Evolutionary Particle Filtering for Sequential Dependency Learning from Video Data

Jun Hee Yoo  
Biointelligence Laboratory  
School of Computer Sci. & Eng.  
Seoul National University  
Seoul 151-742, Korea  
Email: jhyoo@bi.snu.ac.kr

Ho-Sik Seok  
Biointelligence Laboratory  
School of Computer Sci. & Eng.  
Seoul National University  
Seoul 151-742, Korea  
Email: hsseok@bi.snu.ac.kr

Byoung-Tak Zhang  
Biointelligence Laboratory  
School of Computer Sci. & Eng.  
Seoul National University  
Seoul 151-742, Korea  
Email: btzhang@bi.snu.ac.kr

Abstract—We describe a novel learning scheme for hidden dependencies in video streams. The proposed scheme aims to transform a given sequential stream into a dependency structure of particle populations. Each particle population summarizes an associated segment. The novel point of the proposed scheme is that both of dependency learning and segment summarization are performed in an unsupervised online manner without assuming priors. The proposed scheme is executed in two-stage learning. At the first stage, a segment corresponding to a common dominant image is estimated using evolutionary particle filtering. Each dominant image is depicted based on combinations of image descriptors. Prevaling features of a dominant image are selected through evolution. Genetic operators introduce the essential diversity preventing sample impoverishment. At the second stage, transitional probability between the estimated segments is computed and stored. The proposed scheme is applied to extract dependencies in an episode of a TV drama. We demonstrate performance by comparing to human estimations.

Keywords - Evolutionary particle filtering, population codes, video stream learning

I. INTRODUCTION

The ability to understand one’s environment requires dynamic analysis of multimodal streams in realtime. For efficient stream analysis, it is important to observe sequences and understand inherent dependencies in multimodal streams. In this paper, we introduce a novel learning method for multimodal sequential stream. The proposed method divides a sequential stream into a set of segments and extracts hidden dependency structures from the segments set. Each segment is summarized as a set of particles representing dominant features of the segment. Through segmenting process, a sequential stream is converted into a sequence of particle populations and transitional probability between segments is computed. The resulted dependency structure summarizing particles is used to memorize the sequential stream.

In order to extract a dependency structure, we employ an evolutionary particle (PF) filtering [1], [2]. Instead of assuming a prior distribution for a latent variable of a segment, we approximate it by a group of particles. Contrary to conventional particle filter, a group of particles represents dominant image sections. Because each frame in a video stream is continuously influenced by irregular distortions and duration of a segment is not fixed, it is difficult to divide a video stream into segments. This difficulty is alleviated by two-stage learning. At the evolution stage, dominant features are extracted though evolution of particles. At the dependency learning stage, dependency structures are estimated based on the results of evolution stage and represented as a transitional probability matrix. By separating sequence learning into dominant feature extraction and dependency learning, it is possible to analyze streams in a very flexible manner without prior knowledge.

The proposed method is applied to learn a sequence of dominant images in an episode of a TV drama¹. Our aim is to learn sequential dependencies in a drama stream and obtain a set of image descriptors enough to regenerate dominant images. Previous approaches for sequence learning have focused on a stimuli-driven learning process governed by the principle of self-organization [3] or a sequence learning by adaptively combining a set of re-usable primitives [4]. Although these previous works produced very promising results, their approaches are limited due to their application, in these cases, robotics. The “compositionality” for sensory-motor learning [4] is important for a robot because a robot manipulates its world by motion. Contrary to these, it is more important to obtain a proper segmentation of distinct images and a suitable summarization for each dominant image for our work. For image segmentation, restricted characteristics such as fixed order sequence [5] or repeating frames are typically utilized. However, these approaches are unsuitable for our purpose because TV dramas have very flexible structures. Hence we introduce a evolutionary PF based method. The proposed method has potential for analysis of sequential streams due to its transitional probability matrix and segment summarizing particles. We validate our method by segmenting dominant images in a TV drama and comparing the estimated result to human evaluations.

This paper is organized as follows. In Section II, we present some related works. Section III introduces the proposed evolutionary particle filtering and sequential dependency learning scheme. In Section IV, we explain the experimental results. Finally, we discuss the characteristics of the proposed method and summarize the proposed work in Section V.

