Ph.D Dissertation

Attentional Sampling for Efficient Visual Computing

효율적 영상처리를 위한 주의집중 샘플링

By

Hyung Jin Chang

February 2013

SCHOOL OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Abstract

In many practical computer vision scenarios it is possible to use information gleaned from the previous observations through the sampling process. In order to achieve a good performance with small computation, it is desirable that the samples cover the domain of target distribution with the small number of samples as possible via a concept of active or adaptive sampling. Based on the active sampling strategy, sampling could be concentrated on attentional portions, which can improve not only the sampling efficiency but also performances of algorithms. In this thesis, we define three different attentional sampling concepts, *structured attentional sampling*, *empirical attentional sampling* and *selective attentional sampling*. The proposed attentional sampling methods are successfully applied to computer vision problems, by achieving dramatic improvement in the sense of performance as well as computational load.

The *structured attentional sampling* scheme uses an inherent structure to sample an interesting region densely instead of equally distributed sampling over the entire region. This sampling scheme is applied to a tracking failure detection method by imitating human visual system. In this scheme, we adopt a sampling structure based on Log-polar transformation simulating retina structure. Since the log-polar structure shows invariance against rotational changes and intensifies translational changes, it helps to reduce false alarms arising from rotational pose variations and increase true alarms in abrupt translational changes. In addition, foveal predominant property of log-polar structure helps to detect the tracking failing moment by amplifying the resolution around focus (tracking box center) and blurring the peripheries. Each ganglion cell corresponds to a pixel of log-polar image, and its adaptation is modeled as Gaussian mixture model.

The validity of the structured attentional sampling method is illustrated through various experiments.

The *empirical attentional sampling* scheme uses previously obtained empirical knowledge when sampling in current time. The empirical knowledge is modeled by a probability distribution function through an empirical learning process. This empirical sampling scheme is applied to mask generation to speed up conventional background subtraction algorithms for moving object detection. The proposed sampling strategy is designed to focus on attentional region such as foreground regions. The attentional region is estimated by using the detection results in the previous frame in a recursive probabilistic way. We generate a foreground probability map by using foreground properties of temporal, spatial, and frequency properties. Based on this foreground probability map, randomly scattered sampling, spatially expanding importance sampling and surprise pixel sampling are performed sequentially to make the attention sampling mask. The efficiency of the proposed empirical attention sampling method is shown through various experiments. The proposed masking method successfully speeds up pixel-wise background subtraction methods approximately 6.6 times without deteriorating detection performance. Also real-time detection with Full HD video is successfully achieved by various conventional background subtraction algorithms together with the proposed sampling scheme.

The *selective attentional sampling* scheme does not use whole data but selects only important data enough to achieve a given classification objective. This selective sampling scheme is applied to the recognition of pop dances. Pop dances are action streams consisting of diverse actions which cannot be simply annotated. For such “unannotatable” action streams, conventional methods cannot be applied directly due to their complexity and longevity. In order to describe unannotatable action stream effectively, the proposed method employs a novel mid-level “feature flow” with low dimensional embedding. Also, for the purpose of recognition, “attentional motion spots” holding important information about the sequence are automatically selected. The feature values and the temporal locations of each attentional motion spot are modeled with Gaussian mixtures as “Action Charts.” The Action Chart describes the characteristics of an action stream in the spatio-temporal domain. Using the abstract information in the Action Charts, the proposed method efficiently recognizes pop dance sequences. In order to demonstrate the validity of the proposed method, we compare our method against the state-of-the-art methods with a newly built SNU Pop-Dance dataset containing long action streams composed of diverse actions.

Keywords: attentional sampling scheme, structured attentional sampling, empirical attentional sampling, selective attentional sampling, tracking failure detection, speed-up of background subtraction, complex action recognition

Student ID Number: 2006-21280

Contents

List of Figures

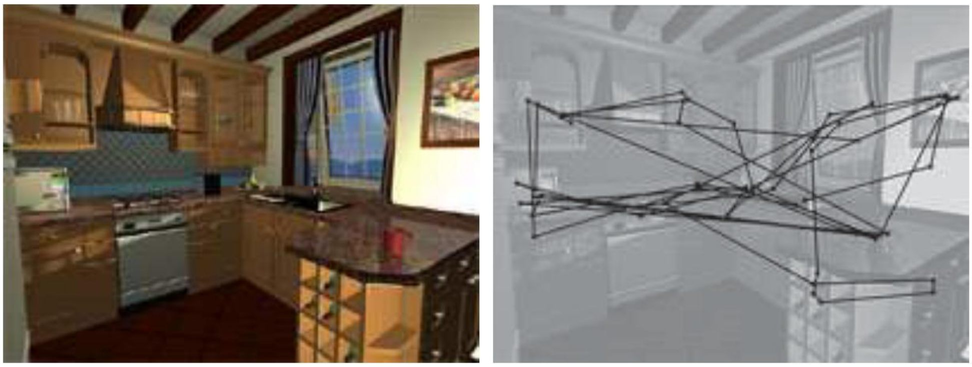
List of Tables

Chapter 1

Introduction

1.1 Motivations

When human perceives the surroundings, one instinctively focuses on important parts and disregards the rest, which is called as attention in human cognition research area[1]. The *attention* is performed in different ways depending on the task. For example as shown in Figure 1.1 [2], when looking at a picture, people look briefly at bland backgrounds and start focusing on “structurally salient” features, like a painting on a wall. Then people can easily recognize that this is a picture of kitchen and start to look closely at “empirically important” parts, such as a gas stove, a sink, or on the table which is empirically known to be related with the kitchen. However, if there is a tail wagging dog on the kitchen floor or if people are asked to find something “selectively” in the picture, then people will focus on it rather than others. Clearly there are different types of attention and there is a feedback between the prior knowlege and the sensing process. One starts by having a rough idea of the entire scene and gather informations using instinctively designed sensing mechanisms, and then has attention on particularly



(a) (b)

Figure 1.1: Attentional region by tracking eye movement. (a) example of a natural scene (b) tracked eye movement during the first seconds of scene perception [2]. (source: MSU Vision Cognition Laboratory)

“attentional” details using empirical knowledges or specific intentions.

The attentional learning and sensing processes are natural abilities for humans however most data sampling processes of artificial learning algorithms and sensing devices do not take advantage of the attentional scheme [2]. For example, all the image pixels of digital camera have the same importance in the image, and most detection algorithms perform full search in the image to detect moving object or faces. This kind of inflexible processes can bring about redundant computational costs or infeasible solutions.

In this thesis we propose attentional schemes to improve the processing efficiency in visual computing such as moving object detection/tracking and action recognition and so on. We define three categories of attentional sampling as *structured attentional sampling*, *empirical attentional sampling* and *selective attentional sampling* (as shown in Figure 1.2). With these attentional sampling methods we can encompass various applications that achieve significant efficiency improvements. In this thesis, the structured, empirical, and selective attentional sampling schemes are applied to tracking failure detection [3, 4, 5, 6],

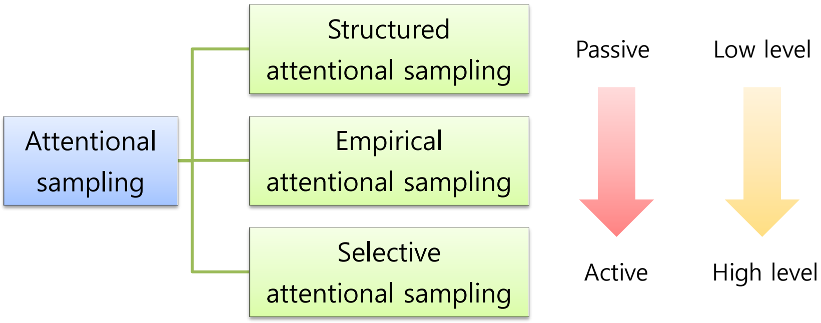

Attentional sampling
Structured attentional sampling
Empirical attentional sampling
Selective attentional sampling
Passive
Active
Low level
High level

Figure 1.2: Types of the proposed attentional sampling schemes.

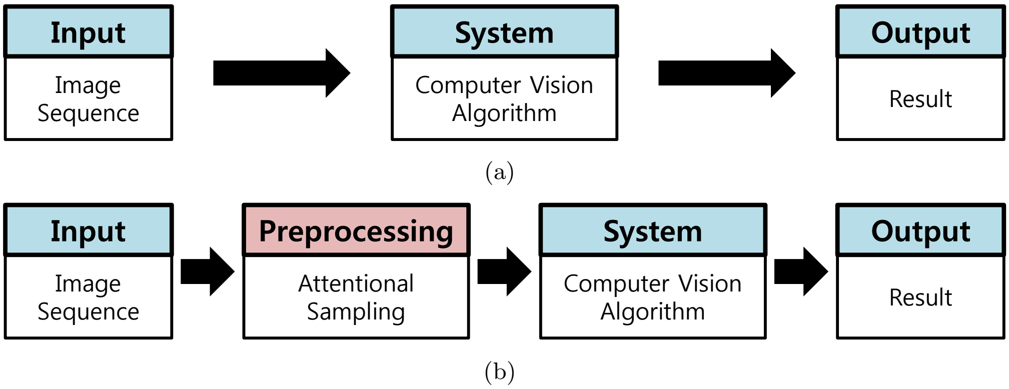
moving object detection [6, 7, 8, 6, 9, 10, 11, 12, 13, 14], and action recognition [15, 16, 17, 18, 19, 20], respectively.

1.2 Contents of Research

The proposed attentional sampling is a kind of preprocessing method which should be placed prior to conventional computer vision algorithms as shown in Figure 1.3. However, unlike the general preprocessing filters, the proposed sampling method is problem-dependent and requires a sophisticated sampling strategy design using prior knowledge about the problem. In the following, we present the basic concepts for the attentional sampling schemes and their applications to visual computing.

1.2.1 Structured Attentional Sampling

Structured attentional sampling performs data sampling according to a pre-designed sampling pattern structure. A sparsely but regularly designed attentional sampling pattern can reduce computational load and achieve a noise reduction effect as well. This sampling method is performed in a passive way comparing to the others. The predetermined sampling pattern does not vary as time goes on, but



(a)

(b)

Figure 1.3: (a) A conventional flow of computer vision system. (b) The location of the proposed attentional sampling method in a system flow.

an anchoring position of the sampling pattern is adaptively changed. This is similar to human eye mechanism. Human can freely change a focusing point but cannot change a distribution of retina ganglion cells which do sampling as photoreceptors. This sampling method is easy to implement and fast, but designing sampling pattern is critical. A prior knowledge about an attacking problem must be reflected on the sampling pattern.

Suppose that we find a sampling point set $\mathbf{X}^t$ at time t. The $\mathbf{X}^t$ is obtained using an anchoring position c, a structured sampling pattern $\mathbf{S}(c;\Psi)$ pre-designed using a prior knowlege Ψ.

- **Structured attentional sampling:** $\mathbf{X}^t$ can be obtained as

$$\mathbf{X}^t = h(\{\mathbf{S}(c;\Psi)\}, U^t), \tag{1.1}$$

  where $h(\cdot)$ is a deterministic function, and U^t accounts for possible randomization of the sampling rule.

In general the function $h(\cdot)$, together with U^t, is called the sampling strategy at time t [2]. This approach is applied to a tracking failure detection which uses log-polar transformation as a sampling structure by imitating human visual system. This part is presented in Chapter 2.

1.2.2 Empirical Attentional Sampling

Several empirical and theoretical results suggest that the use of data collected in early stages can be very helpful for efficient selection of new samples [21, 22]. The *empirical attentional sampling* is designed based on this results. This sampling method is similar to a human sensing mechanism of experience. People can efficiently focus on important parts when they are accustomed to the situation. In order to simulate the experience, we present a probability density map of attention which is squentially updated by the previous results. The sampling in the current state is performed based on the probability density map. Designing of the density map and update rule is a user designing part reflecting prior knowledge on a target problem.

Suppose that we choose a sampling point set $\mathbf{X}^t$ (at time $t \in \mathcal{T} = \{1, ..., T\}$) among current input data image $\mathbf{I}^t$. The sampling points are collected based on the current input data $\mathbf{I}^t$, a previous sampling point set $\mathbf{X}^{t-1}$, its result set $\mathbf{Y}^{t-1}$, and sequentially updated density map $\mathcal{P}^{t-1}$.

- **Empirical attentional sampling:** $\mathbf{X}^t$ is obtained by

$$\mathbf{X}^t = h(\mathbf{I}^t, \mathbf{X}^{t-1}, \mathbf{Y}^{t-1}, \mathcal{P}^{t-1}, U^t), \tag{1.2}$$

$$\mathcal{P}^t = \mathcal{D}(\mathbf{X}^t, \mathbf{Y}^t, \mathcal{P}^{t-1}; \Psi), \tag{1.3}$$

  where $h(\cdot)$ is a deterministic function, and U_t accounts for possible randomization of the sampling rule. $\mathcal{D}(\cdot; \Psi)$ implies a user designed update rule for density map update.

This approach is applied for sampling mask generation to speed up the conventional background subtraction algorithms. The proposed sampling strategy is designed to focus on attentional region such as forground regions. This part is presented in Chapter 3.

1.2.3 Selective Attentional Sampling

Selective attentional sampling selects distinctive points using prior knowlegde about dataset or higher level intention and performs classification and recognition only using the selected samples. The selected samples should include important data enough to achieve a given objective without redundancy. So in this sampling method, the system designer's knowledge is actively reflected in designing the attention scheme. This sampling method simulates active human attention by intention or belief.

Suppose that we select an attentional sampling point set $\mathbf{X}^t$ (at time $t \in \mathcal{T} = \{1, ..., T\}$) among the current input data $\mathbf{I}^t$. The attentional point selection is performed by measuring each data point's importance denoted by $\mathcal{M}(\mathbf{I}; \Psi)$. The importance measure is designed by a system designer reflecting his/her knowledge about the dataset (Ψ). For example, in recognizing a long pop dancing movement, we remember not the whole dancing motions but only some important and characteristic dancing motions. The characteristic motions are mainly determined by motion intensity, interesting poses or syncronization to music etc. These determining features (Ψ) are used for measuring motion importance ($\mathcal{M}(\cdot; \Psi)$) and highly importance motions are selected for recognition.

- **Selective attentional sampling:** Time $t \in \mathcal{T} = \{1, ..., T\}$ and location in data space $p \in \mathcal{N} = \{1, ..., N\}$

$$\mathbf{X}^t = h(\mathcal{M}(\{\mathbf{I}_{\mathcal{N}}^{(1...t)}(p)\}; \Psi), U^t), \tag{1.4}$$

where $h(\cdot)$ is a deterministic function, and $\mathcal{M}^t$ is an importance measure of an attentional feature point.

Measurement of the amount of importance for each point should be defined first and an appropriate classification method for the sparsely sampled points is required. If the sampled data are informative and distinctive enough, then training solely with the sampled data can reduce computational load significantly without performance degeneration. This approach is applied to modeling the sequential flow characteristics of spatio-temporal data patterns for recognizing long and complex action sequences such as pop dances. As a result, a new Action Chart method is developed to recognize Pop dances based on this selective attentional sampling concept. This part is presented in Chapter 4. Finally Chapter 5 gives concluding remarks as well as speculation on future research directions and open problems.

Chapter 2

Structured Attentional Sampling for Tracking Failure Detection

2.1 Introduction

The sampling pattern $\mathbf{S}(c; \Psi$ in Eq.1.1 for the structured attentional sampling can be designed freely depending on the purpose of a target application, and the sampling pattern can meet a purposes such as sparse sampling for computational efficiency and intensive sampling for accuracy only if sufficient prior knowledge (Ψ) about the target application is provided. In this chapter, a designing of a structured attentional sampling strategy for *tracking failure detection* (TFD) is presented. We show that a properly designed sampling pattern can provide not only computational efficiency but also novel properties for the special purpose. In this chapter, we use a log-polar sampling as shown in Figure 2.1 from the basic prior knowledge that object tracking is the same job of placing the tracking

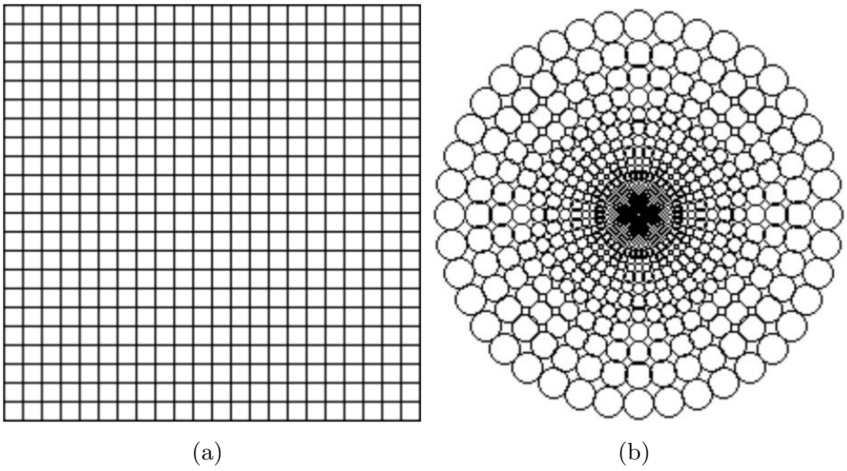


Figure 2.1: (a) Conventional spatially uniform sampling. (b) Structured attentional sampling.

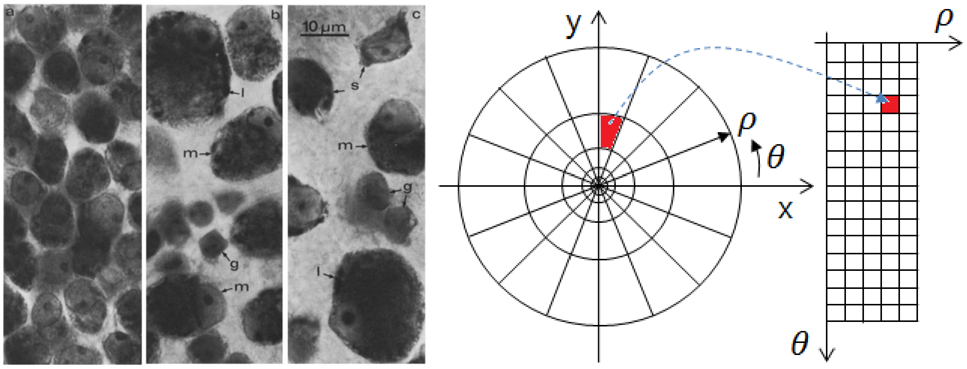
object in the center of tracking box in every frames.

In computer vision, there have been lots of efforts to improve tracking performance, and as a result, most algorithms work well for many challenging situations. Nevertheless, they still lost their tracking object in the long run. Current visual surveillance system [23] restores failed tracker manually. However if we can detect a tracking failure moment, the restoration can be performed automatically. So the TFD is an important component for automatic tracking system.

Most of the existing TFD methods are based on checking similarity measures. [3, 4] detect tracking failure by thresholding a similarity measure of tracker. However, because the similarity measures are not originally designed for TFD, they cannot represent a status of current tracker exactly. Sometimes the similarity measure frequently results in a low value even when the tracking is successful or varies smoothly when the tracking fails by slow changes. So [5] defines a new similarity measure only for TFD. It is assuemed that the boundary of tracking box does not include any pixels of tracking object. However, in actual application, this assumption may be easily violated and as a result it leads to frequent false alarms.

We propose a new approach for TFD by mimicking human visual system. When people look at an object, the attentional area seems clear but peripheries are blurred. This is because of the structure of the retina of a human eye. Fovea [24] is a part of the eye, located in the center of the macula region of the retina. The fovea is responsible for sharp central vision and is surrounded by the parafovea belt and the perifovea outer region. The parafovea and perifovea are composed of sparse ganglion cells [24]. Approximately 50% of the nerve fibers in the optic nerve carry information from the fovea, while the other 50% carry information from the rest of the retina. Log-polar image geometry was first motivated by its resemblance with the structure of the retina [25]. We use a log-polar image for simulating human vision and its characteristics.

Our method focuses on capturing a distinctive feature when tracking fails instead of comparing similarity measures. At an instant when a target object moves out of the area of focus (i.e. tracking fails), the object suddenly becomes blurry, whereas the surroundings gets sharp. We use this sudden sharp and blur view change as an important feature for TFD. Human can detect the changing moment by percepting the amount of fired (stimulated by new color) ganglion cell. We model ganglion cells in the retina as pixels in log-polar image and the adaptation of ganglion cell as Gaussian mixture model (GMM) [6]. So, the perception of the amount of fired ganglion cells is modeled by counting new colored pixels in the log-polar image. This measure is independent of the tracker, so it can be applied to any trackers. The effectiveness of the proposed TFD is shown by several experiments.



(a) (b)

Figure 2.2: (a) Light micropraph of ganglion cells of human retina [24]. (Left) parafoveal region. (Center) midperifovea region. (Right) perifovea region. (b) The log-polar transformation. The radially logarithmic sampling entails that foveal information is represented by a large number of pixels in the log-polar image.[25]

2.2 Characteristics of Log-Polar Image and Tracking Failure

2.2.1 Properties of Log-Polar Image

The log-polar transformation [25] means a conformal mapping (preserves oriented angles between curves and neighborhood relationships) from the point (x, y) on the cartesian plane to point (ρ, θ) in the log-polar plane, where

$$\rho = log(\sqrt{x^2 + y^2}) \tag{2.1}$$

$$\theta = \arctan(y/x). \tag{2.2}$$

The log-polar mapping has three properties. *Biological plausibility, rotation and scaling invariance, and foveal predominance* [25]. As going away from the fovea region, the ganglion cells are sparsely distributed (Fig. 2.2(a)). This charac-

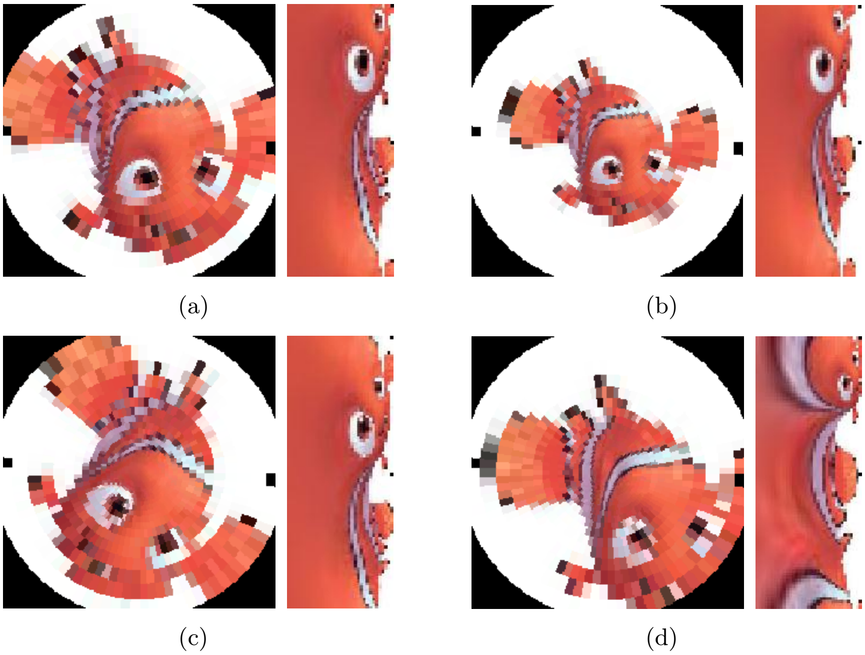
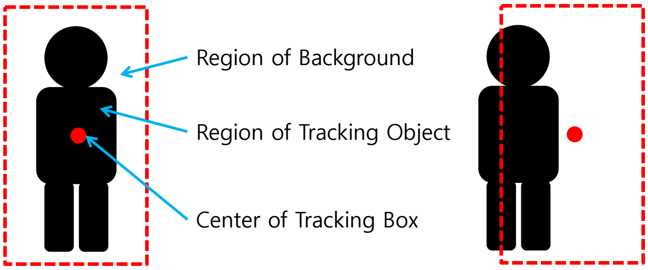

(a)
(b)
(c)
(d)

Figure 2.3: (a) Reference image (b) Scaled by 0.7 (c) Rotated 45 degree in clockwise (d) Translated (20, 20) pixels. Log-polar images (the second image) in (b) and (c) are almost invariant from (a), but that in (d) is largly varying.

teristic is approximated logarithmic-polar law [25] (Fig. 2.2(b)). The translational changes in Cartesian space tends to bring out bigger variations in log-polar images than rotational and scaling changes (see Fig. 2.3). Foveated targets occupy most of pixels in the log-polar image and the background elements are coarsly sampled. On the other hand, if the foveated point is placed in the background area, then background elements are densely sampled, while target object elements are sampled sparsely.

2.2.2 Tracking Failure in Log-Polar Image

Definition 2.2.1 (Tracking failure). *The tracking failure moment is defined as the moment when the center of tracking box (C_{TB}) is moved to background region*



(a) Success (b) Failure

Figure 2.4: The definition of tracking failure.

(R_B) from the region of tracking object (R_{TO}) (see Figure 2.4)..

In the view of C_{TB}, tracking failure appears like crossing over the boundary line between R_{TO} and R_B. It means that, under our definition, translational changes are more important than rotational and scaling changes in the tracking box images.

From the property of area differentiation ($rdrd\theta = dxdy, r = \sqrt{x^2 + y^2}$) and (2.2)(2.2), we derive below relationships.

$$d\rho d\theta = \frac{1}{r^2} dxdy, \tag{2.3}$$

$$d\rho d\theta = \frac{1}{x^2 + y^2} dxdy. \tag{2.4}$$

The C_{TB} corresponds to foveated point of eye. According to Definition 2.2.1, tracking failure appears when the C_{TB} crosses over the boundary line between R_{TO} and R_B. It means that, under our definition, translational changes are more important than rotational and scaling changes in the tracking box images.

In Fig. 2.5, we model the tracking failure situation. The result shows that log-polar transformed image intensifies the changes around C_{TB} and decrease

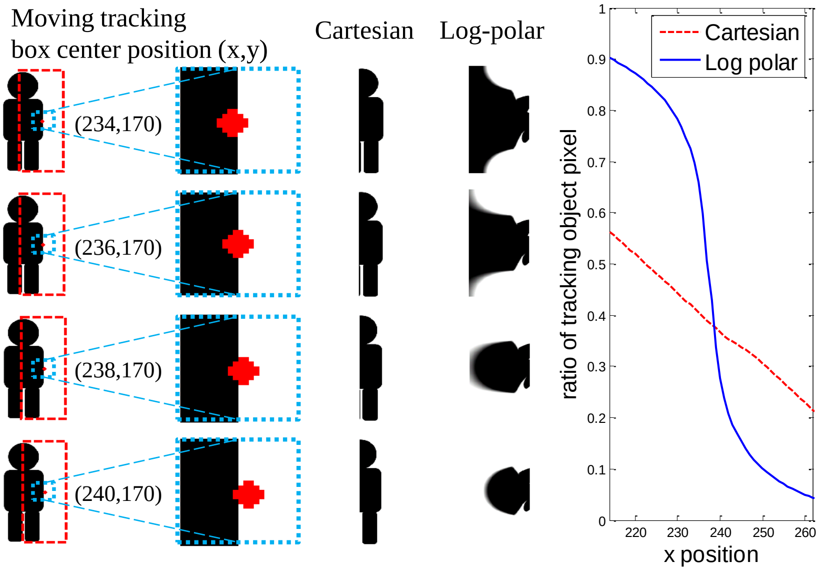


Figure 2.5: Moving a tracking box by two pixels, we check the shape and ratio of tracking object pixels in tracking box. The boundary crossing moment can be detected distinctively in log-polar space comparing to Cartesian space.

the changes of peripheries. This is induced by nonlinear predominance property of log-polar transformation and it helps to capture a boundary crossing moment and ignore other background changes. So, the two properties (*rotation and scaling invariance, and foveal predominance*) of log-polar image are effective for TFD.

