Multimedia Signal Processing for Copyright and Privacy Protection

著作権とプライバシー保護のための マルチメディア信号処理

Yuta Nakashima

Division of Electrical, Electronic and Information Engineering
Graduate School of Engineering
Osaka University
Japan

2011

Preface

This dissertation presents the author's research work on multimedia signal processing for copyright and privacy protection, which was achieved during a Ph.D. course at the Division of Electrical, Electronic, and Information Engineering, Graduate School of Engineering, Osaka University. It is organized as follows:

Chapter 1 describes the problem of copyright and privacy infringement that has arisen with the recent popularization of mobile video cameras to clarify the purpose of this research work.

Chapter 2 introduces an overview of existing technologies for copyright and privacy protection, which is the focus of this research work. The characteristics of the approaches presented in this dissertation are clarified by comparing them with existing technologies.

Chapter 3 describes a digital audio watermarking-based approach for copyright protection against in-theater movie piracy, which is an illegal copy of a movie made in a theater using a mobile video camera. As a clue for identifying pirates, this approach estimates the pinpoint position in a theater where the movie was recorded based on the watermarks embedded in the movie soundtrack. To demonstrate the practical applicability of our approach, the accuracy of the positions estimated in an actual hall is experimentally evaluated, and the acoustic quality of the watermarked movie soundtracks is subjectively assessed by MUSHRA listing tests.

Chapters 4 and 5 deal with the problem of privacy infringement due to the disclosure of persons' appearance in videos. Generally, a video contains human objects in the video frames that correspond to persons. Some human objects are intentionally captured by camera persons while others are accidentally included in the frames. Since intentionally-captured human objects (ICHOs) are essential for the video and permission for capturing can be easily obtained from the persons corresponding to the ICHOs, privacy-protected videos are automatically generated where human objects except ICHOs (non-ICHOs) are obscured.

Chapter 4 presents a method for detecting ICHOs in videos that serves as a basis for automatically generating privacy-protected videos. Based on the observation that the camera persons' behavior in capturing ICHOs is reflected in the camera motion associated with the motion of each human object, all human objects are detected and classified into ICHOs/non-ICHOs using features related to camera and human object motions. The method's performance is evaluated using various videos.

Chapter 5 describes an ICHO detection-based approach for automatically generating privacy-protected videos. Such videos can be generated by first detecting non-ICHOs and obscuring them. However, since non-ICHOs are often only partially captured, which prevents their accurate detection, ICHOs are first detected and other regions are replaced with the estimated background. The validity of the ICHO-based approach is evaluated by a user study, and the quality of the generated privacy-protected videos is also evaluated using several videos.

Chapter 6 concludes this dissertation.

Acknowledgments

The research work described in this dissertation was carried out during my Ph.D. course at the Graduate School of Engineering, Osaka University.

First, I would like to express my deepest gratitude to my supervisor and the chair of my dissertation committee, Professor Noboru Babaguchi of the Graduate School of Engineering, Osaka University, for his patient instruction, encouragement, and valuable comments throughout this research. The countless lessons he has provided beyond the research work have guided and supported every aspect of my life. He has also provided me a number of opportunities that have become invaluable experiences.

I am grateful to my dissertation committee members, Professor Seiichi Sampei of the Graduate School of Engineering, Osaka University, Professor Takashi Washio of the Institute of Scientific and Industrial Research, Osaka University, and Dr. Naoko Nitta, Lecturer of the Graduate School of Engineering, Osaka University, who provided many insightful suggestions and critical comments on this dissertation.

I also thank Professor Kyo Inoue, Professor Zen-ichiro Kawasaki, Professor Ken-ichi Kitayama, Professor Shozo Komaki, and Professor Tetsuya Takine of the Department of Information and Communications Technology of the Graduate School of Engineering, Osaka University, and Professor Riichiro Mizoguchi of the Institute of Scientific and Industrial Research, Osaka University for their thoughtful comments.

Regarding the research of digital audio watermarking-based copyright protection, my sincere appreciation goes to Dr. Ryuki Tachibana of the Tokyo Research Laboratory of IBM Japan. This research is a joint work with him, and his innovative suggestions have guided this research to a successful completion. He has also taught me much about not only digital watermarking techniques but also programming, how to write scientific papers, and so forth.

For the research of intentionally-captured human object detection and privacy protection, I am grateful to Dr. Jianping Fan, Associate Professor of Department of Computer Science, University of North Carolina at Charlotte, who provided beneficial suggestions and comments.

I express my genuine appreciation to Mr. Masahiro Kobayashi, a lawyer at the Hanamizuki Law Office, who kindly explained various rights and legislations involving mobile video cameras and reinforced the related discussions in this dissertation.

I thank all the past and present members of the Media Integrated Communication Laboratory in the Department of Information and Communication Technology, Graduate School of Engineering, Osaka University (Babaguchi Laboratory). They offered warm encouragement and friendship that gave me strength through difficult times. I especially thank Dr. Yoshimichi Ito, Assistant Professor of the Graduate School of Engineering, Osaka University, who provided a number of suggestions on mathematics and mathematical notations. I also express my appreciation to Dr. Xiaoyi Yu, Associate Professor of the School of Software and Microelectronics, Peking University and Dr. Minh Son Dao, Postdoctoral Fellow of Department of Information Engineering and Computer Science, University of Trento for fruitful discussion about my research during their stay in our laboratory. I thank Mr. Ryosuke Kaneto, Mr. Hiroshi Uegaki, and Mr. Tatsuya Koyama with whom I have cooperatively completed papers involving this dissertation.

I am grateful to the Research Fellowships of the Japan Society for the Promotion of Science for Young Scientists, Initiatives for Attractive Education in Graduate Schools of the Division of Electrical, Electronic and Information Engineering, Graduate School of Engineering, Osaka University, and the Hara Research Foundation for financially supporting the research work. This research work was partly supported by a Grant-in-Aid for scientific research from the Japan Society for the Promotion of Science.

Finally, I express my highest gratitude to my parents for their understanding, their selfless support in all aspects of my life, their unlimited kindness, and their encouragement during my entire education.

Contents

1 Introduction 1

2 Technologies to Protect Copyright and Privacy 7

2.1 Introduction . . . 7

2.2 Copyright protection against movie piracy . . . 7

2.3 Privacy protection against disclosure of appearance . . . 13

2.4 Motivations and characteristics of our approaches . . . 18

2.4.1 System for in-theater movie piracy . . . 18

2.4.2 System for accidental privacy infringement . . . 19

2.5 Concluding remarks . . . 21

3 Copyright Protection Using Digital Audio Watermarking 23

3.1 Introduction . . . 23

3.2 Watermarking algorithm . . . 26

3.2.1 Preliminary . . . 26

3.2.2 Watermark embedder . . . 33

3.2.3 Watermark detector . . . 35

3.3 Position estimator . . . 37

3.3.1 Derivation of detection strength model . . . 38

3.3.2 Derivation of position estimator . . . 41

3.3.3 Maximization algorithm to reduce computational cost . . . 42

3.4 Experimental results . . . 44

3.4.1 Estimation accuracy evaluation . . . 45

3.4.2 Watermarking rate versus estimation accuracy . . . 47

3.4.3 Subjective evaluation of acoustic quality . . . 49

3.4.4 Discussion 52
3.5 Concluding remarks 53

4 Intentionally-Captured Human Object Detection 55
4.1 Introduction 55
4.2 Method for intentionally-captured human object detection 57
4.2.1 Human object detection 59
4.2.2 Feature extraction 59
4.2.3 Intentionally-captured human object model 63
4.2.4 Intentionally-captured human object classification incorporating temporal consistency 65
4.3 Experimental results 67
4.3.1 Contributions of features 68
4.3.2 Classification performance evaluation 71
4.3.3 Overall performance evaluation 73
4.3.4 Classification performance evaluation using larger dataset 75
4.3.5 Overall performance evaluation using larger dataset 76
4.4 Concluding remarks 78

5 Automatic Generation of Privacy-Protected Videos 79
5.1 Introduction 79
5.2 Preliminary user study 80
5.3 System for automatically generating privacy-protected videos 84
5.3.1 Background estimation 86
5.3.2 Intentionally-captured human object extraction 90
5.3.3 Intentionally-captured human object substitution 91
5.4 Experimental results 91
5.5 Concluding remarks 95

6 Conclusion 97

List of Figures

1.1 Example of accidental privacy infringement. Intentionally-captured and accidentally-framed-in persons are shown in blue and in red, respectively. Their motion is indicated by blue and red arrows. Camera person moves his/her camera dependent on motion of intentionally-captured person as indicated by black arrow. In this case, privacy of accidentally-framed-in person is infringed. 4

2.1 Flow of movie piracy. 8
2.2 Technologies for copyright protection against movie piracy. 12
2.3 Types of video cameras. 13
2.4 Selection of human objects to be obscured. 14
2.5 Systems for privacy protection against disclosure of persons' appearance. Halderman et al.'s and Brassil's systems are applicable to both fixed and mobile video cameras. 17
2.6 Intentionally-captured person (blue), accidentally-framed-in person (red), and camera person (gray). Corresponding intentionally-captured human object (ICHO) and human object except ICHO (non-ICHO), as well as example of video frame in privacy-protected video are also shown. . . 20

3.1 Scenario for identifying pirates. 24
3.2 Overview of our position estimation system. 25
3.3 Pattern block consists of $W_{\mathrm{B}} \times H_{\mathrm{B}}$ tiles (upper left). Tile is comprised of H_{T} amplitudes of four consecutive frame (upper right). Pattern blocks are arranged on time-frequency plane of host signal (HS) repeatedly (bottom). 29

3.4 Pseudo-random numbers assigned to tiles in pattern block (left). They form pseudo-random number array (PNA). Tile assigned with "+1" (right). In this case, among four consecutive frames of the tile, amplitudes in the first and second frames are increased (represented by "+") and those in the third and fourth frames are decreased (represented by "−"). 30

3.5 (a) Frames $t, t+1, t+2$, and $t+3$ form tiles along frequency axis. (b) Amplitudes of frames. (c) Original amplitudes of frames and amplitudes modified to embed watermark. (d) Difference between frames t and $t+2$. 32

3.6 Example of recorded signal (RS) and frames in watermark detector. . . 35

3.7 (a) Recorded signal (RS) containing multiple watermarks. Pseudo-random numbers assigned to tiles are also shown. (b) Detection strengths calculated from RS. (c) Detection strength blocks. 37

3.8 Examples of (a) $s^c(k)$ and (b) $\mathbf{m}_{k'}$. 39

3.9 Maximization algorithm. 43

3.10 Experimental environment for estimation accuracy evaluation. 44

3.11 Experimental setup. 45

3.12 Estimation errors in auditorium. 47

3.13 Relationship between watermarking rate and estimation error. Means and standard deviations of estimation errors are calculated for various α. 48

3.14 Listening position in room for (a). 51

3.15 Means and 95% confidence intervals for acoustic quality of test signals under (a). 52

3.16 Means and 95% confidence intervals for acoustic quality of test signals under (b). 52

3.17 Means and 95% confidence intervals for acoustic quality of test signals under (c). 53

4.1 Example of intentionally-captured human object (ICHO) surrounded by red rectangles and human object except ICHO (non-ICHO) surrounded by blue rectangles. 57

4.2 (a) Upper body region model and (b) examples of upper body regions. Upper body region is defined as region surrounded by rectangle of upper body region model when it is placed such that circle of the upper body region model surrounds the head of human object. 58

4.3 (a) Positive and (b) negative samples used to train support vector machine (SVM) for human object detection. 58

4.4 (a) Example of frame and (b) capture intention-related (CI) features extracted from human object. 61

4.5 Classification algorithm. 66

4.6 Area under ROC curve (AUC) values for combinations of features from capture intention-related (CI) features: number of used features $N_{\mathrm{F}}^{\mathrm{CI}}$ for (a), (b), (c) or (d) is 1, 2, 6, or 7, respectively. ■ and □ represent corresponding features are used and are not used, respectively. 69

4.7 Area under ROC curve (AUC) values for combinations of features from visual attention-related (VA) features: number of used features $N_{\mathrm{F}}^{\mathrm{VA}}$ for (a), (b), (c), or (d) is 1, 2, 6, or 7, respectively. ■ and □ represent corresponding features are used and are not used, respectively. 70

4.8 Area under ROC curve (AUC) values of intentionally-captured human object (ICHO) classification with ALG for various values of N_{TR} and TH_{ALG}. Horizontal axis is ξ such that $TH_{\mathrm{ALG}} = (2^{-N_{\mathrm{TR}}})^{\xi}$ (see text for details). 71

4.9 ROC curves of SVM-CI, SVM-VA, POST, ALG, and HUMAN over VD1. 72

4.10 Area under ROC curve (AUC) values of intentionally-captured human object (ICHO) detection with ALG for various parameter values using upper body detector. 73

4.11 False positives per frame versus true positive rate curves of SVM-CI, SVM-VA, POST, and ALG. 74

4.12 ROC curves of SVM-CI, SVM-VA, POST, and ALG over VD2. 75

4.13 False positives per frame versus true positive rate curves of SVM-CI, SVM-VA, POST, and ALG over VD2. 77

5.1 Examples of original frames from US1 (a) and US2 (b). 81

5.2 Methods for obscuring human objects except intentionally-captured human objects (non-ICHOs). 82

5.3 Means and standard deviations of acceptability of privacy disclosure for US1 (a) and US2 (b). 83

5.4 Means and standard deviations of adequacy for US1 (a) and US2 (b). . 84

5.5 Overview of our system using intentionally-captured human object (ICHO) detection and background estimation. 85

5.6 Example outputs of each component of our system. 88

5.7 Definitions of (a) Ω_{IH}, Ω_{NH}, and (b) Ω_{O}. 93

5.8 Averaged removal rate and (b) averaged preservation rate. 93

5.9 Example of output frames. For each video, whether intentionally-captured human objects (ICHOs) and human objects except ICHOs (non-ICHOs) are moving or stationary is indicated. 94

List of Tables

2.1 Comparison of position estimation systems. 19

3.1 Test samples used in experiments. 46
3.2 Experimental parameters. 46
3.3 Samples used in subjective assessment of acoustic quality. 49
3.4 Description of test signals used in subjective assessment of acoustic quality. 50
3.5 Summary of conditions under which acoustic quality was assessed. . . . 50

4.1 Values of parameters used in experiments. 67

List of Symbols

Symbol	Description
SV	sound velocity
SF	sampling frequency
W_{B}	number of tiles in pattern block along time axis
H_{B}	number of tiles in pattern block along frequency axis
H_{T}	number of amplitudes in tile along frequency axis
N_{CH}	number of channels
N_{F}	number of samples in audio frame
N_{DS}	number of detection strengths within the duration of a pattern block
N_{CI}	number of video frames for extracting the features for intentionally-captured human object detection
N_{TR}	number of video frames for which human object is tracked
T	number of audio or video frames
Δ	detection shift
x^{F}	horizontal position of the center of video frames
y^{F}	vertical position of the center of video frames
α	watermarking rate
$\omega^c(w,h)$	pseudo-random number for the tile at (w,h) in the c-th channel
$a^c_{\mathrm{HS}}(i)$	i-th sample of a host signal for the c-th channel in time domain
$a^c_{\mathrm{WS}}(i)$	i-th sample of a watermarked signal for the c-th channel in time domain
$a_{\mathrm{RS}}(i)$	i-th sample of a recorded signal
$A^c_{\mathrm{HS}}(t,f)$	f-th Fourier coefficient of the t-th audio frame of a host signal for the c-th channel
$A^c_{\mathrm{WS}}(t,f)$	f-th Fourier coefficient of the t-th audio frame of a watermarked signal for the c-th channel
$A_{\mathrm{RS},k}(t,f)$	f-th Fourier coefficient of the t-th audio frame of a recorded signal
$M^c(t,f)$	inaudible amount of amplitude modification for the f-th Fourier coefficient of the t-th audio frame
$\mathrm{Sign}^c(t,f)$	amplitude modification sign for the f-th Fourier coefficient of the t-th audio frame
$\rho_k(w,h)$	amplitude of the tile at (w,h)
$\overline{\rho}_k$	averaged amplitudes of tiles over a pattern block

$s^c(k)$	k-th detection strength for the c-th channel
$\mathbf{s}_n^c$	n-th detection strength block
S^c	whole detection strengths for the c-th channel
S	detection strengths obtained from recorded signal
β_n^c	parameter to control the peak's height in the n-th detection strength block for the c-th channel
$\boldsymbol{\mu}_{\mathrm{DS}}$	mean of detection strength block
Σ_{DS}	variance of detection strength block
$\mathbf{x}_{\mathrm{mic}}$	microphone position
$\mathbf{x}_{\mathrm{sp}}^c$	position of the loudspeaker for the c-th channel
c_{ref}	index of reference channel
$\kappa^{c_{\mathrm{ref}}}$	peak position of reference channel
$\bar{\kappa}^c(\mathbf{x}_{\mathrm{mic}})$	delay of the c-th channel relative to the c_{ref}-th channel
$\kappa^c(\mathbf{x}_{\mathrm{mic}}, \kappa^{c_{\mathrm{ref}}})$	time position of peaks of detection strengths of the c-th channel
B	set of β_n^c's
$\boldsymbol{\Theta}$	set of parameters for position estimation
$L(\boldsymbol{\Theta})$	likelihood function for position estimation
$L'(\boldsymbol{\Theta}')$	likelihood function in which B is eliminated
$\mathrm{win}(i)$	i-th value of sine window function
$\mathbf{m}$	averaged shape of detection strengths
$\mathrm{mop}(t)$	t-th value of modulus operator
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ and variance σ^2
$\mathbf{v}^{\mathrm{CI}}$	feature vector for capture intention-related features
$\mathbf{v}^{\mathrm{VA}}$	feature vector for visual attention-related features
$\mathbf{v}$	feature vector where $\mathbf{v} = (\mathbf{v}^{\mathrm{CI}}, \mathbf{v}^{\mathrm{VA}})$
$\mathbf{d}_\tau$	human object motion
$\bar{\mathbf{d}}$	human object motion averaged over successive N_{CI} video frames
$\bar{\mathbf{d}}'$	compensated human object motion averaged over successive N_{CI} video frames
ζ_t	scaling factor of camera motion for the t-th video frame
$\mathbf{c}_t$	translation of camera motion for the t-th video frame
$\bar{\mathbf{c}}$	translation of camera motion averaged over successive N_{CI} video frames
λ_τ	similarity between upper body regions in successive two video frames
$\varsigma(\lambda_\tau)$	sigmoid function used to extract capture intention-related features
ϕ_1	scaling for $\varsigma(\lambda_\tau)$
ϕ_2	bias for $\varsigma(\lambda_\tau)$
y	class label for intentionally-captured human object classification
$\mathbf{y}_{t,n}$	sequence of class labels for the n-th human object in the t-th video frame
$V_{t,n}$	sequence of feature vectors for the n-th human object in the t-th video frame
H	label to represent human object
$\mathbf{H}$	sequence of labels to represent human objects