¹We use an American legal drama-comedy, Boston Legal
II. RELATED WORKS

How could one discriminate intervals of various dominant images in a TV drama episode? Xie et al. introduced a layered mixture model for unsupervised stream clustering based on multi-modal feature distributions [6]. A layered dynamic mixture model in [6] utilizes availability of shots in a news stream which alleviating the difficulty of story boundary finding. Contrary to this, our method is a scheme to divide a TV drama episode which is difficult to expect repeating fixed frames. Gershman introduced a generative model in which, on each trial, a single latent cause is responsible for generating all the observation features [7]. We also assume a latent cause which is responsible for generating all scenes sharing a dominant image. However, we approximate this latent cause by a group of particles rather assuming a prior distribution. For a TV drama segmenting, another difficulty is how to construct visual features. Oliva and Torralba introduced a low dimensional representations of the dominant spatial structure of a scene [8]. In this work, we do not assume a set of global features. Instead we select prevailing features from preprocessed images and reselect features when a dominant image is deemed to change.

If one attempts to build a generative model for given streams, a particle filter method is useful for capturing changes [9]. Particle filtering has been widely applied to various time domain tasks such as mobile robot localization [10] or simulating human sentence processing [11]. One of dominant strengths of particle filtering is its inference capability for latent states [7]. Because its simplicity of the principle of local independence - if a latent variable underlies a number of observed variables, then conditionalization on that latent variable will render the observed variables statistically independent [12], a latent variable model is a promising candidate for explaining observed variables [13], [14]. In this work, we propose a modified particle filter based on [1], [2]. In a typical genetic algorithm approaches, the best chromosome tends to dominate a population. Our approach is different from the typical GA. We introduce genetic operators to preserve diversity in a population. In our approach, a number of particles represents an image in a collaborative manner. That is to say, a particle crowd overcomes the limited representation capability of a particle by combining partial representation of a group of particles.

Modelling dependency structure in sequential datasets has received strong research interest because many real world tasks demand the ability to process patterns in which information content depends not only on static or spatial features but also on temporal order [3]. In [15], a correlation structure of vector-valued observations based on sparse Gaussian graphical models was introduced. A sequential stream with multiple changepoints presents a daunting challenge to unsupervised analysis of the stream. For a stream with unknown number \( K \) of partitions, \( \pi_1, \ldots, \pi_K \), such that the data is independent across segments then, the given stream observed from time 1 to time \( T \) can be represented as Eq. 1

\[
P(y_{1:T} | \pi) = \prod_{k=1}^{K} p(y_{\pi_k})
\]  

(1)

(where, \( y_{\pi_k} \) means a set of frames belong to partition \( \pi_k \)

The probability of data from time \( t \) to \( s \) belong to the same segment is defined as Eq. 2 [16].

\[
P(t, s) = Pr(y_{t:s} | t, s \text{ in the same segment}) = \int \prod_{i=t}^{s} f(y_i | \theta) \pi(\theta) d\theta
\]

(2)

(where, \( \theta \) is a parameter associated with each partition \( \pi_k \)).

In [16], the author assumed a prior distribution for the time between successive points as having the characteristics of the negative binomial family. In this paper, we do not aim to derive the best regression model for a given stream. Our goal is to estimate a segments structure of a stream of characteristics making it impractical to assume an underlying distribution. Therefore, we circumvent it by focusing on likelihood of a new frame given the current particle population instead of trying to compute Eq. 2. Our method is capable of inferring change of dominant images. For modelling of changing images, Gaussian mixture model (GMM) is one of classical means [17]. In [17], only pixels satisfy a given confidence value are considered as foreground pixels. We enhance the robustness by estimating changepoints based on visual words produced by SIFT (Scalar invariant feature transform) method [18].

![Fig. 1. Construction of particles and image representation. (a) SIFT feature extraction. (b) Visual word. (c) Chromosome. (d) Image representation.](image)

(a) SIFT feature extraction  
(b) Visual word  
(c) Chromosome  
(d) Image representation

III. EVOLUTIONARY PF AND DEPENDENCY LEARNING

In this section, we describe the proposed dependency learning method. By detecting change of dominant images, the proposed method makes segments for given streams. The
proposed method builds a generational model in which evolved particles summarize associated segments and regenerating the segment. In order to generate a group of particles corresponding to a dominant image and convert the resulting groups of particles into a sequence, we introduce a method composed of an evolutionary particle filtering phase (algorithm 1) and sequential dependence learning phase (algorithm 2).

