mathbf{y})\}_{\mathbf{y} \in I}$, we introduce a pixel-wise state space model characterized by the following two relationships:

$$\begin{aligned} p(s(t, \mathbf{y})|s(t-1, \mathbf{y})) &= \mathcal{G}(s(t, \mathbf{y}); s(t-1, \mathbf{y}), \sigma_{s1}), \\ p(\bar{s}(t, \mathbf{y})|s(t, \mathbf{y})) &= \mathcal{G}(\bar{s}(t, \mathbf{y}); s(t, \mathbf{y}), \sigma_{s2}), \end{aligned}$$

where σ_{si} ($i = 1, 2$) is a model parameter obtained beforehand by using the EM algorithm [10] and $\mathcal{G}(s; \bar{s}, \sigma)$ is the Gaussian probability density function (PDF) with argument s , mean \bar{s} , and variance σ^2 .

$$\mathcal{G}(s; \bar{s}, \sigma) \stackrel{\text{def.}}{=} \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(s - \bar{s})^2}{2\sigma^2}\right\}.$$

The first relationship exploits the temporal characteristic of the human visual system, and the second relationship implies

that a saliency map is observed through a Gaussian density. For brevity, only in this section we will omit the position \mathbf{y} where explicit expression is unnecessary, e.g. $s(t)$ instead of $s(t, \mathbf{y})$.

We employ a Kalman filter to recursively compute the stochastic saliency map. Assume that the density at each position on the stochastic saliency map $s(t-1)$ at time $t-1$ given saliency maps $\bar{s}(1 : t-1)$ up to time $t-1$ is given as the following Gaussian PDF:

$$\begin{aligned} p(s(t-1)|\bar{s}(1 : t-1)) \\ = \mathcal{G}(s(t-1); \hat{s}(t-1|t-1), \sigma_s(t-1|t-1)). \end{aligned}$$

Then, the PDF at each position on the stochastic saliency map $s(t)$ at time t given saliency maps $\bar{s}(1 : t)$ up to time t is obtained as follows:

[Estimation step]

$$\begin{aligned} p(s(t)|\bar{s}(1 : t-1)) &= \mathcal{G}(s(t); \hat{s}(t|t-1), \sigma_s(t|t-1)), \\ \hat{s}(t|t-1) &= \hat{s}(t-1|t-1), \\ \sigma_s^2(t|t-1) &= \sigma_{s1}^2 + \sigma_s^2(t-1|t-1). \end{aligned}$$

[Update step]

$$\begin{aligned} p(s(t)|\bar{s}(1 : t)) &= \mathcal{G}(s(t); \hat{s}(t|t), \sigma_s(t|t)), \\ \hat{s}(t|t) &= \frac{\sigma_s^2(t|t)}{\sigma_s^2(t|t-1)} \hat{s}(t|t-1) + \frac{\sigma_s^2(t|t)}{\sigma_{s2}^2} \bar{s}(t), \\ \sigma_s^2(t|t) &= \frac{\sigma_{s2}^2 \cdot \sigma_s^2(t|t-1)}{\sigma_{s2}^2 + \sigma_s^2(t|t-1)}, \end{aligned}$$

Since both the estimation and update steps can be executed for each pixel independently, they can be directly implemented through stream processing without any special artifices. However, they can execute within a few msec/frame without stream processing. Therefore, we keep the procedure as a CPU calculation.

6. ESTIMATING VISUAL ATTENTION

6.1. Overview

This section describes the main contribution of this report, namely how to estimate eye focusing position $\mathbf{x}(t)$ by integrating stochastic saliency map $S(t)$ and *eye movement pattern* $u(t)$. First, we introduce following transition PDF to estimate the eye focusing position:

$$\begin{aligned} p(\mathbf{x}(t), u(t)|p(S(t)), \mathbf{x}(t-1), u(t-1)) \\ \propto p(\mathbf{x}(t)|p(S(t))) \\ \cdot p(u(t)|u(t-1)) \cdot p(\mathbf{x}(t)|\mathbf{x}(t-1), u(t)), \quad (1) \end{aligned}$$

where \propto stands for the proportional indicator, and the PDF of the stochastic saliency map at time t is represented as $p(S(t))$ for simplicity, namely

$$p(S(t)) = \{p(s(t, \mathbf{y}))\}_{\mathbf{y} \in I},$$

$$p(s(t, \mathbf{y})) = p(s(t, \mathbf{y}) | \bar{s}(1:t, \mathbf{y})) \quad \mathbf{y} \in I.$$