TH_{PR}	threshold of calibrated probability
TH_{TR}	threshold of probability for tracking
TH_{ALG}	threshold of probability for classification algorithm
TH_{C}	threshold of classification algorithm
Q_t	set of SURF features in the t-th video frame
$ANN_{t,t'}$	set of SURF features in Q_t for which approximate nearest neighbors are found in $Q_{t'}$
$\tilde{ANN}_{t,t'}$	subset of $ANN_{t,t'}$ excluding outliers
$sim_{t,t'}$	similarity between the t-and t'-th video frames
$H_{t,t'}$	homography matrix from the t-th video frame to the t'-th video frame
R	set of representative frames
I_t	t-th video frame
$\mathcal{L}$	set of indexes of video frames used in background estimation
z_n	variable for index of video frame in $\mathcal{L}$ for the n-th grid
$\bar{z}_i$	variable representing whether the i-th pixel belongs to intentionally-captured human object
$\tilde{z}_i$	variable for index of video frame in $\mathcal{L}$ for the i-th pixel
$F_n^{\mathrm{BE}}(z_n)$	data term for background estimation
$G_{n,k}^{\mathrm{BE}}(z_n, z_k)$	smooth term for background estimation
$F_i^{\mathrm{IE}}(\bar{z}_i)$	data term for intentionally-captured human object extraction
$G_{i,j}^{\mathrm{IE}}(\bar{z}_i, \bar{z}_j)$	smooth term for intentionally-captured human object extraction
ν_1	parameter of the first term of $F_i^{\mathrm{IE}}(\bar{z}_i)$
ν_2	parameter of the second term of $F_i^{\mathrm{IE}}(\bar{z}_i)$
ϱ_1	parameter of the first term of $G_{i,j}^{\mathrm{IE}}(\bar{z}_i, \bar{z}_j)$
ϱ_2	parameter of the second term of $G_{i,j}^{\mathrm{IE}}(\bar{z}_i, \bar{z}_j)$
Ω_n	set of pixels in the n-th grid
$d_{\Omega_n}(z, z')$	distance of the Ω_n's in video frames associated with z and z'
AG	set of adjacent grids
AP	set of adjacent pixels
E^{BE}	energy to be minimized in background estimation
E^{IE}	energy to be minimized in intentionally-captured human object extraction
$\bar{I}_{\Omega_n}$	averaged pixel value over Ω_n
$\delta(z, z')$	function that gives 1 when $z = z'$ and 0 otherwise
IM_i	i-th pixel of an intention map

Chapter 1

Introduction

Recently, we have witnessed the rapid popularization of mobile video cameras including not only camcorders but also digital cameras and mobile phones with built-in cameras. According to the Cabinet Office of Japan [1], the household penetration rate of camcorders in Japan reached 40.0% in 2010. The penetration rate of mobile phones worldwide was 41.8% in 2006 and 78.0% in 2010 [2]. Gartner, an information technology research and advisory firm, claimed that nearly 50% of these mobile phones would have built-in cameras in 2006 and 81% in 2010 [3].

In addition, the recent popularization of such video sharing services as YouTube[1] and Dailymotion[2] enables camera persons to publish their own videos and distribute them worldwide. Current mobile phones with built-in cameras are especially capable of accessing the Internet. Using such mobile phones, camera persons can take a video and can immediately make it available on the Internet. Mobile video cameras have greatly simplified taking videos and sharing them through the Internet.

However, taking and publishing videos often involve the problems of *copyright* and *privacy*. Copyright is a set of rights that protects the authors of artistic works [4], which are defined in the copyright laws of most nations. Examples of the rights included in the copyright are the rights to permit or prevent the reproduction of artistic works and the distribution of the copies of artistic works. These rights are automatically granted to the authors of artistic works. Any type of artistic work is protected by copyright law, including books, musical compositions, movies, photos, and so forth. Therefore,

[1] http://www.youtube.com/

[2] http://www.dailymotion.com/

copying artistic works without permission infringes on the copyright, possibly resulting in financial losses for the authors.

Privacy involves hiding a person's identity or information about that person. In [5], privacy is divided into three groups: spatial, decisional, and informational. Spatial privacy means that a person's territory should not be invaded by others without permission. Decisional privacy means that a person has the right to make decisions without interference. Informational privacy means that a person has control over the acquisition, the disclosure, and the use of his/her personal information that can identify him/her. In the sense of informational privacy, such information of a person as height, weight, and blood type cannot be acquired, disclosed, or used without permission. Otherwise, a person's privacy is infringed on and he/she may suffer financial losses or mental distress.

Videos taken with mobile video cameras might capture artistic works and persons. For example, if a camera person uses his/her mobile video camera in a theater where a movie, a live musical performance, or a play is being performed, the video is considered a reproduction or a copy of the artistic work. Videos taken without the permission of the authors thus infringe on copyright. Even a video taken in a park or on a street can contain the appearance of persons and can be directly linked to their identities. The video can infringe on the persons' privacy if permission was not obtained.

Laws have been enacted in several nations to protect copyright and privacy. For example, in the United States in 2005, the Family Entertainment and Copyright Act banned the use of mobile video cameras in theaters. In Japan, a similar law has been enforced since 2007. To partly protect privacy, the United States enacted the Video Voyeurism Prevention Act of 2004. However, these laws do not always prevent copyright and privacy infringement because their actual enforcement is imperfect. Technologies are strongly required to protect copyright and privacy.

In this dissertation, we present systems to protect copyright and privacy. These systems must be designed to be easy to use without high initial costs. We aim to develop systems that can be used without modifying mobile video cameras and existing environments such as theaters. By focusing on video that consists of multimedia signals, i.e., audio signals and sequences of images or video frames, we adopt multimedia signal processing techniques that include digital watermarking, object detection, and object segmentation. Our systems act on captured videos so that the systems can be used without modifying mobile video cameras and existing environments, which lead to and increase their actual deployment without high initial costs.

For copyright protection, although several types of artistic works can be infringed on by capturing them with mobile video cameras, we focus on the problem of in-theater movie piracy where movies shown in theaters are captured by mobile video cameras. This is because the movies shown in the theaters themselves are multimedia signals, and thus we can also process them using multimedia signal processing techniques.

To protect copyright against in-theater movie piracy, several countermeasures have been proposed based on digital watermarking techniques, which embed a secret message into multimedia signals as a watermark. Usually a pirated movie, which is an illegally recorded movie, is broadcast through the Internet or other media. By embedding into the movie as a watermark an ID associated with the theater and the date the movie was shown, these systems can automatically find copies of pirated movies and identify the theater and date [6, 7]. However, since the most effective countermeasure against in-theater movie piracy is to identify pirates, our system precisely estimates the pirate position by specifying the pirate's seat. If used with a system that associates seats with the identities of those in them such as a ticketing system, our system can identify pirates.

Considering that a theater usually has at least three loudspeakers, we estimate the pirate position using the delays of a multiple-channel movie soundtrack. Our system embeds a different watermark into each channel of the movie soundtrack using a digital audio watermarking technique. If the movie soundtrack with watermarks is captured by a mobile video camera, the captured video's audio signal contains the watermarks. Using these watermarks, we can calculate the watermark delays, which are proportional to the distances from the loudspeakers in the theater to the microphone attached to the mobile video camera. From these delays, our system estimates the microphone position as the pirate position. In addition, we develop a position estimator based on the maximum likelihood method that statistically improves estimation accuracy.

For privacy protection, we present a system that automatically obscures persons in video frames. When camera persons take videos, they usually have capture intentions that categorize the persons in the videos into intentionally-captured and accidentally-framed-in persons. In Fig. 1.1, the camera person intentionally captures the person shown in blue, moving the mobile video camera to follow him as indicated by the black

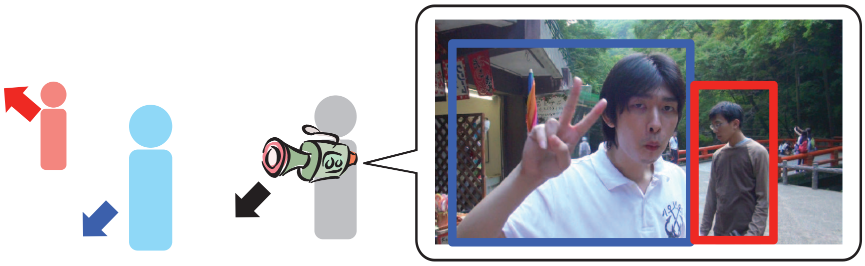


Figure 1.1: Example of accidental privacy infringement. Intentionally-captured and accidentally-framed-in persons are shown in blue and in red, respectively. Their motion is indicated by blue and red arrows. Camera person moves his/her camera dependent on motion of intentionally-captured person as indicated by black arrow. In this case, privacy of accidentally-framed-in person is infringed.

arrow. While capturing, the person in red is accidentally framed in. In most cases, the intentionally-captured person is the camera person's friend or family member, and thus, he/she can easily obtain permission to capture and publish from the intentionally-captured person. However, since the video contains the appearance of the accidentally-framed-in person, as indicated by the red rectangle, it infringes on his privacy, which is referred to as accidental privacy infringement. This is the most common type of privacy infringement. To protect the privacy of accidentally-framed-in persons, we need to obscure their appearance.

In conventional technologies for protecting the privacy of persons captured in videos [8, 9], the appearance of all persons in the video frames are obscured or persons whose appearance is obscured are determined based on their identities. However, these technologies are inappropriate for accidental privacy infringement because they do not consider the camera person's capture intention. In addition, technologies based on person identities cannot be used because obtaining them is extremely difficult under realistic environments.

Therefore, assuming that the camera persons can obtain permission for capturing and publishing from intentionally-captured persons, we realize a system that automatically generates privacy-protected videos. Hereinafter, the regions in the video frames corresponding to persons are called human objects. We refer to human objects who correspond to intentionally-captured persons as intentionally-captured human objects (ICHOs) and to human objects except ICHOs as non-ICHOs. In Fig. 1.1, the human objects surrounded by the blue and red rectangles are an ICHO and a non-ICHO.

Our system only obscures non-ICHOs and presents ICHOs to preserve camera persons' capture intentions.

More specifically, we develop a method for detecting ICHOs to obscure only non-ICHOs. Whether a human object is an ICHO is reflected in how camera persons move their mobile video cameras, i.e., camera motion. We first detect all human objects in the video frames and classify them into ICHOs/non-ICHOs using features related to camera motion. To obscure non-ICHOs, our system estimates the background of the video frames and substitutes it with ICHOs so that non-ICHOs can be obscured without explicitly detecting them, which is usually more difficult than detecting ICHOs.

The outline of this dissertation is as follows: In Chapter 2, we introduce existing technologies to protect copyright and privacy and clarify the characteristics of our approaches by comparing them with these existing technologies. Chapter 3 describes a digital watermarking-based system for copyright protection against in-theater movie piracy by estimating the pirate position. In Chapters 4 and 5, we address the problem of accidental privacy infringement and in Chapter 4 present a method for detecting ICHOs that serves as the basis of our system. In Chapter 5, we describe an ICHO detection-based system for automatically generating privacy-protected videos by only obscuring non-ICHOs. Chapter 6 concludes this dissertation and presents future directions.

Chapter 2

Technologies to Protect Copyright and Privacy

2.1 Introduction

In this chapter, we introduce the existing technologies for copyright and privacy protection. To demonstrate the uniqueness of copyright and privacy infringement caused by mobile video cameras, we present a wider range of technologies not only for in-theater movie piracy and accidental privacy infringement but also for other types of copyright and privacy infringement. For copyright infringement, we present the flow of movie piracy and introduce the existing technologies for protecting copyright against it. For privacy infringement, we first identify the factors that characterize the approaches for privacy protection and introduce the existing systems. We then describe the motivations and characteristics of our approaches for in-theater movie piracy and accidental privacy infringement.

2.2 Copyright protection against movie piracy

Movie piracy is a general term that represents various types of illegal copies of movies including in-theater movie piracy. In this section, we introduce the existing technologies for protecting copyright against movie piracy.

Figure 2.1 shows the flow of movie piracy, which consists of reproduction and dis-

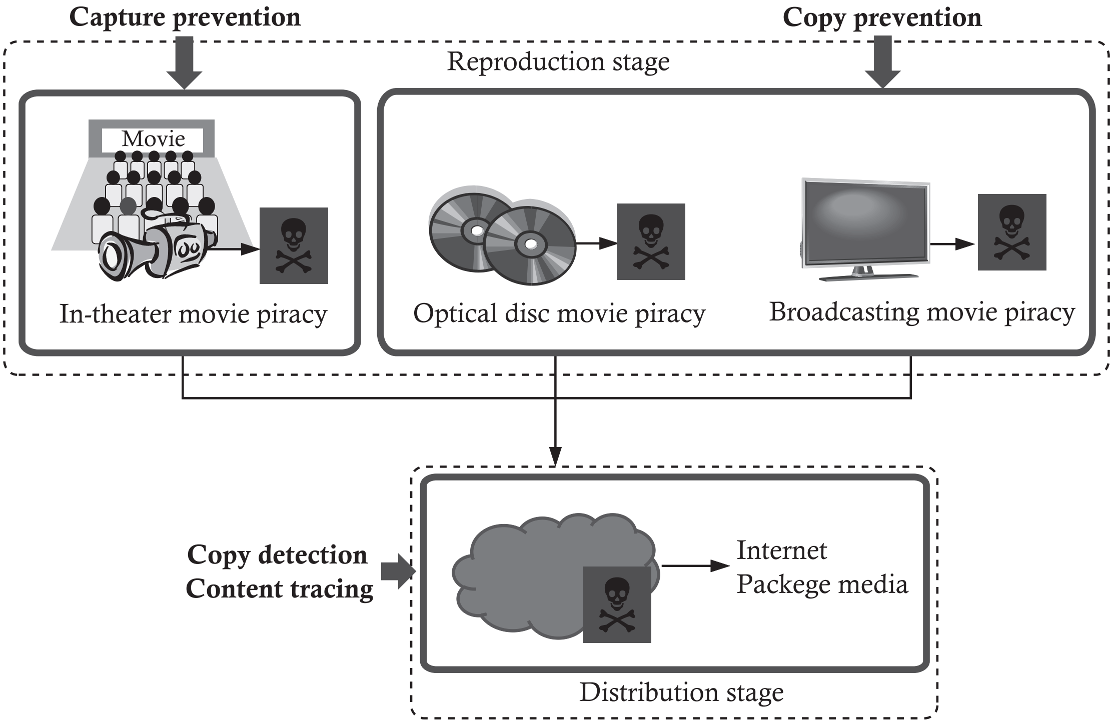


Figure 2.1: Flow of movie piracy.

tribution stages. First, at the reproduction stage, a movie is illegally copied to make a pirated version by capturing it with a mobile video camera (*in-theater movie piracy*), by copying an optical disc containing a movie such as a DVD or a blu-ray disc (*optical disc movie piracy*), or by copying a broadcast (*broadcasting movie piracy*). The pirated movie is distributed through the Internet or as package media at the distribution stage.

The existing systems for protecting copyright against movie piracy are categorized into four groups: (i) capture prevention, (ii) copy prevention, (iii) copy detection, and (iv) content tracing. Capture and copy prevention work at the reproduction stage, and the other two work at the distribution stage.

(i) Capture prevention: Capture prevention, which is a countermeasure against in-theater movie piracy, prevents pirates from capturing movies in theaters. Yamada et al. [10] focused on the difference of the sensitivities of the human visual system and mobile video cameras to infrared light and proposed emitting infrared light from behind a theater's screen. Infrared light is invisible to humans because our visual system is insensitive to it, but it is captured by mobile video cameras. This countermeasure significantly degrades the visual quality of pirated movies so that they cannot be used as a source of further copies. Capture prevention thwarts the act of capture itself, which Yamada et al.'s system fails to do. However, we view this system as capture prevention because it practically inhibits the reproduction of movies. Its limitations are that it can be overcome by infrared filters and devices to emit infrared light need to be installed in theaters.

(ii) Copy prevention: Copy prevention, which inhibits a pirate from copying movies from optical discs and broadcasts, addresses optical discs and broadcasting movie piracy. Most commercially-adopted systems fall into this group. For example, the content scrambling system (CSS) [11] and content protection for recordable media (CPPM) [12] are used for DVDs. For broadcasting, conditional access systems (CASs) have been adopted [13]. These systems are based on such encryption techniques as the advanced encryption standard (AES) [14]. An encryption technique scrambles a movie until it is decrypted with a valid decryption key, and thus, only authorized persons with the key can copy the movie. However, such copy prevention cannot be used for in-theater movie piracy because the movie is decrypted when shown in the theaters.

(iii) Copy detection: Copy detection finds pirated movies distributed through the Internet or other media. Using copy detection, the movie's authors can ask the owner of the pirated movie to remove it. Copy detection can be used for in-theater movie piracy, optical disc movie piracy, and broadcasting movie piracy since it works at the distribution stage (Fig. 2.1).