A. Proposed method

1) Particle representation: In Fig. 1, a chromosome structure of the proposed method is shown. The goal of the proposed method is to extract prevailing features from a segment. For collective representation, each particle \( p_{f_1} \) is constructed as a set of undefined number of SIFT features, \( p_{f_1} = \{ s_j \mid j = 1, \ldots, J \} \) (\( J \): undefined, \( s_j \): SIFT features and location information). At the initial state, the number of SIFT features in all particles are same. However, the number of SIFT features in a particle changes during evolution. Although we do not limit the maximum number of SIFT features in a particle, we did not observe extremely large number of SIFT features in all particles are same. However, the number of SIFT features due to a penalty on number of SIFT features (\( C_{r_e} \) in Eq. 4).

Algorithm 1 Evolutionary PF for stream segmentation

| Input : \( D = \{ y_{1:T} \} \) (\( D \): SIFT transformed T frames) |
| Output : \( PF = \{ P_{f_1} \mid i = 1, \ldots, n \} \) |
| \( P_{f_1} \): a set of particles \( (p_{f_j}) \) for \( i \)th state |
| (here, \( n \) is the number of estimated segments.) |

Begin

\( \circ \) Initialization (for each segment):

1. Draw a set of nodes (C) based on initial frame.
2. Initialize chromosomes with variable orders using C. 

\( \circ \) Sequence estimation and state summarization

- Goal: maximize Q defined by Eq. 3

\[
Q(P_{f_1}|P_{f_{t-1}}, y_{s:t}) = E[Pr(y_{s:t} | P_{f_1}) | P_{f_{t-1}}, y_{s:t-1}]
\]

(3)

- Evolutionary particle filtering

\( \circ \) Stage 1: Selecting particles and applying crossover using fitness as an importance distribution \( q \).

\( \circ \) Stage 2: Applying mutation operator to avoid degeneracy.

- Fitness function

\[
F_1 = f_{C_{r(a,b,c)}} (p_{f_j})
\]

(4)

\( F_1 \) evaluates each chromosome based on \( C_{r_a}, C_{r_b}, C_{r_c} \) (\( C_{r_a} \): inter-closeness, \( C_{r_b} \): coverage, \( C_{r_c} \): penalty)

\( \circ \) Likelihood

\[
F_2 = f(y_{s:t} | P_{f_{t-1}})
\]

(5)

here, \( F_2 \) computes the coverness of \( y_{s:t} \) by \( P_{f_{t-1}} \).

\[
\begin{cases} 
F_2 < \gamma: \text{start a new segment and initialize } P_{f_t} \\
F_2 \geq \gamma: y_{s:t} \text{ is regarded as an element of } P_{f_{t-1}}
\end{cases}
\]

In Fig. 1, each circle represents a particular feature in an image and a square covering a number of circles corresponds to a SIFT feature for the group of particular features. By distilling particles through evolution, it is possible to obtain a group of particles \( (P_{f}) \) representing a given image with essential characteristics.

2) Evolutionary particle filtering: In phase 1 (algorithm 1), the goal is to generate particle filters capable of capturing essential characteristics of a segment. Eq. 3 represents this goal in a temporal manner. When a new frame is observed, the proposed method tries to enhance existing particles by incorporating new features. However, we cannot ignore the possibility of a new segment. In order to resolve this conflict, the proposed method focuses on likelihood of a new frame being generated by the current population.

In the proposed method, each particle is defined as a set of \( N, E > (N \): nodes, \( E \): edges between nodes) in a same manner of hypernetwork [19]. Nodes are selected from visual words and edges link the selected nodes. The novel point is that the introduced particles represent an image in a collective manner. Due to its vast dimensionality, it is impractical to assume a single data type representing an image. A data structure representing a single feature (single visual word) is also too limited. Our approach aims the middle ground. Although a particle has limited representation capability, it is possible to capture prevailing patterns based on a group of particles depicting partial sections of an image. A group of particles with higher weight (better fitness value) are selected to represent an image. In this paper, a population is composed of 100 particles and a particle with better fitness replaces a particle with worse fitness. The search process is biased to reflect the representing capability of a current particle set \( P_{f_1} \), and their structural characteristics. Fitness value of each particle is computed using Eq. 4 and the fitness function has three components. In Eq. 4, \( C_{r_a} \) evaluates distance between nodes in a particle using Eq. 6.