2.3 Tracking Failure Detection Algorithm

2.3.1 Modeling of Ganglion Cell Adaptation

From the *biological plausibility* of log-polar image, each ganglion cell corresponds to each pixel of log-polar image. For dynamic modeling of pixels in tracking box image, we adopt the framework of online GMM method [6].

In tracking box image, majority of pixel values are varying as a target moves. For dynamic modeling of these pixels in tracking box image, we adopt the framework of online method of Stauffer et. al. [6]. At any time t, what is known about at a particular pixel (ρ,θ), is its history

$$\{X(1), ..., X(t)\} => \{I_{log-polar}(\rho, \theta, i) : 1 \leq i \leq t\} \tag{2.5}$$

where $I_{log-polar}$ is the image sequence of log-polar transformed images. $\{X(1), ..., X(t)\}$ is a sequence of log-polar transformed images $I_{log-polar}$ and each image is composed of N pixels $X(t) = \{X_1(t), ..., X_N(t)\}$. The history of pixel n is modeled by a mixture of K Gaussian distributions. The probability of observing a current pixel value is

$$P(X_n(t)) = \sum_{k=1}^{K} \omega_n^k(t) * \eta(X_n(t), \mu_n^k(t), \Sigma_n^k(t)) \tag{2.6}$$

where $\omega_n^k(t)$ is an weight, $\mu_n^k(t)$ is the mean value and $\Sigma_n^k(t)$ is the covariance matrix of each Gaussian in the mixture at time t respectively.

$$\eta(X_n(t), \mu_n^k(t), \Sigma_n^k(t)) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_n^k(t)|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_n(t)-\mu_n^k(t))^T \Sigma^{-1} (X_n(t)-\mu_n^k(t))}. \tag{2.7}$$

In this formulation, $\omega_n^k(t)$, $\mu_n^k(t)$, ${\sigma_n^k(t)}^2$ are updated by following equations as in [6].

$$\omega_n^k(t) = (1-\alpha)\omega_n^k(t-1) + \alpha M_n^k(t) \tag{2.8}$$

where α is a learning rate and $M_n^k(t)$ is 1 for a matched model and 0 for the others.

$$\begin{aligned} \mu_n^k(t) &= (1-\nu)\mu_n^k(t-1) + \nu X_n(t) && (2.9) \\ {\sigma_n^k(t)}^2 &= (1-\nu){\sigma_n^k(t-1)}^2 && (2.10) \\ &\quad +\nu(X_n(t) - \mu_n^k(t))^T (X_n(t) - \mu_n^k(t)), \end{aligned}$$

where ν is $\alpha\eta(X_n(t)|\mu_n^k(t), \sigma_n^k(t))$.

2.3.2 Initialization of GMM

When the tracking starts, we can get color information of tracking box from initial image. Because the tracking box usually wraps tracking object tightly and places C_{TB} in R_{TO}, most pixels within tracking box correspond to the portion of R_{TO}. Also the log-polar tranformation makes peripheral pixels less emphasized. Using the color information of tracking box we set an initial color model of tracking object. We set initial values of $\omega_{init}^k(1), \mu_{init}^k(1), \sigma_{init}^k(1)^2$ using the color information of initial tracking box image $X(1)$. Because we do not know how many colors the tracking object are composed of, mean shift clustering (MSC) [26, 27] method is used to find the number.

With N pixel points $\{X_1(1), ..., X_N(1)\} \in R^3$ (RGB color space), we find K clusters (K color distributions) by means of MSC. Each color distribution $C_{k(k=1...K)}$ is composed of $n_k(\sum_{k=1}^{K} n_k = N)$ pixel points $X_{i(i=1...n_k)}^k(1)$. We model each color distribution as Gaussian distribution and calculate initial parameter values with clustered pixel points. $\omega_{init}^k(1) = n_k/N$ is an weight, $\mu_{init}^k(1) = (\sum_{i=1}^{n_k} X_i^k(1))/n_k$ is the mean. $\Sigma_{init}^k(1) = \sigma_{init}^k(1)^2 I$ (each color space is independent and have the same variance $\sigma_{init}^k(1)^2 = (\sum_{i=1}^{n_k}(X_i^k(1) - \mu_{init}^k(1))^2)/n_k$) is the covariance matrix of each $k^{th}(k = 1...K)$ color distribution respectively. N pixels of $X(t)$ share the same initial values.

2.3.3 Tracking Failure Detection

Every new pixel is checked whether it belongs to existing one of K models and classified as familiar pixel or unseen pixel. The unseen pixel has high probability of being a part of occluding object or background which means out of target. This is similar to every ganglion cell's independent firing (stimulated by new color). New color perception is modeled as checking abruptly changing pixel (ACP). An ACP is defined as a pixel out of 2.5 standard deviations of a existing one of K models.

$$ACP_n(t) = \begin{cases} 0 & \text{if } (X_n(t) - \mu_n^k(t))^2 < 2.5\sigma_n^k(t)^2 \\ & \forall k = 1, ..., K, \\ 1 & \text{otherwise}\,. \end{cases} \tag{2.11}$$

Then, in order to detect tracking failure, ACP ratio ξ_{ACP} in current image $X(t)$ is measured by

$$\xi_{ACP} = \frac{\sum_{n=1}^{N} ACP_n}{N}. \tag{2.12}$$

Using ξ_{ACP}, tracking failure is determined by thresholding:

$$\chi_{TFD}(X(t)) = \begin{cases} 1 & \text{if } \xi_{ACP} > T, \\ 0 & \text{otherwise}\,. \end{cases} \tag{2.13}$$

where T is a threshold value, experimentally defined. Figure 2.6 shows the overall scheme of our proposing method.

2.4 Experimental Results

To evaluate the validity of our TFD algorithm, we conducted some experiments. We implemented our algorithm in MATLAB for simulation with a threshold $T = 0.4$.

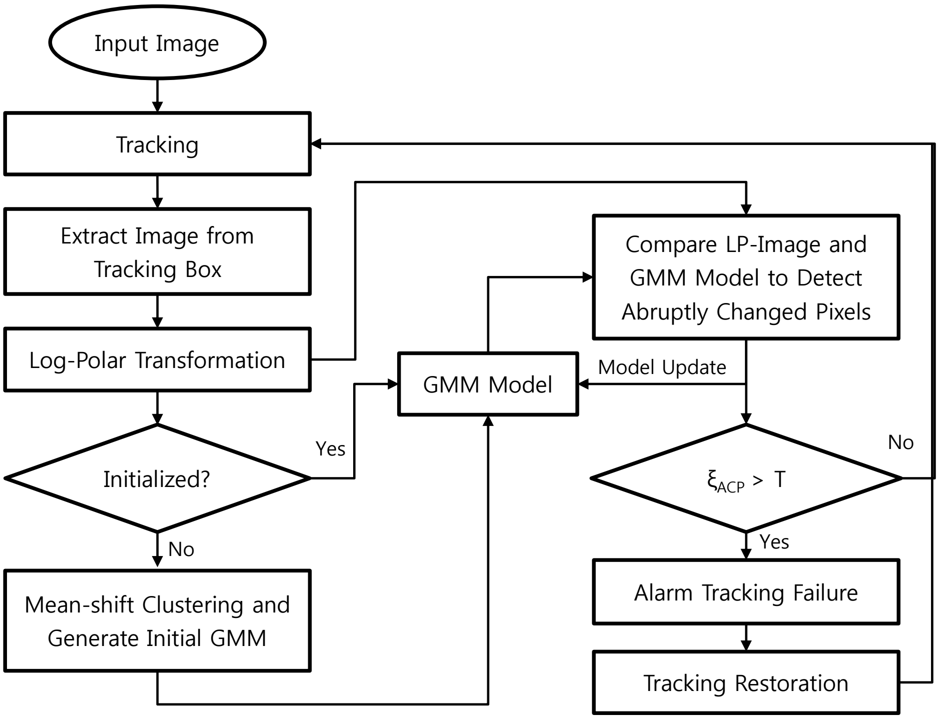
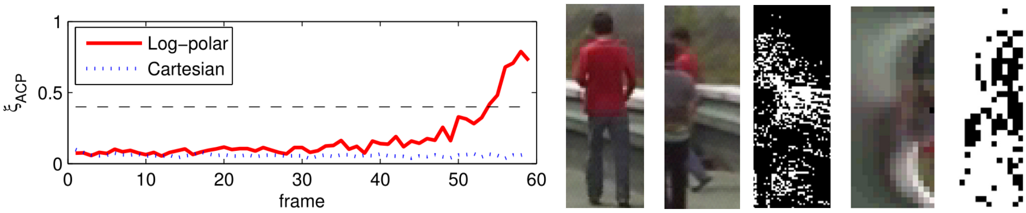


Figure 2.6: Overall TFD algorithm flow chart



(a) (b) (c) (d)

Figure 2.7: (a) shows the ξ_{ACP} comparison. Occlusion occurs as frame 56. (b) is the tracking object image of frame 1. (c) and (d) are images of frame 58 and its ACP image in Cartesian space and log-polar space respectively.

2.4.1 Effectiveness of Log-Polar Transformation and Initial Color Model Generation

We verify our claim that log-polar space is suitable for TFD than cartesian space. Fig. 2.7 shows a comparison between the ACP detection in two different spaces, cartesian space and log-polar space. As we can see in Fig. 2.7(a), the change around C_{TB} is magnified in log-polar space. Also, while the tracking box size in the cartesian space is 110x40, that in the log-polar transformed image is 36x15. So using log-polar space can reduce the computation load 8 times less with better performance. Fig. 2.8 shows the effect of setting initial color model. There are several inner boundaries in R_{TO} which induce false alarms. By setting initial color model for GMM, we could achieve to give less alarms for inner boundaries.

To evaluate the performance of the proposed algorithm, we compare the TFD accuracy with K-means Tracker TFD [5] (Because [5] is a tracker independent TFD measure based method same as ours). As we can see in Fig. 2.9, our method can afford to occlusion and scale changes not giving false alarm until tracker really misses the target.

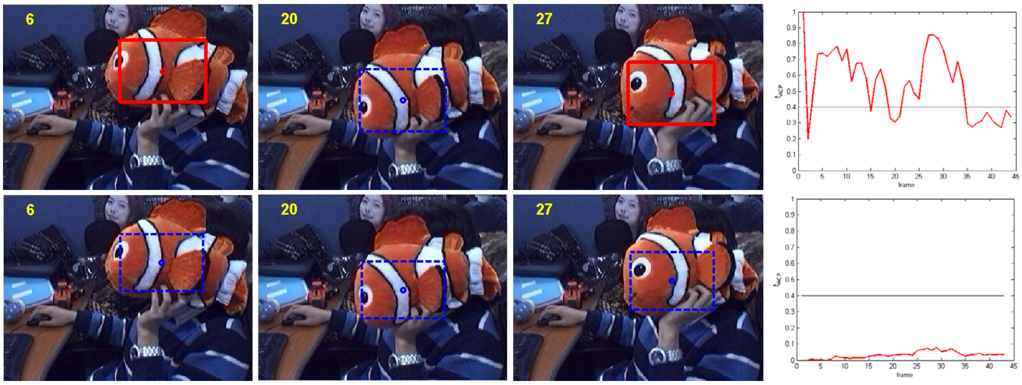
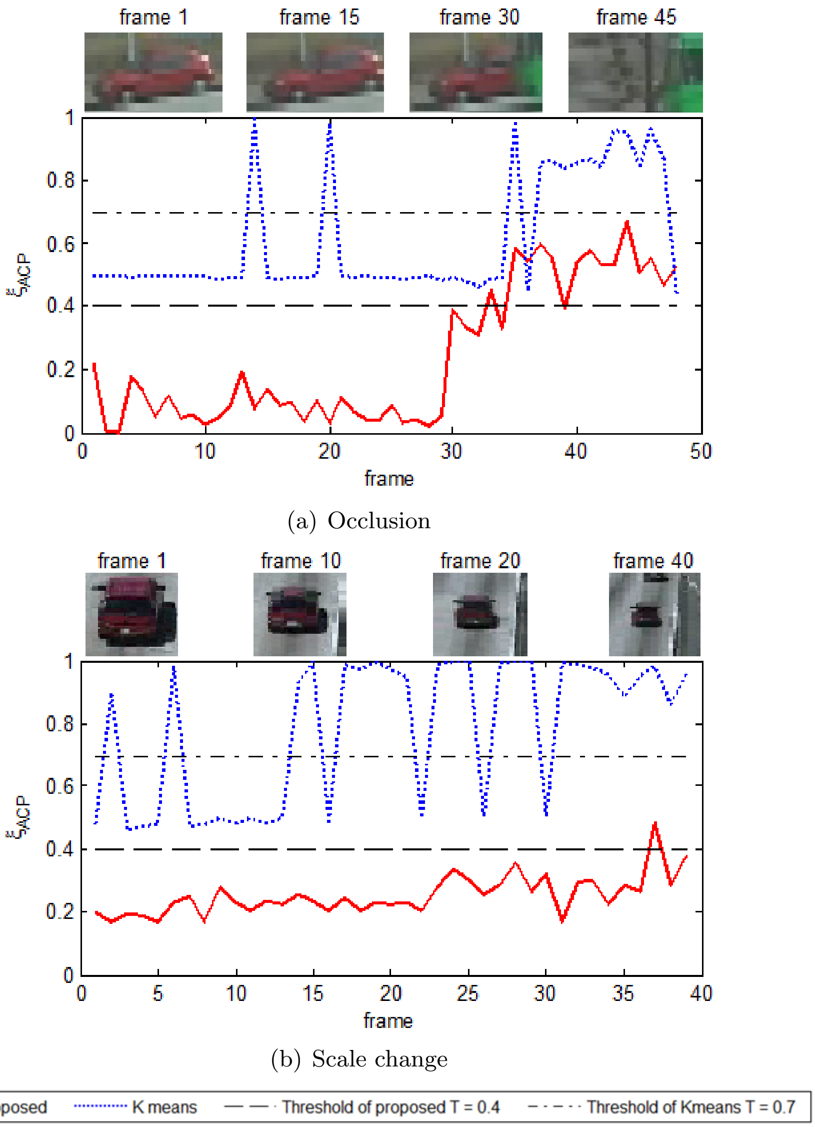


Figure 2.8: The first row represents a TFD result without initial color model generation and the second row is a TFD result by using initial color model.

2.4.2 Combining with various tracking algorithms

The proposed TFD method can be applied to any tracking algorithm. Fig. 2.10 shows combined TFD results with different tracking methods, kernel based tracking [28] and particle filter tracking [29]. Because our method evaluates current tracking status not by an implicit similarity measure of tracker but by an explicit tracking result image (which is analogous to the way of people make a decision), we can see that our TFD method can be successfully combined with any kinds of tracking algorithm.

The proposed method also can be used for enhancing tracking performance by feedback. Fig. 2.11 shows tracking performances of IVT tracker [29] measured by root mean square error (RMSE) comparing to ground truth. When tracking failure measure increases, TFD makes the tracker to stop updating tracking template models and widen particle spreading range. This feedback helps tracking algorithm more robust.



(a) Occlusion

(b) Scale change

Figure 2.9: K-means TFD gives an alarm when the score is over 0.7 (This value is from [5]), and the proposed TFD gives alarm when the score is over 0.4. In ground truth, tracking fail occurred at frames 35 in (a) because of occlusion and frames 42 in (b) because of target becomes too small.

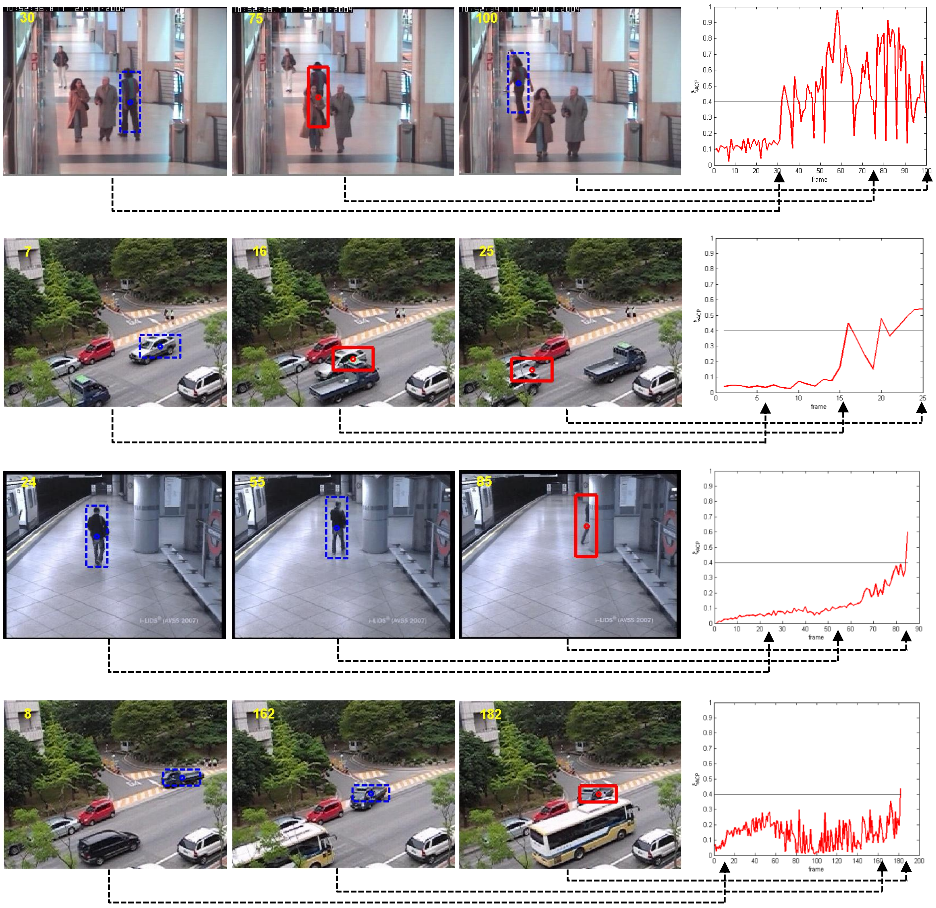


Figure 2.10: The first and second rows show TFD results combined with kernel based tracking [28] and the third and fourth rows are TFD results of particle filter based tracking [29].

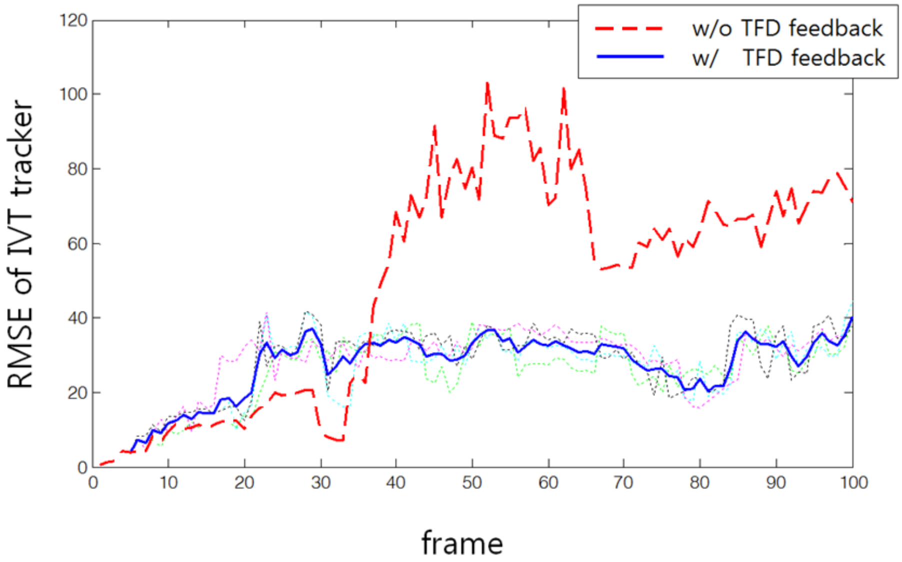


Figure 2.11: Feedback of TFD enhances tracking performance. The blue and red bold lines are averaged values of 10 results. Even the GMM initialization has random factor of MSC, many experimental results show that the TFD can improve the performance.

2.5 Final Remarks and Discussion

In this chapter, we presented a tracking failure detection method using the structured attentional sampling. Based on the knowledge of tracking and human visual system, we designed the log-polar sampling as the sampling pattern of attentional sampling. By adopting log-polar sampling for modeling retina image, we could intensify the translational change. This property makes it possible to detect tracking failure moment easily. We modeled ganglion cell adaptation using online GMM and detected abrupt change in pixels. Experimental results show that the properly designed structured attentional sampling can gives less false alarms with less samples, and can be applied to any tracking methods.

Chapter 3

Empirical Attentional Sampling for Speed-up of Background Subtraction

3.1 Introduction

The empirical attentional sampling scenario allows the sample location to be chosen using the information collected up to that point[2]. So the sampling becomes adaptive and flexible[21]. However, a prior information about the dependency between samples and labels are necessary to design the sampling strategy. In this chapter, a designing of an empirical attentional sampling strategy for background subtraction algorithm is presented.

Background subtraction is a process which aims to segment moving foreground objects from a relatively stationary background[30]. Recently pixel-based probabilistic model methods [7, 8, 6, 9, 10, 11] gained lots of interests and have shown good detection results. There have been many improvements in detection

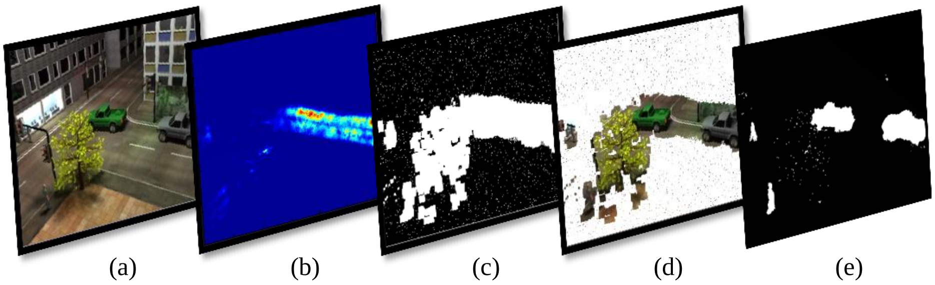


Figure 3.1: Background subtraction by active attentional sampling mask. (a) Input video image (b) Foreground probability map (c) Active attentional sampling mask (d) Sampled pixels (e) Foreground detection result

performance for these methods under various situations, but the computational time still takes too much time. Computation time reduction issue is getting more important in a systematic view, because the background subtraction is generally considered as a low level image processing task, which needs to be done with little computation, and video sizes are getting bigger.

To reduce computation time of background subtraction methods, several approaches have been studied. The first type of approach is based on optimizing algorithms. Although the Gaussian mixture model (GMM) scheme proposed by Stauffer and Grimson[6] works well for various environments, it suffers from slow learning rates and heavy computational load for each frame[11]. Lee [31] makes the convergence fast by using a modified schedule that gradually switches between two stage learning schemes. Zivkovic[9] achieved a significant speed-up by formulating a Bayesian approach to select the required number of Gaussian modes for each pixel in the scene. Gorur[11] modified Zivkovic's method[9] by windowed weight update that minimizes floating point computations. However this optimization approach is hard to be generalized to all background subtraction methods. Also the speed-up ratio is not enough for real-time computation in full HD videos.

The second type of approach is using parallel computation. Multi-core processors in a parallel form, using the OpenMP system are applied for speed-up[32]. Also Graphical Processing Units (GPUs) are used to achieve real-time performance[33] with computationally heavy algorithms. Pham *et al.*[12] perform real time detection even in full HD video using GPU. Until now, allegedly, using GPUs is the only way to perform background subtraction of full HD video in real time[12]. They could successfully achieve speed-up, but special hardware resources are required.

A selective sampling based speed-up method is the third type of approach. Park *et al.*[13] proposed a hierarchical quad-tree structure to decompose an input image. A randomly sampled pixel, which is a node of the tree, is classified as background or foreground. The corresponding node is divided into four child nodes if it is foreground, and then the sampling procedure is carried out recursively. Using the image decomposition, they could achieve the computational complexity reduction. However, their algorithm may miss small objects because they randomly sample from a relatively large region. Kim *et al.*[34] presented a sampling mask designing method which can be readily applied to many existing object detection algorithms. Lee *et al.*[14] proposed a two-level pixel sampling method. They coarsely sampled pixels according to a regularly designed pattern and then it refines the shapes of foreground objects. Their algorithm provides accurate segmentation results without flickering artifacts. Kim *et al.*[34] and Lee *et al.*[14] use compactly designed grid pattern masks to detect small objects, but these grid patterns still cause redundant operations.

In this chapter, we propose a new method of the third type of approach (sampling mask approach) which can be utilized together with the other two approaches. We aim to find an active attentional sampling solution which can be generally applied to most conventional background subtraction methods. We design a foreground probability map based on temporal, spatial and frequency properties of the foreground region. Using previous foreground detection result, the foreground probability map is updated. A sequential coarse-to-fine approach, which involves sparse random sampling and filling in a space in attentional region according to the probability map, achieves a very significant reduction in computation time without degrading the detection performance. Figure 3.1 illustrates the process of the proposed algorithm. By combining with conventional background subtraction methods, our method makes these methods even be able to handle full HD videos in real-time.

3.2 Overview

3.2.1 Motivation

We imitate the selective attention mechanism of human[35], where previously recognized results are reflected in the focusing position of current frame. When a guard monitors a CCTV camera, he/she does not concentrate on whole of the image since he/she has empirically learned that the video image can be categorized into background region, unimportant dynamic scene region and important moving object apprearing region. Then he/she takes his/her attention to the regions which have moving object appearing intentionally and does a sparse scanning to the other regions such as background or dynamic region. The key idea of proposed approach is to simulate this selective attention scheme.

In general, most pixels from surveillance video are background region, and foreground region takes very small portion in both spatially and temporally. We have measured a percentage of the foreground area of commonly used data set in

Data Set	# of tested frames	Mean(%)	Std.
Wallflower	7553	5.03	6.25
VSSN2006	16074	2.30	1.13
PETS2006	41194	1.04	0.26
AVSS2007	33000	3.36	1.02
PETS2009	2581	5.48	1.58
SABS	6400	2.42	1.83
	Average	**2.42**	**1.18**

Table 3.1: Statistical foreground region ratio of several widely used datasets. Only 2.42% of total pixels are foreground pixels.

background subtraction papers. The tested data sets are Wallflower[1], VSSN2006[2], PETS2006[3], AVSS2007 i-LIDS challenge[4], PETS2009[5] and SABS[36][6]. As we can see in Table3.1, the proportions of foreground regions are very small. Hence, if background substraction can be focused on foreground area, necessary calculation would be reduced significantly. In this chapter we try to find attentional region in a current frame considering foreground region detected in a previous frame.