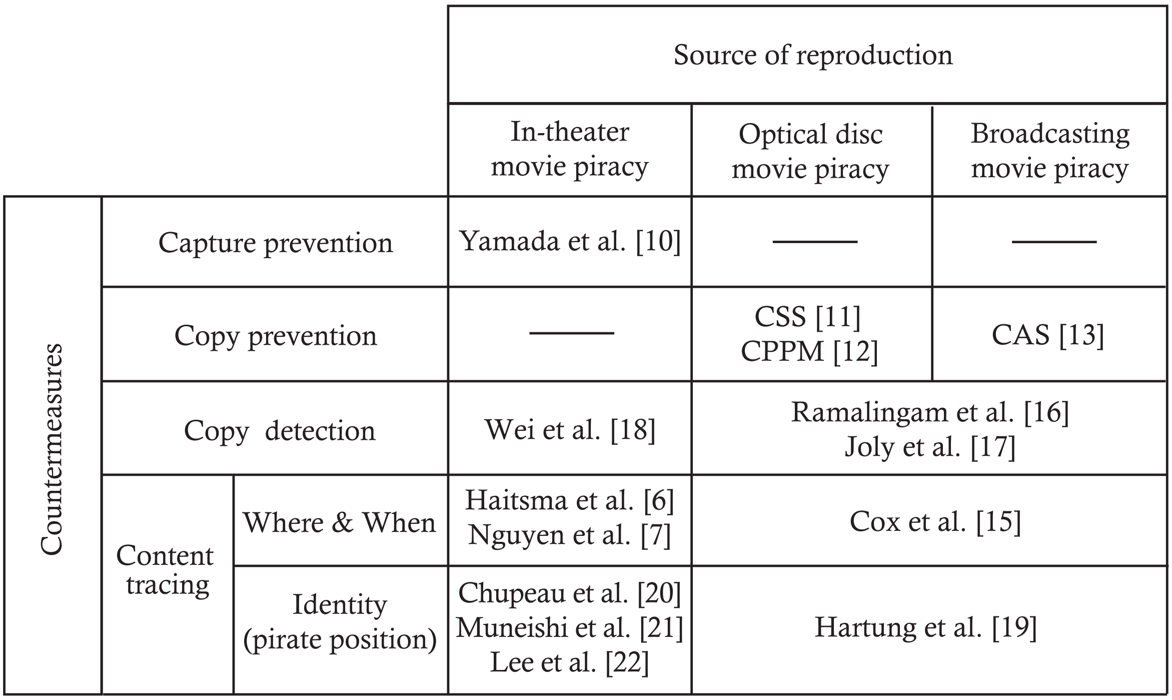
A digital watermarking technique, which embeds a secret message into a multimedia signal as a watermark, can be used for this purpose. A watermark is embedded into the audio signals or the video frames of a movie, and a video is judged as a pirated version if the watermark is detected in it. In optical discs and broadcasting, the movie is compressed or subjected to digital to analog (DA) and analog to digital (AD) conversion, which results in distortion. In addition, if the movie was captured with a mobile video camera, its audio signals and video frames are subjected to excessive distortion. Therefore, watermarks should be robust against such distortion. Cox et al. [15] proposed one of the most well-known watermarking algorithms based on the spread spectrum (SS) technique. Since this algorithm can be used when a movie is compressed or subjected to DA and AD conversion, it is applicable to copy detection for optical disc and broadcasting movie piracy. To realize copy detection for in-theater movie piracy where video frames projected on theater screens are subjected to excessive geometrical distortion caused by being captured with mobile video cameras, Haitsma et al. [6] proposed an algorithm that uses only the time axis of a movie's video frames. A method to compensate the geometrical distortion was proposed by Nguyen et al. [7] so that the watermark can be detected in the captured movie.

A fingerprinting technique is also applicable to copy detection for in-theater movie piracy, optical disc movie piracy, and broadcasting movie piracy. In contrast to the digital watermarking technique, this technique does not embed any message in a multimedia signal but extracts a unique fingerprint of the multimedia signal content from the multimedia signal itself. For copy detection, the fingerprinting technique first extracts the fingerprint from the original movie for which copy detection is required. The fingerprint is then extracted, e.g., from videos on the Internet. Since the fingerprinting technique can extract the same fingerprint as the original movie even from a distorted movie, a video is judged as a pirated version if the extracted fingerprint is identical to the original movie. Compared with the digital watermarking technique, one advantage of the fingerprinting technique is that modification of original multimedia signals is unnecessary, and thus they can be used without any preparation before the movie is shown in theaters or is sold in optical discs. A number of fingerprinting algorithms have been proposed. For example, algorithms for audio signals and video frames have been proposed by Ramalingam et al. [16] and Joly et al. [17]. These algorithms are applicable to copy detection for optical disc and broadcasting movie piracy. Wei et al. proposed an algorithm for video frames [18] and experimentally verified that it is applicable to video captured with a mobile video camera. Therefore, this algorithm is another countermeasure against in-theater movie piracy.

(iv) Content tracing: Content tracing, which identifies the origin of a pirated movie, can be categorized into two groups based on the following information obtained from it: (A) where and when the movie was copied or captured, including the source of the reproduction, i.e., in-theater movie piracy, the optical disc movie piracy, or the broadcasting movie piracy, and (B) the pirate who copied or captured it. The digital watermarking technique [6, 7, 15] can identify (A) by embedding an ID associated with (A) as a watermark. In addition, Hartung et al. [19] argued that the digital watermarking technique can be used to identify (B) for optical disc and broadcasting movie piracy if a different ID is embedded into each copy of the optical disc or into a movie delivered to a specific person through broadcasting, which can be achieved by embedding watermarks in, for example, set-top boxes. However, the conventional digital watermarking technique is incapable of identifying pirates for in-theater movie piracy because we cannot embed a different watermark for each person in the same theater.

To help identify pirates for the content tracing for in-theater movie piracy, several techniques for pirate position estimation have been proposed. Chupeau et al. [20] exploited the geometrical distortion of a movie in captured videos. The video frames projected on the screen are captured in the video with geometrical distortion; i.e., the screen, which is actually a rectangle, is deformed in the captured video. Since the geometrical distortion differs depending on the pirate position, it can provide a clue to the pirate's position. The geometrical distortion is estimated based on correspondences between the feature points in the pirated movie and the original. Muneishi and Iwakiri [21] take a similar approach. However, they need another technique to detect pirated movies and to identify the theater and date on which the movie was shown. Lee et al. [22] estimated the pirate's position using a watermark embedded into the video frames. One of the main characteristics of this technique is that it can find feature points used for estimating the geometrical distortion based on the watermark embedded into the video frames of the movie without accessing the original movie. Since these techniques only estimate the direction from the center of the screen toward the pirate, they need the theater's seating arrangement to specify the seat. We can identify pirates using pirate position estimation with a system that associates the seat with the person who was in the seat. A ticketing system might make such associations. Otherwise, at a movie premiere, most seats are reserved for specific persons, and the associations between the seats and the persons can be leveraged to identify the pirate. Another potential approach is to use surveillance cameras that capture persons in the theater. The problem of privacy infringement can be alleviated by combining pirate position estimation because only the region corresponding to the pirate can be selectively presented.

Figure 2.2 summarizes the existing technologies. For optical disc and broadcasting movie piracy, the same technologies can be used for copy detection and content tracing.

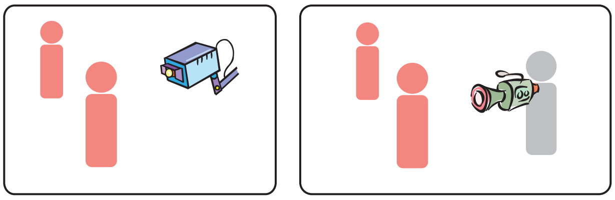


			Source of reproduction		
			In-theater movie piracy	Optical disc movie piracy	Broadcasting movie piracy
Countermeasures	Capture prevention		Yamada et al. [10]	———	———
	Copy prevention		———	CSS [11] CPPM [12]	CAS [13]
	Copy detection		Wei et al. [18]	Ramalingam et al. [16] Joly et al. [17]	
	Content tracing	Where & When	Haitsma et al. [6] Nguyen et al. [7]	Cox et al. [15]	
		Identity (pirate position)	Chupeau et al. [20] Muneishi et al. [21] Lee et al. [22]	Hartung et al. [19]	

Figure 2.2: Technologies for copyright protection against movie piracy.

In contrast, for in-theater movie piracy, which excessively distorts a movie, we need technologies that are robust against such excessive distortion. For copy detection, the fingerprinting algorithms [16, 17] are applicable to optical disc and broadcasting movie piracy, and Wei et al.'s algorithm [18] can be used for in-theater movie piracy. The digital watermarking-based technologies [6, 7, 15] are also applicable to copy detection. Similarly, the technologies [19, 22] for identifying pirates or for estimating their positions can be used for copy detection and for identifying where and when the movie was pirated.

As seen in Fig. 2.2, many technologies have been proposed for copyright protection against movie piracy. Capture and copy prevention are fundamental approaches because they inhibit the reproduction of movies. However, capture prevention [10] can be avoided, as mentioned above, and the systems for copy prevention [11, 12, 13] are also avoidable once the decryption keys are disclosed. Thus countermeasures in the distribution stage are important. Among them, we consider content tracing, which identifies pirates or estimates their positions, the most effective countermeasure. For in-theater movie piracy, techniques have been proposed for pirate position estimation



(a) Fixed video camera (b) Mobile video camera

Figure 2.3: Types of video cameras.

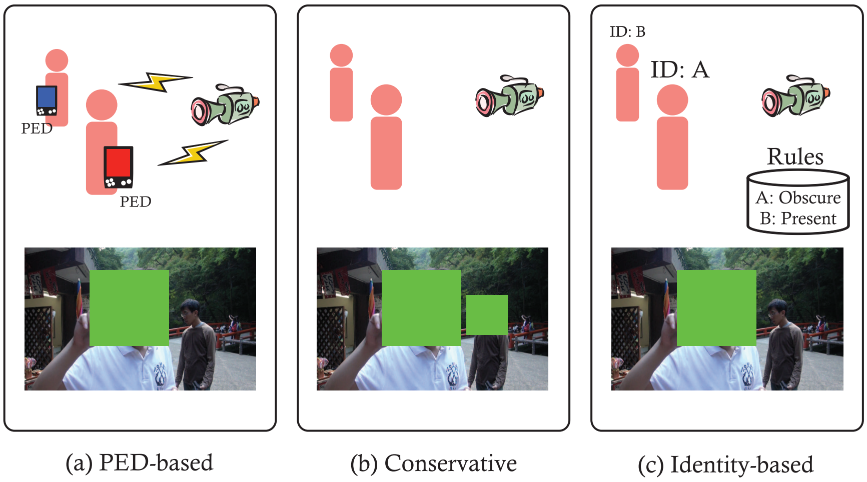
that utilize geometrical distortion [20, 21, 22]. However, the main problem of these technologies is that the geometrical distortion is likely to be compensated for the visibility after the movie is captured, which fundamentally results in failure of the pirate position estimation.

2.3 Privacy protection against disclosure of appearance

To protect privacy in videos, many systems have been proposed. In this section, we introduce the two main factors that characterize them and present existing systems with respect to these factors.

Generally, a system for privacy protection for videos first finds regions in the video frames, which correspond to persons, referred to as human objects. Then, some human objects are selected to be obscured. One crucial factor to characterize a privacy protection system is the type of video cameras. Usually, finding human objects in video frames greatly depends on the type of video cameras. For privacy protection, we categorize the systems to protect privacy for videos into two groups with respect to the type of the video cameras: (A) *fixed* or (B) *mobile*. They are shown in Figs. 2.3 (a) and (b).

Another important factor of a privacy protection system is the selection of human objects to be obscured because it determines the persons whose privacy will be protected in the videos. From the viewpoint of the selection of human objects to be obscured, the existing privacy protection systems for videos can be categorized into three groups as shown in Figs. 2.4 (a)–(c): (i) privacy enabling device (PED)-based, (ii) conservative, and (iii) identity-based systems.



(a) PED-based (b) Conservative (c) Identity-based

Figure 2.4: Selection of human objects to be obscured.

We describe the existing systems for privacy protection based on the selection of human objects to be obscured. The differences caused by the type of video cameras is described within each group of the selection of human objects.

(i) Privacy enabling device-based system: A PED-based system determines the human objects to be obscured based on PEDs, which inform the video cameras of the presence of the persons through wireless communications, assuming that the persons have PEDs with them, as shown in Fig. 2.4 (a). Halderman et al. [23] proposed a system that scrambled an entire video frame using an encryption technique until all persons captured in the video agree to be disclosed. Brassil's system [24] obscures only human objects based on PEDs. In Fig. 2.4 (a), the person who sets his PED to prevent capturing, indicated by red, is obscured, and the other person who permit to be captured, indicated by blue, is presented. One of the main advantages of PED-based systems is that the persons can control whether their appearance is disclosed using PEDs even while they are being captured. In addition, the system is potentially applicable to any type of video camera. However, the assumption that the persons are carrying the PEDs is impractical and the video cameras must be modified to be compliant with the system.

(ii) Conservative system: Assuming that no person grants permission for capturing and publishing, a conservative system obscures all human objects (Fig. 2.4 (b)).

A system in this group usually involves such surveillance tasks as security in public spaces. For example, Park et al. [25], Chattopadhyay et al. [26], and Li et al. [27] obscured all human objects for such surveillance tasks with fixed video cameras.

A conservative system can also be applied to mobile video cameras. Google Street View images are captured with mobile video cameras. For these images, Frome et al. [28] and Flores et al. [29] proposed systems to obscure all human objects. For life-log video cameras, with which people record their personal experiences, Chaudhari et al. [9] proposed a system in this group.

To obscure all human objects, we need techniques to find them. Usually different approaches are taken for fixed and mobile video cameras. For systems using fixed video cameras [25, 26, 27], a background subtraction technique, which identifies moving objects as human objects, is adopted. This technique first constructs a background model, in which the color of each pixel in the video frames without moving objects is represented by a probability distribution. Since the camera is fixed, the background model can be easily constructed. When a moving object appears in a video frame, background subtraction finds the objects by comparing each pixel with the background model and finding pixels that are different from the model. A number of background models have been proposed. For example, Wren et al. [30] modeled pixel color with Gaussian distribution. For further flexibility in the fluctuation of the pixel colors caused by waving trees, e.g., background models have been proposed that represent a pixel by the Gaussian mixture model (GMM) [31, 32] or by a non-parametric model [33].

For systems using mobile video cameras [28, 29], we need alternative techniques to find human objects since adoption of the background subtraction technique is difficult because of the camera motion. In this case, a human object detection technique, which finds human objects based on their appearance, is used instead. Many algorithms have been proposed for human object detection. Extensive surveys can be found in [34, 35]. Here, we introduce the most well-known and widely adopted algorithms [36, 37]. Generally, many algorithms for human object detection first extract features that represent well human objects from a region of a video frame and determine whether the region corresponds to a human object using a classifier trained with training data. Since this process is repeated for various window sizes and positions, object detection is usually computationally expensive. Viola and Jones [37] used simple features that represent well each part of a face, such as eyes, noses, or mouth. They also developed a fast algorithm to calculate the features from regions of various sizes as well as a computationally-efficient classifier to realize fast detection of faces. Dalal and Triggs [36] proposed features that are suitable to represent pedestrians in a video frame. Since Viola and Jones [37] use features representing each part on a face, their algorithm is sensitive to face orientations; i.e., a classifier, which is trained to find frontal faces, does not find profile faces. In contrast, the features in [36] represent the rough shape of pedestrians and thus are insensitive to their orientations.

(iii) Identity-based system: In this group, human objects are detected using background subtraction or human object detection techniques and are selectively obscured based on the identities of the persons that correspond to the human objects and predetermined rules (Fig. 2.4 (c)). The identities of the persons can be obtained using radio frequency identification (RFID) tags and readers or face recognition techniques that identify people from corresponding human objects. The rules are preliminarily determined before using the system based on, for example, whether a person is authorized to enter a restricted area in a building [38]. If the person is authorized, the corresponding human object is obscured. Identity-based systems resemble PEDs-based systems in the sense that the appearance of a person is obscured while the appearance of others is presented; however, in identity-based systems, people cannot control the disclosure of their appearance while they are being captured.

This group is mainly used for surveillance tasks in such specific environments as offices and hospitals to which only limited persons have access. Several systems for fixed video cameras are included in this group [38, 39, 40, 41]. These systems adopt background subtraction techniques to find all human objects. The systems proposed by Wickramasuriya et al. [38], Zhang et al. [39], and Yu et al. [40] use RFID tags and readers while Tansuriyavong and Hanaki's system [41] adopts a face recognition technique [42].

A system for mobile video cameras was proposed by Kitahara et al. [8]. This system supposes a specific environment in which fixed video cameras and RFID readers are installed and people have RFID tags. Using the videos from fixed video cameras, it finds human objects with a background subtraction technique and projects their positions into the view of the mobile video camera to locate human objects. The identities of the persons corresponding to human objects are obtained using RFID

		Type of video cameras	
		Fixed video cameras	Mobile video cameras
Selection of human objects to be obscured	PED-based	Halderman et al. [23] Brassil [24]	
	Conservative	Park et al. [25] Chattopadhyay et al. [26] Li et al. [27]	Frome et al. [28] Flores et al. [29] Chaudhari et al. [9]
	Identity-based	Wickramasuriya et al. [38] Zhang et al. [39] Yu et al. [40] Tansuriyavong et al. [41]	Kitahara et al. [8]

Figure 2.5: Systems for privacy protection against disclosure of persons' appearance. Halderman et al.'s and Brassil's systems are applicable to both fixed and mobile video cameras.

tags, and the human objects to be obscured are determined based on their identities and predetermined rules.

The main problem of the systems in this group is that they require a technique to identify the persons such as RFID readers and tags or face recognition techniques, all of which impose a strict limitation on the availability of the systems; face recognition techniques also remain error-prone.

The above discussion is summarized in Fig. 2.5. A relatively large number of systems have been proposed for fixed video cameras. In contrast, the number of systems that address privacy protection for mobile video cameras is small. One reason for this difference is the difficulty in selecting human objects to be obscured. Most systems for mobile video cameras are designed for such special applications as Google Street View [28, 29] and life-log video cameras [9]. In them, the assumption that all human objects should be obscured is reasonable.