The stochastic saliency map $S(t)$ controls ‘‘covert shifts of attention’’ (shifts without saccadic eye movements) through the PDF $p(\mathbf{x}(t) | p(S(t)))$ ¹. On the other hand, the eye movement pattern $u(t)$ controls the degree of ‘‘overt shifts of attention’’ (shifts of attention with saccadic eye movements). We assume two types of eye movement patterns: 1) the passive state $u(t) = 0$ in which eyes tend to remain near one particular position, and 2) the active state $u(t) = 1$ in which eyes move around actively. In what follows, we call a pair $z(t) = (\mathbf{x}(t), u(t))$, consisting of an eye focusing position and an eye movement pattern, as the *eye focusing state* $z(t)$ for brevity. The following density of eye focusing positions $\mathbf{x}(t)$ given a PDF $p(S(1:t))$ of stochastic saliency maps up to time t characterizes an *eye focusing density map* at time t :

$$p(\mathbf{x}(t) | p(S(1:t))) = \sum_{u(t)=0,1} p(z(t) | p(S(1:t))), \quad (2)$$

$$p(z(t) | p(S(1:t))) = \int_{z(t-1)} p(z(t-1) | p(S(1:t-1))) \cdot p(z(t) | p(S(t)), z(t-1)) dz(t-1). \quad (3)$$

Note that the second term of Eq. (3) is the same as Eq. (1).

Since the formula for computing Eq. (2) cannot be derived, we introduce a sampling-based approach instead. The density of eye focusing states shown in Eq. (3) can be approximated by samples of eye focusing states $\{z_n(t)\}_{n=1}^N$ and the associated weights $\{w_n(t)\}_{n=1}^N$ as

$$p(z(t) | p(S(1:t))) \approx \sum_{n=1}^N w_n(t) \cdot \delta(z(t) - z_n(t)), \quad (4)$$

where N is the number of samples, and $\delta(\cdot)$ represents the Dirac delta function.

The procedure for estimating eye focusing density maps can be separated into two steps: 1) Computing $p(\mathbf{x}(t) | p(S(t)))$ estimated from a stochastic saliency map, and 2) computing $p(u(t) | u(t-1))$ and $p(\mathbf{x}(t) | \mathbf{x}(t-1), u(t))$ estimated from an eye movement pattern. We now describe each step in detail.

6.2. Estimating attention from stochastic saliency maps

The first term of Eq. (1) represents the fact that the eye focusing position is selected based on the signal detection theory, where the position at which the stochastic saliency takes the maximum is determined as the eye focusing position. In other words, this term computes the probability at each position that the stochastic saliency takes the maximum.

$$\underline{p(\mathbf{x}(t) | p(S(t)))}$$

¹The notation $p(\mathbf{x}(t) | p(S(t)))$ seems to be unusual, however, the PDF of eye focusing positions $\mathbf{x}(t)$ estimated from the stochastic saliency map $S(t)$ can be determined by the PDF of the stochastic saliency map, not the stochastic saliency map itself, as shown in Section 6.2.