3.2.2 Overall Scheme of Proposed Algorithm

Figure 3.2 shows the overall scheme of the proposed method. To get active sampling mask for background substraction, we use three properties of foreground; temporal, spatial, frequency properties. The temporal property is that a pixel is more likely to be a part of the foreground region if it has been a foreground pixel previously. The spatial property is that a pixel has a high probability of being a foreground pixel if its surrounding pixels are of the foreground. The probability is

[1] http://research.microsoft.com/~jckrumm/wallflower/testimages.htm

[2] http://mmc36.informatik.uni-augsburg.de/VSSN06$_$OSAC

[3] http://www.cvg.rdg.ac.uk/PETS2006/data.html

[4] http://www.eecs.qmul.ac.uk/~andrea/avss2007$_$ss$_$challenge.html

[5] http://www.cvg.rdg.ac.uk/PETS2009/a.html

[6] http://www.vis.uni-stuttgart.de/index.php?id=sabs

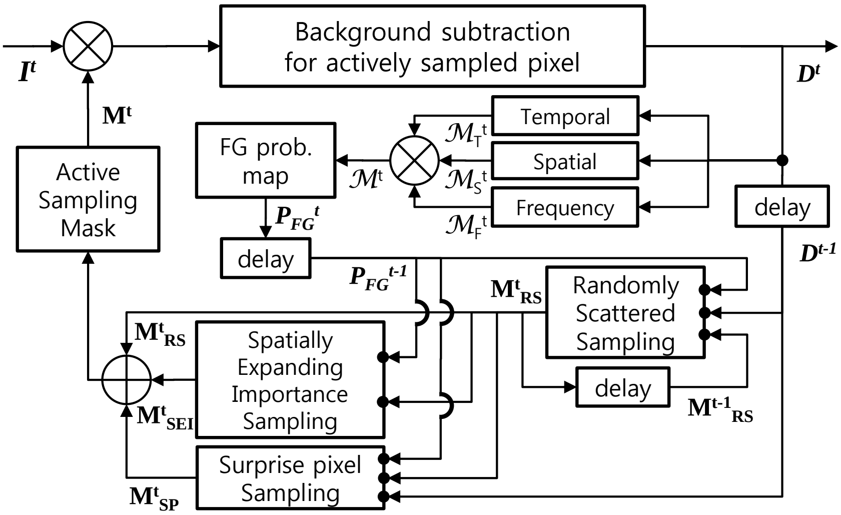


Figure 3.2: Overall scheme of the proposed algorithm.

proportional to the number of surrounding foreground pixels. This spatial ergodic property was also used in [37][30] for background modeling. The frequency property is that if foreground/background index of a pixel is changed too frequently, then the pixel is more likely to be a noise or dynamic background region. So the probability of being a stable foreground region is low. Based on the properties, we make a foreground probability map P_{FG} (described in Section 3.3).

The active sampling strategy is updated in every frame according to the foreground probability map $(P_{FG})^{t-1}$. The strategy is composed of three sampling strategies such as *randomly scattered sampling*, *spatially expanding importance sampling*, and *surprise pixel sampling*, which are performed sequentially. We make the sampling mask $\mathbf{M}^t$ at every frame (described in Section 3.4). Using sampling mask $\mathbf{M}^t$, selective pixel-wise background subtraction is performed, only for the pixels of $\mathbf{M}^t(n) = 1$ where n indicates the pixel index. This sampling mask can be combined with any kind of pixel-wise background subtraction methods.

In addition, newly updated foreground probability map $(P_{FG})^t$ in the current frame is also used to refine the detection result. Detected pixels of low probability region are filtered out except surprise pixels. This refining step reduces many false alarms caused by dynmic background movements (regular movements such as tree waving, fountain and water ripples) and small noises.

The background subtraction task finds a sequence of detection masks $\{D^1, ..., D^T\}$ using a sequence of video frames $\{I^1, ..., I^T\}$ and sampling mask $\{\mathbf{M}^1, ..., \mathbf{M}^T\}$. Each video image I^t, sampling mask $\mathbf{M}^t$ and detection mask D^t are composed of N pixels $\{I^t(1), ..., I^t(N)\}$, $\{\mathbf{M}^t(1), ..., \mathbf{M}^t(N)\}$ and $\{D^t(1), ..., D^t(N)\}$ respectively. All the masks are binary masks. In this chapter, selective pixel-wise background subtraction is performed, only for the pixels of $\mathbf{M}^t(n) = 1$. The detection mask at pixel n shall be denoted with the symbol $D(n)$: $D(n) = 0$ if pixel n belongs to the background and $D(n) = 1$ if it belongs to the foreground.

There are several empirical and theoretical results suggesting that use of data collected in early stages can be significantly more efficient to guide the selection of new samples [21, 22]. Conventional background subtraction algorithms are based on passive sampling. The collection of sample points is chosen independent to the labels, and a prior probability distribution of foreground is assumed uniform. So, in order to detect unexpected foreground, the sampling becomes a full search regardless of previous observations.

On the other hand, the active sampling scenario allows the sample location to be chosen using the information collected up to that point[2]. So the sampling becomes adaptive and flexible[21]. However, a prior information about the dependency between samples and labels are necessary to design the sampling strategy. In the following sections, we describe a way how to design the sampling strategy using the properties of attentional foreground region.

3.3 Foreground Probability Map Generation

3.3.1 Estimation of Foreground Properties

Estimation models are proposed to measure the temporal, spatial, and frequency properties of each pixel. The three property measures are referred to as $\{\mathcal{M}_T, \mathcal{M}_S$, and $\mathcal{M}_F\}$. The temporal property measure $\mathcal{M}_T$ is estimated by the recent history of detection results. The spatial property $\mathcal{M}_S$ is estimated by the number of foreground pixels around each pixel. The frequency property $\mathcal{M}_F$ is estimated by the ratio of detection result flipping over a period of time. All estimation models are updated by a running average method, with learning rates α_T, α_F and α_S (all learning rates are between 0 and 1). The estimation models for the measures of the properties are given in the following.

- **Temporal property** $\mathcal{M}_T$: At each location n, a recent history of detection mask results at that location are averaged to estimate the property.

  $$\mathcal{M}_T^t(n) = (1 - \alpha_T)\mathcal{M}_T^{t-1}(n) + \alpha_T D^t(n). \tag{3.1}$$

  As the value of $\mathcal{M}_T^t(n)$ comes close to 1, the possibility of foreground appearance at the pixel is high.

- **Spatial property** $\mathcal{M}_S$: Detection results of nearby pixels are used to measure the spatial coherency of each pixel n.

  $$\mathcal{M}_S^t(n) = (1 - \alpha_S)\mathcal{M}_S^{t-1}(n) + \alpha_S s^t(n), \tag{3.2}$$
  $$(s^t(n) = \frac{1}{w^2} \sum_{i \in \mathcal{N}(n)} D^t(i)),$$

  where $\mathcal{N}(n)$ denotes a spatial neighborhood around pixel n ($w \times w$ square region centered at n). $\mathcal{M}_S^t(n)$ closer to 1 means high probability of being a part of the foreground.

- **Frequency property** $\mathcal{M}_F$: If detection results have been changed twice during previous three frames, we consider it as a clue of dynamic scene.

$$\begin{aligned} \mathcal{M}_F^t(n) &= (1-\alpha_F)\mathcal{M}_F^{t-1}(n) + \alpha_F f^t(n), \\ f^t(n) &= \begin{cases} 1 & (D^{t-2}(n) \neq D^{t-1}(n)) \\ & \&(D^{t-1}(n) \neq D^t(n)) \\ 0 & \text{otherwise}\,. \end{cases} \end{aligned} \tag{3.3}$$

where $f^t(n)$ denotes a frequently changing property at n. Unlike the other measures, the pixel n has a high probability of being a foreground, as the value $\mathcal{M}_F^t(n)$ is close to 0.

3.3.2 Foreground Probability Map: P_{FG}

Background detection considers only the background model, and the foreground probability map is usually considered to be uniform, which means no prior shape of orientation of the foreground is explicitly assumed. Our aim is to replace the naive, uniform foreground probability density model with a more accurate estimate based on the previous detection results and the three foreground properties. By estimating the three foreground properties, we get the three measurements, $\mathcal{M}_T, \mathcal{M}_S$, and $\mathcal{M}_F$. Every measurement has a value between 0 and 1. So we define the foreground probability for a pixel n at frame t as

$$P_{FG}^t(n) = \mathcal{M}_T^t(n) \times \mathcal{M}_S^t(n) \times (1 - \mathcal{M}_F^t(n)). \tag{3.4}$$

The foreground probability map P_{FG}^t is a composition of $\{P_{FG}^t(n)\}_{n=1}^N$.

3.4 Active Sampling Mask Generation

The sampling mask $\mathbf{M}^t$ is obtained by a combination of three masks by a pixel-wise 'OR' operation ($\oplus$) as

$$\mathbf{M}^t = \mathbf{M}^t_{RS} \oplus \mathbf{M}^t_{SEI} \oplus \mathbf{M}^t_{SP}, \tag{3.5}$$

where $\mathbf{M}^t_{RS}$, $\mathbf{M}^t_{SEI}$ and $\mathbf{M}^t_{SP}$ are sampling masks of *randomly scattered sampling ($\mathbf{S}_{RS}$)*, *spatially expanding importance sampling ($\mathbf{S}_{SEI}$)* and *surprise pixel sampling ($\mathbf{S}_{SP}$)* respectively.

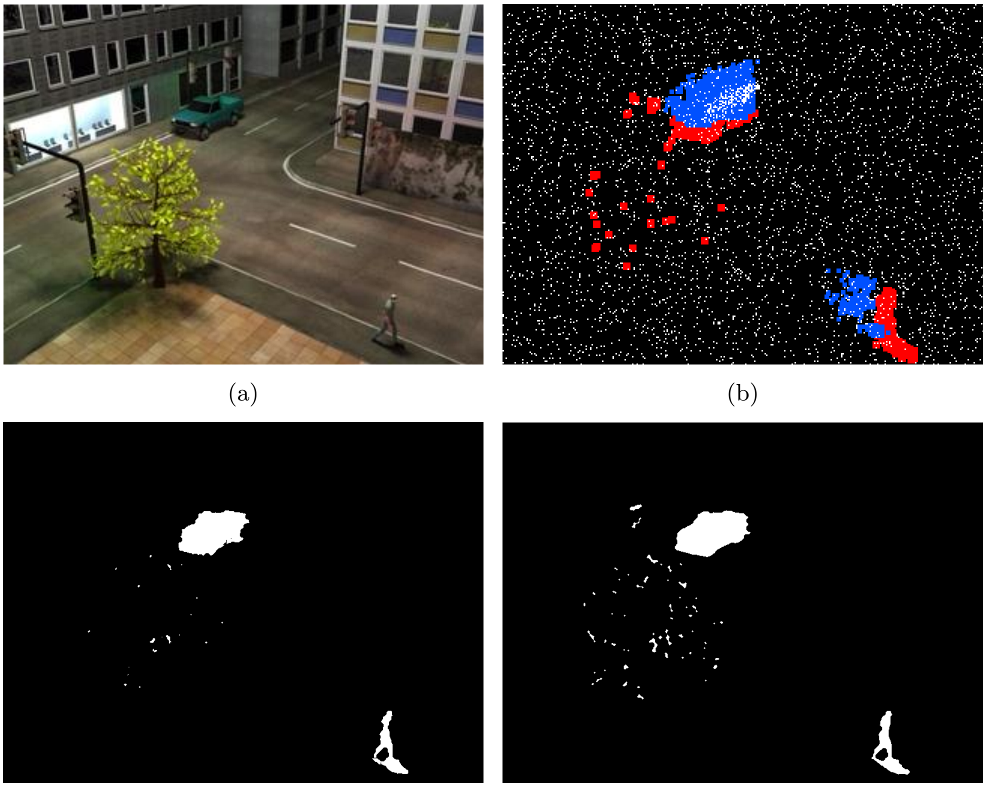
At each sampling stage, the sampling masks are generated based on the foreground probability map P_{FG} and foreground detection result D. We design the active sampling strategies as

$$\begin{aligned}
\mathbf{M}^t_{RS} &= \mathbf{S}^t_{RS}(\mathbf{M}^{t-1}_{RS}, D^{t-1}, P^{t-1}_{FG}), && (3.6)\\
\mathbf{M}^t_{SEI} &= \mathbf{S}^t_{SEI}(\mathbf{M}^t_{RS}, P^{t-1}_{FG}), && (3.7)\\
\mathbf{M}^t_{SP} &= \mathbf{S}^t_{SP}(\mathbf{M}^t_{RS}, D^{t-1}, P^{t-1}_{FG}). && (3.8)
\end{aligned}$$

Figure 3.4 shows the foregorund property measurements, corresponding sampling mask $\mathbf{M}^t$ and foreground detection results with and without $\mathbf{M}^t$. In the following, we describe the details on the sampling strategies in (3.6), (3.7), and (3.8).

3.4.1 Randomly Scattered Sampling

First, $100 \times \rho\%$ (usually ρ 0.05 to 0.1) pixels of the entire pixels are selected through randomly scattered sampling. Uniform random sampling approximates that every pixel is checked probabilistically on average once among $1/\rho$ frames.



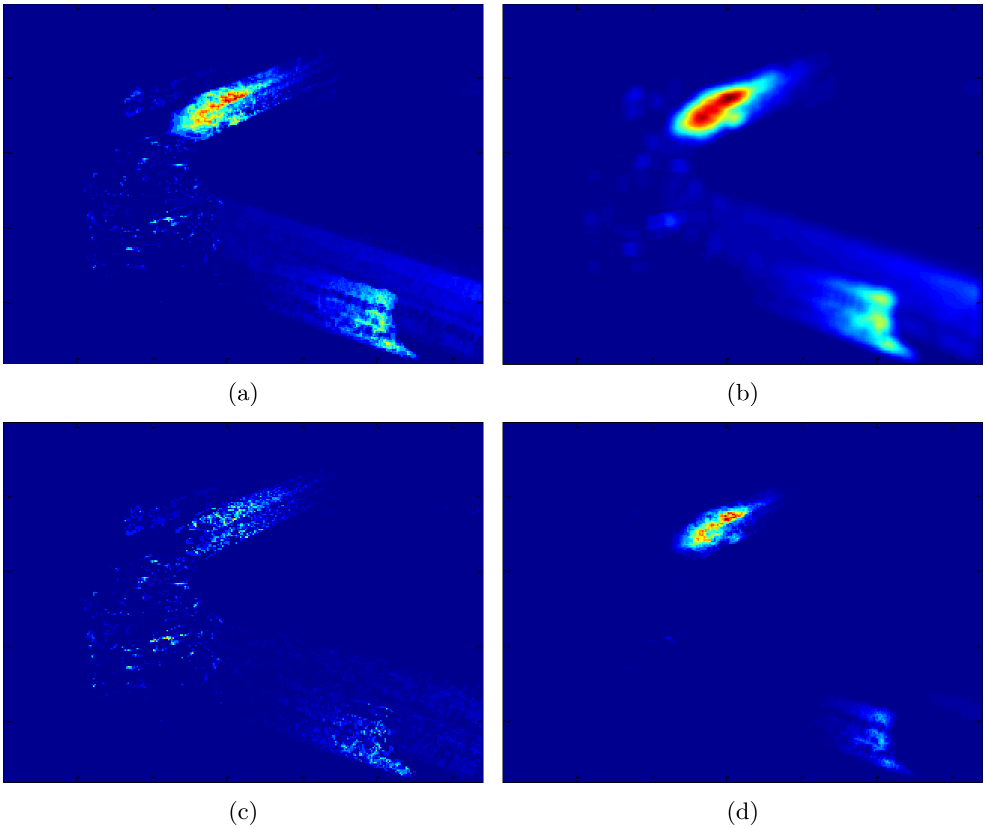
(a)

(b)

(c)

(d)

Figure 3.3: Active attentional mask generation. (a) is a current input video image. (b) shows the active attentional mask used for background subtraction. The white points are randomly scattered sampling mask $\mathbf{M}^t_{RS}$. The blue pixels represent $\mathbf{M}^t_{SEI}$ and the red regions are $\mathbf{M}^t_{SP}$. As we can see in (b), most of mask $\mathbf{M}^t$ become zeros. The mask, whose redundancy is removed, optimizes the necessary computational load. (c) Foreground detection result by GMM method [6] with the active mask. (d) Foreground detection result by GMM method [6] without the mask.



(a)

(b)

(c)

(d)

Figure 3.4: Foreground probability map generation. (a) Temporal property $\mathcal{M}_T^t$. (b) Spatial property $\mathcal{M}_S^t$. (gc) Frequency property $\mathcal{M}_F^t$. (d) Foreground probability map P_{FG}^t

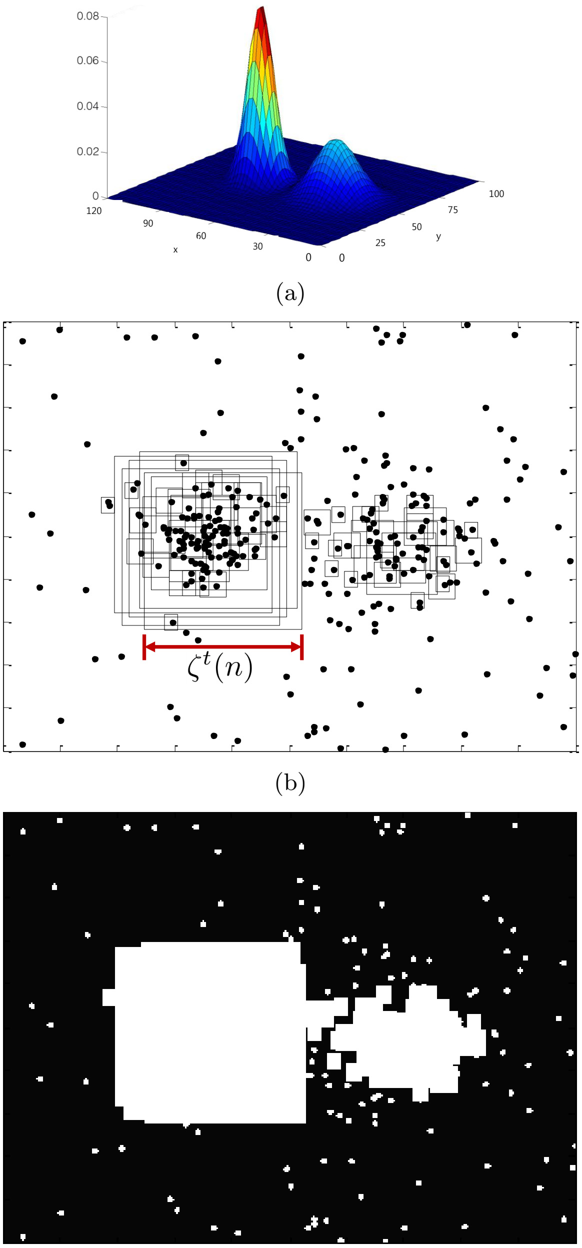
The number of random samples N_s is ρN. This number is constant for all frames. However, some of the random points generated in the previous frames are worth to be preserved. The determination of these points are based on the amount of information measured by the foreground probability P_{FG}^{t-1}. A sample point n which was $\mathbf{M}_{RS}^{t-1}(n) = D^{t-1}(n) = 1$ is used again in current frame($\mathbf{M}_{RS}^{t}(n) = 1$). Therefore, the number of reused samples N_{reuse} changes adaptively. Then, $N_s - N_{reuse}$ samples are resampled randomly across the entire image.

3.4.2 Spatially Expanding Importance Sampling

The randomly sampled mask $\mathbf{M}_{RS}^{t}$ is too sparse to construct a complete foreground region and might miss small objects. It is therefore necessary to fill the space between sparse points in the foreground region. In order to fill the space, we develop an appropriate importance sampling solution focusing only on necessary region compactly.

Conventional importance sampling[38] draws samples densely where the importance weight is high. In our case, the sampling mask should cover all of the foreground pixels and so the dense sampling is not enough in the foreground region. To solve this full coverage sampling problem, we propose a *spatially expanding importance sampling* method which expands the sampling area proportional to the importance weight at every point of $\mathbf{M}_{RS}^{t} = 1$ as shown in Figure 3.5. The shape of the expanded region is a square with width of ζ^t which depends on the importance weight at the i^{th} randomly scattered sample. Even though the square regions are overlapped, they are depicted by one region with $\mathbf{M}_{SEI}^{t} = 1$ as shown in Figure 3.5.

If the proposal distribution is assumed as an uniform distribution, importance weight of each randomly scattered sample i (where $\mathbf{M}_{RS}^{t}(i) = 1$) becomes $r^t(i) = P_{FG}^{t}(i)$. Proportional to $r^t(i)$, we expand the sampling region $\mathcal{N}(i)$ with size of



(a)

(b)

(c)

Figure 3.5: Spatially expanding importance sampling mask $\mathbf{M}_{SEI}$ generation by foreground probability map P_{FG}. (a) is P_{FG}. (b) For each point of M_{RS}, the spatially expanding region width ζ_s is calculated. (c) The mask $\mathbf{M}_{SEI}$ is generated by setting all the inside points of the square to 1.

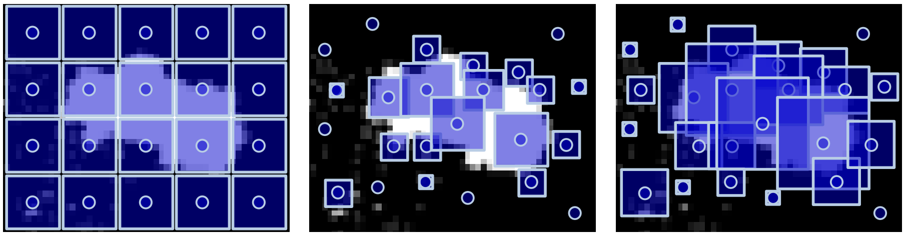


Figure 3.6: The effect of the parameter k. (a) $r^t = 1, k = 1$. (b) $r^t = P^t_{FG}, k = 1$. (c) $r^t = P^t_{FG}, k = \sqrt{3}$.

$\zeta^t(i) \times \zeta^t(i)$ centered at pixel i, i.e.

$$\mathbf{M}^t_{SEI}(\mathcal{N}(i)) = 1. \tag{3.9}$$

The spatially expanding width $\zeta^t(i)$ is determined as

$$\zeta^t(i) = round(r^t(i) \times \omega_s), \tag{3.10}$$

$$\omega_s = k\sqrt{N/N_s}. \tag{3.11}$$

ω_s is an expanding constant with parameter k (usually k is $\sqrt{3}$ or $\sqrt{5}$). Figure 3.6 shows how ω_s is designed and the effect of the parameter k. As shown in Figure 3.6(a), the ω_s with $k = 1$ and $r^t = 1$ implies a width of one square under an assumption that the image is equally decomposed into N_s squares centered at regularly distributed N_s samples. However, in actual situation, the N_s samples are not distributed regularly and most of r^t are less than 1. So the sampling mask $\mathbf{M}^t_{SEI}$ can not cover the estimated foreground region compactly as shown in Figure 3.6(b). The parameter k (larger than 1) expands the sampling masks so that the masks cover the foreground region compactly (Figure 3.6(c)). As we can see in Figure 3.3(b), high foreground probability regions are widely sampled and most of $\zeta^t(n)$ are 0 in low probability region.

3.4.3 Surprise Pixel Sampling Mask

Even if we estimate the foreground probability correctly, the foreground detection still has unpredictability intrinsically. Abnormal foreground is caused by spontaneousness. For example, a person or a car suddenly appears from a new direction, or a thief enters into a restricted area. These surprisingly appearing moving objects should be detected successfully. In addition, rarely appearing very fast moving objects could be lost, because the spatially expanded region may not be wide enough.

The randomly scattered samples become important when capturing these unpredictable cases. A pixel is defined as a *surprise pixel* if it was foreground in the previous frame even though its foreground probability is small. Because the foreground object is not expected to exist there, the observation of foreground pixel is very surprising. So by widening the sampling area around the pixel in a current frame can find new foreground pixels. For pixel i (where $\mathbf{M}_{RS}^{t}(i) = 1$), the *surprise pixel* index $\xi^t(i)$ is given by

$$\xi^t(i) = \begin{cases} 1 & (P_{FG}^{t-1}(i) < \theta_{th}^{t-1}) \& (D^{t-1}(i) = 1) \\ 0 & \text{otherwise}\,. \end{cases} \tag{3.12}$$

where $\theta_{th}^{t-1} = max(P_{FG}^{t-1}/\omega_s)$. Surprise pixel sampling mask is generated as $\mathbf{M}_{SP}^{t}(\mathcal{N}(i)) = 1$ for $\mathcal{N}(i)$ region ($\omega_s \times \omega_s$ region centered at i if $\xi^t(i) = 1$).

3.5 Computational Efficiency Boundary

We calculate a computational efficiency of the proposed method (C_P) comparing to the conventional full search method (C_F). α and α_{std} imply an average ratio (from 0 to 1) of foreground pixels and its standard deviation in a video, respectively. β is a computational complexity ratio of each computation block (such as $P_{FG}, \mathbf{M}^t_{RS}, \mathbf{M}^t_{SEI}, \mathbf{M}^t_{SP}$ generation) of proposed method comparing to original detection method. β_{max} and β_{min} are the largest and smallest value, respectively. Other parameters (ρ and k) are described above. The detailed derivation of the efficiency boundary is given in Appendix A of this thesis. The derived efficiency boundary is given

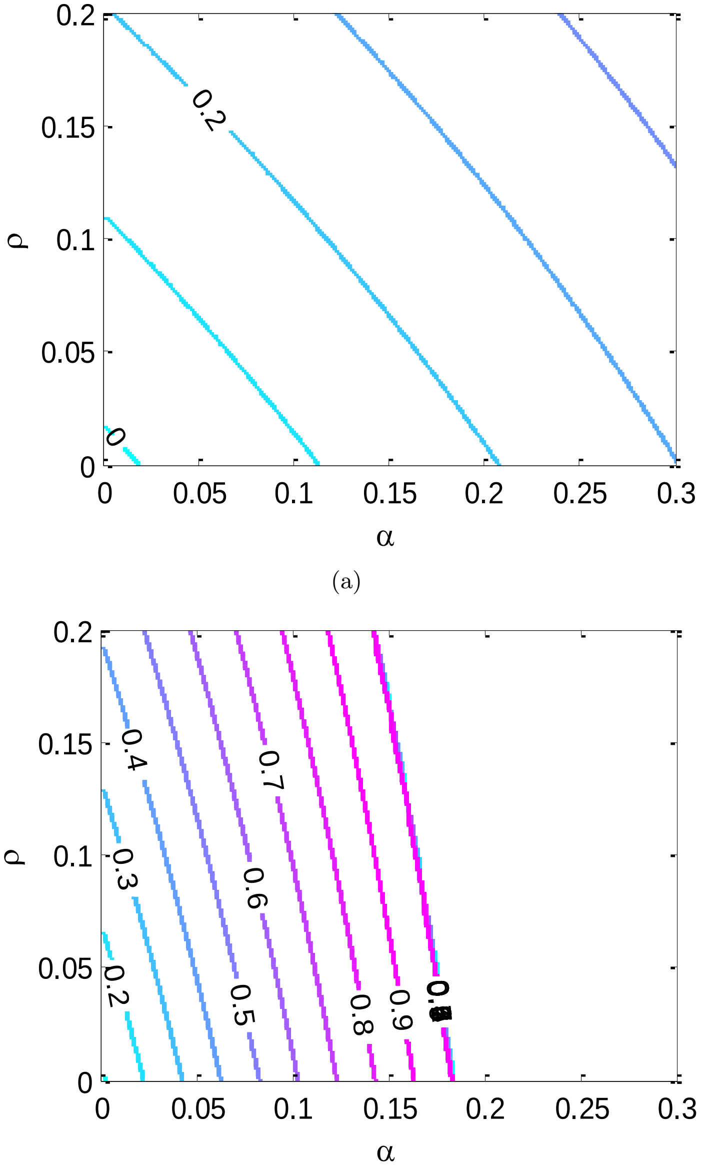
$$\begin{aligned}&(\alpha - \alpha_{std}) \left\{\beta_{min} + (1-\rho)(1+\beta_{min})\right\} + \rho(1+2\beta_{min}) \\ &< \frac{C_P}{C_F} < (\alpha + \alpha_{std})k^2 \left\{\beta_{max} + (1-\rho)(1+\beta_{max})\right\} + \rho(1+2\beta_{max}).\end{aligned} \tag{3.13}$$

Figure 3.7 is a simulated result of the efficiency boundary. We have validated the analysis result (3.13) through actual experimental values. In our implementation GMM[6] method and SABS dataset [36] is used with $\beta_{min} = 0.03, \beta_{max} = 0.33, k = \sqrt{3}$ and $\rho = 0.05$. In this case, actual C_P/C_F is 0.25 which is between lower bound (0.06) and upper bound (0.29) of analysis (3.13). Also we verified that actually measured computational efficiency is placed in the middle of the derived lower bound and upper bound as shown in Figure 3.8.