However, when a camera person takes a video, he/she usually has a capture intention, i.e., what he/she wants to present in the video. In this case, the video becomes meaningless if all human objects are obscured because the capture intention is spoiled. Generally, in such a video, the persons captured in the video are divided into intentionally-captured and accidentally-framed-in persons. In many cases, the intentionally-captured persons are friends and family members, and thus, permission for capturing and publishing can be obtained. However, accidentally-framed-in persons are usually passers-by from whom obtaining permission is difficult; therefore, the video might infringe on their privacy, which is referred to as accidental privacy infringement. For accidental privacy infringement, the assumption in [9, 28, 29] is not reasonable because intentionally-captured persons can be presented.

2.4 Motivations and characteristics of our approaches

In this section, we describe the motivations and characteristics of our approaches for in-theater movie piracy and accidental privacy infringement.

2.4.1 System for in-theater movie piracy

To protect copyright against in-theater movie piracy, the most effective countermeasure is pirate position estimation [20, 21, 22]. In this dissertation, we present a digital audio watermarking-based system to help identify pirates. Our system estimates their positions in theaters by precisely specifying their seats using watermarks embedded into movie soundtracks. As mentioned in Section 2.2, pirate position estimation, which utilizes the geometrical distortion of pirated movies [20, 21, 22], fails when geometrical distortion is compensated for visibility. In contrast, our system embeds watermarks into multiple-channel movie soundtracks to calculate the delay of the audio signal of each channel emitted from a separate loudspeaker. Therefore, we can estimate the pirate position as long as the mobile video camera receives audio signals from at least three loudspeakers.

In addition to [20, 21, 22], systems for estimating indoor positions using various technologies have been proposed. We investigated the applicability of these systems to pirate position estimation. Table 2.1 summarizes the systems for indoor and pirate position estimation with respect to estimation accuracy (Accuracy), the technology used to estimate the positions (Technology), and the applicability to in-theater movie piracy (Applicability). Yim et al. [43] used the received signal strengths of WLAN.

Table 2.1: Comparison of position estimation systems.

System	Accuracy	Technology	Applicability
Yim et al. [43]	3 m	WLAN	Inapplicable
LANDMARK [44]	1 m	RFID	Inapplicable
Cricket [45]	0.02 m	Ultrasonic	Inapplicable
Chupeau et al. [20]	Approx. 1 m	Feature point of images	Applicable
Muneishi et al. [21]	Approx. 1 m	Feature point of images	Applicable
Lee et al. [22]	Approx. 1 m	Digital image watermarking	Applicable
Our system	0.44 m	Digital audio watermarking	Applicable

Ni et al.'s system called LANDMARK [44] uses RFID tags deployed in the target environment. However, these systems require the mobile video cameras to be modified so that they can receive WLAN or RFID signals and can record them for position estimation. Therefore, these systems are practically inapplicable to in-theater movie piracy. Priyantha et al. [45] proposed an ultrasonic-based system called Cricket that has outstanding accuracy. Although low-frequency ultrasonic that is inaudible to the human auditory system can be captured by microphones attached to mobile video cameras, it can be easily filtered out from recorded audio signals without significantly degrading their audible part. To the best of our knowledge, no system estimates microphone position using audible audio signals except ours. Note that the accuracy of these systems cannot be directly compared with each other. Some [43, 44, 45] estimate three-dimensional positions, but ours estimates two-dimensional positions. Other systems [20, 21, 22] estimate the direction of the pirate and calculate three-dimensional positions based on a theater's seating arrangement as mentioned in Section 2.2. Our system can be easily modified to estimate three-dimensional positions based on seating arrangements. In this case, accuracy does not change significantly because the vertical position can be uniquely determined given the two-dimensional horizontal position in the theater, and thus the estimation of three-dimensional positions can be reduced to the estimation of two-dimensional positions.

2.4.2 System for accidental privacy infringement

Generally, when a camera person takes a video with a mobile video camera, the camera person has a capture intention, which divides the persons captured in the video into

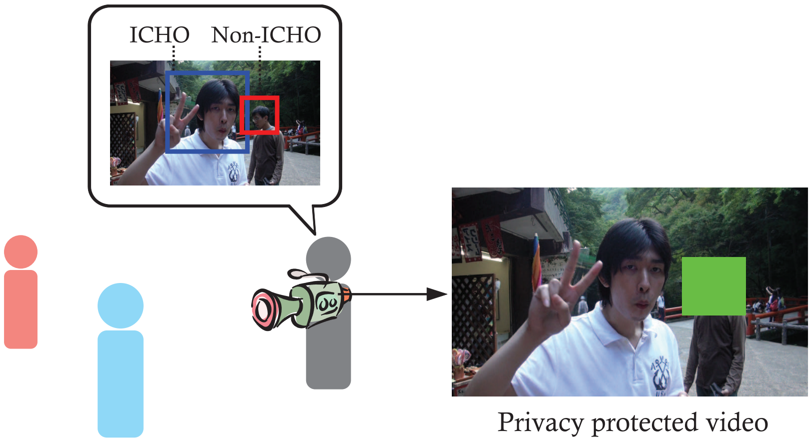


Figure 2.6: Intentionally-captured person (blue), accidentally-framed-in person (red), and camera person (gray). Corresponding intentionally-captured human object (ICHO) and human object except ICHO (non-ICHO), as well as example of video frame in privacy-protected video are also shown.

two types: intentionally-captured and accidentally-framed-in persons (Fig. 2.6). The regions in the video frames that correspond to the intentionally-captured persons are called intentionally-captured human objects (ICHOs), and those that correspond to the accidentally-framed-in persons are called non-ICHOs.

In accidental privacy infringement, which is a problem peculiar to videos taken with mobile video cameras, the disclosure of non-ICHOs infringes on the privacy of accidentally-framed-in persons. In many cases, the intentionally-captured persons are friends or family members. Therefore, permission for capturing them or publishing the video can be easily obtained, or at least, the camera person can negotiate for such permission. In contrast, accidentally-framed-in persons might simply be passers-by from whom obtaining permission is difficult. Hence, non-ICHOs should be obscured (Fig. 2.6). However, as summarized in Fig. 2.5, no existing system considers this point.

In this dissertation, assuming that permission for capturing and publishing is obtained from intentionally-captured persons, we present a system that automatically generates privacy-protected videos where only non-ICHOs are obscured. By considering whether human objects are ICHO/non-ICHO is reflected in the camera motion, our system classifies human objects into ICHOs/non-ICHOs based on camera motion and obscures the non-ICHOs. The following are the advantages of our system: (i) In contrast to identity-based systems, the human objects to be obscured can be determined without RFID readers and tags or face recognition techniques. (ii) Since the ICHOs are presented in the privacy-protected videos, the camera person's capture intention can be maintained. (iii) Even though PED-based systems [23, 24] can potentially achieve more flexible privacy protection than our system, they require modification of mobile video cameras and assume the penetration of PEDs. Our system does not require such modification or assumption.

2.5 Concluding remarks

In this section, we introduced the existing technologies for copyright and privacy protection. By comparison with these existing technologies, we investigated the uniqueness of our approaches. For in-theater movie piracy, we adopt a digital audio watermarking technique that precisely estimates pirate positions as long as theaters have at least three loudspeakers. For accidental privacy infringement, camera persons' capture intentions are considered to determine the human objects to be obscured to protect the privacy of accidentally-framed-in persons.

Chapter 3

Copyright Protection Using Digital Audio Watermarking

3.1 Introduction

With the technical advances in mobile video cameras, in-theater movie piracy, where movies are captured from theaters to make pirated movies, has become a serious problem. The Motion Picture Association claims that the annual loss from pirated movies exceeds six billion dollars and that over 90% of the pirated movies can be traced to in-theater movie piracy [46, 47]. In-theater movie piracy is explicitly banned in many nations. For instance, in the United States, the Family Entertainment and Copyright Act, which became a law in 2005, bans mobile video cameras in theaters. In Japan, in response to the significant losses of box-office revenues, a similar law has been enforced since 2007 that prohibits capturing movies even for private use, which was permitted by previous copyright law.

Several technologies against in-theater movie piracy have been proposed for copy detection and content tracing [6, 7]. However, the most efficient countermeasure is to identify the pirates. To this end, we consider the following scenario (Fig. 3.1): (i) A pirate illegally captures a watermarked movie and uploads it to the Internet. (ii) A conventional digital watermarking-based system such as [6] finds the pirated movie and analyzes the embedded message to determine the theater and date it was recorded. (iii) A position estimation system estimates the pirate's position in the theater precisely

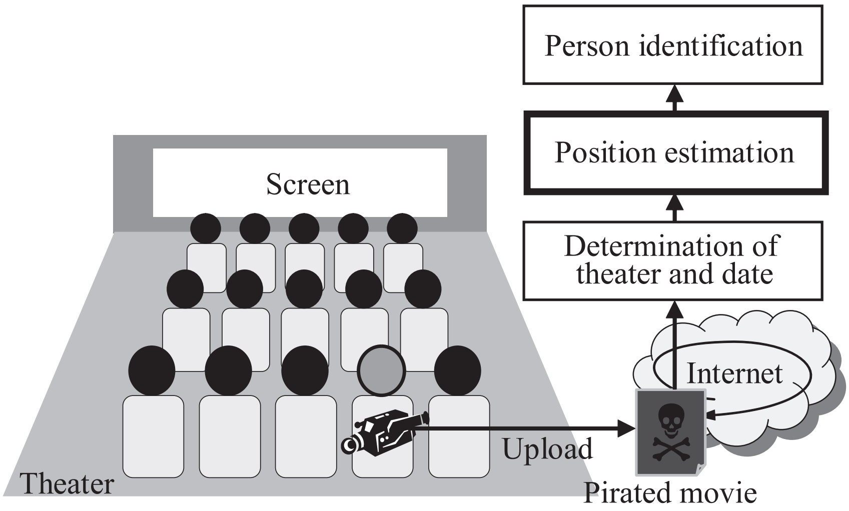


Figure 3.1: Scenario for identifying pirates.

enough to specify the seat. (iv) A person identification system identifies the pirate by associating the seat with the person in it. A ticketing system or a video surveillance system may be used as the person identification system. In this chapter, we focus on the position estimation system surrounded with thick, black lines in Fig. 3.1, which is a key component of this scenario.

Our system embeds watermarks into movie soundtracks for estimating pirate positions, even though this is difficult. In fact, most digital watermarking-based systems designed as countermeasures against in-theater movie piracy embed watermarks into the video frames of movies. The difficulty comes from the nature of movie soundtracks, which are composed of several types of audio, such as music, sound effects, voices, and silent parts. The voices and silent parts dominate movie soundtracks. In these parts, the watermarks cannot be embedded sufficiently to accurately estimate the pirate position because they are embedded by modifying the movie soundtracks and the amount of modification is limited in the voices and silent parts to maintain acoustic quality. Therefore, our system statistically improves estimation accuracy by exploiting the long duration of movie soundtracks instead of embedding the watermarks sufficiently for accurate position estimation with spoiling the acoustic quality.

An overview of our system is shown in Fig. 3.2 that explains how our system works. Since a movie soundtrack consists of multiple channels, a theater has multiple loudspeakers. We refer to each channel of the movie soundtrack as a *host signal* (HS).

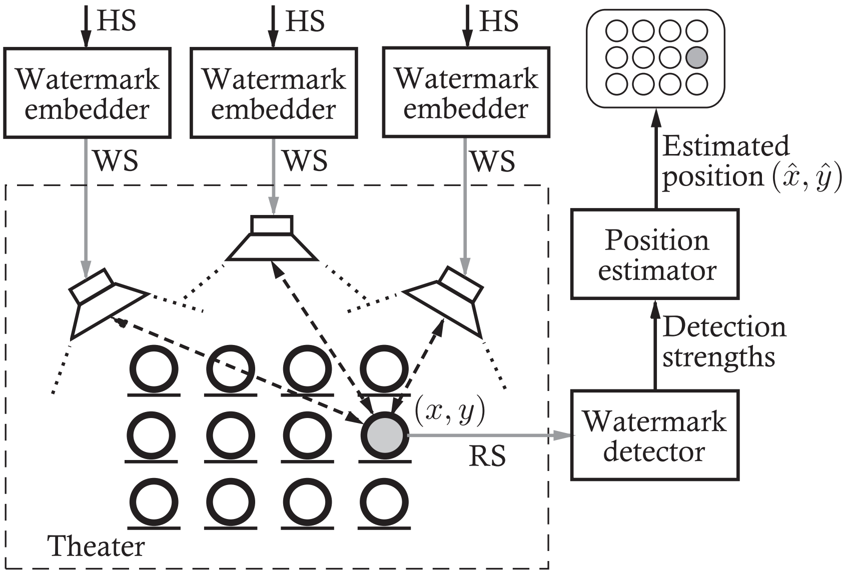


Figure 3.2: Overview of our position estimation system.

Our watermarking algorithm is based on the spread spectrum technique, which uses pseudo-random numbers to embed a watermark. We use different pseudo-random numbers for each HS. The watermark embedder generates a watermark for each HS and embeds it into the HS. The HS with the watermark is referred to as a *watermarked signal* (WS). Each WS is emitted into the air from a separate loudspeaker. If the movie is captured with a mobile video camera to make a pirated movie, the audio signal of the pirated movie, which we call a *recorded signal* (RS), is a monaural signal consisting of a mixture of all WSs. In the RS, the WS from each loudspeaker is delayed in proportion to the distance from the loudspeaker to the mobile video camera's microphone. For each watermark, the watermark detector calculates the *detection strengths*, which are defined as the correlations between the pseudo-random numbers of the watermark and the RS. Thus, the detection strengths form a peak at a particular time dependent on the delay. We construct a probabilistic model of the detection strengths with respect to the microphone position and estimate it as the pirate position using a position estimator based on the maximum-likelihood method.

In the following sections, we describe the watermarking algorithm, i.e., the watermark embedder and the watermark detector in Section 3.2. Section 3.3 presents the position estimator. We experimentally evaluate the accuracy of the estimated positions as well as the acoustic quality of the WSs in Section 3.4. Concluding remarks are given in Section 3.5. This chapter is related to the work published in [48, 49, 50, 51, 52, 53, 54, 55, 56, 57].

3.2 Watermarking algorithm

In this section, we first describe digital audio watermarking techniques and introduce the basic ideas of our watermarking algorithm, which is based on [58]. We then present our watermarking algorithm for pirate position estimation.

3.2.1 Preliminary

A movie consists of a sequence of video frames and a multiple-channel movie soundtrack. Each channel of the movie soundtrack is an audio signal. Many digital watermarking techniques have been proposed as countermeasures against movie piracy, as introduced in Section 2.2. In this section, we describe the digital watermarking techniques for audio signals, which are called digital audio watermarking, because we adopt them to estimate a pirate's position.

A watermarking algorithm consists of a watermark embedder and a watermarking detector. An audio signal into which a watermark is embedded is called a HS. The watermark embedder embeds a secret message into the HS as a watermark, generating a WS. The WS may be subjected to such distortion as cropping, pitch shifting, noises, and compression. Some types of distortion are caused by malicious attack to destroy the watermark. The distortion can also be caused by capturing the movie with a mobile video camera. Digital to analog (DA) and analog to digital (AD) conversion as well as propagation of the WS in air significantly distorts the WS. The watermark is then detected by the watermark detector.

The following are the requirements for a watermarking algorithm as a countermeasure against movie piracy:

Robustness: The WS can be subjected to various types of distortion. The watermark in the WS should be robust against such distortion so that it can be detected even after the WS is distorted.

Inaudibility: The watermark is embedded into the movie soundtrack by modifying it. Therefore, to maintain the movie's quality, modification to embed it should not degrade the HS's acoustic quality. In other words, the watermark should be inaudible.

Generally, the HS is represented using pulse-code modulation (PCM) where the analog audio signal is sampled and quantized. Each sample is stored as a binary number. The simplest algorithm to embed a watermark modifies the least significant bits of each sample. That is, a secret message embedded into the HS is encoded in binary numbers, and each bit of the message replaces one of the least significant bits of the samples. A watermark by this algorithm is inaudible if the least significant bits to be modified are appropriately chosen. However, this algorithm's watermark is not robust because it can be easily destroyed by changing the least significant bits. For example, the watermark does not survive DA and AD conversion, which significantly changes the least significant bits.

Considering these disadvantages of the algorithm based on the modification of the least significant bits, spread spectrum (SS)-based algorithms have been proposed [15], which use pseudo-random numbers to embed a watermark. Let $a_{\mathrm{HS}}(i)$ and $\omega(i) \in \{+1, -1\}$ denote the i-th sample of the HS and the i-th pseudo-random number. One of the most basic algorithms can be represented by

$$a_{\mathrm{WS}}(i) = a_{\mathrm{HS}}(i) + \alpha\omega(i), \tag{3.1}$$

where $a_{\mathrm{WS}}(i)$ is the i-th sample of the WS. Parameter α is a *watermarking rate* that controls the watermark's robustness and inaudibility. The watermark embedded using (3.1) cannot contain a secret message. However, it can be adopted to content tracing to identify where and when the movie was shown because we can use arbitrary pseudo-random numbers to embed the watermark and the pseudo-random numbers themselves can be the ID associated with the theater and date. In addition, this algorithm can be easily extended to contain a secret message.

A watermark by this basic algorithm is detected by

$$\sum_i a_{\mathrm{WS}}(i)\omega(i) = \sum_i a_{\mathrm{HS}}(i)\omega(i) + \sum_i \alpha\omega(i)^2. \tag{3.2}$$

The second term becomes large compared with the first term when the summation is calculated for a sufficient number of samples or α is sufficiently large. If the watermark detector is applied to an audio signal without the watermark, the second term vanishes. Therefore, an audio signal can be judged to be watermarked if the value given by (3.2) exceeds a certain threshold. This algorithm can be used in such transformed domains as the frequency domain.