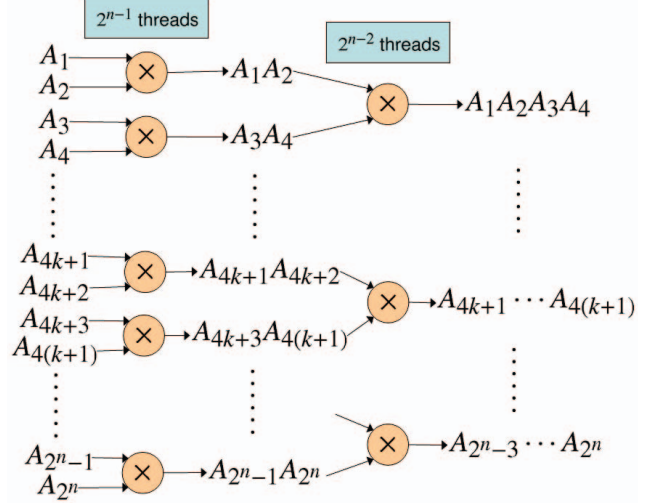


Fig. 3. Tree-based multiplication for computing the product $A_1A_2 \cdots A_{2^n}$

$$= \int_{-\infty}^{\infty} p(s(t, \mathbf{x}(t)) = s) \prod_{\tilde{\mathbf{x}} \neq \mathbf{x}(t)} P(s(t, \tilde{\mathbf{x}}) \leq s) ds, \quad (5)$$

where $P(s(t, \tilde{\mathbf{y}}) \leq s)$ is the cumulative distribution function (CDF) that corresponds to the PDF $p(s(t, \tilde{\mathbf{y}}))$.

Since direct computation of Eq. (5) is ill-suited for stream processing, we introduce an alternative expression that is applicable to stream processing.

$$p(\mathbf{x}(t) | p(S(t))) = \int_{-\infty}^{\infty} \frac{p(s(t, \mathbf{x}(t)) = s)}{P(s(t, \mathbf{x}(t)) \leq s)} \prod_{\tilde{\mathbf{x}} \in I} P(s(t, \tilde{\mathbf{x}}) \leq s) ds. \quad (6)$$

The latter part of Eq. (6) does not depend on the position $\mathbf{x}(t)$, which implies that it can be calculated in advance for every s . This calculation can be executed in $\mathcal{O}(\log |I|)$ time through a tree-based multiplication and parallelization at each level (cf. Fig. 3). Also, the former part of Eq. (6) can be calculated independently for each position $\mathbf{x}(t)$. Therefore, once the calculation of the latter part has finished, Eq. (6) can be calculated in $\mathcal{O}(\log |S|)$ time with a combination of tree-based addition and pixel-wise parallelization, where $|S|$ stands for the resolution of the integral in Eq. (6).

6.3. Integrating eye movement patterns

Fig. 4 depicts the old and new strategies for calculating eye focusing density maps. The previous strategy [10, 11] utilizes rejection sampling (See e.g. [20]), and samples are drawn directly from the whole transition PDF (Eq. (1)). However, the rejection sampling method is quite costly and ill-suited for stream processing since its computational cost for generating a sample highly depends on the nature of the PDF to be

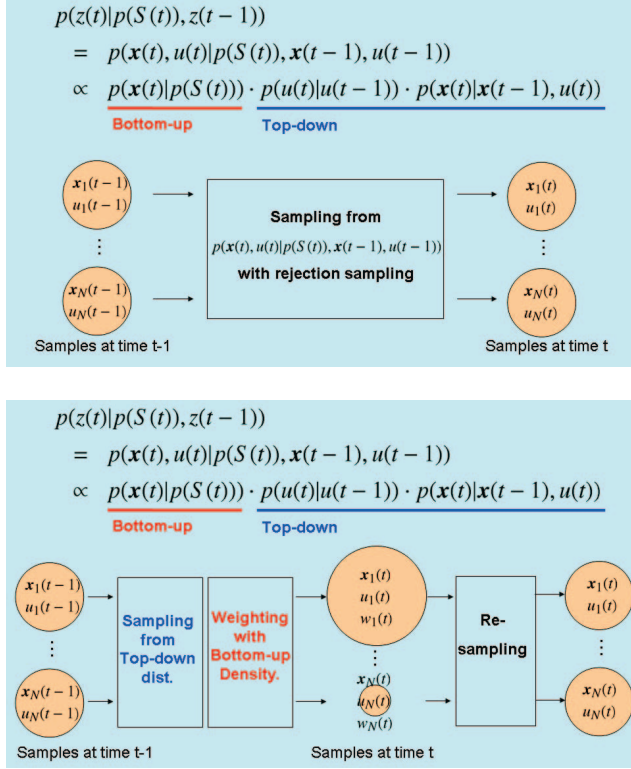


Fig. 4. Old and new strategies for calculating eye focusing density maps (Top) old strategy (Bottom) new strategy

sampled, and therefore, the slowest kernel for generating samples keeps other kernels waiting for a long time. Instead, the new strategy introduces a technique inspired by a particle filter with Markov chain Monte-Carlo (MCMC) sampling. The procedure of the new strategy is as follows:

(1) Suppose that samples $\{z_n(t-1)\}_{n=1}^N$ of eye focusing states at time $t-1$ have already been obtained. Then, samples $\{z_n(t)\}_{n=1}^N$ at time t are drawn by using the second and third terms of Eq. (1) with the Metropolis algorithm [21], a standard MCMC sampling strategy.

$$u_n(t) \sim p(u(t)|u_n(t-1)), \quad (7)$$

$$\mathbf{x}_n(t) \sim p(\mathbf{x}(t)|\mathbf{x}_n(t-1), u_n(t)). \quad (8)$$

The density of Eq. (7) is characterized by the transition probability of eye movement patterns defined by a 2×2 matrix. The density of Eq. (8) represents the transition PDF of eye focusing positions governed by the eye movement pattern at the current time, defined as

$$p(\mathbf{x}(t)|\mathbf{x}(t-1), u(t)) = \mathcal{L}(\mathbf{x}(t); \mathbf{x}(t-1), \gamma_{x,u(t)}, \sigma_{x,u(t)}),$$

where γ_{xi} and σ_{xi} ($i = 1, 2$) are model parameters, and $\mathcal{L}(\mathbf{x}; \bar{\mathbf{x}}, \gamma, \sigma)$ is a shifted 2D Gaussian PDF with argument

\mathbf{x} , mean $\bar{\mathbf{x}}$, indent γ and variance σ^2 such that

$$\mathcal{L}(\mathbf{x}; \bar{\mathbf{x}}, \gamma, \sigma) \propto \exp \left\{ -\frac{(\|\mathbf{x} - \bar{\mathbf{x}}\| - \gamma)^2}{2\sigma^2} \right\}.$$

The model parameters and the transition matrix can be obtained beforehand by using Viterbi learning [10].

(2) As the second step, sample weights are updated based on the first term $p(\mathbf{x}(t)|p(S(t)))$ of Eq. (1), which has been derived from the procedure shown in Section 6.2. Formally, the weight $w_n(t)$ of the n -th sample $z_n(t)$ at time t can be calculated as

$$w_n(t) \propto p(\mathbf{x}(t) = \mathbf{x}_n(t)|p(S(t))).$$

As shown in Eq. (4), samples $\{z_n(t)\}_{n=1}^N$ and the associated weights $\{w_n(t)\}_{n=1}^N$ comprise an eye focusing density map at time t .

(3) Finally, i is performed to eliminate samples with small weights and multiply samples with large weights. This step enables us to avoid “degeneracy” problem, namely, to avoid the situation where all but one of the weights are close to zero. Although the effective number of samples [22] is frequently used as a criterion for resampling, we execute resampling at regular time intervals.

Note that a computational cost for each step (propagation, update and resampling) does not depend on the nature of the PDF to be estimated so much. This implies that the proposed approach is quite feasible for stream processing.

7. EVALUATION

7.1. Conditions

We evaluated the performance of our new method by comparing it with the Itti model [2] and the previous stochastic model [10, 11] in terms of execution time and estimation accuracy.

For the accuracy evaluation, we used CRCNS eye-1 database² created by University of South California. This database includes 100 video clips (MPEG-1, 640 × 480 pixels, 30fps) and eye traces when showing these video clips to 8 human subjects (4-6 available eye traces for each video clip, 240fps). Please refer to the document included in the database for the detail. In this evaluation, we used 50 video clips (about 25 minutes in total) called “original experiment” and associated eye traces.