3.6 Experimental Results

We evaluated the performance of the proposed method on several video sequences of various resolutions and situations to prove its practical applicability. The results are compared to the existing background subtraction methods such as GMM[6], KDE[8], efficient GMM[31], shadow GMM[10], Zivkovic[9][7], and Gorur[11].

[7] implementation from author: `www.zoranz.net`



(a)

(b)

Figure 3.7: Derived efficiency bound of C_P/C_F. (a) is a lower bound and (b) is a upper bound.

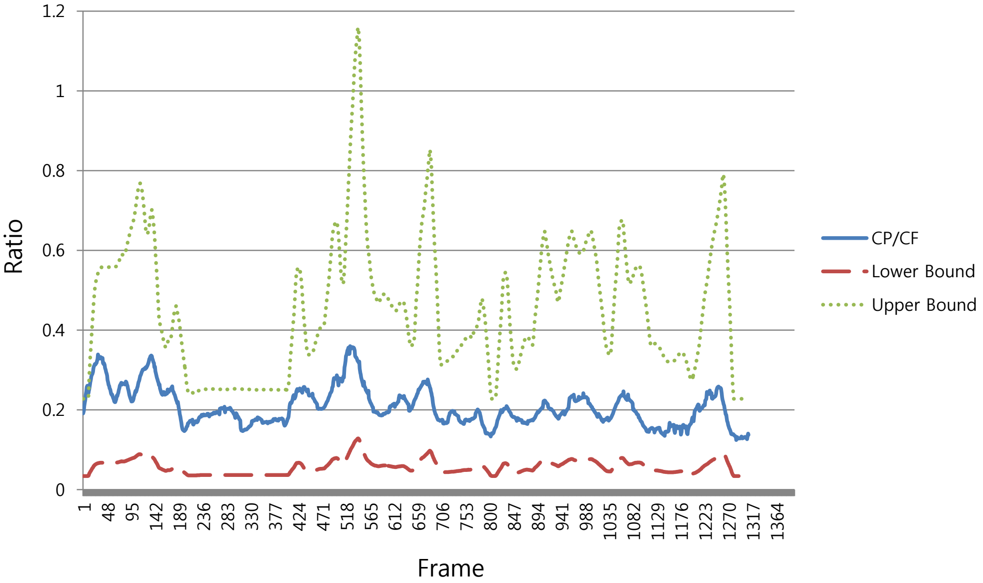


Figure 3.8: Experimental verification of the derived efficiency boundaries.

We implemented our algorithm in C++ for simulation with Intel Core i7 2.67GHz processor and 2.97GB RAM. Throughout the whole experiments, we do not use any kind of parallel processing methods, such as GPUs, OpenMP, pthread, and SIMD(single instruction multiple data). We have implemented the algorithm to be computed in a sequential way in a single core, to show its efficiency. The parameters of background subtraction methods are optimized one by one for various videos as was in [36], but the parameters of the proposed method are the same regardless of combining detection methods and testing videos. The used parameters are $\alpha_T = 0.1, \alpha_F = 0.01, \alpha_S = 0.05, \rho = 0.05$ and $k = \sqrt{3}$.

3.6.1 Efficiency of Active Attentional Sampling

We have monitored sequential intensity changes of two pixels (**A** and **B**) in Figure 3.9(a) (AVSS i-LIDS dataset is used). **A** is from a road and **B** is a pixel of a building wall. Active attentional sampling resulted in different number of samples.

Sampling Method	**A**	**B**
Uniform Sampling	20.09	3.04
Proposed Sampling	9.64	3.79

Table 3.2: Estimation accuracy comparison in RMSE.

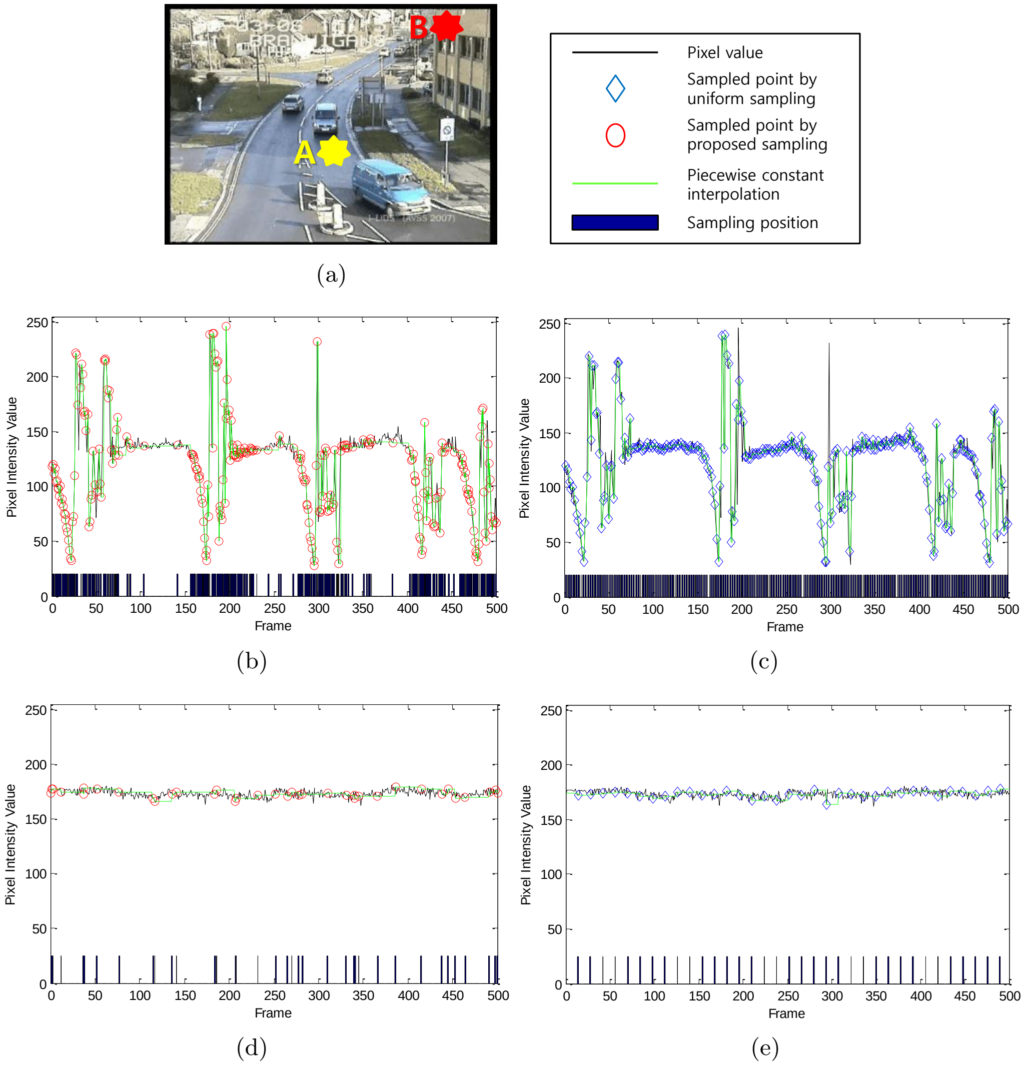
As we have expected, the road pixels are more frequently sampled. Also the effectiveness of active attentional sampling is compared with uniform sampling. As shown in Figure 3.9, the proposed sampling does not miss critical points (such as radically changing values). We have measured the RMSE (root mean squared error) of two different sampling methods in Table 3.2. The results show that the proposed sampling catches pixel value changing moment adaptively and accurately with much less samples.

3.6.2 Detection Performance Comparison

The SABS dataset[36] is used to test detection performance of the proposed method over various situations. The SABS dataset is an artificial dataset for pixel-wise evaluation of background subtraction method. For every frame of each test sequence, ground-truth annotation is provided as foreground masks. Even though it is generated artificially, there are realistic scenarios such as light reflection, shadows, traffic lights and waving trees. When considering the fact that the best F_1*-Measure* in [36] is just 0.8, SABS datasets are difficult enough to evaluate the performance of algorithm. The correctness of foreground detection is expressed by F_1*-Measure* as in [36] which is a harmonic mean of *recall* and *precision*,

$$F_1 = 2 \times \frac{recall \times precision}{recall + precision}. \tag{3.14}$$

Detection results are optimally tuned and the value of Figure 3.10 is an average of each frames's F_1*-Measure* over whole sequences. The proposed method can



(a)

(b)

(c)

(d)

(e)

Figure 3.9: The intensity of pixel **A** changes frequently because of the crossing cars. The value of **B** remains almost unchanged. The graphs show the intensity values and bars under the graphs indicate the sampled positions. For pixel **A**, the active attentional sampling samples 256 times and 25 times for pixel **B** during 500 frames. The same number of samples are generated uniformly for each sequence, and the piecewise constant interpolation is performed to reconstruct the sequence. (b) and (d) show estimated intensity graphs by proposed sampling method for A and B, respectively. (c) and (e) are reconstructed graphs by uniform sampling. We can see that the proposed one concentrate the sampling on the foreground pixels in frames with moving objects.

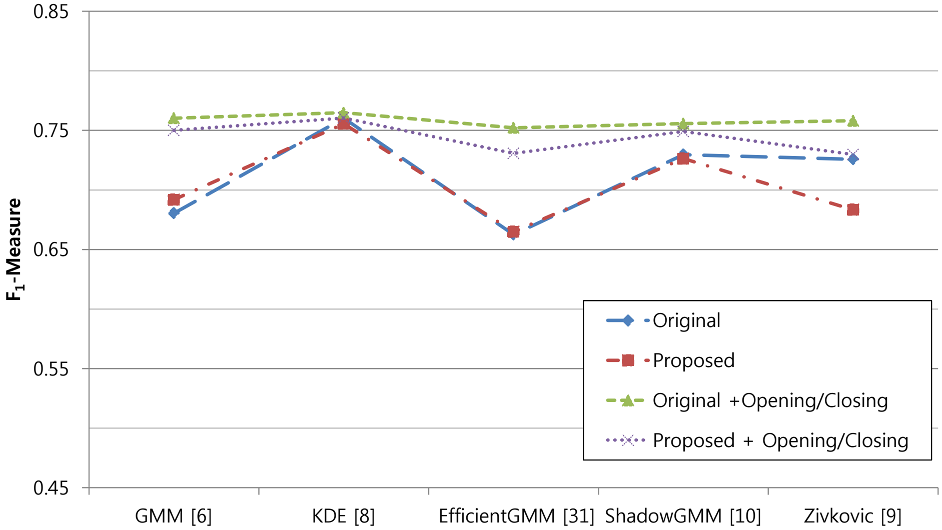


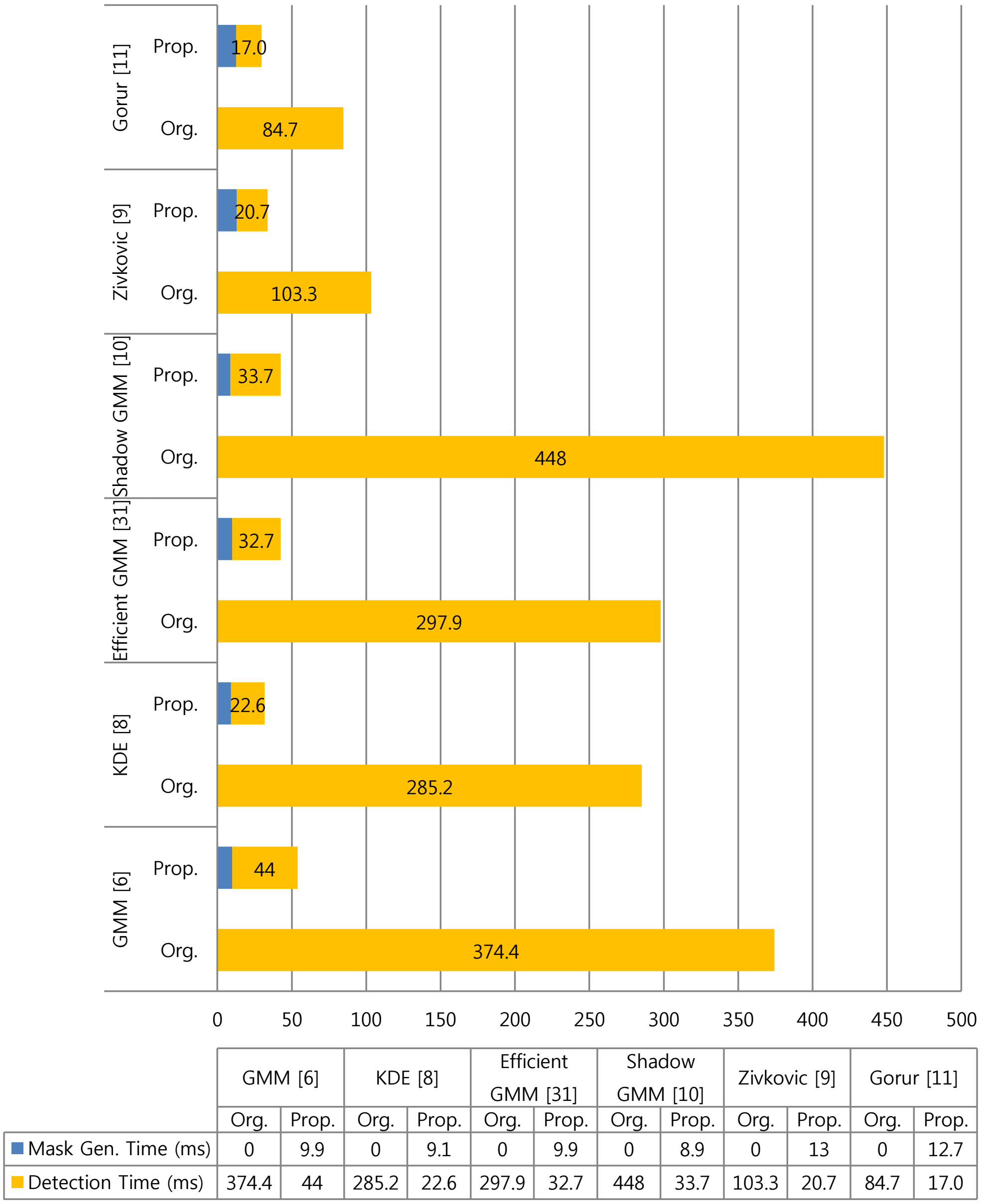
Figure 3.10: Best F_1-*Measure* for various background subtraction methods. Post image processing methods, such as opening/closing, also can be used.

be successfully combined with various background subtraction methods and post image processing methods without performance degradation.

3.6.3 Speed-up Performance Comparison

Figure 3.11 shows computation time speed-up results. The proposed method significantly shortens the detection time (on average 6.6 times). Fast detection algorithms show relatively small speed-up ratio than computationally heavy algorithms. This is because the mask generation time becomes relatively large compared to the detection time.

Figure 3.12 shows computation time changes over frames. GMM[6] method and SABS video[36] (bootstrap video) are used for the test. The computational time of the proposed method increases as the ratio of foreground region becomes large. However, the original GMM also takes more time when the foreground region increases. So the ratio of speed-up is maintained uniformly.



	GMM [6]		KDE [8]		Efficient GMM [31]		Shadow GMM [10]		Zivkovic [9]		Gorur [11]	
	Org.	Prop.	Org.	Prop.	Org.	Prop.	Org.	Prop.	Org.	Prop.	Org.	Prop.
🟦 Mask Gen. Time (ms)	0	9.9	0	9.1	0	9.9	0	8.9	0	13	0	12.7
🟨 Detection Time (ms)	374.4	44	285.2	22.6	297.9	32.7	448	33.7	103.3	20.7	84.7	17.0

Figure 3.11: Comparisons of the computational time speed-up. The tests were performed with full HD videos. The speed-up ratio of computationally heavy algorithms, such as GMM[6], shadow GMM[10] and KDE[8], is approximately 8.5 and the speed-up ratio of fast detection algorithms, such as Zivkovic[9] and Gorur[11], is approximately 3.

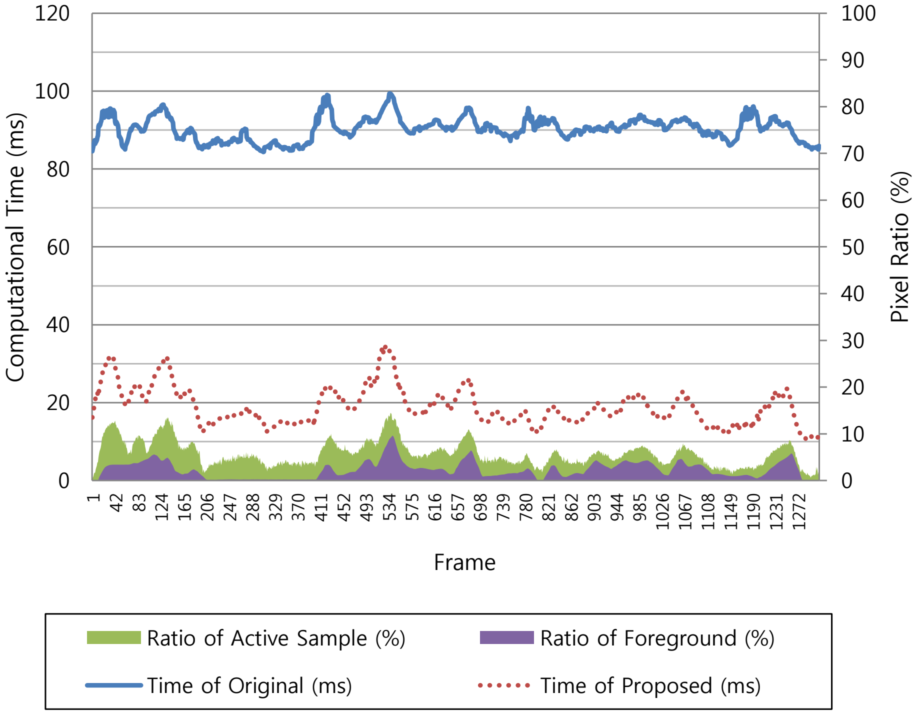
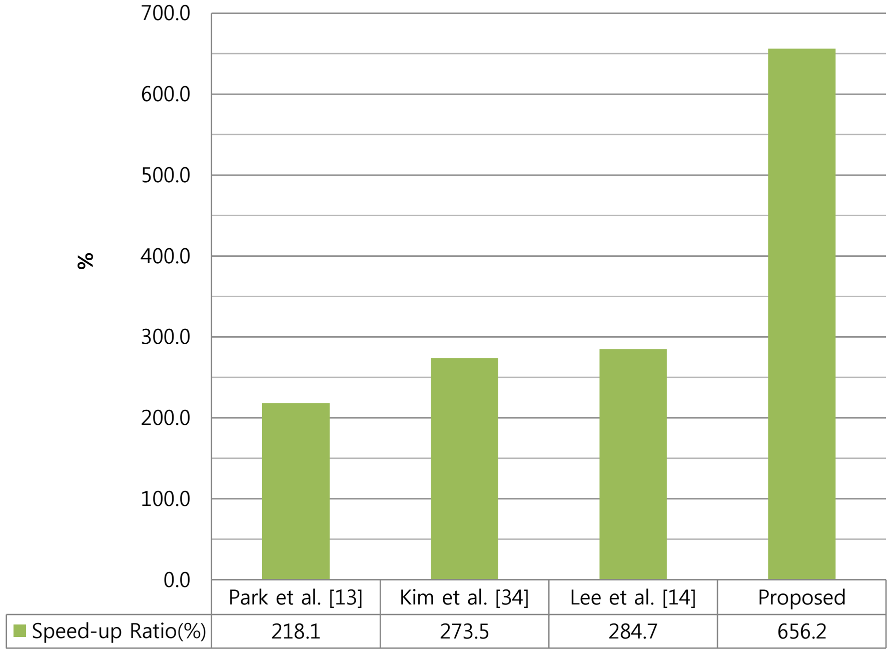


Figure 3.12: Computational time changes over foreground region ratio. The foreground region varies from 0 % to 10%. Not only the proposed method but also the original detection [6] takes more time as the ratio of foreground region increases.



	Park et al. [13]	Kim et al. [34]	Lee et al. [14]	Proposed
🟩 Speed-up Ratio(%)	218.1	273.5	284.7	656.2

Figure 3.13: Comparison of selective sampling-based speed-up methods. All the methods were commonly applied to GMM [6].

Also, we have compared the computational complexity reduction performance with similar selective sampling-based methods; Park et al.[13], Kim et al.[34] and Lee et al.[14]. All speed-up performance data are based on the optimized values of the original paper. Figure 3.13 show the average speed-up performances. The speed-up ratio of our method outperforms the others. The other subsampling strategies are pre-designed regardless of video situation. So many unnecessary samplings are inevitable because of the regularly designed sampling pattern. This causes redundant calculations. The sampling strategy of our method is totally different from the grid pattern based subsampling approach. Proposed probabilistic sampling approach is more adaptive to various video situations and becomes more efficient by reducing redundant calculations.

Method	Original(FPS)	**Proposed(FPS)**
GPU [12]	78.9	-
GMM[6]	1.6	**18.6**
KDE[8]	3.5	**31.5**
Efficient GMM[31]	3.4	**23.5**
Shadow GMM[10]	2.2	**23.5**
Zivkovic[9]	9.7	**29.7**
Gorur[11]	11.8	**33.7**

Table 3.3: Comparisons of detection time in full HD videos (1920 × 1080) in terms of frame rate (FPS).

3.6.4 Real-time Detection in Full HD Video

Until now, allegedly, using GPU is the only solution of real time detection in full HD video[12]. However, as shown in Table 3.3, our method makes it possible for the conventional pixel-wise background subtraction methods to be used for high resolution videos in real-time. The experiments are performed with GeForce GTS 250 (128 CUDA cores) for GPU version [12][8] and a single core processor for the others. Every detection method is applied to a full HD video (1920 × 1080) with optimal parameters and detection time is measured with and without our method, seperately.

3.7 Final Remarks and Discussion

The computational time problem of background subtraction is very critical because it is generally considered as a lower level image processing task and the video size is getting bigger. In this chapter, we proposed a speed-up method of conventional background subtraction algorithms using temporal attentional sampling mask generation method based on empirical attentional sampling con-

[8] implementation from `http://www.codeproject.com/KB/GPU-Programming/cubgs.aspx`

cept. The motionless background region can be skipped by attentional sampling. We designed a foreground probability map by measuring three foreground region properties, and active attentional sampling is performed to make a sampling mask. Various experiments show that the proposed method can speed up about 6.6 times without detection performance deterioration. Also our method makes it possible for the conventional background subtraction algorithms to perform real-time detection in Full HD videos with a single core processor.

Chapter 4

Selective Attentional Sampling for Recognition of Pop Dances

4.1 Introduction

The selective attentional sampling strategy is similar to feature data selection. This sampling can be useful if some characteristic data points are appeared repetitively with a similar pattern. By designing the importance measure function of each data point, $\mathcal{M}(\mathbf{I}; \Psi)$ in section 1.2.3 with proper prior knowledge about the target data, the selective attentional sampling can filter out many redundant data. So this sampling scheme requires a highest level prior knowledge about the target application among the proposed three attentional sampling methods, because the selected few sample points should be informative enough to represent the characteristics of the rest points. Therefore the selective attentional sampling is appropriate to high level vision problems which are based on high level informative features such as action recognition, object recognition and scene understanding. In this chaper, we present how the selective attentional sampling scheme can be used for complex action recognition, especially Pop dance recognition. Figure 4.1 is a conceptual illustration of the presented scheme.

Action recognition has been widely studied for decades and there are many successful approaches in recognizing simple actions [15]. Recently, following the success of simple action recognition, more realistic and complex activity recognition tasks have been dealt with. Research on complex activities has progressed to the recognition of real videos such as internet videos [16] or surveillance videos [39], human interactions [40, 15], group activities [41, 15] and temporally composed action sequences [17]. However, the current status of the research on complex activities is in its initial phase, far from the recognition ability of human.

In our work, we are interested in a new class of complex activity, namely recognizing action streams that are natural, temporally long, not repeated, and not able to be simply annotated into parts (unannotatable). Example of such action streams are dances, pantomimes, and monodramas (illustrated in Figure 4.2). Difficulty in recognizing such action streams arises from the flexibility and the high dimensionality of the human body and motion. Also, even the same actions are expressed differently depending on the body shape and habits of the actor. Dance is a good example of these kinds of complex activities. Even people dancing together to the same music, such as the famous *'Macarena'* or *'Gangnam Style,'* may show different pose, motion duration, speed, etc. However, people can easily recognize the name of dance and catch distinctive motion parts without much effort. A reliable and efficient solution to this problem would be useful for various areas, such as dance video categorization, abstraction and retrieval in YouTube, or real-time dance scoring games.