A watermark by this algorithm is robust against such distortion as noises and compression. However, this algorithm has the following drawbacks: (i) The watermark's inaudibility can be spoiled if a large α is used. (ii) To correctly detect the watermark by (3.2), we need strict synchronization between the WS and the pseudo-random numbers; i.e., the watermark cannot be detected if $a_{\mathrm{WS}}(i)$ is delayed even by one sample.

For drawback (i), a SS-based algorithm that exploits the properties of the human auditory system (HAS) has been proposed [59]. HAS is insensitive to acoustic stimuli immediately before and after loud acoustic stimuli, which are called temporal masking effects. In addition, when HAS receives acoustic stimuli with a certain frequency, it is insensitive to acoustic stimuli with similar frequencies, which are called frequency masking effects. This algorithm modifies $\alpha\omega(i)$ based on a model of such effects, which is called a psychoacoustic model, to make the watermark inaudible.

Drawback (ii) is especially serious for SS-based algorithms in the time domain because a delay of a WS by only one sample makes the watermark undetectable, as mentioned above. To overcome this drawback, we need to repeatedly apply (3.2) by shifting the pseudo-random numbers to synchronize them with the WS, which is computationally expensive. The synchronization drawback is not serious for algorithms in the frequency domain because a delay by several samples hardly changes the frequency components of the WS.

Another serious problem related to synchronization can be caused by pitch shifting, which can be applied to a WS as a malicious attack. Pitch shifting, which resamples the WS in a different sampling frequency and stores the resampled WS in the original sampling frequency, changes its duration and pitch. For a watermark in the time domain, this results in continuous change of the synchronization position along the time. Even in the frequency domain, pitch shifting makes the watermark undetectable because the WS's frequency components are altered.

Pitch shifting is not likely to be applied to movie soundtracks because the change in

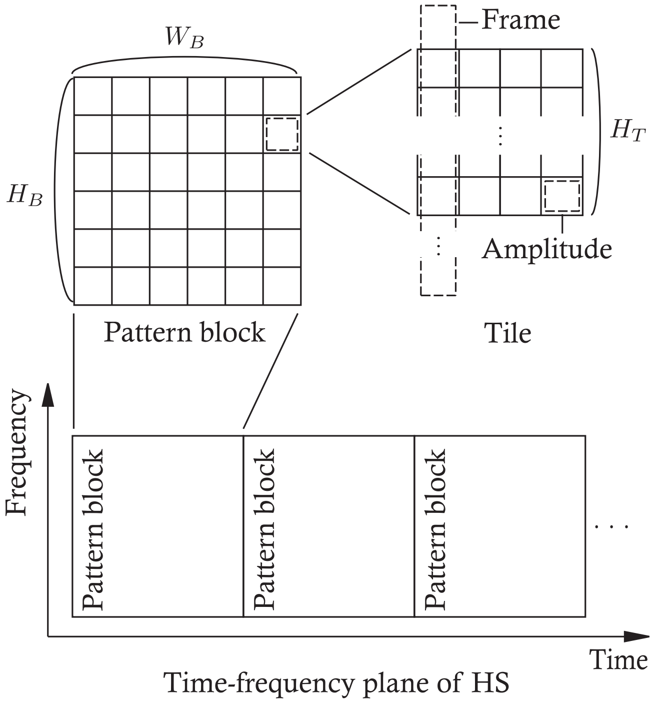


Figure 3.3: Pattern block consists of $W_{\mathrm{B}} \times H_{\mathrm{B}}$ tiles (upper left). Tile is comprised of H_{T} amplitudes of four consecutive frame (upper right). Pattern blocks are arranged on time-frequency plane of host signal (HS) repeatedly (bottom).

the duration of the WS spoils its synchronization with the video frames. However, for example, randomly cropping and inserting samples can yield a similar effect to pitch shifting without changing the WS's duration. It causes the fluctuation of the WS. In some parts of the audio signal, the duration becomes longer and the pitch becomes lower, and in other parts, the duration becomes shorter and the pitch becomes higher.

Some algorithms [60, 58] address the problem related to synchronization including pitch shifting as well as random cropping and insertion of samples. To solve the problem, the algorithms use pseudo-random numbers and introduce redundant representation of them in the time-frequency plane of the HS. They also maintain the inaudibility of the watermark by adopting psychoacoustic models.

Tachibana et al.'s algorithm [58], which is a basis of our watermarking algorithm, is an example of such algorithms. Their watermark embedder divides the HS into audio frames, each of which consists of N_{F} samples. The audio frames overlap each other by $N_{\mathrm{F}}/2$ to alleviate the discontinuity of the resulting WS. Discrete Fourier transform

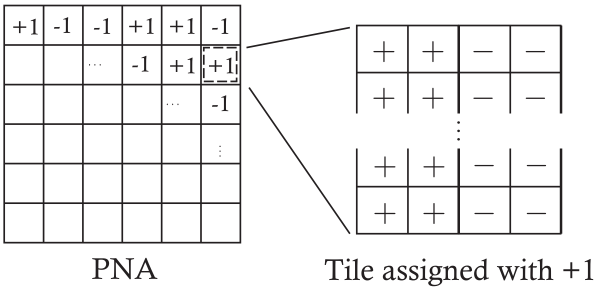


Figure 3.4: Pseudo-random numbers assigned to tiles in pattern block (left). They form pseudo-random number array (PNA). Tile assigned with "+1" (right). In this case, among four consecutive frames of the tile, amplitudes in the first and second frames are increased (represented by "+") and those in the third and fourth frames are decreased (represented by "−").

(DFT) is applied to them to construct the time-frequency plane of the HS. To embed a watermark, the amplitudes of each segmented region in the plane called a *pattern block*, which is shown in Fig. 3.3, are modified. A pattern block has $W_\mathrm{B} \times H_\mathrm{B}$ *tiles*, each of which consists of the H_T amplitudes of four consecutive audio frames. The tile in the w-th column and in the h-th row are denoted by the tile at (w, h). The pattern blocks are arranged repeatedly along the time axis of the HS as in Fig. 3.3. The amplitudes in each tile are modified based on the pseudo-random number in $\{+1, -1\}$ assigned to the tile. The pseudo-random numbers of the tiles in a pattern block form a two-dimensional *pseudo-random number array* (PNA) as shown in Fig. 3.4. The pseudo-random number for the tile at (w, h) is denoted by $\omega(w, h)$.

In this algorithm, each pseudo-random number is redundantly represented by a tile consisting of $4 \times H_\mathrm{T}$ amplitudes. This redundancy reduces the fluctuation influence of the WS caused by random cropping and insertion of samples if the fluctuation in time and the frequency axes is small so that most parts of the tiles in the WS can overlap the original tiles. In addition, since the PNA consists of a small number of pseudo-random numbers, the computational costs to synchronize the PNA with the WS can be reduced.

However, the small number of the pseudo-random numbers in a PNA complicates detection. The watermark detector in this algorithm can be modeled by (3.2) where the summation is calculated over the PNA. In this case, the value given by the second term of the right hand side of (3.2) is small, and the watermark cannot be detected because the value given by the second term is buried in the noise due to the first term. Since α cannot be too large for the inaudibility of the watermark, an alternative approach to alleviate this problem is to reduce the value given by the first term.

For this purpose, Tachibana et al. [58] introduced a *modulus operator*. The modulus operator and the pseudo-random number assigned to a tile determine how the amplitudes in the tile are modified. Modulus operator mop(t) for the t-th frame is defined by

$$\mathrm{mop}(t) = \begin{cases} +1 & \text{if } t \bmod 4 = 0 \text{ or } 1 \\ -1 & \text{otherwise} \end{cases} . \tag{3.3}$$

The signs of the amplitude modifications for the first, second, third, and fourth frames in the tile at (w, h) are determined by $\omega^c(w, h)\mathrm{mop}(0)$, $\omega^c(w, h)\mathrm{mop}(1)$, $\omega^c(w, h)\mathrm{mop}(2)$, and $\omega^c(w, h)\mathrm{mop}(3)$, respectively. This means that $\omega^c(w, h) = +1$ increases the amplitudes in the first and second frames of the tile and decreases those in the third and fourth frames. In the opposite case, $\omega^c(w, h) = -1$ decreases the amplitudes in the first and second frames and decreases those in the third and fourth frames. The modulus operator in [58] modifies two consecutive frames with the same direction to make the watermark more insensitive to the synchronization. Actually, in [58], the watermark is detected by repeatedly calculating (3.2) by shifting the PNA by $N_{\mathrm{F}}/2$ samples.

The modulus operator reduces the influence of the first term of the right hand side of (3.2). Figures 3.5 (a)–(d) describe how the modulus operator alleviates the influence of the first term of the right hand side of (3.2). Figure 3.5 (a) shows the tiles along the frequency axis, which consist of the t-th $(t + 1)$-th, $(t + 2)$-th, and $(t + 3)$-th frames, as well as the pseudo-random numbers assigned to them. The amplitudes of these consecutive four frames are similar (Fig. 3.5 (b)). The watermark is embedded by modifying the amplitudes as mentioned above, which is shown in Fig. 3.5 (c). When detecting the watermark, the difference between the frames, e.g., between the t-th and the $(t + 2)$-th frames, is calculated as in Fig. 3.5 (d). The original amplitudes in these frames are canceled and the pseudo-random numbers for the tiles become significant. The watermark detector calculates (3.2) from the signal derived from the difference. Therefore, the value of the first term on the right-hand side of (3.2) becomes small.

To make the watermarks inaudible, Tachibana et al.'s algorithm [58] uses a psychoacoustic model based on ISO-MPEG1 audio psychoacoustic model 2 for layer 3 [61] to determine the amount of amplitude modifications.

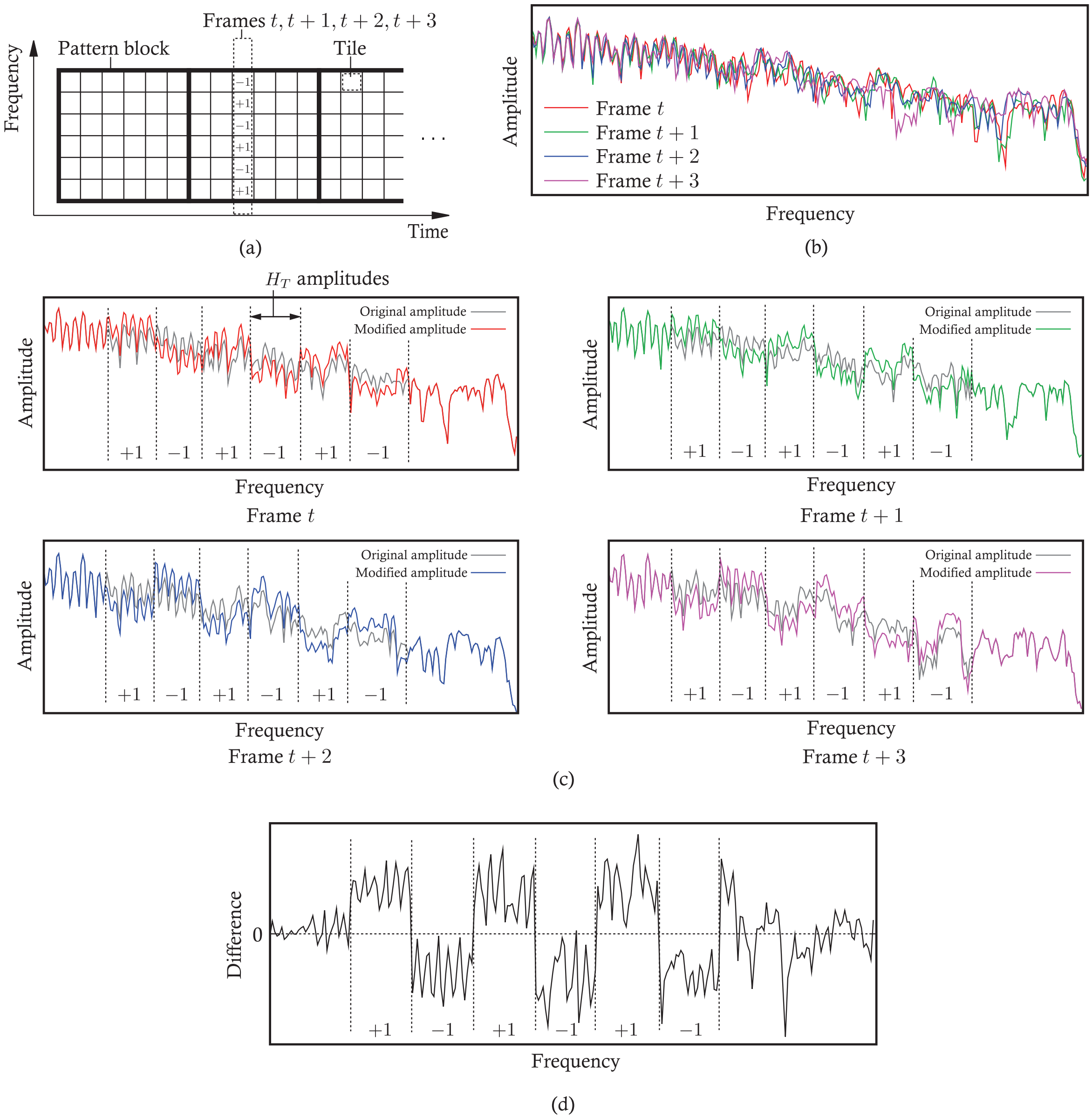


Figure 3.5: (a) Frames $t, t+1, t+2$, and $t+3$ form tiles along frequency axis. (b) Amplitudes of frames. (c) Original amplitudes of frames and amplitudes modified to embed watermark. (d) Difference between frames t and $t+2$.

Our watermarking algorithm is based on Tachibana et al.'s algorithm, which we modify to realize the pirate position estimation. For position estimation, we need multiple watermarks in a RS to calculate the delay of each channel of the movie soundtrack. Therefore, we use a different PNA to embed the watermark into each HS, which corresponds to a channel of the movie soundtrack. The pseudo-random number assigned to the tile at (w, h) for the c-th HS $(c = 1, 2, \cdots, N_{\mathrm{CH}})$ is represented as $\omega^c(w, h)$.

In Tachibana et al.'s algorithm, precise synchronization is unnecessary between the RS and the PNA. However, since position estimation requires accurate delays, which can be obtained from the synchronization position of the PNAs, we calculate the detection strengths at a fine resolution, called *fine detection*. To achieve fine detection, (3.2) is calculated by shifting the PNA by Δ samples. *Detection shift* Δ determines the resolution of the detection strengths. We use sufficiently small Δ for accurate position estimation.

Since we adopt fine detection, excessive redundancy along the time axis is unnecessary. Therefore, although Tachibana et al. [58] uses four consecutive frames to represent a tile, we only use two. Accordingly, the definition of modulus operator $\mathrm{mop}(t)$ is modified to

$$\mathrm{mop}(t) = \begin{cases} +1 & \text{if } t \bmod 2 = 0 \\ -1 & \text{otherwise} \end{cases}. \tag{3.4}$$

3.2.2 Watermark embedder

The watermark embedder generates a WS. The watermark's energy is spread over a pattern block using a PNA. The WS for the c-th HS, $a^c_{\mathrm{WS}}(i)$, is generated in the following steps.

1. The HS in the time domain $a^c_{\mathrm{HS}}(i)$ is divided into audio frames, each of which consists of N_{F} samples, using the sine window. Adjacent frames are overlapped with each other by $N_{\mathrm{F}}/2$ samples to avoid the discontinuities in the WS. The i-th sample of the t-th frame is represented as

   $$\tilde{a}^c_{\mathrm{HS}}(t, i) = a^c_{\mathrm{HS}}(i + tN_{\mathrm{F}}/2)\mathrm{win}(i), \tag{3.5}$$

   where $\mathrm{win}(i)$ is the sine window defined as

   $$\mathrm{win}(i) = \begin{cases} \sin\left(\frac{\pi i}{N_{\mathrm{F}}}\right) & \text{for } 0 \leq i \leq N_{\mathrm{F}} - 1 \\ 0 & \text{otherwise} \end{cases}. \tag{3.6}$$

2. The frames are transformed into the frequency domain using the N_{F}-point DFT.

The f-th Fourier coefficient in the t-th frame $A^c_{\mathrm{HS}}(t, f)$ is obtained as

$$A^c_{\mathrm{HS}}(t, f) = \mathrm{DFT}[\tilde{a}^c_{\mathrm{HS}}(t, i)](f). \tag{3.7}$$

The amplitude and phase of the Fourier coefficient are denoted by $|A^c_{\mathrm{HS}}(t, f)|$ and $\arg A^c_{\mathrm{HS}}(t, f)$.

3. The psychoacoustic model determines the inaudible amount of amplitude modification $M^c(t, f)$.

4. Amplitude modification sign $\mathrm{Sign}^c(t, f)$ for the amplitude in the tile at (w, h) is calculated as

$$\mathrm{Sign}^c(t, f) = \omega^c(w, h)\mathrm{mop}(t), \tag{3.8}$$

where (w, h) is transformed to the corresponding (t, f).

5. The amplitude of the WS, $A^c_{\mathrm{WS}}(t, f)$, is obtained as

$$A^c_{\mathrm{WS}}(t, f) = |A^c_{\mathrm{HS}}(t, f)| + \alpha M^c(t, f)\mathrm{Sign}^c(t, f), \tag{3.9}$$

where α is the watermarking rate to control the tradeoff between the acoustic quality of the WS and the position estimation accuracy.

6. The time-domain representation of the WS in each frame is constructed with the inverse DFT (IDFT) with the original phases of the HS:

$$\tilde{a}^c_{\mathrm{WS}}(t, i) = \mathrm{IDFT}[A^c_{\mathrm{WS}}(t, f) \exp\{\sqrt{-1} \arg A^c_{\mathrm{HS}}(t, f)\}](i). \tag{3.10}$$

7. The final WS in the time domain, $a^c_{\mathrm{WS}}(i)$, is generated by the overlap-and-add technique using the sine window as follows.

$$a^c_{\mathrm{WS}}(i) = \sum_{t=0}^{T-1} \tilde{a}^c_{\mathrm{WS}}(t, i - tN_{\mathrm{F}}/2)\mathrm{win}(i - tN_{\mathrm{F}}/2), \tag{3.11}$$

where T is the number of frames in the HS.