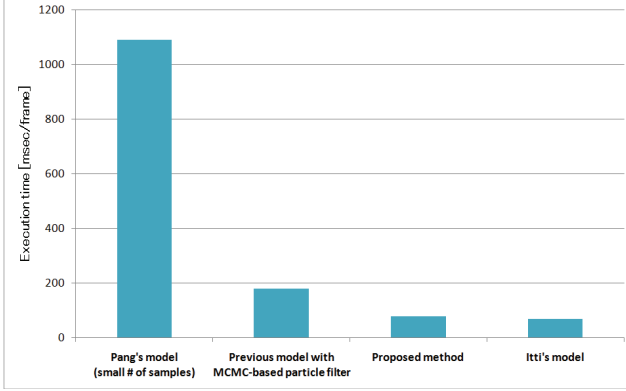
Model parameters shown in Section 5 and 6.3 were derived in advance with the learning algorithm presented in [10]. In this time, we used 5-fold cross validation so that 40 video clips and associated eye traces were used as the training data for evaluating the remaining data.

As a metric to quantify how well a model predicts the actual human eye focusing positions, we used the normalized scan-path saliency (NSS) [23]. Let $R_j(t)$ be a set of all pixels

²<http://crcns.org/data-sets/eye/eye-1>

Table 1. Platform used in the evaluation

OS	Windows Vista Ultimate
Development platform	Microsoft Visual Studio 2008 C++ OpenCV 1.1pre & NVIDIA CUDA 2.1
Optimization	none
CPU	Intel Core2 Quad Q6600 (2.40GHz)
RAM	4.0GB
GPU	NVIDIA GeForce8800 GT ×2 SLI

**Fig. 5.** Total execution time [msec/frame]

in a circular region centered on the eye focusing position of test subject j with a radius of 30 pixels. Then, the NSS value at time t is defined as

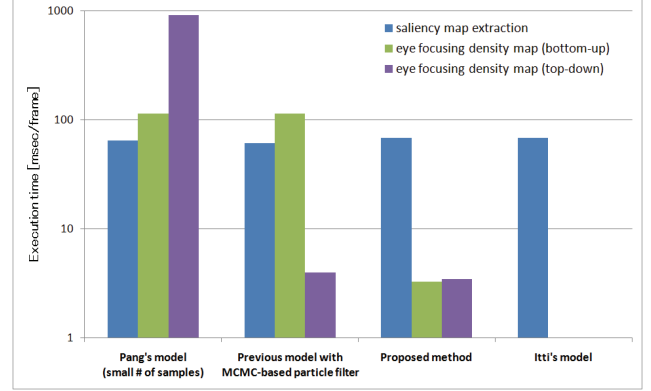
$$NSS(t) = \frac{1}{N_s} \sum_{j=1}^{N_s} \frac{1}{\sigma(p(\mathbf{x}))} \left\{ \max_{\mathbf{x}(t) \in R_j(t)} p(\mathbf{x}(t)) - \bar{p}(\mathbf{x}) \right\},$$

where N_s is the total number of subjects, $\bar{p}(\mathbf{x})$ and $\sigma(p(\mathbf{x}))$ are the mean and the variance of the pixel values of the model's output, respectively. $NSS(t) = 1$ indicates that the subjects' eye positions fall in a region whose predicted density is one standard deviation above average. Meanwhile, $NSS(t) \leq 0$ indicates that the model performs no better than picking a random position on the map.

The platform used in this evaluation is listed in Table 1.

7.2. Results

Fig. 5 shows the total execution time of 1) the previous stochastic model, 2) the model including only MCMC-based particle filter described in Section 6.3, 3) our new method including all the ideas, and 4) Itti's model [2]. The result indicates that our model performed more than 10 times faster than the previous stochastic model. The result also indicates that the introduction of the MCMC-based particle filter significantly accelerated the implementation. Consequently, the

**Fig. 6.** Execution time for each step [log msec/frame]

proposed method has achieved near real-time estimation (70-80 msec/frame), and almost the same processing time as the one for Itti's model.