Recognizing these unannotatable action streams, having natural, diverse, and flexible motions, is not an easy task. We need good features and strong and efficient classifiers able to deal with long and complex sequences. Various features

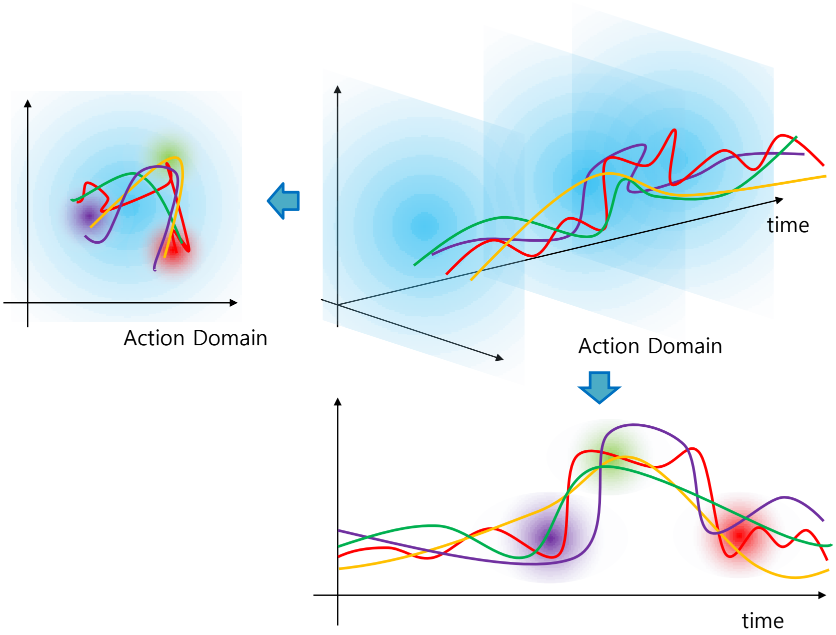


Figure 4.1: A conceptual illustration of selective attentional sampling in spatio-temporal space for the complex action recognition. Among many action data flows, shaded regions are characteristic regions which are found by the importance measure function $\mathcal{M}(\mathbf{I}; \Psi)$. Action data points only in the characteristic regions are sampled for recognition.

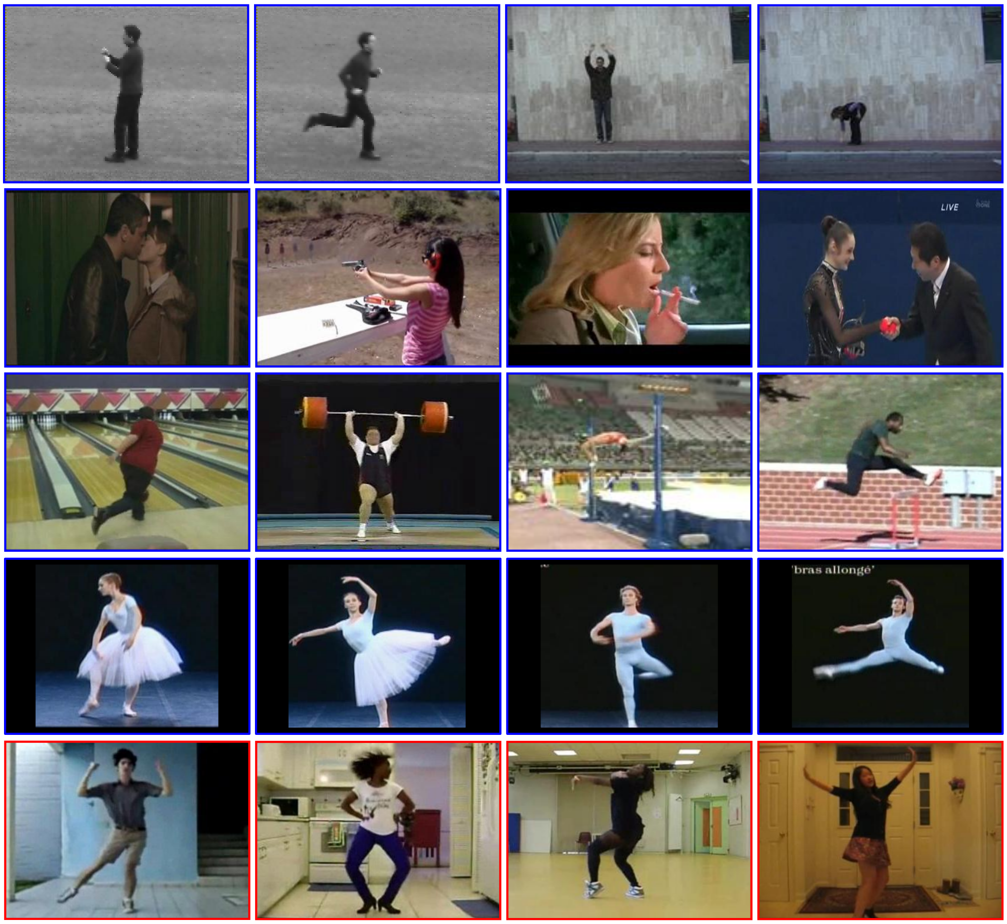


Figure 4.2: Sample frames of action recognition datasets. The first row is from KTH [42] and Weizmann [43] dataset. The secont and third rows are from HMDB51 [44] and Olympic Sports dataset [17] respectively. The fourth row is from ballet dataset [45]. The last row is from the proposed Pop-Dance dataset. Frames of KTH, HMDB51 and Olympic Sports dataset are easy to be named such as punching, running, kissing, shooting gun, smoking, high jump etc. Especially the ballet motions and postures have very specific names *'Pas de chat', Arabesque, Fouetté en tournant, and Grand jeté* (from the left). Compared to these actions, all actions of the Pop-Dance dataset are unannotatable. They only can be named as a part of each dance.

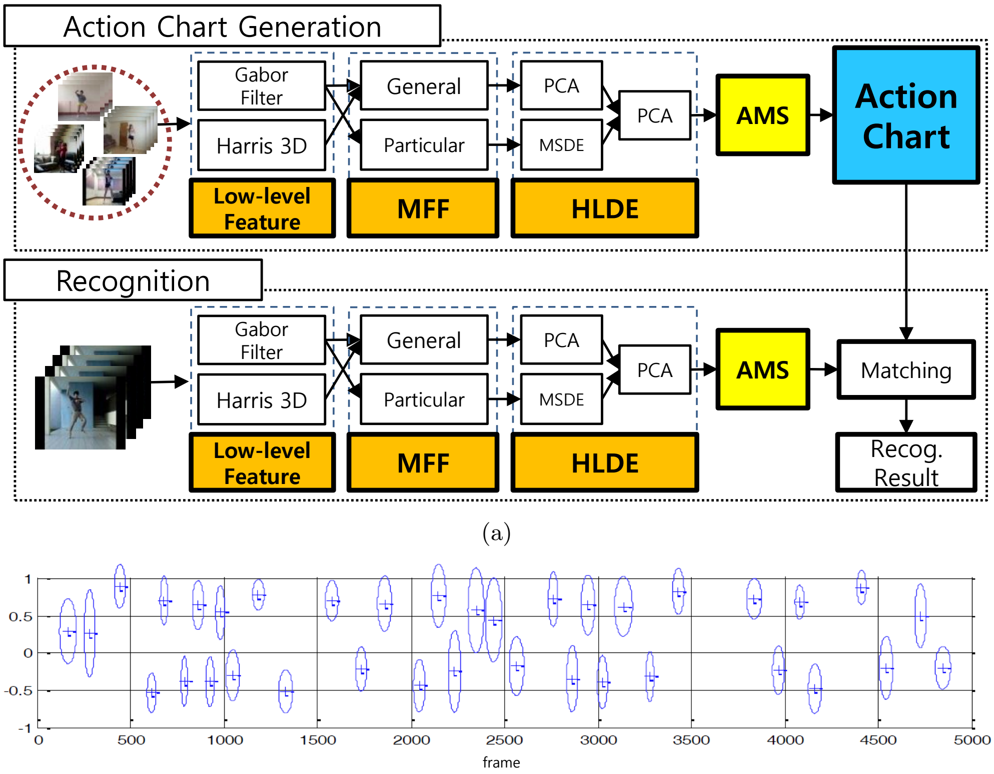
have been proposed for behavior recognition until now, but they cannot be used directly in our case. Local features, such as HoG/HoF [18] and cuboids [19], represent local spatio-temporal changes as a vector. The local features can represent motion changes with small numbers of features, but they cannot imply temporal ordering and arrangement of features in the action sequence [17]. Global spatio-temporal templates such as spatio-temporal shapes [43] and motion history [20] have been proposed to contain such temporal ordering of motions and represent human body pose changes along temporal sequence. Also Fathi *et al.* [46] proposed mid-level motion features which are built from low-level optical flow information by a learning method. These methods contain more motion information than local features, but are not appropriate for long sequence since they require extensive memory and are computational complex.

Classifiers used for traditional methods are also not suitable in our case. Dynamic time warping (DTW) algorithm and its variations [47, 48, 49, 50, 15] is one of the successful methods used for behavior recognition. However the DTW algorithm takes polynomial time and memory complexity finding the optimal nonlinear match between two feature flows. Also some probabilistic state transition models, such as Hidden semi-Markov Models (HSMMs) [51] and Conditional Random Fields (CRFs) [52], have been used for modeling temporal structure but they require predefined states which are not straight forward in our case. Recently, [17, 16] modeled the temporal structure using latent SVM. However, the number of low-level events/actions should be predetermined which is also not possible for our case of the unannotatable and undecomposable action sequences.

In this chapter, we propose a flexible and efficient method for recognizing pop dances, which is a presentable example of unannotatable action streams with natural, diverse, and flexible motions. Since conventional low-level and mid-level features are not enough, we propose a new method for mid-level feature generation from various local features representing diverse motion characteristics. The method characterizes global and sequential motion changes by a feature flow which require small memory, suitable for long sequences. To overcome the limitations of traditional classifiers, we propose a novel recognition method which *catches and focuses* on distinctive instances along the complex motion flow and efficiently recognizes long and complex sequences with promising performance.

To *catch* distinctive instances in the motion sequences, we propose a method based on zero-velocity points. There have been lots of researches [53, 54, 20, 55] using the zero-velocity or zero-crossing points of the stream of motion feature. They are usually used for motion segmentation in relatively short sequences, because a little noise will result in many false segments. This makes the zero-velocity based method be applicable to long sequences robustly. However, in our method, we propose a new filtering method (refered to as "attention measure method") for removing false detections, thus making the zero-velocity based method to be applicable for long sequences. After the filtering process, instances that survive are distictive instances decribing the action stream. We will refer to these instances as Attentional Motion Spot (AMS), which are automatically determined in our scheme.

Our recognition method then *focuses* on this AMS. AMSs appear in a similar spatio-temporal pattern for the same class of dances. We group nearby AMSs together and model each group using generative Gaussian mixture models (GMM) in spatio-temporal space. The temporal sketch of this model looks like a music chart, thus we name our model as *Action Chart*. Action Charts describe motion types and temporal motion sequences as if they are notes and rhythms of songs. In order to test the validity of our method, we have built a new dataset composed of various dancing sequences, which are difficult to be discriminated by the existing methods. Several experiments are conducted to analyze the effect of each mid-level



(a)

(b)

Figure 4.3: (a) Overall scheme of building *Action Chart* and using it for action recognition. (b) shows the generated *Action Chart* of *"You and I"* by IU.

feature and the results show that our method has good recognition performance with low computational complexity.

4.2 Action Chart

The proposed *Action Chart* is obtained by the following five steps: (1) extracting low-level features, (2) generating a mid-level motion feature using extracted low-level features, (3) embedding the mid-level feature to a low dimensional vector in order to represent the activity as temporal motion feature flow, (4) detecting

AMS in the feature flow, and (5) constructing *Action Chart* by modeling the AMS action distribution in spatio-temporal domain. Figure 4.3(a) shows the overall scheme of the proposed method.

4.2.1 Motion Feature Flow (MFF)

In order to generate the *Action Chart*, the complex and long motion variations should be represented as a feature sequence. The feature sequence should contain abundant motion properties in low dimensions. In order to achieve this requirement, we develop a new mid-level motion feature constructing method which uses low-level local feature information. We named the mid-level motion feature as motion feature flow (MFF).

Conventional low-level local feature detectors such as Gabor filtering [19] and Harris-3D [18] find local motion changes in spatio-temporal space, and local features such as cuboids [19] and HoG (Histogram of Gradient) and HoF (Histogram of Flow) descriptors [56] are independent to each other in space and time. For the description and recognition of long actions, global temporal motion change information are more important rather than accurate portrayal of short time motion. So the local features are unsuitable to recognize long and complex action sequences which should be represented as consecutive actions.

Low-level Features

To build a good mid-level feature, we use both the Gabor filtering detector [19] and the Harris-3D feature point detector [18] with HoG/HoF descriptor [56]. The two detectors behave differently having their own strong points. As shown in [57], Gabor filtering finds much more features than Harris-3D, with filter responses available for each feature point. On the other hand, Harris-3D detector shows good recognition performances when it is combined with HoG/HoF descriptors [57]. We extract them both in our case to use the strong points of both features.

The feature points are detected in a stack of images denoted by $I = \{I(x, y, t)|t = 1, ..., N\}$. Feature point sets of each frame detected by Gabor filtering are represented by $\{P^1, ..., P^N\}$ and each P^t at frame t not only contains the feature location information p_x and p_y, but also has the filter response value r (i.e. $P^t = \{p^t_x(i), p^t_y(i), r^t(i)|0 \leq i \leq n^t_p\}$ where n^t_p is the number of features detected). Also the other local features, which are detected by Harris-3D detector and described by HoG/HoF descriptors, are quantized by a descriptor codebook, which is obtained by k-means clustering (we set k as 1000 in the experiments.) of the descriptors in the training set.

MMF Generation

The MFF ($\mathcal{M}$) is composed of general ($\mathcal{M}_G$) and particular ($\mathcal{M}_P$) motion features. The MFF $\mathcal{M}$ is a temporal flow of N features, that is, $\mathcal{M} = \{\mathcal{M}^t|t = 1, \ldots, N\}$. Each $\mathcal{M}^t$ is represented as a 21-dimentional vector. The general motion feature $\mathcal{M}_G$ is composed of five measurements; motion intensity (m_I), motion extent (m_E), motion speed (m_S), motion distinctiveness (m_{DIS}) and motion diversity (m_{DIV}). The motion intensity, extent, and speed represent quantitative property of motion while the distinctiveness and diversity reflect qualitative property of motion. At every t^{th} frame, the five measurements are obtained in the following.

- **Motion intensity** m^t_I: Gabor filtering finds a pixel whose intensity has been changed for short time. If we assume that the intensity change is caused by motion, the number of detected feature points would be proportional to motion intensity. More local features will be detected around a high intensity motion area. So we measure the motion intensity by the number of cuboid features as

$$m_I^t = n_p^t. \tag{4.1}$$

- **Motion extent** m_E^t: Motion extent measures how widely current motion is occurring and it is measured by spatial distribution of feature points. We measure the motion extent by a norm of standard deviations of feature point locations as

$$m_E^t = (\frac{1}{n_p^t}\{\sum_{i=1}^{n_p^t}(p_x^t(i) - \mu_x^t)^2 + \sum_{i=1}^{n_p^t}(p_y^t(i) - \mu_y^t)^2\})^{\frac{1}{2}}, \tag{4.2}$$

where $\mu_x^t = \frac{1}{n_p^t}\sum_{i=1}^{n_p^t} p_x^t(i)$ and $\mu_y^t = \frac{1}{n_p^t}\sum_{i=1}^{n_p^t} p_y^t(i)$.

- **Motion speed** m_S^t: Gabor filtering detector is tuned to fire whenever variations in local image intensities contain periodic frequency components or spatio-temporal corners. This means that action in the same period of the filter will give a strong response to the filter, and slow action or pure translation motion will induce small response. We set the period of the filter to 15 frames/sec which means the motion with period of 0.5 seconds in 30 frames/sec video will give the strongest response. However human motion is not faster than this in general, so we consider small response is only caused by slower motion than the period time. So the motion speed is measured by sum of filter response values as

$$m_S^t = \sum_{i=1}^{n_p^t} r^t(i). \tag{4.3}$$

- **Motion distinctiveness** m^t_{DIS}: Motion distinctiveness measures how much current motion (at t frame) is changed comparing to the previous motion ($t-1$ frame). It is well known that the histogram of bag-of-words codebook can describe a short motion as a vector [56, 57]. h^t represents a normalize histogram of codebook memberships of 100 frames centered at time t. So we measure the motion changes by chi-square distance between h^{t-1} and h^t as follows,

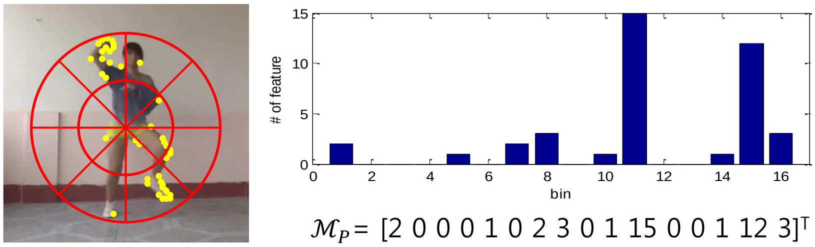
$$m^t_{DIS} = \chi^2(h^{t-1}, h^t) = \sum_{i=1}^{k} \frac{(h^{t-1}(i) - h^t(i))^2}{h^{t-1}(i) + h^t(i)}. \tag{4.4}$$

- **Motion diversity** m^t_{DIV}: Various codebook memberships of h^t imply that the current motion is composed of diverse local motions. So we measure the codebook diversity by entropy of h^t as

$$m^t_{DIV} = -\sum_{i=1}^{k} h^t(i) \log h^t(i) \tag{4.5}$$

Each measurement is one-dimensional data sequence. By concatenating the five motion measurements $\mathcal{M}^T_G = [m_I, m_E, m_S, m_{DIS}, m_{DIV}]$, the general motion feature $\mathcal{M}_G$ becomes a five-dimensional data sequence ($\mathcal{M}_G \in R^{5\times N}$).

The particular motion feature $\mathcal{M}_P = \{m_P(n)|n = 1, ..., 16)$ represents the number of local feature points $p_{(x,y)}$ and their relative location using a concentric 16-bin histogram method as shown in Figure 4.4. We place the center of the concentric circular bin at the estimated center of human and the radius of the circle is the same as the half of human height. The center position and the height can be estimated using foreground information or human detection algorithm [55]. In this chapter we estimate the values of current frame using the locational information of feature points of the previous 100 frames. The mean of feature point



$$\mathcal{M}_P = [2\ 0\ 0\ 0\ 1\ 0\ 2\ 3\ 0\ 1\ 15\ 0\ 0\ 1\ 12\ 3]^T$$

Figure 4.4: Particular motion feature $\mathcal{M}_P$.

locations is estimated as the center and the mean of the maximum distances from the center is estimated as the half of human height.

Finally, the MMF at time t becomes

$$(\mathcal{M}^t)^T = [\mathcal{M}_G^t, \mathcal{M}_P^t] = [m_I^t, m_E^t, m_S^t, m_{DIS}^t, m_{DIV}^t, m_P^t(1, \ldots, 16)]. \qquad (4.6)$$

The measured data flow is too peaky because of noises, so we smooth the data flow using local polynomial regression fitting [58] with a low degree of smoothing (span=0.03).

4.2.2 Hierarchical Low Dimensional Embedding

To avoid the curse of dimensionality in analyzing motion streams, the MFF $\mathcal{M}$ is required to be embedded to a lower dimensional space. There have been some efforts to find the most appropriate low-dimensional embedding method for action recognition [59, 60]. Wang *et al.* [55] have experimentally verified that linear methods (such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Locality Preserveing Projections [55]) outperform nonlinear methods (Locally Linear Embedding (LLE) [61] and Laplacian Eigenmaps (LE) [62]) in action recognition. This conventional linear embedding method considers all dimensions of high-dimensional data at the same level and linearly combines them.

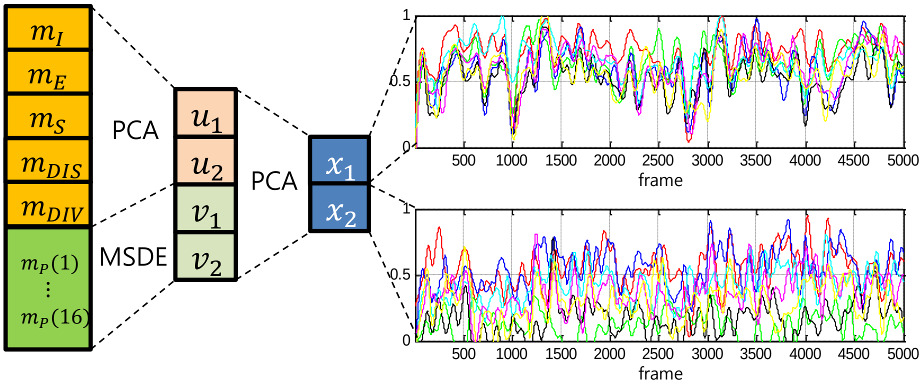


Figure 4.5: Concept of the HLDE and the embedded data sequences of *"Gee"* by SNSD. Different color implies different people.

However $\mathcal{M}$ consists of two information groups; $\mathcal{M}_G$ and $\mathcal{M}_P$. They have different numbers of dimensions and each dimension has different amounts of information. Each dimension of $\mathcal{M}_G$ implies distinctive motion information while 16 dimensions of $\mathcal{M}_P$ represent only one motion information in a combination. In other words, the importance of each dimension is different.

To handle this problem, we propose a hierarchical low dimensional embedding (HLDE) method. First, we embed five dimensional general feature vector $\mathcal{M}_G$ into a two dimensional vector $U^T = [u_1, u_2]$ using PCA and we simultaneously apply Mean and Standard deviation-distance embedding (MSDE) [63] to reduce the particular feature vector $\mathcal{M}_P$ having 16 dimensions into a two dimensional vector $V^T = [v_1, v_2]$. Then, we perform PCA again on the four dimensional vector $W^T = [U, V] = [u_1, u_2, v_1, v_2]$ reducing it to a two dimensional feature vector $X^T = [x_1, x_2]$. In figure 4.5, we show the concept of the proposed HLDE and an example of the embedded results $X^T = [x_1, x_2]$. As we can see, the principal dimensional data x_1 contains more characteristic motion information than the second dimension x_2. Since our goal is to recognize the dance class, using only the principal dimension x_1 for recognition ($X = \{x_1\}$) is sufficient.

4.2.3 Attentional Motion Spot Selection

Psychological study [64, 65, 66] reports that dividing ongoing activity into meaningful actions is essential for perception and later memory, and the segmentation is strongly related to motion changes [64]. By mimicking the human perception mechanism, we propose a method to catch and focus on distinctive instances along the motion flow X. The motion feature data X is a sequential data $X = \{x^t | t = 1...N\}$. We define these distinctive instances as attentional motion spot (AMS) and we use velocity (first derivative) of X to find the AMS, similar to human using motion changes as a clue for segmentation. We define a zero-velocity point set $Z = \{z_1, z_2, \ldots\} = \{t | \Delta_t x^t = x^{t+1} - x^t = 0\}$ and *convexity index* ξ^t as

$$\xi^t = \begin{cases} 1 & \Delta_t^2 x^t \leq 0 \\ -1 & \Delta_t^2 x^t > 0\,, \end{cases} \tag{4.7}$$

where $\Delta_t^2 x^t = \Delta_t x^t - \Delta_t x^{t-1}$. The number of zero-velocity point is determined adaptively. To avoid the false detection problem which other methods using zero-velocity [53, 20, 55] suffer from, we introduce an attention measure η at j^{th} zero-velocity point z_j defined as

$$\eta(z_j) = \left| \frac{x^{z_j} - x^{z_{j-1}}}{z_j - z_{j-1}} \right| + \left| \frac{x^{z_{j+1}} - x^{z_j}}{z_{j+1} - z_j} \right|. \tag{4.8}$$

We use η to filter out the noisy zero-velocity points by thresholding and make an *attentional point* set $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_n\} = \{z_j | \eta(z_j) > \epsilon\}$ (ϵ is 0.005 in the experiments) as shown in the first chart of Figure 4.6. n is the number of attentional points determined automatically. The i^{th} AMS of X is composed of two components; (for about i^{th} AMS) attentional point τ_i and its corresponding convexity indexed x^{τ_i} value. The AMS set Ψ is defined as

$$\Psi = \{\psi_1, \psi_2, \ldots, \psi_n\}, \tag{4.9}$$

where $\psi_i^T = [\tau_i, x^{\tau_i}\xi^{\tau_i}]$ (see the second chart of Figure 4.6).

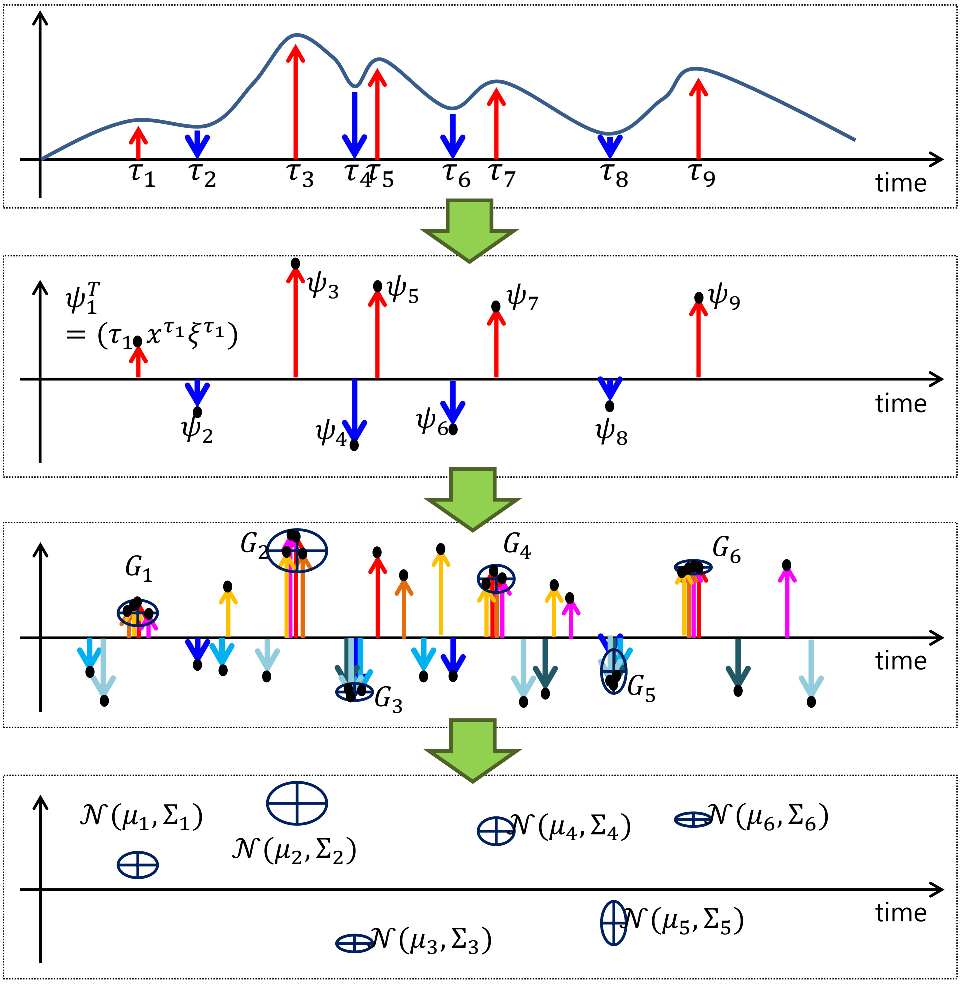
4.2.4 Action Chart Generation and Recognition

The *Action Chart* is the multivariate Gaussian model of AMSs in spatio-temporal domain. In order to generate the *Action Chart* of each action class $c \in 1...C$, training motion streams X_s^c ($s \in 1...S^c$) are temporally aligned using DTW [67] and the AMS set Ψ_s^c is generated independently. After all AMSs are generated from all motion streams for an action class, they are placed on a spatio temporal space. We then group the closely placed AMSs based on temporal proximity as shown in the third row of Figure 4.6.