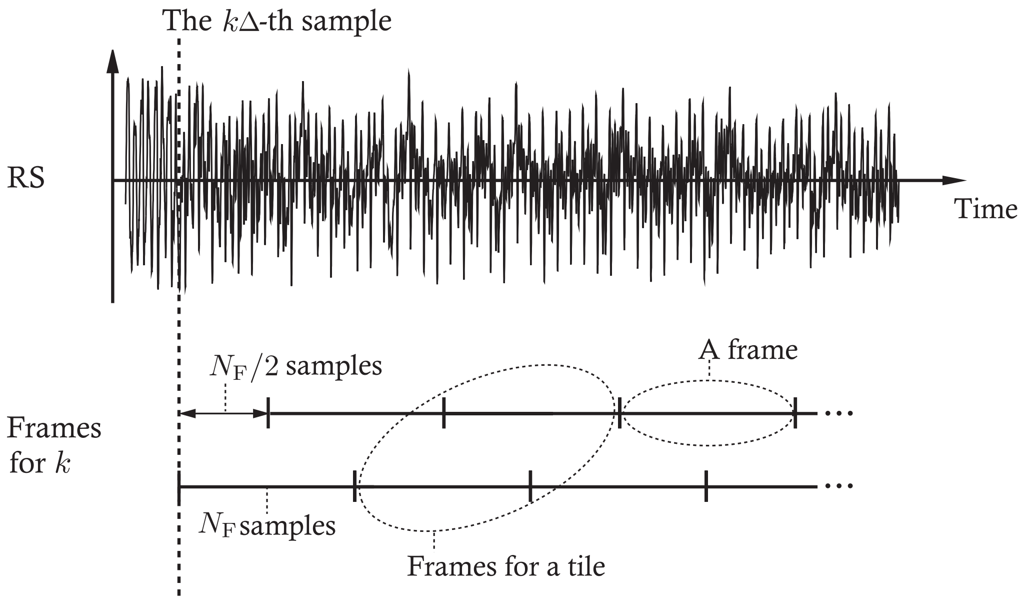


Figure 3.6: Example of recorded signal (RS) and frames in watermark detector.

3.2.3 Watermark detector

The watermark detector detects multiple watermarks with different PNAs in the RS by calculating detection strengths, which can be modeled by (3.2), in a fine resolution. The detection strength of the c-th WS with $k\Delta$-sample delay $s^c(k)$ is calculated from the RS by placing a pattern block starting at the $k\Delta$-th sample as follows:

1. The RS $a_\mathrm{RS}(i)$ is divided into audio frames by the sine window. Each frame is comprised of N_F samples and overlaps with each other by $N_\mathrm{F}/2$ samples. The first frame of the pattern block starts at the $k\Delta$-th sample of the RS (Fig. 3.6):

$$\tilde{a}_{\mathrm{RS},k}(t,i) = a_\mathrm{RS}(i + tN_\mathrm{F}/2 + k\Delta)\mathrm{win}(i). \tag{3.12}$$

2. The frames for k are transformed into the frequency domain by the DFT.

$$A_{\mathrm{RS},k}(t,f) = \mathrm{DFT}[\tilde{a}_{\mathrm{RS},k}(t,i)](f). \tag{3.13}$$

3. The amplitudes are normalized as

$$\overline{A}_{\mathrm{RS},k}(t,f) = \frac{|A_{\mathrm{RS},k}(t,f)|}{\frac{1}{N_{\mathrm{F}}/2}\sum_{f=0}^{N_{\mathrm{F}}/2-1}|A_{\mathrm{RS},k}(t,f)|}. \tag{3.14}$$

4. The difference between the log amplitudes of the two frames in a tile at (w,h), $D_k(w,f)$, is calculated as

$$D_k(w,f) = \log \overline{A}_{\mathrm{RS},k}(2w,f) - \log \overline{A}_{\mathrm{RS},k}(2w+1,f). \tag{3.15}$$

   This alleviates the influence of the HS because its amplitudes in consecutive frames have almost the same values, while the watermark is enhanced by the modulus operator.

5. The amplitude of the tile at (w,h), $\rho_k(w,h)$, is given by

$$\rho_k(w,h) = \sum_f D_k(w,f). \tag{3.16}$$

   The summation is computed for f, which is included in the tile at (w,h).

6. The k-th detection strength of the c-th channel $s^c(k)$ is calculated as

$$s^c(k) = \frac{\sum_{(w,h)} \omega^c(w,h)\left[\rho_k(w,h) - \overline{\rho}_k\right]}{\sqrt{\sum_{(w,h)}\left\{\omega^c(w,h)\left[\rho_k(w,h) - \overline{\rho}_k\right]\right\}^2}}, \tag{3.17}$$

   where

$$\overline{\rho}_k = \frac{1}{W_{\mathrm{B}}H_{\mathrm{B}}}\sum_{(w,h)}\rho_k(w,h). \tag{3.18}$$

   The summations in the above equations are calculated for (w,h) in a pattern block.

From the central limit theorem, $s^c(k)$ follows the Gaussian distribution. If a pattern block of the watermark does not start around the $k\Delta$-th sample of the RS, since the standard deviation of the numerator of (3.17) is given by the denominator, $s^c(k)$ asymptotically follows the standard Gaussian distribution $\mathcal{N}(0,1)$. In contrast, if it

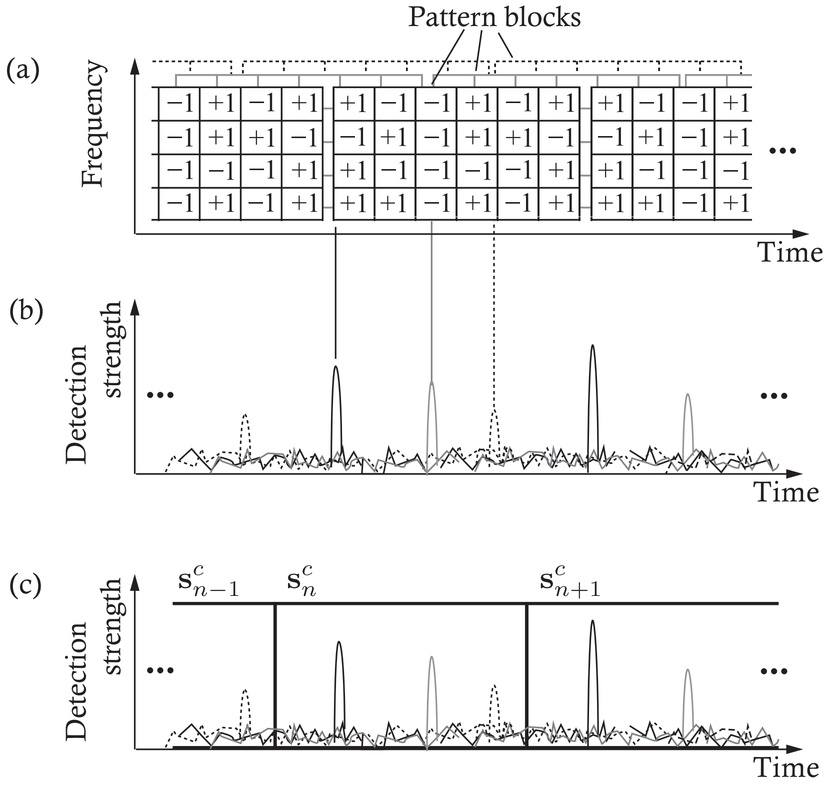


Figure 3.7: (a) Recorded signal (RS) containing multiple watermarks. Pseudo-random numbers assigned to tiles are also shown. (b) Detection strengths calculated from RS. (c) Detection strength blocks.

starts around the $k\Delta$-th sample of the RS, the numerator does not approach zero. Therefore, $s^c(k)$ does not follow the standard Gaussian distribution.

3.3 Position estimator

In this section, we describe the maximum likelihood method-based position estimator, which is based on a *detection strength model*. We also present an algorithm that reduces the computational cost to maximize the likelihood function using a pruning technique based on an upper bound of the likelihood function.

3.3.1 Derivation of detection strength model

As described above, the detection strengths asymptotically follow the Gaussian distribution. Hence, we model them as random values that follow the Gaussian distribution. In this section, we describe how the mean and the variance of the distribution are determined.

Since pattern blocks appear repeatedly on the time-frequency, as shown in Fig. 3.7 (a), the detection strengths form a peak at the beginning of each pattern block (Fig. 3.7 (b)). We divide them into *detection strength blocks* (Fig. 3.7 (c)), each of which consists of N_{DS} detection strengths such that each detection strength block has a single peak. Let $\mathbf{s}_n^c$ denote the n-th detection strength block:

$$\mathbf{s}_n^c = (s^c(nN_{\mathrm{DS}}+0), s^c(nN_{\mathrm{DS}}+1), \cdots, s^c(nN_{\mathrm{DS}}+N_{\mathrm{DS}}-1))^\top, \tag{3.19}$$

where $\top$ represents the transpose. N_{DS} is the number of the detection strengths in a detection strength block, which is given by

$$N_{\mathrm{DS}} = W_{\mathrm{B}} N_{\mathrm{F}} / \Delta, \tag{3.20}$$

where $W_{\mathrm{B}} N_{\mathrm{F}}$ is the number of samples within a pattern block. We also define

$$S^c = \{\mathbf{s}_1^c, \mathbf{s}_2^c, \cdots\} \tag{3.21}$$

and

$$S = \{S^1, S^2, \cdots, S^{N_{\mathrm{CH}}}\}. \tag{3.22}$$

We model $\mathbf{s}_n^c$ by the multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_{\mathrm{DS}}, \Sigma_{\mathrm{DS}})$.

Mean $\boldsymbol{\mu}_{\mathrm{DS}}$ of the distribution depends on the microphone position and the recording conditions. Since we adopt fine detection, detection strength $s^c(k)$ is calculated for every Δ samples, which is smaller than N_{F}. Therefore, not only at the exact time position at which the pattern block starts but also around that time position, strong correlation values appear as shown in Fig. 3.7 (b). We refer to the time position at which the pattern block starts as the peak position. The shape of the peak depends on the watermarking algorithm and the PNA used for embedding. The peak position is determined by the microphone position. Furthermore, the recording conditions (i.e.,

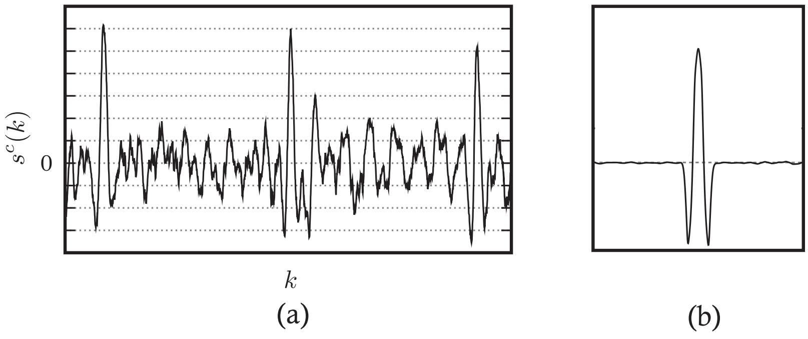


Figure 3.8: Examples of (a) $s^c(k)$ and (b) $\mathbf{m}_{k'}$.

volume, background noises, etc.) and the HS affect the peak height. Considering these factors, we compute the averaged shape of the detection strength peak over various PNAs, and mean $\boldsymbol{\mu}_{\mathrm{DS}}$ given the microphone position and peak height is determined using the averaged shape.

The averaged shape can be obtained as follows. First, we generate a WS for a HS whose samples are zero. The psychoacoustic model is not applied to the HS, and the amount of amplitude modification $M^c(t, f)$ is set to 1. Then the watermark detector is applied to the WS. In the calculation of (3.17), since the watermark embedder arranges pattern blocks repeatedly and thus the detection strengths are periodic, we calculate the detection strengths in the duration of a single pattern block, i.e., $k = 0$ to $N_{\mathrm{DS}} - 1$. Since the first pattern block in the WS starts at the beginning of the WS, the peak is at $k = 0$. We repeat this process using various PNAs and calculate the average of the k-th detection strength over the PNAs. The averaged shape is denoted by

$$\mathbf{m} = (m_0, m_1, \cdots, m_{N_{\mathrm{DS}}-1})^\top. \tag{3.23}$$

The circular shift of $\mathbf{m}$ by k' is given by

$$\mathbf{m}_{k'} = (m_{N_{\mathrm{DS}}-k'}, \cdots, m_{N_{\mathrm{DS}}-1}, m_0, \cdots, m_{N_{\mathrm{DS}}-k'-1})^\top, \tag{3.24}$$

where a non-integer value of k' is rounded to the nearest integer. Figures 3.8 (a) and (b) show an example of actual $s^c(k)$ and $\mathbf{m}_{k'}$. The shape of $\mathbf{m}_{k'}$ resembles the shapes of the peaks in $s^c(k)$.

When the peak position of the n-th detection strength block in the c-th WS is at

k', which means that the pattern block starts at $k = k'$, we obtain the mean of the Gaussian distribution for the detection strength block as

$$\boldsymbol{\mu}_{\text{DS}} = \beta_n^c \mathbf{m}_{k'}, \tag{3.25}$$

where β_n^c is a parameter that determines the height of the peak dependent on the recording condition and the HS.

The value of k' is determined by the microphone and loudspeaker positions. Since we have no information on when the recording was started, we calculate the relative delay of the c-th WS with respect to the WS of the reference channel c_{ref} in the RS. Let $\mathbf{x}_{\text{mic}}$ and $\mathbf{x}_{\text{sp}}^c$ denote the microphone position and the loudspeaker position for the c-th WS. The relative delay of the c-th WS is given by a function of $\mathbf{x}_{\text{mic}}$ as

$$\bar{\kappa}^c(\mathbf{x}_{\text{mic}}) = \frac{SF(\|\mathbf{x}_{\text{sp}}^c - \mathbf{x}_{\text{mic}}\|_2 - \|\mathbf{x}_{\text{sp}}^{c_{\text{ref}}} - \mathbf{x}_{\text{mic}}\|_2)}{SV\Delta}, \tag{3.26}$$

where SV is the sound velocity and SF is the sampling frequency. From this equation, the time position of the peak of the c-th WS is given as

$$\kappa^c(\mathbf{x}_{\text{mic}}, \kappa^{c_{\text{ref}}}) = \kappa^{c_{\text{ref}}} + \bar{\kappa}^c(\mathbf{x}_{\text{mic}}), \tag{3.27}$$

where $\kappa^{c_{\text{ref}}}$ is the time position of the peak of the reference channel. Therefore, for given $\mathbf{x}_{\text{mic}}$ and $\kappa^{c_{\text{ref}}}$, the peak position of the c-th channel is $k' = \kappa^c(\mathbf{x}_{\text{mic}}, \kappa^{c_{\text{ref}}})$, and thus, the mean of the detection strength model for the n-th detection strength block in the c-th channel is given by

$$\boldsymbol{\mu}_{\text{DS}} = \beta_n^c \mathbf{m}_{\kappa^c(\mathbf{x}_{\text{mic}}, \kappa^{c_{\text{ref}}})}. \tag{3.28}$$

To simplify the notation, we omit $(\mathbf{x}_{\text{mic}}, \kappa^{c_{\text{ref}}})$, which is common to any c.

As mentioned in Section 3.2.3, the variance of a detection strength is asymptotically 1 for k not around the peak because the mean of the numerator of (3.17) is 0, and thus the denominator is a sample standard deviation of the numerator. This is not true for k around the peak. However, for simplicity, we assume that the variance of the detection strengths is 1 for all k. We also assume that the detection strengths are independent given $\mathbf{x}_{\text{mic}}$ and $\kappa^{c_{\text{ref}}}$. Hence variance Σ_{DS} is the $N_{\text{DS}} \times N_{\text{DS}}$ identity matrix.

3.3.2 Derivation of position estimator

In this section, we derive the maximum-likelihood estimator of the microphone position. First, we calculate the probability density of $\mathbf{s}_n^c$. As mentioned in Section 3.3.1, $\mathbf{s}_n^c$ is modeled by the multivariate Gaussian distribution. Given microphone position $\mathbf{x}_{\mathrm{mic}}$ and time position of reference channel $\kappa^{c_{\mathrm{ref}}}$, the mean of the distribution is $\beta_n^c \mathbf{m}_{\kappa^c}$ and the variance is the identity matrix. Therefore, the conditional probability density of $\mathbf{s}_n^c$ is given by

$$p(\mathbf{s}_n^c | \mathbf{x}_{\mathrm{mic}}, \kappa^{c_{\mathrm{ref}}}, \beta_n^c) = \frac{1}{(2\pi)^{N_{\mathrm{DS}}/2}} \exp\left[-\frac{(\mathbf{s}_n^c - \beta_n^c \mathbf{m}_{\kappa^c})^\top (\mathbf{s}_n^c - \beta_n^c \mathbf{m}_{\kappa^c})}{2} \right]. \tag{3.29}$$

The conditional probability density of S is given by

$$\begin{aligned} p(S|\mathbf{\Theta}) &= \prod_c p(S^c|\mathbf{\Theta}) && (3.30)\\ &= \prod_c \prod_n p(\mathbf{s}_n^c|\mathbf{\Theta}), && (3.31) \end{aligned}$$

where $\mathbf{\Theta} = \{\mathbf{x}_{\mathrm{mic}}, \kappa^{c_{\mathrm{ref}}}, B\}$ and $B = \{\beta_n^c | c = 1, 2, \cdots, N_{\mathrm{CH}}; n = 0, 1, \cdots\}$. Thus, we define log-likelihood function $L(\mathbf{\Theta})$ as

$$L(\mathbf{\Theta}) = -\sum_c \sum_n \frac{(\mathbf{s}_n^c - \beta_n^c \mathbf{m}_{\kappa^c})^\top (\mathbf{s}_n^c - \beta_n^c \mathbf{m}_{\kappa^c})}{2}. \tag{3.32}$$

This log-likelihood function consists of the squared Euclidean distance between $\mathbf{s}_n^c$ and $\beta_n^c \mathbf{m}_{\kappa^c}$. Since the distance is accumulated over all n and thus the effect of noises in $\mathbf{s}_n^c$ is alleviated, the estimation accuracy can be improved. Eliminating β_n^c by setting $\partial L(\mathbf{\Theta})/\partial \beta_n^c = 0$ and ignoring the irrelevant terms, we obtain the following maximization criterion equivalent to (3.32):

$$L'(\mathbf{\Theta}') = \sum_c \sum_n (\mathbf{m}_{\kappa^c}^\top \mathbf{s}_n^c)^2, \tag{3.33}$$

where $\mathbf{\Theta}' = \{\mathbf{x}_{\mathrm{mic}}, \kappa^{c_{\mathrm{ref}}}\}$.