Fig. 6 shows the detailed execution time for each step, where each bar represents the execution time for calculating saliency maps (Section 4), eye focusing density maps from stochastic saliency maps (Section 6.2) and eye movement patterns (Section 6.3), respectively. Note that the execution time for estimating stochastic saliency maps (Section 6.2) was omitted since the calculation was very fast (a few msec) without the installation of any stream processing. The result indicates that the introduction of the MCMC-based particle filter greatly reduced the time needed to perform the most expensive procedure. It should be noted that our new method employed many more samples for estimating eye focusing density maps than the previous method: Previously, we could use a maximum of 500 samples owing to time limitations. On the other hand, the new method enables us to handle more than 5000 samples with reasonable calculation costs. Tree-based multiplication described in Section 6.2 also reduced the processing time for calculating eye focusing density maps from stochastic saliency maps to 1/10.

Fig. 7 shows the model accuracy measured by the NSS score averaged over video clips and human subjects, and Fig. 8 shows the NSS score averaged over human subjects for each video clip. We compared 1) the previous stochastic model, 2) our new method including all the ideas, and 3) Itti model [2]. Note that the introduction of tree-based multiplication does not provide any changes in eye focusing density maps, and therefore the model including only MCMC-based particle filter is equivalent to our new method from the viewpoint of the model accuracy. The result indicates that our new method achieved almost the same NSS score as the previous stochastic model and significantly better NSS score than Itti model. This result implies that our new method can estimate human visual attention with high accuracy.