\[
C_{r_a} = \sum_{j=1}^{J} |S_j - \text{Cent}|^2
\]

(6)

\( (S_j): j\text{th node in a particle, Cent: a centroid point of all nodes in a particle)\)

Using \( x - y \) coordinate in each node, it is possible to compute distance between nodes. With too long average distance, it is likely that the particle is not robust enough because some part of a particle would disappear at the next frame. Therefore, \( C_{r_a} \) prefers lower average distance in a particle. \( C_{r_b} \) measures representing capability of a particle. \( C_{r_c} \) is a penalty term. If a particle has too many nodes, it is possible that this particle has too detailed information. In order to obtain more abstract level of representation, we prefer shorter particles. Because a group of particles represents an image in a collaborative manner, Eq. 5 is a function of a particle group \( P_{f} \). In contrary fitness is a measure for an individual, hence Eq. 4 is a function of a particle, \( p_{f} \).

In order to approximate a true distribution, particle filtering requires an importance distribution \( Q \). The proposed method samples particles based on a distribution of fitness value from
Eq. 4.

New particles are generated through evolution. There is a well known problem called the degeneracy [9] caused by excessive resampling. We overcome the degeneracy problem by introducing new particles through genetic operations (Eq. 7). In order to maintain the fixed population size \( N_s \), new population is constructed based on fitness of offsprings and particles of the previous generation.

\[
\begin{align*}
  P_{a_i}' &= P_{a_i} \odot P_{a_j} \quad (i \neq j \text{ and } \odot : \text{crossover}) \\
  P_{a_i}'' &= m(P_{a_i}) \quad (m(\cdot) : \text{mutation}) \\
  P_f_{t+1} &= P_a + P_a' + P_a'' \quad (\text{here, } N_s \text{ is fixed})
\end{align*}
\]

An image is regenerated based on a set of particles. Thus, an image regeneration could be explained as a searching procedure for an image with maximum likelihood for a given group of particles. If one have a library of all possible images, then regeneration is a process depicted by Eq. 8. This approach is too burdensome, hence we represent an image by using SIFT feature in each node. This scheme is represented by \( F_2(c|P_f) \) in Eq. 8.

\[
Image = \arg \max_{c \in C} Pr(C|P_f)
\propto F_2(c|P_f) \tag{8}
\]

The evolved particles represent a dominant image in a segment but a new set of particles \( P_f' \) should be generated when a frame belonging to a new segment is observed. If likelihood by Eq. 5 is lower than a threshold, it is determined as a new segment. Ideally, the likelihood of a new frame \( y_{t+1} \) should be estimated using Eq. 9.

\[
p(y_{t+1}|y_{t+1},m) = \int_{\theta} \prod_{i=0}^{t+1} p(y_i|y_{i-1},\theta,m)p(\theta|m)p(m)d\theta
\tag{9}
\]

(here, \( \theta \) has same meaning with \( \theta \) in Eq. 2.)

In the proposed method, we do not assume any specific distribution for each term in Eq. 9. Therefore, covering ratio of current \( P_f \) by Eq. 5 substitutes likelihood by Eq. 9.

3) Sequential dependency learning: After phase 1, \( Pf = \{P_f_i : i = 1, \cdots, n\} \) is obtained. In the second phase (algorithm 2), we transform \( Pf \) into a transitional probability matrix of \( T \). Input to Algorithm 2 is \( Pf = \{P_f_i : i = 1, \cdots, n\} \) obtained from Algorithm 1. Output of Algorithm 2 is a transitional probability matrix of estimated segments. The
A. Data and preprocessing

We requested 10 human participants (10 undergraduate students) to evaluate 3 episodes of a TV series *Boston Legal*. Human participants estimated changes of dominant images. We did not provide exact definition of a dominant image. Total play time of test materials is 125 minutes 30 seconds and we sampled total 7530 scenes to construct test materials.