The microphone position is estimated by finding the parameters that maximize this criterion. The maximum-likelihood estimator of $\mathbf{\Theta}'$ is

$$\hat{\mathbf{\Theta}}' = \arg\max_{\mathbf{\Theta}'} L'(\mathbf{\Theta}'), \tag{3.34}$$

and its element, $\hat{\mathbf{x}}_{\mathrm{mic}}$, is the maximum-likelihood estimator of $\mathbf{x}_{\mathrm{mic}}$. The simplest solution for this maximization problem is an exhaustive search in the set of possible values of $\mathbf{\Theta}'$.

3.3.3 Maximization algorithm to reduce computational cost

Finding the maximum of $L'(\mathbf{\Theta}')$ by an exhaustive search is computationally too expensive because the parameter space is three-dimensional when $\mathbf{x}_{\mathrm{mic}}$ is two-dimensional, and each possible $\mathbf{\Theta}'$ requires evaluation of (3.33). In this section, we present an algorithm that can drastically reduce the computational cost using an upper bound of $L'(\mathbf{\Theta}')$. We calculate the upper bound for each value of $\kappa^{c_{\mathrm{ref}}}$. If the upper bound is lower than the maximum that was obtained by that time in the search, the algorithm omits further search with the value of $\mathbf{x}_{\mathrm{mic}}$.

Let $l^c(\kappa^c)$ denote the inner summation of the right-hand side of (3.33):

$$l^c(\kappa^c) = \sum_n (\mathbf{m}_{\kappa^c}^\top \mathbf{s}_n^c)^2. \tag{3.35}$$

By separating $l^{c_{\mathrm{ref}}}(\kappa^{c_{\mathrm{ref}}})$, which is irrelevant to $\mathbf{x}_{\mathrm{mic}}$, the maximization criterion (3.33) can be rewritten as

$$L'(\mathbf{\Theta}') = \sum_c l^c(\kappa^c) = l^{c_{\mathrm{ref}}}(\kappa^{c_{\mathrm{ref}}}) + \sum_{c \neq c_{\mathrm{ref}}} l^c(\kappa^c). \tag{3.36}$$

In the exhaustive search, we choose a value of $\kappa^{c_{\mathrm{ref}}}$, and then find the value of $\mathbf{x}_{\mathrm{mic}}$ that maximizes $L'(\mathbf{\Theta}')$ given the value of $\kappa^{c_{\mathrm{ref}}}$. Since the first term of the rightmost side of (3.36) is irrelevant to $\mathbf{x}_{\mathrm{mic}}$, we can write this maximization as

$$\max_{\mathbf{x}_{\mathrm{mic}}} \left[l^{c_{\mathrm{ref}}}(\kappa^{c_{\mathrm{ref}}}) + \sum_{c \neq c_{\mathrm{ref}}} l^c(\kappa^c) \right] = l^{c_{\mathrm{ref}}}(\kappa^{c_{\mathrm{ref}}}) + \max_{\mathbf{x}_{\mathrm{mic}}} \sum_{c \neq c_{\mathrm{ref}}} l^c(\kappa^c). \tag{3.37}$$

Since the maximum of the last term is less than or equal to the sum of the maxima of

```
Maximization algorithm
$\hat{\kappa}^c \leftarrow \arg\max_{\kappa^c} l^c(\kappa^c)$
$c_{\text{ref}} \leftarrow \arg\max_c l^c(\hat{\kappa}^c)$
current_maximum $\leftarrow 0$
for all $\kappa^{c_{\text{ref}}}$ starting from $\hat{\kappa}^{c_{\text{ref}}}$ do
  if current_maximum $< l^{c_{\text{ref}}}(\kappa^{c_{\text{ref}}}) + U$ then
    $\check{\mathbf{x}}_{\text{mic}} \leftarrow$ search possible $\mathbf{x}_{\text{mic}}$ exhaustively
    $\check{\mathbf{\Theta}}' \leftarrow \{\kappa^{c_{\text{ref}}}, \check{\mathbf{x}}_{\text{mic}}\}$
    if $L'(\check{\mathbf{\Theta}}') >$ current_maximum then
      current_maximum $\leftarrow L'(\check{\mathbf{\Theta}}')$
      $\mathbf{\Theta}'_{\text{cand}} \leftarrow \check{\mathbf{\Theta}}'$
    end if
  end if
end for
return $\mathbf{\Theta}'_{\text{cand}}$
```

Figure 3.9: Maximization algorithm.

$l^c(\kappa^c)$'s, we obtain the following inequality:

$$\max_{\mathbf{x}_{\text{mic}}} \sum_{c \neq c_{\text{ref}}} l^c(\kappa^c) \leq \sum_{c \neq c_{\text{ref}}} \max_{\kappa^c} l^c(\kappa^c) = U. \tag{3.38}$$

The maximization of $l^c(\kappa^c)$ is inexpensive because it involves only one parameter: κ^c. In other words, although κ^c is determined by $\mathbf{x}_{\text{mic}}$ and $\kappa^{c_{\text{ref}}}$, the maximization of $l^c(\kappa^c)$ simply finds the value of κ^c, regardless of $\mathbf{x}_{\text{mic}}$ and $\kappa^{c_{\text{ref}}}$. This maximization is done only once because it is irrelevant to the value of $\kappa^{c_{\text{ref}}}$. Thus, we obtain an upper bound of $L'(\mathbf{\Theta}')$ given $\kappa^{c_{\text{ref}}}$ as

$$L'(\mathbf{\Theta}') \leq l^{c_{\text{ref}}}(\kappa^{c_{\text{ref}}}) + U. \tag{3.39}$$

Now we can prune the search of $\mathbf{x}_{\text{mic}}$ for $\kappa^{c_{\text{ref}}}$ if $l^{c_{\text{ref}}}(\kappa^{c_{\text{ref}}}) + U$ is less than the maximum value that we computed for a different value of $\kappa^{c_{\text{ref}}}$. Figure 3.9 shows the maximization algorithm using the upper bound. This algorithm drastically reduces the number of possible $\kappa^{c_{\text{ref}}}$'s while the exhaustive search must find $\mathbf{x}_{\text{mic}}$ that maximizes $L'(\mathbf{\Theta}')$ for all $\kappa^{c_{\text{ref}}}$'s. Furthermore, the earlier we obtain large values, the more effective

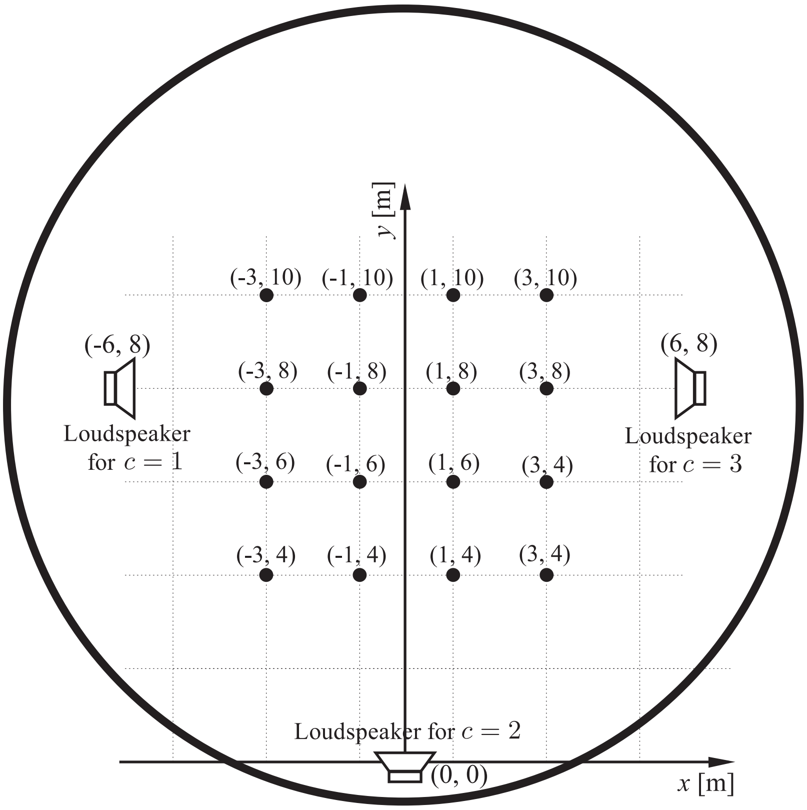


Figure 3.10: Experimental environment for estimation accuracy evaluation.

the pruning of the algorithm becomes. Therefore, we choose the reference channel as

$$c_{\text{ref}} = \arg\max_{c} l^c(\hat{\kappa}^c), \tag{3.40}$$

where

$$\hat{\kappa}^c = \arg\max_{\kappa^c} l^c(\kappa^c), \tag{3.41}$$

and the search is begun from $\hat{\kappa}^{c_{\text{ref}}}$, where $L'(\boldsymbol{\Theta}')$ is expected to be large.

3.4 Experimental results

In this section, we evaluate the estimation accuracy of our system in a circular auditorium with 250 seats. The effect of watermarking rate α, which controls the audibility of the watermarks, on estimation accuracy is investigated by simulation experiments.

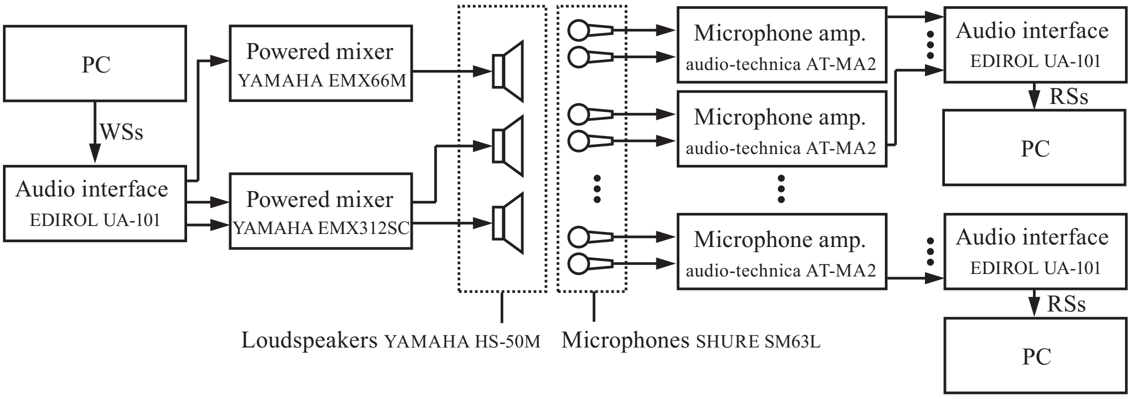


Figure 3.11: Experimental setup.

We also subjectively assess the acoustic quality of WSs by MUSHRA listening tests [62].

3.4.1 Estimation accuracy evaluation

To evaluate the estimation accuracy of our system in a semi-realistic environment, we conducted experiments at the Hankyu Sanwa Conference Hall in the Alumnus Union Building of the Osaka University Medical School[1]. This 8.8-m radius, circular auditorium has 250 seats. Three loudspeakers and 16 microphones (represented by dots) were arranged in the same plane (Fig. 3.10). The experimental setup is shown in Fig. 3.11. We simultaneously recorded audio signals with all 16 microphones. The volume of the two powered mixers was manually adjusted to be the same.

In an actual theater, the audience may affect accuracy for the following possible reasons: (a) It can block the direct paths from the loudspeakers to the microphone, resulting in false peaks. (b) It can make noise, resulting in decreased peak heights. Since the loudspeakers in theaters are usually attached to the upper side of walls, the influence of (a) is considered insignificant. For (b), we conducted an experiment with a modified version of the position estimation system in [48] and experimentally demonstrated that noise hardly affected accuracy. Hence, in the following experiments, we ignored audience influence.

The test samples used in these experiments are listed in Table 3.1. They are excerpts

[1] http://www.med.osaka-u.ac.jp/pub/general/alumni/intro.html

Table 3.1: Test samples used in experiments.

Label	Title	Start at	RMS [dB] for channel		
		[sec]	$c = 1$	$c = 2$	$c = 3$
DS1	Saw	1,034	-31.1	-29.9	-31.0
DS2	Pretty Woman	3,998	-42.8	-29.6	-42.5
DS3	The Bourne Identity	3,050	-28.7	-25.6	-28.8
DS4	Harry Potter and the Goblet of Fire	2,246	-29.3	-25.5	-24.8
DS5	RENT	2,290	-26.5	-21.0	-27.1

Table 3.2: Experimental parameters.

Number of tiles in a column of a pattern block	W_{B}	20
Number of tiles in a row of a pattern block	H_{B}	24
Number of amplitudes in a tile along the frequency axis	H_{T}	6
Number of channels	N_{CH}	3
Frame length [samples]	N_{F}	512
Detection shift [samples]	Δ	16
Sampling frequecny [Hz]	SF	44100
Sound velocity [m/s]	SV	340

from the right ($c = 1$), the center ($c = 2$), and the left ($c = 3$) channels of the original movie soundtracks. The starting positions were randomly chosen. The duration of each test sample was 1,800 seconds (30 minutes). The root mean square (RMS) values of each test sample are also listed in Table 3.1. The parameters used in the experiments are listed in Table 3.2. Watermarking rate α was set to 1.0.

Figure 3.12 shows the estimation error for each microphone position $\mathbf{x}_{\mathrm{mic}}$, which is given as the Euclidean distance between the microphone and estimated positions, i.e., $\|\mathbf{x}_{\mathrm{mic}} - \hat{\mathbf{x}}_{\mathrm{mic}}\|_2$, where $\hat{\mathbf{x}}_{\mathrm{mic}}$ is the estimated position. Almost all microphone positions are accurately estimated except for positions $(3, 4)$, $(1, 4)$, and $(-1, 4)$ for DS2. The estimation errors for these microphone positions are large. One reason is that there are not enough watermarks of the first and third WSs in the RS to form peaks in the detection strengths since the energies of the first and third HSs of the DS2 are low (Table 3.1). The directional characteristics of the loudspeakers and the distances from them to these positions enhance this energy imbalance. Furthermore, the effect

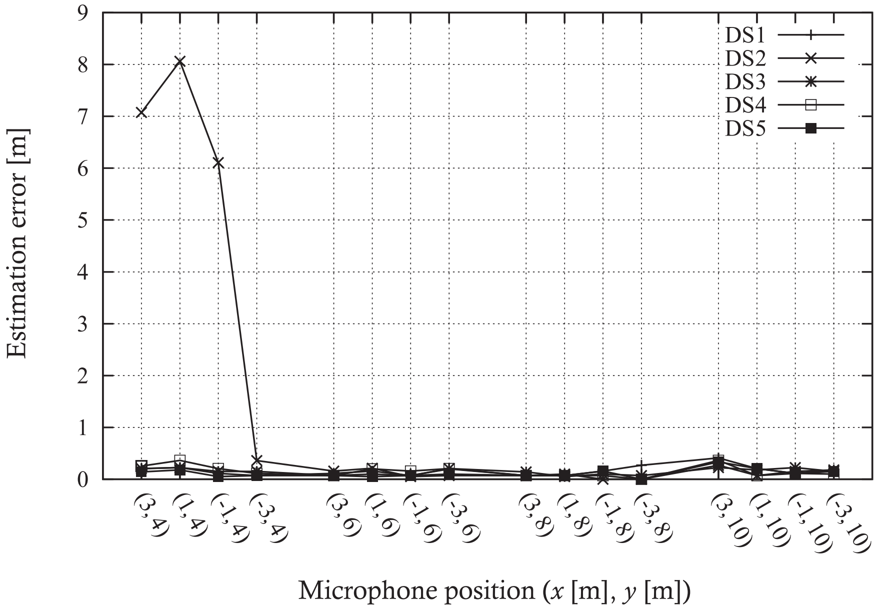


Figure 3.12: Estimation errors in auditorium.

of cross-correlation among the three PNAs may enlarge the error. For example, if the PNAs for the first and second WSs have correlation, the strong watermark in the second WS forms a false peak in the detection strengths for the first WS even when the correlation among the PNAs is weak. If the false peak is larger than the actual peak, the estimation fails. Therefore, in practical use, we need to adaptively control watermarking rate α. The mean and standard deviation of the estimation errors for all microphone positions are 0.40 m and 1.33 m, respectively. Although the standard deviation is large due to the large errors of DS2, it almost identifies a seat.

3.4.2 Watermarking rate versus estimation accuracy

In the previous section, we showed that our system accurately estimated the microphone positions for $\alpha = 1.0$. However, the acoustic quality is degraded as α becomes large. To maintain acoustic quality, the watermarking rate should be small. However, this may cause larger estimation errors. We investigated the relationship between α and the estimation errors by simulation experiments.