We group AMSs which are temporally placed within δ frames (δ is 50 in the experiments). We denote the group set as G. The number of groups of c class N_G^c and the number of AMSs of g^{th} group $N_E^{(c,g)}$ are automatically decided. Each group is modeled as a weighted multivariate Gaussian model $\omega_g^c \mathcal{N}(\mu_g^c, \Sigma_g^c)$, where,

$$\begin{aligned}
\omega_g^c &= \frac{N_E^{(c,g)}}{S^c}, && (4.10)\\
\mu_g^c &= \frac{1}{N_E^{(c,g)}} \sum_{\psi_i^c \in G_g^c} \psi_i^c, && (4.11)\\
\Sigma_g^c &= \frac{1}{N_E^{(c,g)}} \sum_{\psi_i^c \in G_g^c} (\psi_i^c - \mu_g^c)(\psi_i^c - \mu_g^c)^T && (4.12)
\end{aligned}$$

The class of the test action stream X_{test} is determined through maximum likelihood estimations. The AMS set of X_{test} is obtained and represented as $\Psi^{test} = \{\psi_1^{test}, ..., \psi_{n^{test}}^{test}\}$. The class recognition is performed by matching the



τ_1
τ_2
τ_3
τ_4
τ_5
τ_6
τ_7
τ_8
τ_9
time
ψ_1^T
$= (\tau_1, x^{\tau_1} \xi^{\tau_1})$
ψ_3
ψ_5
ψ_7
ψ_9
ψ_2
ψ_4
ψ_6
ψ_8
time
G_1
G_2
G_4
G_6
G_3
G_5
time
$\mathcal{N}(\mu_1, \Sigma_1)$
$\mathcal{N}(\mu_2, \Sigma_2)$
$\mathcal{N}(\mu_4, \Sigma_4)$
$\mathcal{N}(\mu_6, \Sigma_6)$
$\mathcal{N}(\mu_3, \Sigma_3)$
$\mathcal{N}(\mu_5, \Sigma_5)$
time

Figure 4.6: Illustration of AMS selection and *Action Chart* generation

Ψ^{test} and the generated *Action Chart* of each class $c \in 1...C$ one by one,

$$\begin{aligned} p(\Psi^{test}|c) &= \frac{1}{N_G^c} \sum_{i=1}^{n^{test}} \sum_{g=1}^{N_G^c} \omega_g^c \mathcal{N}(\psi_i^{test}|\mu_g^c, \Sigma_g^c) && (4.13) \\ C(X_{test}) &= \underset{c}{\operatorname{argmax}}\, p(\Psi^{test}|c). && (4.14) \end{aligned}$$

4.3 Experimental Results

To evaluate the validity of the proposed method, we have compared our method with other well known methods [56, 17]. The evaluation was performed with synthesized complex actions using the Weizmann dataset [43] as in [17] and our own action dataset named Pop-Dance dataset. We implemented our algorithm and the method by Niebles *et al.* [17] in Matlab for simulation with Intel Core i7 3.40GHz processor and 16.0GB RAM. The parameters and inital values in Niebles *et al.* [17] were carefully optimized one by one for various datasets, while the parameters of the proposed method were set to the same regardless of datasets. We used the binaries provided by [56] to extract Harris-3D feature point and the HOG/HOF feature descriptors, and matlab code for Gabor filter based feature detector were provided by the author of [19]. VLFeat library [68] was used to obtain the bag-of-words codebook and SVM was implemented using LIBSVM [69].

4.3.1 Pop-Dance Dataset

Well-known action datasets such as KTH [42], Weizmann [43] and HMDB51 [44] datasets are relatively short and contain only one action in a video clip. Therefore, they are not appropriate for evaluating an algorithm for recognition of complex

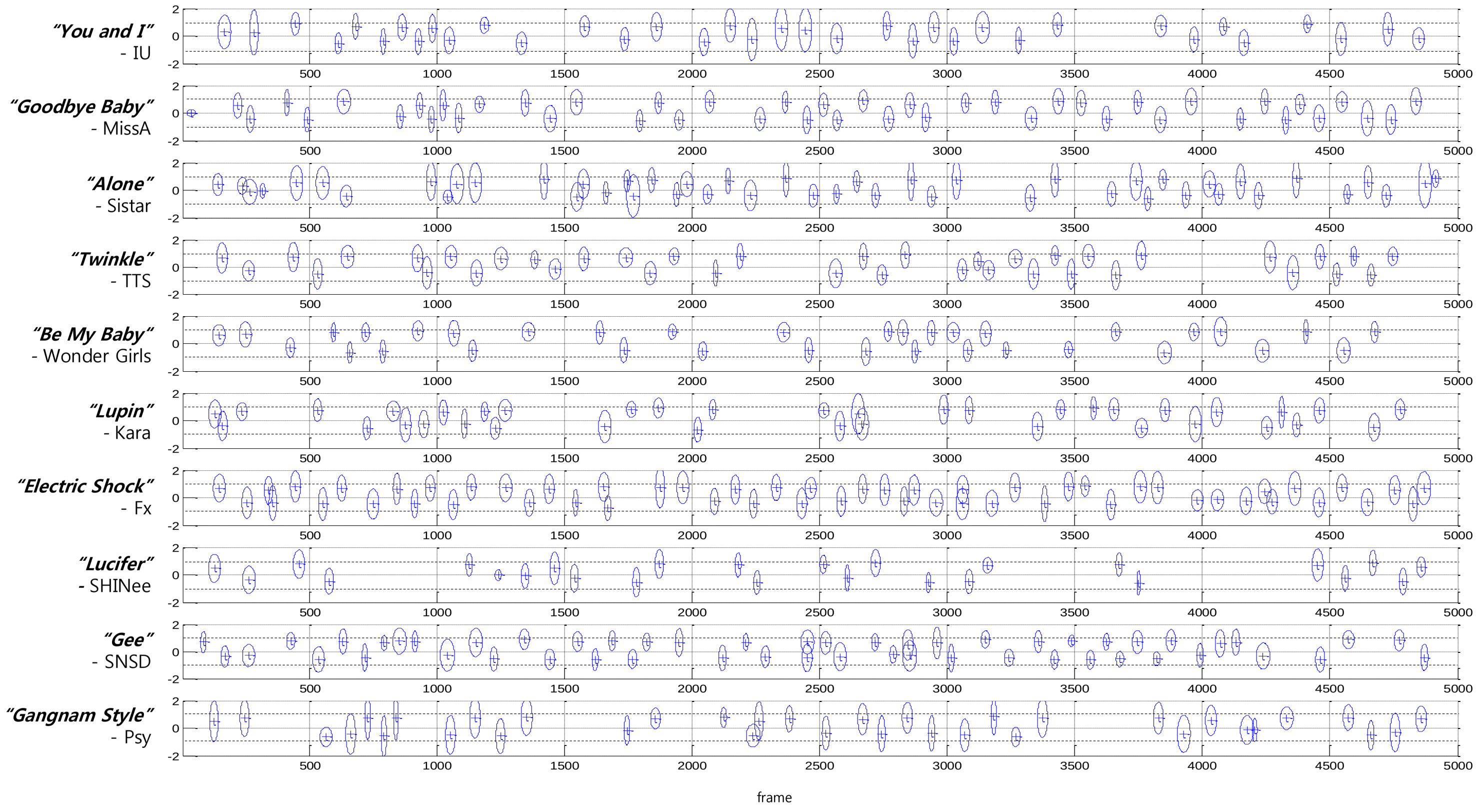


Figure 4.7: Generated *Action Charts* for the Pop-Dance dataset.

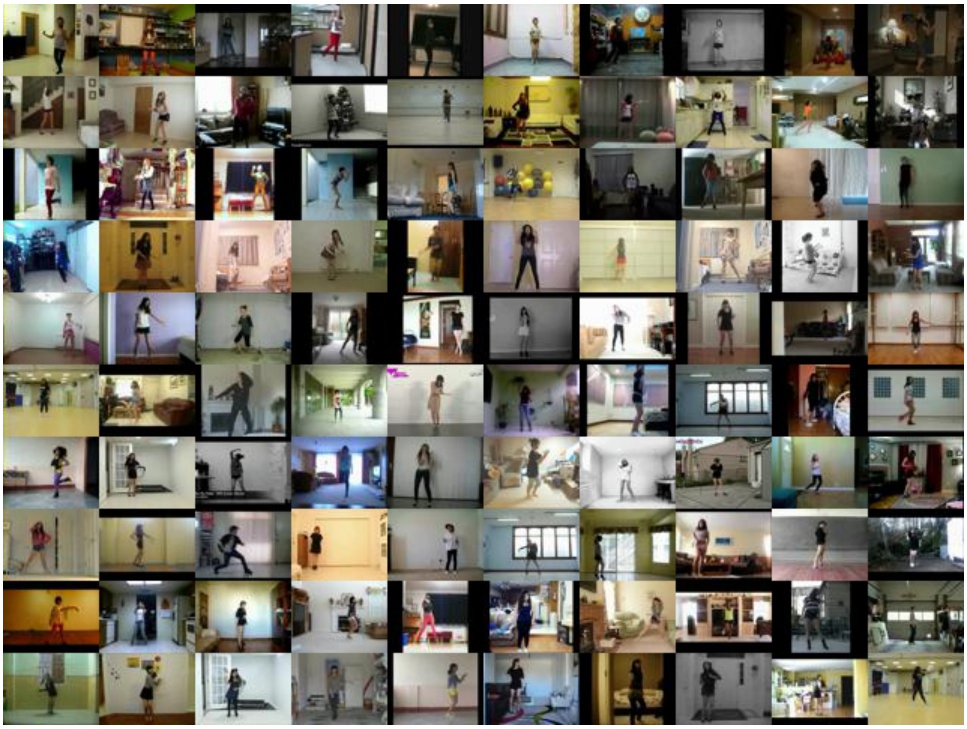


Figure 4.8: Sample frames of Pop Dance Dataset.

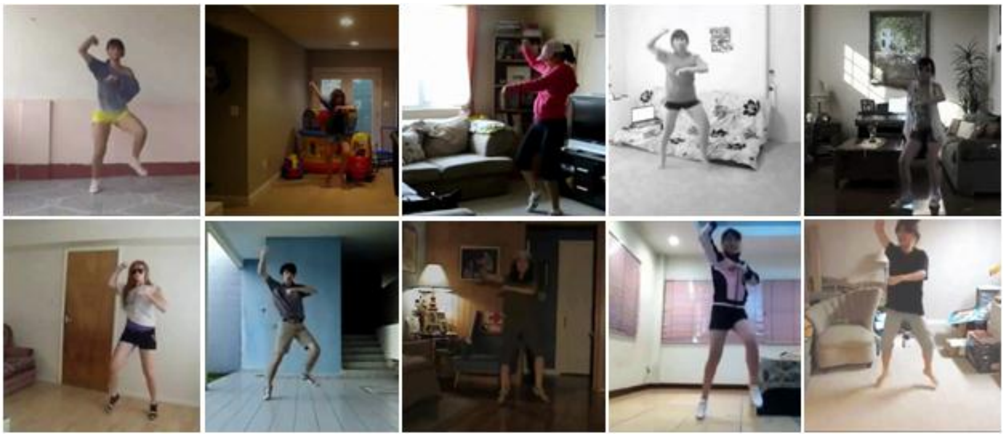


Figure 4.9: Sample frames of Pop-Dance dataset (Gangnam Style by Psy). Even people are dancing the same part of dance, they look different. The dataset will be made available online.

action sequences. Olympic Sports dataset [17] contains complex motions that go beyond simple punctual or repetitive actions, but still, the number of atomic motions is small (3 to 4) and the motions are simple. The case of dancing contains relatively complex and diverse motions with large degrees of freedom. To evaluate complex activity recognition algorithms, we built up a new dataset which contains motion sequences of people dancing following Pop songs.

The dataset is consisted of video clips of people dancing downloaded from YouTube. Video clips in the dataset are relatively long, composed of diverse actions. Each person in the dataset dances differently in his/her own style to the same music. Also the dance motions show large variations depending in camera view point, human scale, appearance, clothes, shadow and illumination conditions as shown in Figure 4.9. The dataset contains 10 dances: *"You and I"*-IU, *"Goodbye Baby"*-MissA, *"Alone"*-Sistar, *"Twinkle"*-TTS, *"Be My Baby"*-Wonder Girls, *"Lupin"*-Kara, *"Electric Shock"*-Fx, *"Lucifer"*-SHINee, *"Gee"*-SNSD, and *"Gangnam Style"*-Psy. Each dance was performed by 10 different people. In total, the dataset is composed of 100 dancing video sequences of 100 different people as

Data Set	**year**	**action class #**	**video clip #**	**ave. # of frms per video clip**	**resol.**	**video source**
KTH [42]	2004	6	600	483	160x120	Recorded
Weizmann [43]	2005	10	93	61	180x144	Recorded
UCF-Sports [71]	2009	9	150	64	480x360 720x576	BBC ESPN
Hollywood2 [72]	2009	12	3669	340	480x360 720x576	69 movies
Olympic [17]	2010	16	784	233	320x240 1280x720	YouTube
UCF50 [73]	2012	50	6680	200	320x240	YouTube
HMDB51 [44]	2011	51	6766	94	height: 240	Internet
Pop Dance	**2012**	**10**	**100**	**6190**	**640x480**	**YouTube**

Table 4.1: Comparison with widely used datasets.

shown in Figure 4.8. The average length of the video clips in the dataset is 6190 frames long (specific lengths of each dance classes are shown in Table 4.2. To the best of our knowledge, this is the longest action video clip of one person acting in the vision community (Table 4.1). Also our Pop-Dance dataset is much more difficult than the existing Ballet dataset [70, 45], because of the diverse types of motions within the dataset. Ballet is usually composed of sequential annotatable ballet poses (Figure 4.2), but dance poses of Pop-Dance dataset are neither annotatable nor separable.

4.3.2 Validation of Proposed Features

To verify the effects of the mid-level features and the low dimensional embedding method used in our thesis, we measured the classification performance of the proposed method under various configurations of mid-level features. In this experiment, we tested the effect of a mid-level feature by leave-one-out (LOO) strategy. As shown in Figure 4.10, without all general motion features ($\mathcal{M}_G$),

Singer	Song	Number of frames
IU	You and I	6860
MissA	Goodbye Baby	6583
Sistar	Alone	6000
TTS	Twinkle	6100
Wonder Girls	Be My Baby	6034
Kara	Lupin	5595
Fx	Electric Shock	5772
SHINee	Lucifer	7020
SNSD	Gee	5710
Psy	Gangnam Style	6225
	Average	6190

Table 4.2: Informations of Pop dance dataset.

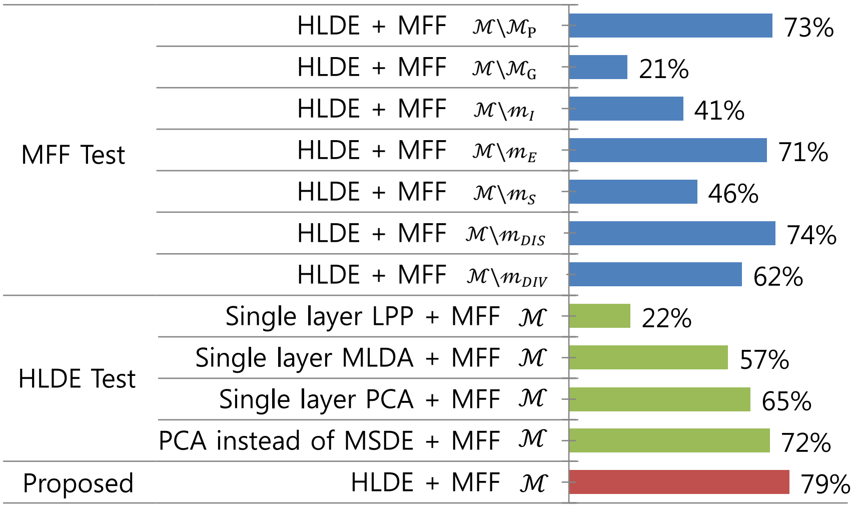


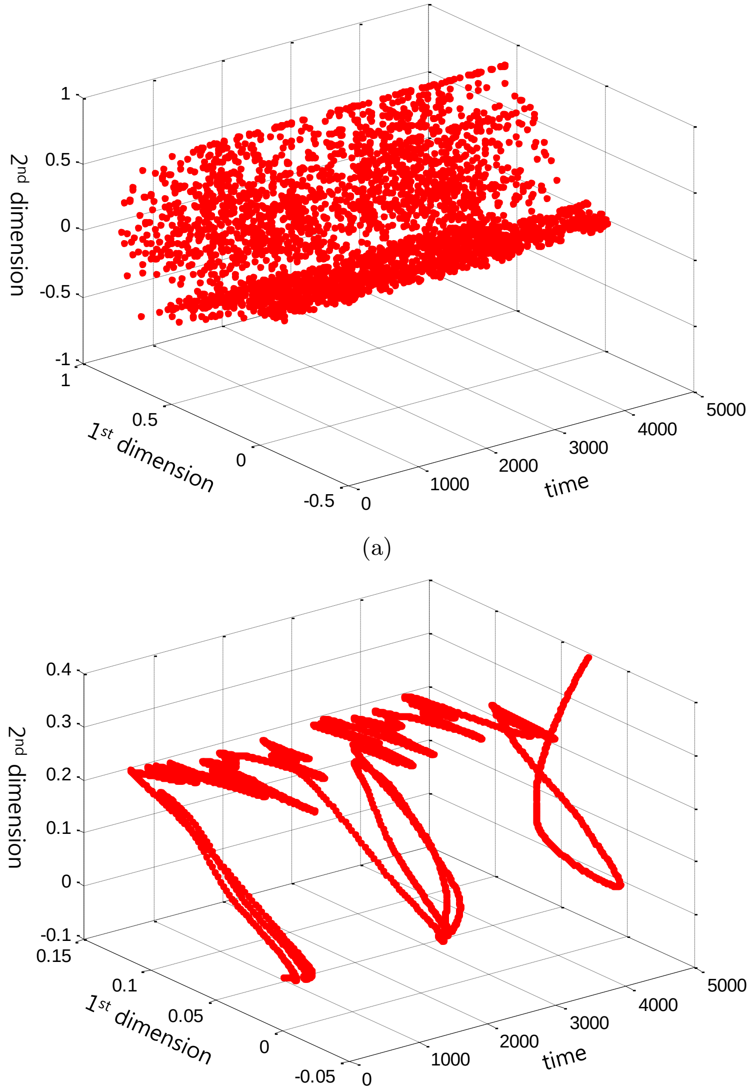
Figure 4.10: Performance of our method with different configurations.

large degradation in performance (21%) is shown which implies the general features take a significant role in recognition performance. Among the general features, motion intensity(m_I) and motion speed (m_S) are shown to be influential to the performance. The result shows the best performance when using the all features proposed in our thesis. Also, to show the effect of HLDE, we tested it with different dimension reduction schemes with MFF. As shown in Figure 4.10, with various configurations our method shows different performances, demonstrating the effects of each components of the proposed method. Especially for the multi-class linear discriminant analysis (MLDA) [74], even the MLDA finds axes (the number of classes-1=9) that best separate the categories, we only use the first principal dimension only not the whole 9 dimensions for the algorithm. The result shows that the proposed HLDE outperforms all other configurations.

Furthermore we verify the nonlinear embedding method MSDE by comparing with linear embedding method PCA. The motion data MFF especially particular motion features ($\mathcal{M}_P$) has a nonlinear property, so linear embedding method such as PCA is inappropriate for representing data in a low dimension as shown in Figure 4.11.

4.3.3 Recognition Performance

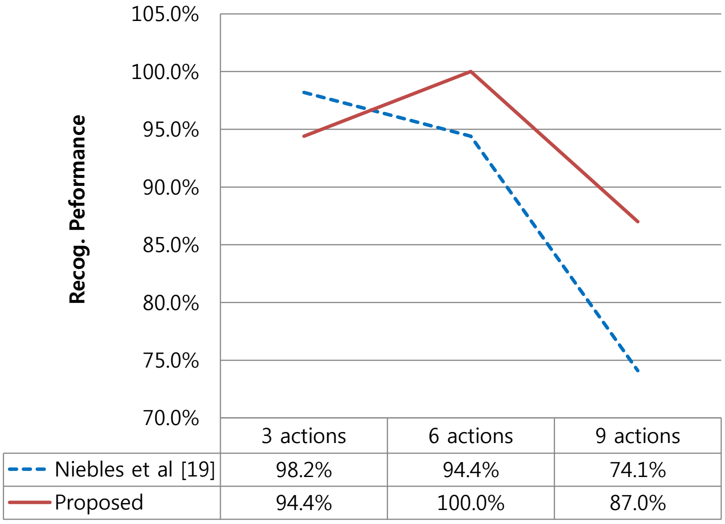
The recognition performance of our method was compared to other well known methods in three ways. First, the method was tested with a set of synthesized complex actions using the discriminative simple actions from the Weizmann dataset [43]. Second, we used the proposed Pop-Dance dataset with the whole sequence as the query using LOO strategy. Third, we used the Pop-Dance dataset with only a part of the sequence as the query. The same codebook, generated beforehand for each dataset, was used for all methods compared.



(a)

(b)

Figure 4.11: Low dimensional embedding result comparisons between (a) linear embedding method PCA and (b) nonlinear embedding method MSDE.



	3 actions	6 actions	9 actions
Niebles et al [19]	98.2%	94.4%	74.1%
Proposed	94.4%	100.0%	87.0%

Figure 4.12: Recognition performance comparison using synthesized Weizmann dataset.

Synthesized Complex Actions

Using synthesized complex actions for measuring the performance was also performed in [17]. A synthesized set of complex action sequences is constructed by concatenating 3 simple motions from the Weizmann action database [43]: ‘jump’, ‘wave’ and ’jack’. In [17] only 6 complex action classes are generated using 3 simple motions, but we increased the number of complex action classes by allowing repetition of the 3 atomic motions. Figure 4.12 shows the recognition performance with respect to the number of atomic actions in the sequence compared to [17]. As the number of atomic actions increases, our method shows better recognition performance than [17].

Pop-Dance Dataset Recognition

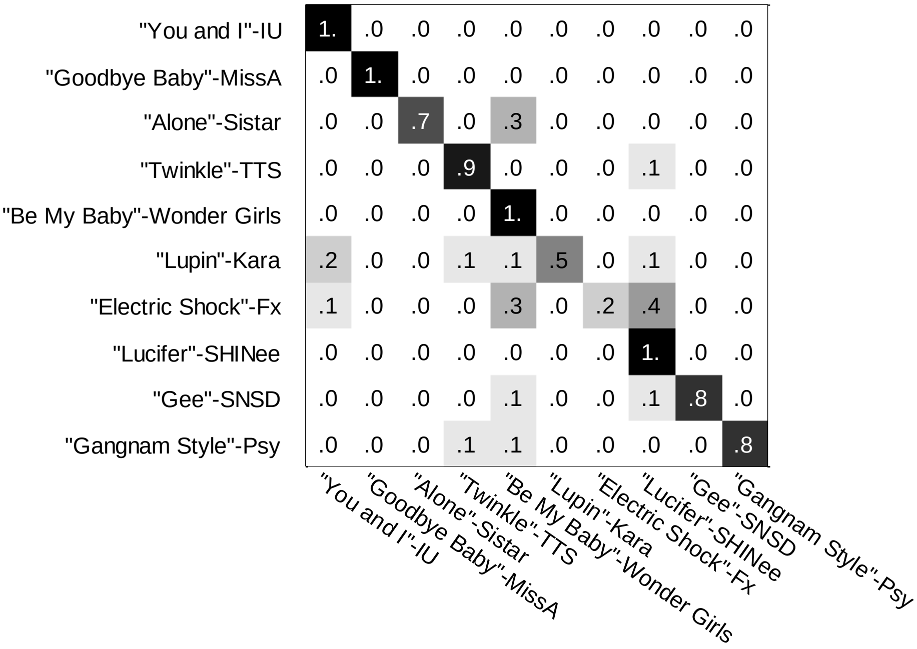
Performance comparison results for the Pop-Dance dataset using the whole sequence as the query is shown in Table 4.3. We compared our method with four

Algorithm	Perf. (%)	Total Test Time (sec)
MFF+HLDE+SVM	15	94818.0
MFF+HLDE+DTW	64	17089.0
Laptev [56]	25	20.9
Niebles *et al.* [17]	66	31235.0
Proposed	**79**	**57.4**

Table 4.3: Recognition performance and classification time comparison with widely used methods.

methods; SVM-based classification with MFF (separate SVM classifiers were trained for each class using RBF kernel), DTW with MFF (similar to methods used in [49, 75]), the method by Laptev *et al.* [56], and the state-of-the-art method by Niebles *et al.* [17]. We used a linear kernel for [17] and a χ^2 kernel for [56]. All the tests were conducted using LOO validation except for Niebles *et al.* [17] we used ten-fold validation, since it took too much time for model training.

We obtained the best recognition performance as well as a very short computational time compared to other methods. These results show that the proposed *Action Charts* well model each sequences in an abstract manner. The fastness of our method comes from the fact that we only use AMS for evaluating the fitness. This is similar to looking at the "charts" of a song to determine which song a person is listening to, which can be done efficiently. Note that DTW achieves better recognition result than SVM. This is not surprising because the MFF itself is a temporal feature flow. Confusion matrix for the results of our method on the Pop-Dance dataset is shown in Figure 4.13. Our model performs relatively poorly for the *"Lupin"*-Kara and *"Electric Shock"*-Fx classes. This can be due to the weak discriminative power of the features extracted from these videos.