Fig. 9 shows snapshots of outputs from Itti model (the

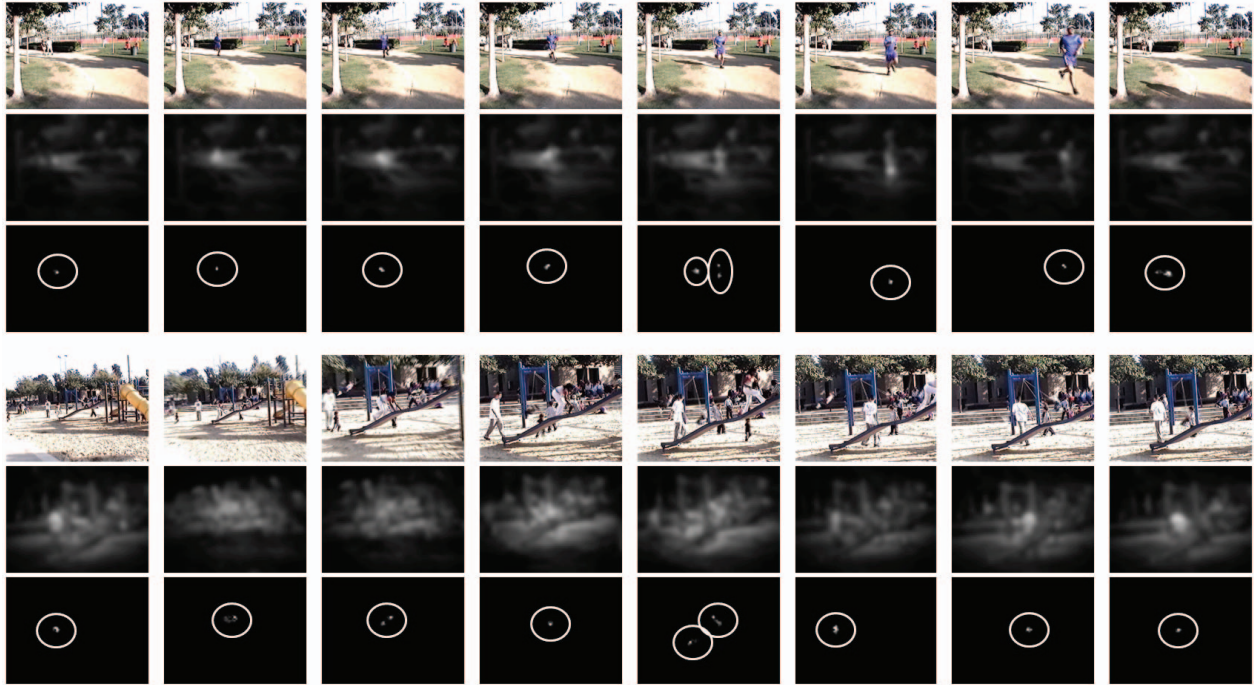


Fig. 9. Snapshots of model outputs

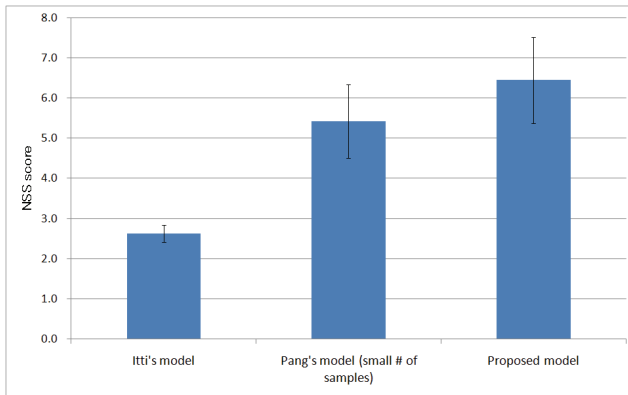


Fig. 7. Average NSS score

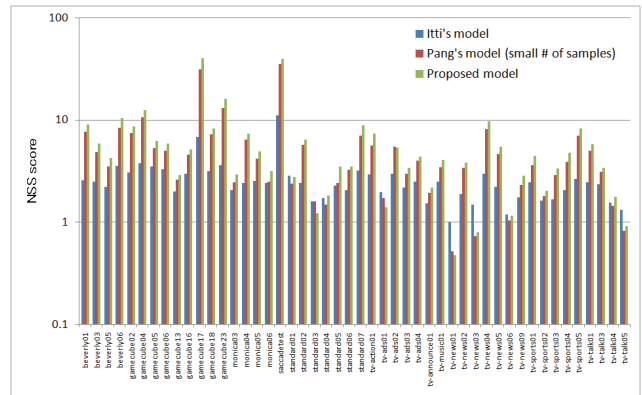


Fig. 8. Average NSS score for each video

second and fifth rows) and our new method (the third and sixth rows). Some video demonstrations will be available on the web³. Note that we manually put some circles on outputs from our new method to show estimated eye focusing areas clearly. This results indicates that outputs from Itti model included several large salient regions. On the other hand, outputs from our new method included only a few small eye focusing areas. This implies that our new method picked up probable eye focusing areas accurately.

³<http://www.brl.ntt.co.jp/people/akisato/saliency-2.html>

8. CONCLUSION

We have proposed a new method for estimating human visual attention with a much shorter execution time. Our main contribution is the incorporation of an MCMC-based particle filter into our stochastic model of saliency-based human visual attention to accelerate the estimation through the GP-GPU framework. Some other modifications including tree-based multiplication were simultaneously realized to make the approach suitable for parallel computing. Experimental results

indicated that the proposed method can estimate human visual attention over 10 times faster than the previous implementation. The results also implied that our new method properly estimated probable eye focusing areas.

The acceleration enables us to integrate the new method as a front-end process of certain applications such as salient region extraction [24], active vision, generic object recognition and content-based video retrieval. A better integration of the bottom-up and the top-down information, and a better saliency model for extracting (deterministic) saliency maps also constitutes promising future work.

9. ACKNOWLEDGMENT

The authors thank Prof. Laurent Itti of University of South California, Prof. Minho Lee of Kyungpook National University, and Dr. Tatsuto Takeuchi of NTT Communication Science Laboratories for their valuable discussions and helpful comments, which led to improvements of this work. The first author spent his summers working as research internship students at NTT Communication Science Laboratories. The authors thank Dr. Yoshinobu Tonomura, Dr. Naonori Ueda, Dr. Hiroshi Sawada, Dr. Kenji Nakazawa and Dr. Kunio Kashino of NTT Communication Science Laboratories for their support of the internship.