<table>
<thead>
<tr>
<th>Time</th>
<th>Episode 1</th>
<th>Episode 2</th>
<th>Episode 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.84s</td>
<td>2.10s</td>
<td>1.88s</td>
</tr>
<tr>
<td>10</td>
<td>1.03s</td>
<td>1.15s</td>
<td>1.03s</td>
</tr>
<tr>
<td>20</td>
<td>1.00s</td>
<td>1.00s</td>
<td>1.00s</td>
</tr>
<tr>
<td>30</td>
<td>8.00s</td>
<td>7.00s</td>
<td>9.00s</td>
</tr>
</tbody>
</table>

*Fig. 3. (a) Evaluation results by humans for the circle-shaped interval in (b). Each line means a changepoints interval. (b) Evaluation results of 10 participants for an episode.*

We summarize the evaluation results by human in Fig. 3, Fig. 4, and Table I.

IV. EXPERIMENTAL RESULTS

In this section, we report experimental results and discuss their meaning. We focus on the estimated accuracy, fitness curves, and image regeneration performance. Prior to provide experimental results, we report about human estimations.
As expected, human did not agree on changepoints and several changepoints constitute an interval (refer to Fig. 3 and Fig. 4). For all the human evaluated changepoints, we verified similar changepoints manually and assigned an interval to a group of changepoints denoting a same change. Table I shows the statistical characteristics of those intervals. The average length of the intervals in each episode are 1.84s, 2.10s, and 1.88s respectively. A minimum interval length of 1.0s is acceptable but a maximum interval length of 9.0s and 8.0s seems to be too long. These maximum intervals correspond to segments where frames change continuously in such cases showing Boston skylines.

Although the number of intervals is unknown, the atomic elements such as visual words should be preprocessed for image analysis. Images are preprocessed using SIFT (Scale Invariant Feature Transform) method [18] to provide change-robust features.

**B. Estimation results**

We proceed experiment with the parameters in Table II and we guaranteed minimal evolution time 0.1s for each segment.

<table>
<thead>
<tr>
<th>TABLE II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
</tr>
<tr>
<td>Crossover</td>
</tr>
<tr>
<td>Rate</td>
</tr>
</tbody>
</table>

**TABLE III**

**COMPARISON OF THE PROPOSED METHOD AND HUMAN EVALUATIONS**

<table>
<thead>
<tr>
<th></th>
<th>Episode 1</th>
<th>Episode 2</th>
<th>Episode 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of changepoints</td>
<td>981</td>
<td>798</td>
<td>818</td>
</tr>
<tr>
<td>Number of intervals</td>
<td>560</td>
<td>444</td>
<td>498</td>
</tr>
<tr>
<td>Estimation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of changepoints</td>
<td>2206</td>
<td>1746</td>
<td>1576</td>
</tr>
<tr>
<td>Precision</td>
<td>0.273</td>
<td>0.410</td>
<td>0.461</td>
</tr>
<tr>
<td>Recall</td>
<td>0.613</td>
<td>0.896</td>
<td>0.889</td>
</tr>
</tbody>
</table>

Table III provides human evaluation results and estimation performance by the proposed method on two episodes. For episode 1, 10 participants evaluated 560 intervals. For episode 2, there are 444 intervals. For episode 3, human participants evaluated 498 intervals. Our method estimated 2206 changepoints for episode 1, 1746 changepoints for episode 2, and 1576 changepoints for episode 3. In the video stream analysis, it is hard to expect that an estimated changepoint and an evaluated changepoint coincides exactly. Therefore, we regard an evaluated changepoint as a correct one if it belongs to an interval of human evaluated points. In order to demonstrate the performance in a more traditional way, we provide recall and precision.

\[
\text{Precision} = \frac{\#(\text{Correctly estimated changepoints})}{\#(\text{Total estimated changepoints})}
\]

In the case of precision, the proposed method achieves 0.273, 0.410, and 0.461 for episode 1, episode 2 and episode 3 respectively. Recall performance is 0.613, 0.896, and 0.889 for each episode. In order to improve precision, we need to reduce the number of estimated changepoints. Likelihood computed by \(F_2\) has much room to improve the performance. For example, it is possible to focus on likelihood tendency. Although inferior to human evaluations, the proposed method achieves significant results considering the fact that only SIFT transformed features were used for image preprocessing.