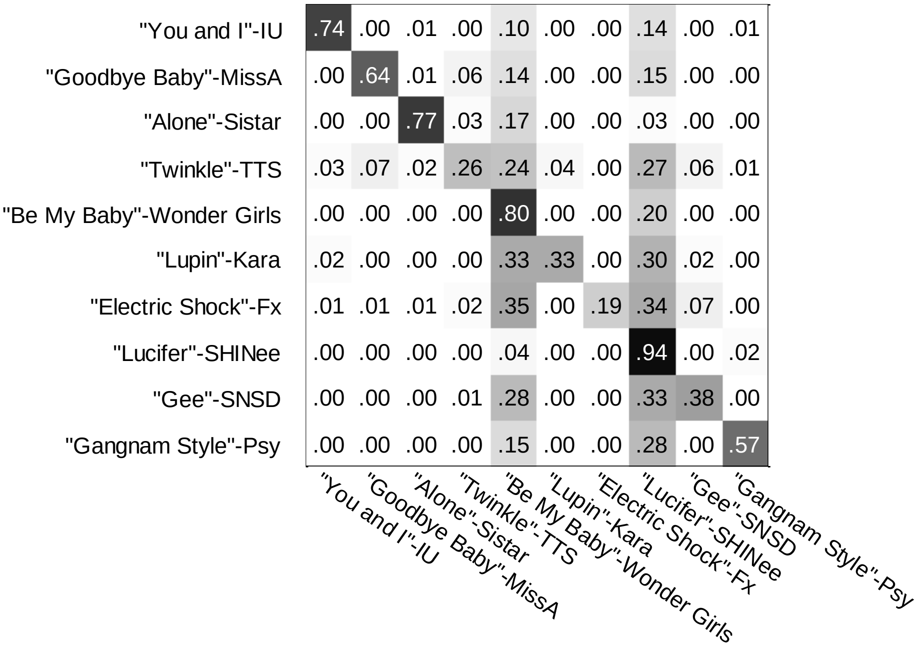


	"You and I"-IU	"Goodbye Baby"-MissA	"Alone"-Sistar	"Twinkle"-TTS	"Be My Baby"-Wonder Girls	"Lupin"-Kara	"Electric Shock"-Fx	"Lucifer"-SHINee	"Gee"-SNSD	"Gangnam Style"-Psy
"You and I"-IU	1.	.0	.0	.0	.0	.0	.0	.0	.0	.0
"Goodbye Baby"-MissA	.0	1.	.0	.0	.0	.0	.0	.0	.0	.0
"Alone"-Sistar	.0	.0	.7	.0	.3	.0	.0	.0	.0	.0
"Twinkle"-TTS	.0	.0	.0	.9	.0	.0	.0	.1	.0	.0
"Be My Baby"-Wonder Girls	.0	.0	.0	.0	1.	.0	.0	.0	.0	.0
"Lupin"-Kara	.2	.0	.0	.1	.1	.5	.0	.1	.0	.0
"Electric Shock"-Fx	.1	.0	.0	.0	.3	.0	.2	.4	.0	.0
"Lucifer"-SHINee	.0	.0	.0	.0	.0	.0	.0	1.	.0	.0
"Gee"-SNSD	.0	.0	.0	.0	.1	.0	.0	.1	.8	.0
"Gangnam Style"-Psy	.0	.0	.0	.1	.1	.0	.0	.0	.0	.8

Figure 4.13: Confusion matrix of the proposed method.

Recognizing with Parts of the Sequence

Our method can be applied not only for the case when the whole action sequence is given as the query, but also for the case when only parts of the sequences are given. Recognizing with parts is important for practical applications such as video retrieval. In our test setting, whole sequences were used for training and 1000 random portions (with random length longer than 1000 frames and random positions) were used for testing. We compared our method only with DTW since all other methods in Section 4.3.3 are not appropriate for this kind of testing. We applied our method by finding the highest similarity score for each class model by sliding the query on the model generated for the whole sequence, and then selecting the class with highest score. Average performance is shown in Table 4.4 and Figure 4.14 is the confusion matrix for the recognition results of our method. Our method shows promising results both in recognition performance



	"You and I"-IU	"Goodbye Baby"-MissA	"Alone"-Sistar	"Twinkle"-TTS	"Be My Baby"-Wonder Girls	"Lupin"-Kara	"Electric Shock"-Fx	"Lucifer"-SHINee	"Gee"-SNSD	"Gangnam Style"-Psy
"You and I"-IU	.74	.00	.01	.00	.10	.00	.00	.14	.00	.01
"Goodbye Baby"-MissA	.00	.64	.01	.06	.14	.00	.00	.15	.00	.00
"Alone"-Sistar	.00	.00	.77	.03	.17	.00	.00	.03	.00	.00
"Twinkle"-TTS	.03	.07	.02	.26	.24	.04	.00	.27	.06	.01
"Be My Baby"-Wonder Girls	.00	.00	.00	.00	.80	.00	.00	.20	.00	.00
"Lupin"-Kara	.02	.00	.00	.00	.33	.33	.00	.30	.02	.00
"Electric Shock"-Fx	.01	.01	.01	.02	.35	.00	.19	.34	.07	.00
"Lucifer"-SHINee	.00	.00	.00	.00	.04	.00	.00	.94	.00	.02
"Gee"-SNSD	.00	.00	.00	.01	.28	.00	.00	.33	.38	.00
"Gangnam Style"-Psy	.00	.00	.00	.00	.15	.00	.00	.28	.00	.57

Figure 4.14: Cropped video recognition results represented as a confusion matrix.

Algorithm	Perf. (%)	Recog. Time per Video (sec)
DTW	24	129.4
Proposed	**56.3**	**39.7**

Table 4.4: Recognition performance and computation time for recognizing one cropped video.

and computational time.

4.3.4 Automatic Action Abstraction

Automatic action abstraction is performed by concatenating frames around attentional parts. This is a reasonable way to create abstracts of videos, since attentional parts are very much similar to the human concept of characteristic points in the video. We have experimentally validated that this is true by comparing automatically found attentional points with manually indicated characteristic parts of each dance. Figure 4.15 and Figure 4.16 show the abstraction results of two different dances. This evaluation method is similar to the methodologies used for psychological studies [64]. Comparison with the human annotation of characteristic parts coincide by 76.6(±8.0)%, being quite similar. This shows that our abstraction method is reasonable.

4.4 Conclusions

In this chapter, we showed how the selective attentional sampling scheme can be applied for recognizing long and unannotatable motion streams such as a dance. For the recognition, we proposed a new motion feature flow generation method using local features and hierarchical low-dimensional embedding method in order to represent the motion changes as one dimensional feature flow. We designed the importance measuring function for sample selection and named the selected points as attentional motion spots (AMS). The AMSs are adaptively detected based on significant temporal changes in motion flow. Spatio-temporal groups of AMSs are modeled as weighted Gaussian models. The modeling results look similar to musical chart, so we named our model as *Action Chart*. In order to validate the proposed method, we generate a new complex action dataset; Pop-Dance dataset. The experimental results show that the selective attentional sampling strategy gives a promising recognition performance with a very low computational load. Also it can be used for abstracting a long video sequence.

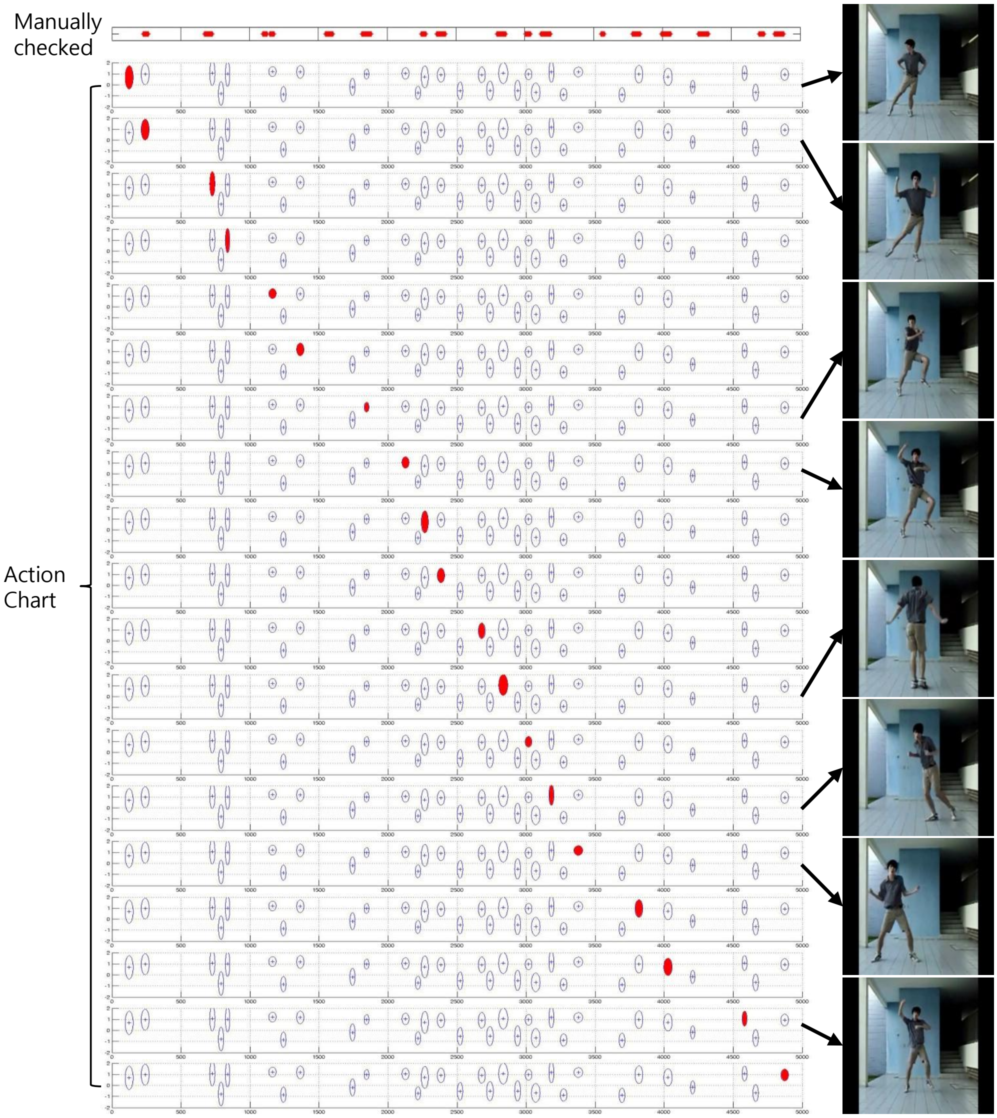


Figure 4.15: Video abstraction using generated *Action Chart* for “Gangnam Style” by Psy. The generated *Action Chart* are quite similar to manually checked points. The accuracy is 80.4% and the video is compressed to 23.2% in length.

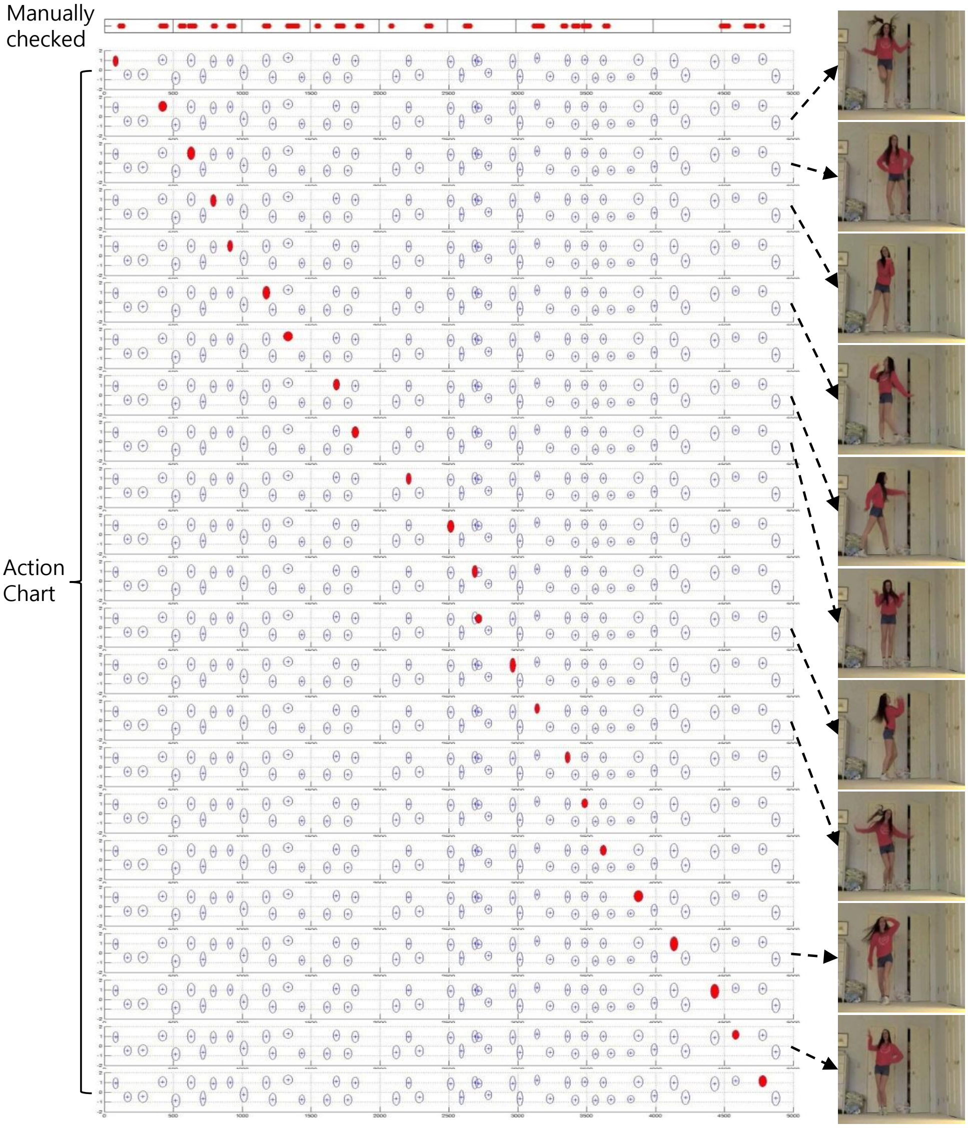


Figure 4.16: Video abstraction using generated *Action Chart* for “Gee” by SNSD. The generated *Action Chart* are quite similar to manually checked points. The accuracy is 69.7% and the video is compressed to 27.0% in length.

Chapter 5

Concluding Remarks

In this thesis, we proposed a generalized attentional sampling framework and defined the attentional sampling as three categories, *structured attentional sampling*, *empirical attentional sampling*, and *selective attentional sampling*. Each attentional sampling concept was explicitly defined and applied to computer vision applications. Although the potential gains of attentional sampling seemed very intuitive, there was a lack of understanding of its categories and properties. This thesis contributed to that understanding by clarifying in a general way when attentional sampling helps, and how much it helps. In the thesis it was shown that attentional sampling could dramatically improve performance and efficiency. The key contributions of this thesis are summarized as follows.

- Robust Tracking Failure Detection using Structure Attentional Sampling: Chapter 2 is about *structured attentional sampling* and its application to design a new scheme for detecting tracking failure moment. In order to mimick human visual sensing structure, log-polar transformation to tracking image is adopted. As a result, we could achieve a significantly improved TFD performance. Experimental results shows that our method could give much less false alarm and be more robust to target appearance change, and that our TFD method could be applied to any tracking methods.

- Speed-up of Background Subtraction using Empirical Attentional Sampling: Chapter 3 we showed how the *empirical attentional sampling* could be used to reduce computational load. We proposed a speed-up method of conventional background subtraction algorithms using active attention sampling mask generation method based on empirical attention concept. The motionless background region could be skipped by attention sampling. We designed a foreground probability map by measuring three foreground region properties, and active attention sampling was performed to make a sampling mask. Various experiments showed that the proposed method could speed up about 6.6 times without detection performance deterioration. Also our method made it possible for the conventional background subtraction algorithms to perform real-time detection in Full HD videos with a single core processor.

- *Action Chart* generation for recognizing Pop dances using Selective Attentional Sampling: In chapter 4 we proposed *Action Chart* for recognizing long and complex action sequences and its generation method using *selective attentional sampling.* We proposed a new motion feature flow and hierarchical low-dimensional embedding method. Attentional motion spots were adaptively selected based on significant temporal changes in motion flow and were modeled as weighted Gaussian models. For validation we built a new complex action dataset; Pop-Dance dataset. The experimental results showed that the *Action Chart* gave a promising recognition performance with a very low computational load by focusing only informative data region. Also it could be used for abstracting a long video sequence.

There are still many open questions regarding attentional sampling, in particular good “recipes” for the construction of realistic and general attentional sampling algorithms are still unknown. Also various applications are possible by properly combining different attentional sampling schemes depending on the problem. For example, speed-up of head detection in a video might be possible if the search space is drastically reduced by properly using the *structured attentional sampling* and the *empirical attentional sampling* together. This thesis opens up many avenues for future research.

Bibliography

[1] Asher Cohen, *Selective Attention*, John Wiley and Sons, Ltd, 2006.

[2] Rui M. Castro, *Active Learning and Adaptive Sampling for Non-Parametric Inference*, Ph.D. thesis, Rice University, Houston, TX, 2007.

[3] M. Gelgon, P. Bouthemy, and T. Dubois, "A region tracking method with failure detection for an interactive video indexing environment," *Lecture Notes in Computer Science (LNCS)*, vol. 1614, pp. 261–269, 1999.

[4] D.S. Bolme, J.R. Beveridge, B.A. Draper, and Y.M. Lui, "Visual object tracking using adaptive correlation filters," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010.

[5] C. Hua, H. Wu, Q. Chen, and T. Wada, "K-means tracker: A general algorithm for tracking people," *Journal of Mutimedia*, vol. 1, no. 4, pp. 46–53, July 2006.

[6] C. Stauffer and W.E.L Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999, pp. 246–252.

[7] T. Bouwmans, F. El Baf, and B. Vachon, "Statistical background modeling for foreground detection: A survey," *Handbook of Pattern Recognition and Computer Vision World Scientific Publishing*, vol. 4, pp. 181–199, Jan. 2010.

[8] Ahmed Elgammal, Ramani Duraiswami, David Harwood, Larry S. Davis, R. Duraiswami, and D. Harwood, "Background and foreground modeling using nonparametric kernel density for visual surveillance," in *Proceedings of the IEEE*, 2002, pp. 1151–1163.

[9] Zoran Zivkovic and Ferdinand van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Patten Recognition Letters*, vol. 27, no. 7, pp. 773–780, 2006.

[10] JinMin Choi, Yung Jun Yoo, and Jin Young Choi, "Adaptive shadow estimator for removing shadow of moving object," *Computer Vision and Image Understanding (CVIU)*, vol. 114, pp. 1017–1029, Sep 2010.

[11] Pushkar Gorur and Bharadwaj Amrutur, "Speeded up gaussian mixture model algorithm for background subtraction," in *8th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, Sept. 2011, pp. 386–391.

[12] Vu Pham, Phong Vo, Vu Thanh Hung, and Le Hoai Bac, "GPU implementation of extended gaussian mixture model for background subtraction," in *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2010 IEEE RIVF*, Nov. 2010.

[13] Johnny Park, Amy Tabb, and Avinash C. Kak, "Hierarchical data structure for real-time background subtraction," in *Proceeding of International Conference on Image Processing (ICIP)*, 2006.

[14] Dae-Youn Lee, Jae-Kyun Ahn, and Chang-Su Kim, "Fast background subtraction algorithm using two-level sampling and silhouette detection," in *Proceeding of International Conference on Image Processing (ICIP)*, Nov. 2009, pp. 3177–3180.

[15] J.K.Aggarwal and M.S.Ryoo, "Human activity analysis: A review," *ACM Computing Surveys*, vol. 43, no. 3, Apr. 2011.

[16] Kevin Tang, Li Fei-Fei, and Daphne Koller, "Learning latent temporal structure for complex event detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.

[17] Juan Carlos Niebles, Chih-Wei Chen, , and Li Fei-Fei, "Modeling temporal structure of decomposable motion segments for activity classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2010.

[18] Ivan Laptev, "On space time interest points," *International Journal of Computer Vision (IJCV)*, vol. 64, no. 2/3, pp. 107–123, 2005.

[19] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, October 2005.

[20] Daniel Weinland, *Action Representation and Recognition*, Ph.D. thesis, Institut National Polytechnique De Grenoble, oct 2008.

[21] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan, "Active learning with statistical models," *Journal of Artificial Intelligence Research*, vol. 4, pp. 129–145, 1996.

[22] Yoav Freund, H. Sebastian Seung, Eli Shamir, and Naftali Tishby, "Selective sampling using the query by committee algorithm," in *Machine Learning*, 1997, pp. 133–168.

[23] I. Saleemi, K. Shafique, and M. Shah, "Probabilistic modeling of scene dynamics for applications in visual surveillance," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 31, no. 8, pp. 1472–1485, Aug. 2009.

[24] R. Hebel and H.Hollander, "Size and distribution of ganglion cells in the human retina," *Anatomy and Enbryology*, pp. 125–136, 1982.

[25] V.J. Traver and A. Bernardino, "A review of log-polar imaging for visual perception in robotics," *Robotics and Autonomous Systems*, vol. 58, pp. 378–398, Apr. 2010.

[26] Keinosuke Fukunaga and Larry D. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, no. 1, pp. 32–40, Jan. 1975.

[27] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 24, no. 5, pp. 603–619, May 2002.

[28] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 25, pp. 564–577, 2003.

[29] J. Lim, D. Ross, R.S. Lin, and M.H Yang, "Incremental learning for visual tracking," in *Advances in Neural Information Processing Systems (NIPS)*. 2004, pp. 793–800, MIT Press.

[30] J. M. McHugh, J. Konrad, V. Saligrama, and P. M. Jodoin, "Foreground-adaptive background subtraction," *Signal Processing Letters, IEEE*, vol. 16, no. 5, pp. 390–393, May 2009.

[31] Dar-Shyang Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 27, 2005.

[32] Grzegorz Szwoch, "Performance evaluation of the parallel codebook algorithm for background subtraction in video stream," *Multimedia Communications, Services and Security, 2011, Springer*, vol. 149, pp. 149–157, 2011.

[33] Li Cheng, Minglun Gong, Dale Schuurmans, and Terry Caelli, "Real-time discriminative background subtraction," *IEEE Transactions on Image Processing*, vol. 20, no. 5, May 2011.

[34] Hyo-Kak Kim, Suryanto, Dae-Hwan Kim, Dongni Zhang, and Sung-Jea Ko, "Fast object detection method for visual surveillance," in *IEEE International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC) 2008*, 2008.

[35] Oren Griffiths and Chris J. Mitchell, "Selective attention in human associative learning and recognition memory," *Journal of Experimental Psychology General*, vol. 137, pp. 626–648, 2008.

[36] Sebastian Brutzer, Benjamin Hoferlin, and Gunther Heidermann, "Evaluation of background subtraction techniques for video surveillance," in *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, June 2011, pp. 1937–1944.

[37] Pierre-Marc Jodoin, Max Mignotte, and Janusz Konrad, "Statistical background subtraction using spatial cues," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 12, pp. 1758 –1763, Dec. 2007.

[38] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[39] Helmut Grabner Fabian Nater and Luc Van Gool, "Temporal relations in videos for unsupervised activity analysis," in *British Machine Vision Conference (BMVC)*, 2011.

[40] M. S. Ryoo and J. K. Aggarwal, "Recognition of composite human activities through context-free grammar based representation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, june 2006.

[41] M.S.Ryoo and J.K.Aggarwal, "Stochastic representation and recognition of high-level group activities," *International Journal of Computer Vision (IJCV)*, vol. 93, no. 2, pp. 183–200, June 2011.

[42] Christian Schuldt, Ivan Laptev, and Barbara Caputo, "Recognizing human actions: A local svm approach," in *International Conference on Pattern Recognition (ICPR)*, 2004.

[43] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *International Conference on Computer Vision (ICCV)*, oct. 2005.

[44] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: A large video database for human motion recognition," in *International Conference on Computer Vision (ICCV)*, nov 2011, pp. 2556 –2563.

[45] Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri, "Actions as space-time shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 29, no. 12, pp. 2247–2253, December 2007.

[46] Alireza Fathi and Greg Mori, "Action recognition by learning mid-level motion features," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, june 2008.

[47] T. Darrell and A. Pentland, "Space-time gestures," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 1993, pp. 335 –340.

[48] A. Veeraraghavan, R. Chellappa, and A.K. Roy-Chowdhury, "The function space of an activity," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 959 –968.

[49] Nazli Ikizler and Pinar Duygulu, "Human action recognition using distribution of oriented rectangular patches," in *Worshop on Human Motion, LNCS*, 2007, pp. 271–284.

[50] Ronald Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976 – 990, 2010.

[51] Pradeep Natarajan and Ramakant Nevatia, "Coupled hidden semi markov models for activity recognition," in *IEEE Workshop on Motion and Video Computing (WMVC)*, 2007.

[52] A. Quattoni, S. Wang, L. p Morency, M. Collins, T. Darrell, and Mit Csail, "Hidden-state conditional random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2007.

[53] Yong Rui and P. Anandan, "Segmenting visual actions based on spatio-temporal motion patterns," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000.

[54] Ajo Fod, Maja J. Matarić, and Odest Chadwicke Jenkins, "Automated derivation of primitives for movement classification," *Autonomous Robots*, vol. 12, no. 1, pp. 39–54, Jan. 2002.

[55] Liang Wang and David Suter, "Learning and matching of dynamic shape manifolds for human action recognition," *IEEE Transaction on Image Processing*, vol. 16, no. 6, june 2007.

[56] I. Laptev, M Marszalek, C Schmid, and B Rozenfeld, "Learning realistic human actions from movies," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, june 2008.

[57] Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, and Cordelia Schmid, "Evaluation of local spatio-temporal features for action recognition," in *British Machine Vision Conference (BMVC)*, 2009.

[58] William S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American Statistical Association*, vol. 74, no. 368, pp. pp. 829–836, 1979.

[59] Behrouz Saghafi and Deepu Rajan, "Human action recognition using pose-based discriminant embedding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 96 – 111, 2012.

[60] Jernej Barbic, Alla Safonova, Jia-Yu Pan, Christos Faloutsos, Jessica K. Hodgins, and Nancy S. Pollard, "Segmenting motion capture data into distinct behaviors," in *In Graphics Interface*, 2004, pp. 185–194.

[61] A. Elgammal and Chan-Su Lee, “Inferring 3d body pose from silhouettes using activity manifold learning,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[62] Cristian Sminchisescu and Allan Jepson, “Generative modeling for continuous non-linearly embedded visual inference,” in *International Conference on Machine Learning (ICML)*, 2004, pp. 759–766.

[63] Yoonho Hwang, Bohyung Han, and Hee-Kap Ahn, “A fast nearest neighbor search algorithm by nonlinear embedding,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.

[64] Jeffrey M. Zacks and Khena M. Swallow, “Event segmentation,” *Current Directions in Psychological Science*, vol. 16, no. 2, pp. 80–84, 2007.

[65] J. M. Zacks, T. S. Braver, M. A. Sheridan, D. I. Donaldson, A. Z. Snyder, J. M. Ollinger, R. L. Buckner, and M. E. Raichle, “Human brain activity time-locked to perceptual event boundaries,” *Nature Neuroscience*, vol. 4, no. 6, pp. 651–655, 2001.

[66] Thomas F. Shipley and Jeffrey M. Zacks, *Understanding Events: From Perception to Action*, Oxford University Press, Feb 2008.

[67] Veeraraghavan A, A.K. Roy-Chowdhury, and R. Chellappa, “Matching shape sequences in video with applications in human movement analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 27, no. 12, pp. 1896 –1909, dec 2005.

[68] A. Vedaldi and B. Fulkerson, “VLFeat: An open and portable library of computer vision algorithms,” `http://www.vlfeat.org/`, 2008.

[69] Chih-Chung Chang and Chih-Jen Lin, “LIBSVM: A library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[70] Alexei A. Efros, Alexander C. Berg, Greg Mori, and Jitendra Malik, “Recognizing action at a distance,” in *International Conference on Computer Vision (ICCV)*, 2003, pp. 726–733.

[71] M.D. Rodriguez, J. Ahmed, and M. Shah, “Action mach a spatio-temporal maximum average correlation height filter for action recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, june 2008.

[72] M. Marszalek, I. Laptev, and C. Schmid, “Actions in context,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, june 2009, pp. 2929 –2936.

[73] Kishore K. Reddy and Mubarak Shah, “Recognizing 50 human action categories of web videos,” *Machine Vision and Applications*, 2012.