10. REFERENCES

- [1] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, pp. 219–227, 1985.
- [2] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, November 1998.
- [3] C. M. Privitera and L. W. Stark, "Algorithms for defining visual regions-of-interest: Comparison with eye fixations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 9, pp. 970–982, 2000.
- [4] E. Gu, J. Wang, and N.I. Badler, "Generating sequence of eye fixations using decision-theoretic attention model," in *Proc. Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2005, pp. 92–99.
- [5] S. Jeong, S. Ban, and M. Lee, "Stereo saliency map considering affective factors and selective motion analysis in a dynamic environment," *Neural Networks*, vol. 21, pp. 1420–1430, October 2008.
- [6] D. Gao and N. Vasconcelos, "Decision-theoretic saliency: Computational principles, biological plausibility, and implications for neurophysiology and psychophysics," *Neural Computation*, vol. 21, no. 1, pp. 239–271, January 2009.
- [7] L. Itti and P. Baldi, "A principled approach to detecting surprising events in video," in *Proc. Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2005, pp. 631–637.
- [8] C. Leung, A. Kimura, T. Takeuchi, and K. Kashino, "A computational model of saliency depletion/recovery phenomena for the salient region extraction of videos," in *Proc. International Conference on Multimedia and Expo (ICME)*, July 2007, pp. 300–303.
- [9] S. Ban, I. Lee, and M. Lee, "Dynamic visual selective attention model," *Neurocomputing*, vol. 71, pp. 853–856, March 2007.
- [10] D. Pang, A. Kimura, T. Takeuchi, J. Yamato, and K. Kashino, "A stochastic model of selective visual attention with a dynamic Bayesian network," in *Proc. International Conference on Multimedia and Expo (ICME)*, June 2008, pp. 1076–1079.
- [11] A. Kimura, D. Pang, T. Takeuchi, J. Yamato, and K. Kashino, "Dynamic Markov random field for stochastic modeling of visual attention," in *Proc. International Conference on Pattern Recognition (ICPR)*, December 2008, p. Mo.BT8.35.
- [12] D. Mallinson and M. DeLoura, "CELL: A new platform for digital entertainment," in *Game Developers Conference*, March 2005.
- [13] B. Chapman, G. Jost, and R. Van der Pas, *Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation)*, MIT Press, 10 2007.
- [14] NVIDIA Corporation, *CUDA programming guide Ver.2.0*, 2008, http://www.nvidia.co.jp/object/cuda_home.jp.html.
- [15] T. Koike and J. Saiki, "Stochastic saliency-based search model for search asymmetry with uncertain targets," *Neurocomputing*, vol. 69, pp. 2112–2126, October 2006.
- [16] T. Avraham and M. Lindenbaum, "Esaliency (extended saliency): Meaningful attention using stochastic image modeling," Tech. Rep., Israel Institute of Technology, 2007.
- [17] G. Boccignone, "Nonparametric bayesian attentive video analysis," in *Proc. International Conference on Pattern Recognition (ICPR)*, Dec. 2008, pp. 1–4.
- [18] P. Longhurst, K. Debattista, and A. Chalmers, "A GPU based saliency map for high-fidelity selective rendering," in *Proc. International Conference on Computer Graphics, Virtual Reality, Visualization and Interaction in Africa (AFRIGRAPH)*, January 2006, pp. 21–29.
- [19] B. Han and B. Zhou, "High speed visual saliency computation on GPU," in *Proc. International Conference on Image Processing (ICIP)*, October 2007, pp. 361–364.
- [20] C. P. Robert and G. Casella, *Monte Carlo statistical methods*, Springer-Verlag, 2004.
- [21] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *Journal of Chemical Physics*, vol. 21, pp. 1087–1092, 1953.
- [22] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*, Artech House Publishers, Boston, 2004.
- [23] R. J. Peters and L. Itti, "Beyond bottom-up: Incorporating task-dependent influences into a computational model of spatial attention," in *Proc. Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2007, pp. 1–8.
- [24] K. Fukuchi, K. Miyazato, A. Kimura, S. Takagi, and J. Yamato, "Saliency-based video segmentation with graph cuts and sequentially updated priors," in *Proc. International Conference on Multimedia and Expo (ICME)*, June 2009, to appear.