![Fitness curve](image)

**Fig. 5. Fitness curve**

Fig. 5 shows a fitness curve of the best particle for a segment. Because the best particle of the previous generation is swapped only if there exists a better one in the offsprings, Fig. 5 has a stepwise form. It took 60 generations to estimate the target segment. Because we aim to develop a scheme analyzing a stream in real time, we cannot afford unlimited number of generations. Therefore, we need to investigate the optimal way to calibrate balance between diversity and convergence for video stream analysis.

<table>
<thead>
<tr>
<th>TABLE IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>EFFECTS OF SUB-FITNESS COMPONENTS</td>
</tr>
<tr>
<td>(C_{r_a})&amp;(C_{r_b})&amp;(C_{r_c})</td>
</tr>
<tr>
<td>Precision</td>
</tr>
</tbody>
</table>

In order to verify the effect of sub-components in Eq. 4, we report the effects of each component in Table IV. For 30 minutes portion of episode 1, we estimated precision for cases of using \(C_{r_a}\)&\(C_{r_b}\), \(C_{r_a}\)&\(C_{r_c}\) and \(C_{r_b}\)&\(C_{r_c}\), respectively. Table IV indicates that \(C_{r_b}\) has the strongest effect and \(C_{r_a}\) has the weakest effect. Without \(C_{r_b}\), the precision plummeted over 11% compared to the case of without \(C_{r_a}\). Because the proposed method estimates image changes based on the regeneration performance, the importance of \(C_{r_b}\) could be easily understood.

Fig. 6 shows an image generated from an evolved population. This populations is composed of 20 particles. With only 20 particles, it was possible to generate an plausible image. Because each population has enough information to regenerate
a dominant image of the segment, the proposed scheme can be interpreted as a method capable of summarizing a given stream.

A sequence of regenerated images in Fig. 7 was created based on the transitional probability matrix (matrix) by Algorithm 2. In Fig. 7, left images depict images of the original sequence and the right images correspond to images representing the estimated segments. The images were created by the following way. When an image seed is given, the next segment for the seed image is estimated based on the matrix. For the estimated segment, the next segment is re-estimated also based on the matrix. We attempted to estimate 3 segments and validate the sequence by regenerating a representative image for each segment. Among 3 segments in Fig. 7, only the second segment is not correct one. However the second image is also a partially correct answer because a dominant character (a bald man) appears in the both of the original image and the estimated image. If we regard the second image as a partially correct one, then it is apparent that the proposed method estimated a correct segment sequence. This is another contribution of the proposed method. By constructing a transitional probability matrix, the proposed method can represent a video stream in much smaller and simpler data structure and regenerate a partial segments based on the transitional probability matrix.

V. CONCLUSION

We have described a method to make segments in a video stream and estimate their dependency. For segmenting a stream, we modified particle filtering method to estimate segments based on a group of particles in a collaborative manner.

Through evolution, dominant visual words were incorporated into particles. By representing a segment based on a set of particles, it was possible to tolerate some minor distortions due to camera works or light. The segmented sequences were used to learn a temporal dependencies in a video stream. The obtained segments sequence was transformed into a transitional probability matrix by clustering similar segments into a state and this transitional probability matrix was used to anticipate a next segment for a given cue. The resulted state sequence and associated particle groups actually constituted a compressed version of the given video stream.

The performance of the proposed method was verified against human evaluations. Compared to human evaluations, the proposed method reported the following characteristics. The proposed method estimated four times as much change-points than human participants and the estimation accuracy is between $0.273 \sim 0.461$. Although this performance is promising one considering the task difficulty, we need more improvement especially in terms of reducing the number of estimated changepoints.

The proposed method is the first step towards more ambitious goal of real-world stream contents analysis and on-
line recommendation of contents. In order to accomplish these goals, following further works are needed. Firstly, the proposed method needs a measure to control the evolution procedure adaptively. Depending on an initial population and a sequence of images, number of necessary generations should be changed. Secondly, we need a new search bias to reach a sub-optimal state more quickly. Secondly, we need a more robust method to compare particle populations.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MEST) (No. 2011-0016483), the Industrial Strategic Technology Development Program (10035348) funded by the Korean government (MKE), and the BK21-IT Program. The ICT at Seoul National University provides research facilities for this study.

REFERENCES