[74] Tao Li, Shenghuo Zhu, and Mitsunori Ogihara, “Using discriminant analysis for multi-class classification: and experimental investigation,” *Knowledge and Information Systems*, vol. 10, no. 4, pp. 453–472, 2006.

[75] U. Gaur, Y. Zhu, B. Song, and A. Roy-Chowdhury, “A “string of feature graphs” model for recognition of complex activities in natural videos,” in *International Conference on Computer Vision (ICCV)*, 2011, pp. 2595–2602.

Appendix A

Derivation of Computational Efficiency Boundaries

In this appendix, we provide a derivation of the computational efficiency of the proposed method (C_P) comparing to the conventional full search method (C_F).

A.1 Definition of Notations

Let's start with defining some notations first.

- Notations for pixel numbers
  - Number of total pixels: N
  - Number of randomly scattered sampling pixels: $N_s = \rho N$
  - Number of seed pixels for adaptively expanding sampling: N_a (where $\mathbf{M}^t_{RS} = 1$)
  - Number of sampling pixels of active sampling mask: N_A (where the sampling mask $\mathbf{M}^t = 1$)

- Notations of algorithms
  - Ratio of foreground pixel to whole image: α $(0 \leq \alpha \leq 1)$
  - Standard deviation of α values through whole video: α_{std}
  - Spatially expanding constant: ω_s $(\omega_s = k\sqrt{N/N_s})$
  - Parameter of ω_s: k (usually k is $\sqrt{3}$ or $\sqrt{5}$)
  - Maximum spatially expanding width: ζ_{max}
- Notations for representing calculation costs
  - Cost of randomly scattered sampling for one pixel: C_r
  - Cost of adaptively expanding sampling for one pixel: C_a
  - Cost of updating foreground model for one pixel: C_f
  - Cost of foreground detection for one pixel: C_D
  - Cost of background subtraction using conventional full sampling: C_T
  - Cost of background subtraction using the proposed active sampling: C_P

A.2 Derivative of the efficiency boundary

We designed the *spatially expanding importance sampling* expands the sampling area proportional to the importance weight, and the expanding constant w_s is defined as $\omega_s = k\sqrt{N/N_s}$. Also the spatial expanding width ζ is proportional to foreground probability and expanding constant. Because the foreground probability is between 0 and 1, the ζ_{max} becomes ω_s when the probability is 1. The expected number of foreground pixels of current image becomes αN. Because the N_s pixels are sampled uniformly throught the whole image, we assume the N_a becomes as

$$N_a \approx \alpha \times N_s. \tag{A.1}$$

It is assumed that the final active sampling mask fully covers the foreground region ($\alpha N \leq N_A$). At the same time, the active sampling mask is usually smaller than the case of each of the N_as has maximum spatial expanding width ζ_{max}.

$$N_A \leq \zeta_{max}^2 N_a = \omega_s^2 N_a = (k^2 \frac{N}{N_s}) \times \alpha N_s = \alpha k^2 N. \tag{A.2}$$

Using the two inequality properties, we can get the following inequality.

$$\alpha N \leq N_A \leq \alpha k^2 N. \tag{A.3}$$

The α is a measurement value of each frame by background subtraction methods. However the background subtraction methods can not perfectly detect all the foreground pixels, so the α is essentially inaccurate. By considering this measurement inaccuracy, the inequality margin of A.3 is widened by α_{std} as follows.

$$(\alpha - \alpha_{std})N < N_A < (\alpha + \alpha_{std})k^2 N. \tag{A.4}$$

The final goal of this derivation is to find a boundary of computational ratio (C_P/C_T). The totally computational cost of background subtraction method using a conventional full sampling (C_T) is

$$C_T = N \times C_D \tag{A.5}$$

which does not require sampling mask generation cost. On the other hand, the calculation cost of the proposed active sampling based background method (C_P) is composed of several sub computational parts, such as randomly scattered sampling cost (C_rN_s), adaptively expanding cost (C_aN_A), foreground model update cost ($C_f(N_s - \frac{N_A}{N} \cdot N_s + N_A)$) and foreground detection cost ($C_D(N_s - \frac{N_A}{N} \cdot N_s + N_A)$).

$$\begin{aligned}
C_P &= C_rN_s + C_aN_A + C_f(N_s - \frac{N_A}{N} \cdot N_s + N_A) + C_D(N_s - \frac{N_A}{N} \cdot N_s + N_A) \\
&= C_rN_s + C_aN_A + C_fN_s - C_f\frac{N_A}{N} \cdot N_s + C_fN_A + C_DN_s \\
&\quad -C_D\frac{N_A}{N} \cdot N_s + C_DN_A \\
&= C_rN_s + C_fN_s + C_DN_s + C_aN_A + C_fN_A + C_DN_A \\
&\quad -C_f\frac{N_A}{N} \cdot N_s - C_D\frac{N_A}{N} \cdot N_s \\
&= N_s(C_r + C_f + C_D) + N_A(C_a + C_f + C_D - C_f\frac{N_s}{N} - C_D\frac{N_s}{N}) \\
&= N_s(C_r + C_f + C_D) + N_A(C_a + C_f + C_D - \frac{N_s}{N}(C_f + C_D)) \\
&= N_s(C_r + C_f + C_D) + N_A(C_a + (1 - \frac{N_s}{N})(C_f + C_D)). \qquad \text{(A.6)}
\end{aligned}$$

From A.6

$$N_A = \frac{C_P - N_s(C_r + C_f + C_D)}{C_a + (1 - \frac{N_s}{N})(C_f + C_D)}. \qquad \text{(A.7)}$$

We assume that every intermediate calculation costs such as C_r, C_a and C_f are smaller than foreground detection cost C_D.

$$C_r = \beta_rC_D \qquad \text{(A.8)}$$

$$C_a = \beta_aC_D \qquad \text{(A.9)}$$

$$C_f = \beta_fC_D \qquad \text{(A.10)}$$

$$0 < \beta_{min} \leq \{\beta_r, \beta_a, \beta_f\} \leq \beta_{max} < 1. \qquad \text{(A.11)}$$

By substituting (A.7) to (A.4), we can get

$$(\alpha - \alpha_{std})N < \frac{C_P - N_s(C_r + C_f + C_D)}{C_a + (1 - \frac{N_s}{N})(C_f + C_D)} < (\alpha + \alpha_{std})k^2 N. \tag{A.12}$$

First we consider the left inequality of (A.12).

$$(\alpha - \alpha_{std})N < \frac{C_P - N_s(C_r + C_f + C_D)}{C_a + (1 - \frac{N_s}{N})(C_f + C_D)} \tag{A.13}$$

$$(\alpha - \alpha_{std})N \cdot \left\{ C_a + (1 - \frac{N_s}{N})(C_f + C_D) \right\} < C_P - N_s(C_r + C_f + C_D) \tag{A.14}$$

$$\begin{aligned} C_P &> (\alpha - \alpha_{std})N \cdot \left\{ C_a + (1 - \frac{N_s}{N})(C_f + C_D) \right\} + N_s(C_r + C_f + C_D) \\ &> (\alpha - \alpha_{std})N \cdot \left\{ \beta_a C_D + (1 - \frac{N_s}{N})(\beta_f C_D + C_D) \right\} \\ &\quad + N_s(\beta_r C_D + \beta_f C_D + C_D) \\ &> (\alpha - \alpha_{std})N \cdot \left\{ \beta_{min} C_D + (1 - \frac{N_s}{N})(\beta_{min} C_D + C_D) \right\} \\ &\quad + N_s(\beta_{min} C_D + \beta_{min} C_D + C_D) \\ &> (\alpha - \alpha_{std})N \cdot \left\{ \beta_{min} C_D + (1 - \frac{N_s}{N})(1 + \beta_{min}) C_D \right\} \\ &\quad + N_s(1 + 2\beta_{min}) C_D \\ &> (\alpha - \alpha_{std})N C_D \cdot \left\{ \beta_{min} + (1 - \frac{N_s}{N})(1 + \beta_{min}) \right\} \\ &\quad + N_s(1 + 2\beta_{min}) C_D \\ &> N C_D \left[(\alpha - \alpha_{std}) \left\{ \beta_{min} + (1 - \frac{N_s}{N})(1 + \beta_{min}) \right\} + \frac{N_s}{N}(1 + 2\beta_{min}) \right]. \end{aligned} \tag{A.15}$$

By using (A.5)

$$(\alpha - \alpha_{std}) \left\{ \beta_{min} + (1 - \frac{N_s}{N})(1 + \beta_{min}) \right\} + \frac{N_s}{N}(1 + 2\beta_{min}) \leq \frac{C_P}{C_T}. \tag{A.16}$$

In the same way, the right inequality of (A.12) is

$$(\alpha + \alpha_{std})k^2 N > \frac{C_P - N_s(C_r + C_f + C_D)}{C_a + (1 - \frac{N_s}{N})(C_f + C_D)}. \tag{A.17}$$

$$(\alpha + \alpha_{std})k^2 N \cdot \left\{ C_a + (1 - \frac{N_s}{N})(C_f + C_D) \right\} > C_P - N_s(C_r + C_f + C_D). \tag{A.18}$$

$$\begin{aligned}
C_P &< (\alpha + \alpha_{std})k^2 N \cdot \left\{ C_a + (1 - \frac{N_s}{N})(C_f + C_D) \right\} + N_s(C_r + C_f + C_D) \\
&< (\alpha + \alpha_{std})k^2 N \cdot \left\{ \beta_a C_D + (1 - \frac{N_s}{N})(\beta_f C_D + C_D) \right\} \\
&\quad + N_s(\beta_r C_D + \beta_f C_D + C_D) \\
&< (\alpha + \alpha_{std})k^2 N \cdot \left\{ \beta_{max} C_D + (1 - \frac{N_s}{N})(\beta_{max} C_D + C_D) \right\} \\
&\quad + N_s(\beta_{max} C_D + \beta_{max} C_D + C_D) \\
&< (\alpha + \alpha_{std})k^2 N \cdot \left\{ \beta_{max} C_D + (1 - \frac{N_s}{N})(1 + \beta_{max}) C_D \right\} \\
&\quad + N_s(1 + 2\beta_{max}) C_D \\
&< (\alpha + \alpha_{std})k^2 N C_D \cdot \left\{ \beta_{max} + (1 - \frac{N_s}{N})(1 + \beta_{max}) \right\} \\
&\quad + N_s(1 + 2\beta_{max}) C_D \\
&< N C_D \left[(\alpha + \alpha_{std})k^2 \left\{ \beta_{max} + (1 - \frac{N_s}{N})(1 + \beta_{max}) \right\} + \frac{N_s}{N}(1 + 2\beta_{max}) \right].
\end{aligned} \tag{A.19}$$

$$\frac{C_P}{C_T} < (\alpha + \alpha_{std})k^2 \left\{ \beta_{max} + (1 - \frac{N_s}{N})(1 + \beta_{max}) \right\} + \frac{N_s}{N}(1 + 2\beta_{max}). \tag{A.20}$$

So we can get the computational efficiency of the proposed method comparing to the conventional full sampling method by combining two inequalities (A.16) and

(A.20) as

$$\begin{aligned}
&(\alpha - \alpha_{std}) \left\{ \beta_{min} + (1 - \frac{N_s}{N})(1 + \beta_{min}) \right\} + \frac{N_s}{N}(1 + 2\beta_{min}) \\
&< \frac{C_P}{C_T} < (\alpha + \alpha_{std}) k^2 \left\{ \beta_{max} + (1 - \frac{N_s}{N})(1 + \beta_{max}) \right\} + \frac{N_s}{N}(1 + 2\beta_{max}).
\end{aligned} \tag{A.21}$$

Using the $N = \rho N_s$, the final inequality becomes

$$\begin{aligned}
&(\alpha - \alpha_{std}) \{ \beta_{min} + (1 - \rho)(1 + \beta_{min}) \} + \rho(1 + 2\beta_{min}) \\
&< \frac{C_P}{C_F} < (\alpha + \alpha_{std}) k^2 \{ \beta_{max} + (1 - \rho)(1 + \beta_{max}) \} + \rho(1 + 2\beta_{max}).
\end{aligned} \tag{A.22}$$

국문 초록

컴퓨터 비전 문제는 영상 획득 장치를 통해 픽셀 단위로 수치화된 데이터를 샘플링 하는 것으로부터 시작된다. 가장 기본이 되는 데이터인 픽셀 값들을 그대로 사용하는 경우도 있고, 이 픽셀 값들을 조합하여 새로운 의미를 가진 데이터들을 구성하고 샘플링 하여 사용하기도 한다. 좋은 성능을 얻기 위해서는 최대한 많은 수의 데이터를 샘플링 하는 것이 필요하지만 이럴 경우 필요로 하는 연산량이 급격히 증가하는 문제가 있다. 반대로 연산량 만을 고려해 최소한의 데이터만 샘플링 하여 사용하는 경우 좋은 성능을 기대하기 어렵다. 그러므로 효율적인 연산량으로 최적의 성능을 얻기 위해서는, 이미지가 바뀜에 따라 혹은 시간이 흐름에 따라 문제를 풀기에 충분한 최소한의 데이터만 찾아내어 샘플링 하는 능동 샘플링(active sampling) 개념이 필요하다. 이러한 능동 샘플링 개념을 현실화하기 위해서는 문제를 해결하는데 중요한 데이터들을 찾아내는 과정이 매우 중요하며, 찾아낸 데이터들을 어떻게 집중하여 샘플링 하는가가 중요해진다. 본 논문에서는 서로 다른 세 가지의 주의집중 샘플링(attentional sampling) 방법, 즉 구조적 주의집중 샘플링(structured attentional sampling), 경험적 주의집중 샘플링(empirical attentional sampling), 선택적 주의집중 샘플링(selective attentional sampling)을 제안하였다. 제안된 각각의 주의집중 샘플링 방법들은 주의집중이 필요한 중요 데이터들을 찾기 위해 문제의 특성에 대한 사전 지식(prior knowledge)을 적용하는 세가지 방법을 제안하고 있으며, 그에 따라 적응적으로 샘플링 하는 방법들이다. 제안된 주의집중 샘플링 방법들은 컴퓨터 비전 문제들에 성공적으로 적용되어 연산 효율뿐만 아니라 각 알고리즘의 성능을 크게 향상 시켰다.

첫 번째 구조적 주의집중 샘플링(structured attentional sampling)은 문제의 특성에 맞춰 미리 구조화된 샘플링 패턴에 따라 샘플링을 수행하는 방법이다. 이러한 구조적 주의집중 샘플링 방법을 사람 눈의 구조를 흉내 내어 물체 추적

실패를 탐지하는 데 적용하였다. 사람 눈 망막 위의 시신경 세포(ganglion cells)의 분포를 근사화한 log-polar 패턴 구조로 이미지 픽셀 샘플링을 수행하여 사람 눈의 유용한 특성을 흉내 내었다. Log-polar 패턴으로 샘플링 된 이미지는 회전(rotation) 변화에 의한 영향은 감소되어 나타나고, 좌우나 위아래로의 병진(translation) 변화는 증폭되어 나타나는 특성이 있다. 이러한 특성은 회전에 의해 나타나는 포즈 변화들로 인해 발생하는 추적 실패에 대한 거짓 경보(false alarm)들은 줄이고, 급격한 위치 변화로 인한 추적 실패에 대한 참 경보(true alarm)를 증가시킬 수 있다. 게다가 log-polar 구조의 특징인 중심와(fovea) 선명화 특성(predominant property)은 초점이 맞춰진 중심 부분(추적 물체의 중심 부분)의 선명도는 증가시키고 그 이외의 주변부(추적 물체 바깥 부분)는 흐릿하게 함으로써 추적 실패의 순간을 정확하게 탐지할 수 있도록 도와준다. 또한 망막 위의 시신경 세포 하나하나는 log-polar 변환 이미지의 각 픽셀에 대응시켜, 각 세포가 빛에 적응하는 방식과 유사하게 각 픽셀의 추적 물체의 색상에 대한 적응을 가우시안 혼합 모델(Gaussian mixture model)을 이용하여 모델링 하였다. 이러한 방식으로 제안된 추적 실패 탐지를 위한 구조적 주의집중 샘플링의 유용성은 다양한 실험을 통해 검증되었다.

두 번째 경험적 주의집중 샘플링(empirical attentional sampling)은 이전에 획득된 경험적 지식을 현재 단계 샘플링에 사용하는 방식이다. 경험적 지식은 경험 학습 과정을 통하여 확률 분포로 모델링 된다. 이러한 경험적 샘플링 개념은 움직이는 물체 탐지를 위해 일반적으로 사용되는 배경 제거 방법들에 픽셀 단위의 선택적 연산 마스크를 적용하여 연산 속도를 향상시키는 방식으로 적용되었다. 제안된 샘플링 방법은 전경 지역(foreground region)과 같이 주의집중을 필요로 하는 영역에 초점이 맞춰져 샘플링이 진행되도록 설계되었다. 주의집중 영역은 전경 확률 지도(foreground probability map)로 표현되고, 이 확률 지도는 이전 프레임에서의 탐지 결과를 이용하여 재귀적(recursive) 확률 업데이트 방식으로 추정된다. 전경 확률 지도는 전경 부분의 시간적(temporal), 공간적(spatial), 주파수적(frequency) 특성을 이용하여 생성되었다. 생성된 전경 확률

지도를 이용하여, 무작위 샘플링(randomly scattered sampling), 공간 확장 방식의 중요 샘플링(spatially expanding importance sampling), 놀람 픽셀 샘플링(surprise pixel sampling) 방법들이 순차적으로 진행되면서 주의집중 샘플링 마스크를 생성한다. 제안된 경험적 주의집중 샘플링 방법의 효율성은 다양한 실험을 통해 검증되었다. 제안된 방법은 기존의 픽셀 단위의 배경 제거 방법의 연산 속도를 탐지 성능 저하 없이 약 6.6배 향상 시켰다. 또한 기존의 배경 제거 알고리즘을 이용하여 full HD 영상(1920x1080)에서 실시간으로 움직이는 물체를 탐지할 수 있도록 하였다.

선택적 주의집중 샘플링(selective attentional sampling)은 주어진 데이터와 목적에 대한 사전 정보를 이용하여 문제의 해결을 위해 꼭 필요로 하는 중요 데이터만 미리 선택하여 문제 해결의 효율성을 높이는 방식이다. 본 논문에서는 이러한 선택적 샘플링 방식을 이용하여 일반인이 추는 유명 대중가요의 춤을 인식하는 방법을 제안하였다. 대중가요 춤은 일반적으로, 발레나 리듬 체조의 춤 동작과는 달리 하나하나를 따로 이름을 붙일 수 없는 짧고 복잡하며 다양한 행동의 연속으로 나타난다. 특히 춤에 대한 일정한 제약이 없다 보니, 동작의 정확성보다는 추는 사람의 개성과 자유로움에 따라 동일한 춤도 다양하게 표현이 된다. 이러한 행동의 자유로움과 다양함, 그리고 시간적으로 긴 행동의 길이 때문에 기존의 행동 인식 알고리즘은 직접적으로 적용할 수 없다. 본 논문에서는 명확하게 구분할 수 없을 정도로 자유로운 행동의 흐름 특징을 효과적으로 표현하고 인식 알고리즘에 적용할 수 있도록 하기 위해 새로운 행동 특징 표현 방법을 제안하고, 이를 효과적으로 낮은 차원 데이터로 표현하는 방법을 제안하였다. 또한 효율적인 인식을 위해 특징적인 시공간적 행동의 변화 지점을 주의집중적 행동 지점(attentional motion spot)라 명명하고 이를 자동을 선택하는 방법을 제안하였다. 이 특징 점들의 시공간적 분포를 혼합 가우시안(Gaussian) 분포로 모델링하고, 이렇게 표현된 모델링 방법을 행동 악보(Action Chart)라고 명명하였다. 이 행동 악보는 시공간적인 행동의 흐름을 음악 악보처럼 중요 행동의 시간적 발생 지점과 종류, 지속 시간을 표현하고 있다. 이렇게 표현된 행동 악보를 이용하여

새롭게 제작된 대중 가요 춤 데이터 세트를 효율적이고 효과적으로 인식하였다. 제안된 방법을 검증하기 위하여 제안된 방법을 구성하는 세부 알고리즘 하나 하나를 실험적으로 검증하여 각 부분의 필요성을 보였고, 현재 존재하는 길고 복잡한 행동을 인식하는 방법을 직접 구현하여 동일한 데이터 세트를 이용하여 제안된 방법이 인식 성능과 연산 시간측면에서 월등히 뛰어남을 검증하였다. 또한 더 나아가 행동 악보를 이용하면 긴 춤 동작을 사람이 하는 것과 거의 유사한 성능으로 요약 가능함을 보였다.

주요어: 주의집중 샘플링, 구조적 주의집중 샘플링, 경험적 주의집중 샘플링, 선택적 주의집중 샘플링, 추적 실패 탐지, 배경 제거 알고리즘 속도 향상, 복잡 행동 인식, 대중가요 춤 인식

학번: 2006-21280

감사의 글

길고 길었던 학생의 길을 이제 마무리하려 합니다. 십 년 넘는 이곳 관악에서의 행복했던 삶도 이제는 접고, 새로운 인생의 시작점에 서 있는 지금 만감이 교차합니다. 오랜 시간 쏟았던 열정이 컸던 만큼 너무나도 소중하고 행복했던 대학원 생활이기에, 헤어짐을 준비하는 이 순간이 더욱더 큰 아쉬움으로 남는 것 같습니다. 그 동안 연구실 생활을 통해 배우고, 익히고, 깨달은 모든 것들 소중히 간직하고 발전시켜나가 더 나은 모습의 제가 되기 위해 노력하겠습니다. 칠 년의 땀과 노력으로 작은 결실을 맺어 마무리하는 이 순간 그 동안 사랑과 관심으로 지켜봐 주시고 소중한 추억을 함께 나눴던 분들께 이렇게 지면으로나마 감사의 마음을 전하고자 합니다.

우선 오랜 시간 한결같은 모습으로 참된 스승의 모습을 보여주신 최진영 교수님 감사합니다. 교수님을 만난 것은 하나님께서 주신 제 인생 가장 큰 축복 중에 하나인 것 같습니다. 교수님께서 가르쳐주신 것들 하나하나 가슴에 깊이 새기고, 부지런히, 하지만 애쓰지 않고 순간순간 감사하며 살겠습니다. 항상 믿어주시고, 격려해주시고, 조언해주시고, 이끌어주셔서 진심으로 감사합니다. 교수님을 본받아 좋은 리더, 좋은 스승이 되도록 노력하겠습니다. 항상 건강하시고 행복하시길 기도합니다.

또한 심사를 통해 많은 관심을 가져주시고 귀중한 조언들을 해 주신 최종호 교수님, 조남익 교수님, 오성회 교수님, 임종우 교수님께도 깊은 감사를 드립니다. 바쁘신 중에도 귀중한 시간을 내어주셔서 저의 부족한 논문을 마무리 지을 수 있었습니다. 감사합니다.

인지지능 연구실에서 매일매일 함께한 연구실 선후배 분들께 감사의 뜻을 전합니다. 항상 미소와 격려로 보듬어주시는 따뜻한 민수형, 회사 생활과 함께 졸업을 준비하느라 고생 많으셨습니다. 졸업 진심으로 축하 드립니다. 조용하면서도 약간은 엉뚱한 그러면서도 끈기 있는 선정이 누나 좋은 연구 결과 있길

기대합니다. 차분하며 배려심 깊은 매력남 수완이, 능력자라는 말이 누구보다 잘 어울리는 광무, 오랜 시간 함께 과제하느라 고생 정말 많았습니다. 좋은 논문 많이 쓰고 멋진 연구 성과를 내며 누구보다 빛날 앞날을 기대해 봅니다. 넉살 좋고 붙임성 좋으며 연구에 있어서는 누구보다 열정적인 하욱이, 번뜩이는 재치와 입담을 가진 한주, 조용하면서 꾸준한 문섭이, 관심 있는 일에는 누구보다 적극적인 기민이, 철저한 자기 관리 가운데 엉뚱함이 매력인 영준이, 넉살 좋고 인상 좋은 상두, 꼼꼼함과 끈기로 연구에 많은 도움을 준 고마운 지윤이, 항상 여유로워 보이는 병호, 연구실을 옮겨 힘들 테지만 열심히 하는 장욱이, 똘똘한 인도 친구 Tushar 그리고 새로운 식구 기경이와 병주 모두 함께 할 수 있어서 즐거웠습니다. 다들 좋은 연구 결과를 내며 즐겁게 연구실 생활 하시길 바랍니다.

소중한 추억을 함께 나누었던 여러 선배님들께도 감사의 마음을 전하고 싶습니다. 그 어느 때 보다 열정적으로 살았던 대학원 신입생 시절 힘든 과제를 함께하며 서로 마음을 나누었던 표재형과 동성이형, 부족한 저에게 많은 것들을 가르쳐주고 많은 얘기들 나누며 연구실 생활에 큰 버팀목이 되어준 우성이형, 진희형, 윤석민형 감사합니다. 함께 생활하며 많은 것을 배울 수 있었던 명수형, 정환이형, 한석민형, 진민이형, 호석이형, 영민이, 영은이 형에게도 감사합니다. 형님들이 있기에 든든했고, 함께해서 즐거웠습니다.

넥스리얼 형님들께도 감사의 마음을 전하고 싶습니다. 한결 같은 모습과 성실함으로 넥스리얼을 이끌어가시는 석호형, 비범한 문혁이형, 항상 따뜻하게 관심 가져주시고 조언을 아끼지 않아 주시는 홍석이형, 밝고 친근한 준석이형, 성실함의 대명사 인수형 모두 모두 감사합니다. 앞으로도 즐겁게 개발하시고, 즐겁게 사업하셔서 날로 날로 번창하시길 기원합니다.

인생의 절반 이상 함께하고 옆에서 응원해 줬던 소중한 친구 주인이와, 가장 젊었고 푸르렀던 대학생 시절 잊지 못할 추억과 즐거움을 함께했던 친구 상훈이, 홍래, 인재, 상익이에게도 고마움을 전하고 싶습니다. 연구실은 다르지만 연구에 많은 도움을 준 정찬이와, 학교 선배이자 힘들지만 재밌었던 훈련소 생활을 함께한 현구형과 동우형에게도 감사의 뜻을 전합니다.

아직은 학생이라는 신분이라 많이 부족한 저를 기쁘게 가족으로 받아주시고 늘 배려해 주시는 사랑하는 장인, 장모님, 그리고 우리 멋진 처남들 감사합니다. 그리고 제 인생의 가장 처음 순간부터 지금까지 가장 오랜 시간 아낌없는 사랑과 헌신으로 보살펴주신 사랑하고 존경하는 아버지, 어머니 그리고 바쁘다는 핑계로 많이 신경 써주지 못해 늘 미안하고 고마운 사랑하는 동생 욱진이에게 깊은 감사의 마음과 사랑을 전합니다.

마지막으로, 십 년 가까운 시간 동안 한결같이 옆에서 응원해주고 믿어주고 가장 큰 힘이 되어 준 사랑하는 아내 혜리와 세상 무엇과도 바꿀 수 없는 소중한 아들 지온이에게 가슴 속 깊은 감사와 사랑을 전하고, 모든 순간 순간 인도해주시고 동행해주신 사랑하는 하나님께 이 논문을 바칩니다.