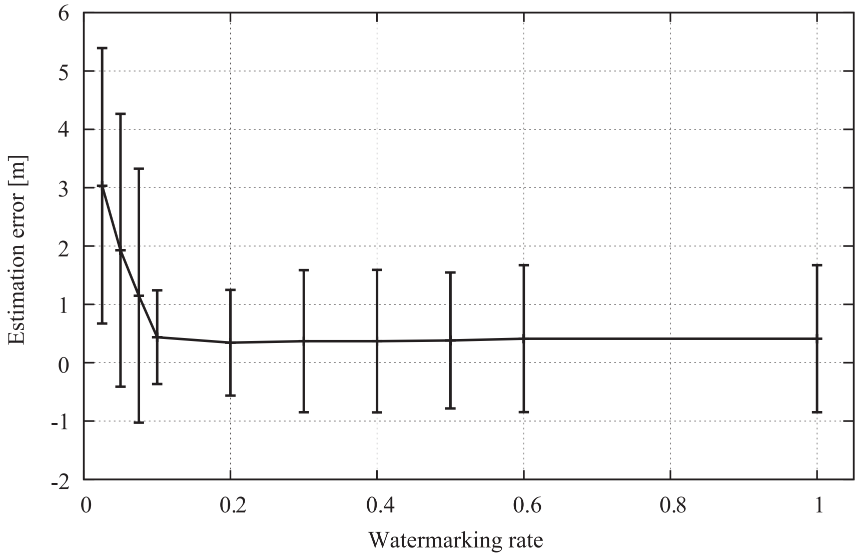


Figure 3.13: Relationship between watermarking rate and estimation error. Means and standard deviations of estimation errors are calculated for various α.

First, we model a RS, which is received by the microphone at $\mathbf{x}_{\text{mic}}$, as

$$\tilde{a}_{\text{RS}}(i) = \sum_{c} a^{c}_{\text{WS}}(i) * h^{c}_{\mathbf{x}_{\text{mic}}}(i) + \eta(i), \tag{3.42}$$

where $h^{c}_{\mathbf{x}_{\text{mic}}}(i)$ is the impulse response of the path from the loudspeaker for the c-th WS to the microphone at $\mathbf{x}_{\text{mic}}$, $\eta(i)$ represents the background and thermal noises, and "$*$" is the convolution operator. We measured impulse response $h^{c}_{\mathbf{x}_{\text{mic}}}(i)$ by the time stretched pulse method [63] under the same experimental setup discussed in Section 3.4.1. Noise $\eta(i)$ is assumed to follow the Gaussian distribution $\mathcal{N}(0, \sigma^2)$ and its variance σ^2 is determined from the RS without any audio signals from the loudspeakers.

Applying (3.42) to the WSs with various α, we generated simulated versions of the RSs. The other parameter values were the same as in Section 3.4.1. The mean and standard deviation of the estimation errors were calculated for each α.

The result is shown in Fig. 3.13. The mean and standard deviation of the estimation error for $\alpha = 1.0$ are 0.41 m and 1.26 m. These values are close to the result in Section 3.4.1; the mean and standard deviation are 0.40 m and 1.33 m, and thus, the result of this simulation experiment is reliable. The mean of the estimation errors is large for $\alpha < 0.1$. Meanwhile, the microphone positions are estimated with small errors

Table 3.3: Samples used in subjective assessment of acoustic quality.

Label	Excerpt from	Starts at	Ends at
SUB1	DS2	454 [s]	473 [s]
SUB2	DS3	111 [s]	129 [s]
SUB3	DS4	1,229 [s]	1,248 [s]
SUB4	DS5	326 [s]	349 [s]
SUB5	DS2	1,229 [s]	1,046 [s]

for $\alpha \geq 0.1$, although the standard deviations are relatively large due to the large estimation errors of DS2, as mentioned in Section 3.4.1. The mean of the estimation error for $\alpha = 0.1$ was 0.44 m, indicating that, in this experimental environment, the peak of the detection strengths is buried in noise for $\alpha < 0.1$. In other words, we can reduce the value of α as small as 0.1 without significant estimation errors. Note that an appropriate value of α may depend on the frequency response of the acoustical system of the auditorium including background noises.

To show the effectiveness of the algorithm for reducing the computational cost, in this experiment, we measured the time to estimate the positions using a PC with an Intel Core 2 Duo processor running at 1.6 GHz with 1 Gbyte of memory. The average time over all trials of the position estimation is 596 seconds. For comparison, we also measured the time to estimate the positions with an exhaustive search. However, since this was time consuming, the estimation was executed only twice. The average time over these two estimations is 179,573 seconds. Hence, the proposed algorithm achieves the 99.7% execution time reduction compared to the exhaustive search.

3.4.3 Subjective evaluation of acoustic quality

We subjectively assessed the acoustic quality of the WSs by MUSHRA listening tests [62]. This method assesses the acoustic quality of audio signals that undergo audio signal processing techniques. In this assessment, subjects listened to multiple audio signals, including not only the WSs but also the original audio signal called hidden references and others for comparison. We used the samples listed in Table 3.3, which are excerpts from the test samples used in Section 3.4.1. These samples were processed as in Table 3.4. For each test sample, 17 inexperienced subjects graded all test signals

Table 3.4: Description of test signals used in subjective assessment of acoustic quality.

Label	Description
REF	Reference signal
HREF	Hidden reference
ALPF	Low pass filtered signal as an anchor
AM48	Compressed signal using MP3 48 kbps as an anchor
AM32	Compressed signal using MP3 32 kbps as an anchor
WR01	Watermarked signal with $\alpha = 0.1$
WR03	Watermarked signal with $\alpha = 0.3$
WR05	Watermarked signal with $\alpha = 0.5$

Table 3.5: Summary of conditions under which acoustic quality was assessed.

	Listening method	
	Loudspeaker	Headphones
Office	(a)	(b)
Auditorium	—	(c)

after training sessions where they were exposed to all of the audio signals used in this assessment.

Since MUSHRA listening tests take a long time, we could not conduct the assessment in the auditorium with loudspeakers. Instead, the subjects assessed the test signals under the following conditions.

(a) Assessment in a small office with three loudspeakers. The subjects sat at the listening position corresponding to $(3, 3)$ in a 6×6 m^2 office (Fig. 3.14), and assessed the test signals from three loudspeakers.

(b) Assessment of simulated listening in the office using headphones. The test signals were convolved by the impulse responses measured by a dummy head at the listening position in the same office as used for (a), and the subjects listened to the simulated signals with headphones.

(c) Assessment of simulated listening in the auditorium using headphones. This condition is almost the same as (b), but the impulse responses were measured at $(0, 6)$ in the auditorium in Fig. 3.10.

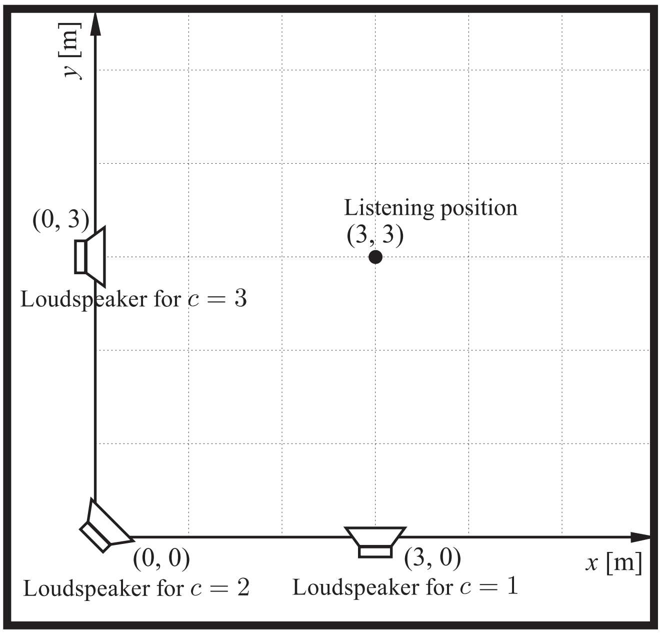


Figure 3.14: Listening position in room for (a).

These conditions are summarized in Table 3.5. Since the test signals for (b) were generated using the impulse responses measured in the same office as used for (a), the results of (a) and (b) should be similar. If this is satisfied, the results of (c) are considered to be similar to those of the subjective assessment when the subjects actually listened to the test signals from the loudspeakers in the auditorium.

The means and 95% confidence intervals for the acoustic quality of the test signals under (a) and (b) are shown in Figs. 3.15 and 3.16. The degradation of the acoustic quality for WR01 and WR03 is almost imperceptible, and it is perceptible for WR05, although it remains acceptably small. The subjective acoustic quality under (a) and (b) is almost the same. Therefore, the results under (c) should be similar to the results when the subjects actually assessed the acoustic quality in the auditorium.

Figure 3.17 shows the means and 95% confidence intervals for the acoustic quality of the test signals under (c). Although the watermarks are relatively audible compared to (a) or (b), the subjective acoustic quality of WR01 remains good enough for practical use.

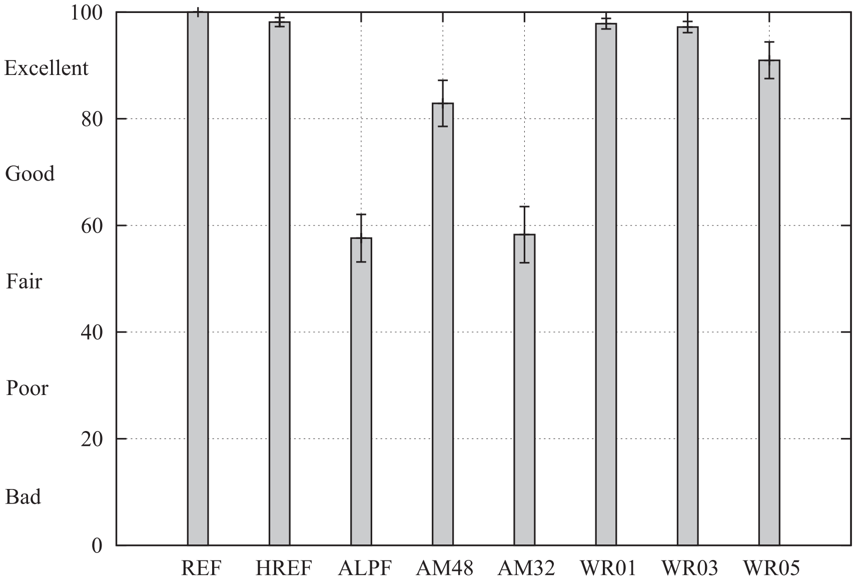


Figure 3.15: Means and 95% confidence intervals for acoustic quality of test signals under (a).

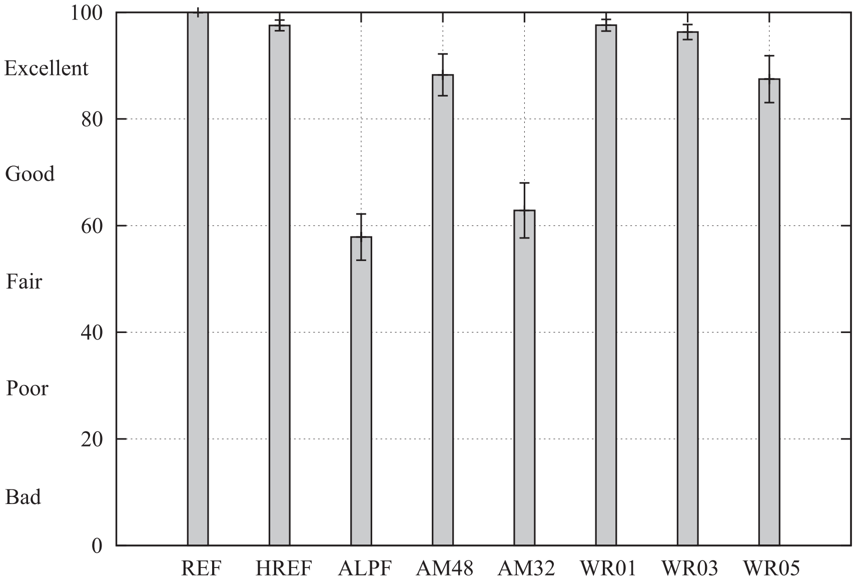


Figure 3.16: Means and 95% confidence intervals for acoustic quality of test signals under (b).

3.4.4 Discussion

From the results of Sections 3.4.2 and 3.4.3 with $\alpha = 0.1$, our system can estimate microphone positions with mean estimation error of 0.44 m, and the subjective acoustic

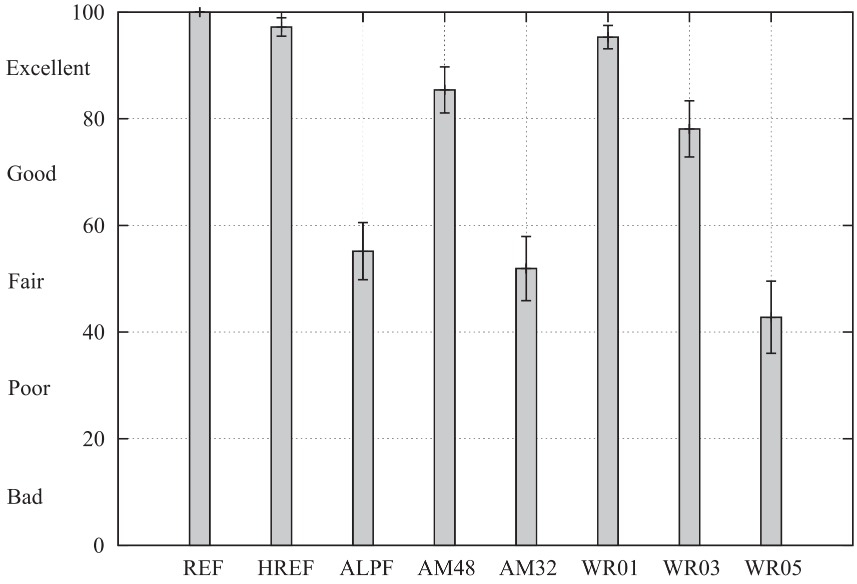


Figure 3.17: Means and 95% confidence intervals for acoustic quality of test signals under (c).

quality is in the *excellent* range. By increasing α to 0.3, the mean estimation error can be reduced to 0.34 m at the expense of degrading the acoustic quality to the *good* range. Therefore, we successfully showed that our system is able to estimate microphone positions without significantly spoiling the acoustic quality of movie soundtracks.

However, the difference between the results of (b) and (c) indicates that acoustic quality largely depends on the environments in which the system is used. Estimation accuracy probably depends on the frequency response of the auditorium, the background noise, and so forth. Hence, we need a preliminary experiment in the actual environment before practical use to determine the appropriate α.

3.5 Concluding remarks

In this chapter, we presented a position estimation system to prevent in-theater movie piracy by a new application of the digital audio watermarking technique. The core idea of our system utilizes delays of the watermarks embedded into multiple channel movie soundtracks. The presented watermarking algorithm is designed to accurately obtain the delays. We also described a maximum likelihood-based position estimator using a probabilistic model of the detection strengths that exploits the long duration of movie soundtracks to improve estimation accuracy.

Our experimental results show that our system can estimate the microphone position with mean estimation errors of 0.44 m without significantly spoiling the acoustic quality assessed by MUSHRA listening tests. However, the acoustic quality depends on the environment in which the system is used. To clarify the effect of such environmental factors as the frequency responses of auditoriums and background noise on acoustic quality and estimation accuracy, we need more experiments in various environments. Furthermore, we must investigate the robustness of our system against such attacks as lossy compression.

Chapter 4

Intentionally-Captured Human Object Detection

4.1 Introduction

In this and the next chapters, we present a system that automatically generates privacy-protected videos. We focus on accidental privacy infringement where the privacy of persons who are accidentally framed in is infringed on by capturing them in a video. In this dissertation, we refer to the regions in video frames corresponding to persons as human objects.

Generally, when camera persons take videos, they have capture intentions [64], which divide the human objects into two groups: intentionally-captured human objects (ICHOs), which correspond to those who are intentionally captured by the camera persons, and human objects except the ICHOs (non-ICHOs). The non-ICHOs correspond to those who are accidentally framed in. ICHOs are essential for the camera persons' capture intentions, and thus videos may become meaningless without them.

The problem of accidental privacy infringement is the disclosure of non-ICHOs. Therefore, for accidental privacy infringement, we assume that a camera person can obtain permission to capture the intentionally-captured persons and to publish the video. This assumption is reasonable because, in most cases, there are only a few intentionally-captured persons and the camera person can at least negotiate with them for permission. Furthermore, the intentionally-captured persons are often friends or family members. In contrast, it is usually infeasible to get permission from accidentally framed-in persons because they are often merely passing by while the camera person captures the video. Therefore, considering that non-ICHOs are inessential for the camera person's capture intention, non-ICHOs should always be obscured. The goal of privacy protection against accidental privacy infringement is to realize a system that obscures only non-ICHOs.

Some privacy protection systems for videos obscure all human objects [9, 25, 26, 27]. Other systems selectively determine the human objects to be obscured based on predetermined rules and the identities of the persons corresponding to the human objects [8, 38, 39, 41, 65]. However, for accidental privacy infringement, such approaches are inappropriate because ICHOs are essential for videos taken by a camera person who can obtain permission for capturing and publishing from intentionally-captured persons. In addition, a human object in a video can be both an ICHO and a non-ICHO dependent on the transition of the camera person's capture intention. Therefore, we need a technique to find ICHOs in videos.

In this chapter, we present a method for *ICHO detection* that automatically detects the ICHOs in videos. ICHO detection is one technique for important region determination, and visual attention models have been extensively studied for this purpose. One of the most well-known visual attention models for still images was proposed by Itti et al. [66]. For an image, it generates a saliency map that represents the extent to which each pixel in the image attracts viewer attention. Itti et al.'s model, inspired by the behavior and the neuronal architecture of the visual system of primates, generates saliency maps by integrating multiple feature maps, each of which represents the image's saliency based on intensity, color, or orientation features. Ma and Zhang's model for images is based on the observation that regions with large changes in color or luminance are most likely to attract viewer attention [67]. Itti and Baldi proposed a bottom-up visual attention model for videos [68, 69] based on the definition of surprise using the Bayesian theorem. Hu et al. proposed another visual attention model based on the idea that moving objects may attract viewer attention [70].

Compared with these techniques, the following is the novelty of ICHO detection. Since the visual attention models simulate the responses of animals' visual systems against visual stimuli, they are considered important regions for the viewers of images/videos. Conversely, ICHO detection can be regarded as important regions for

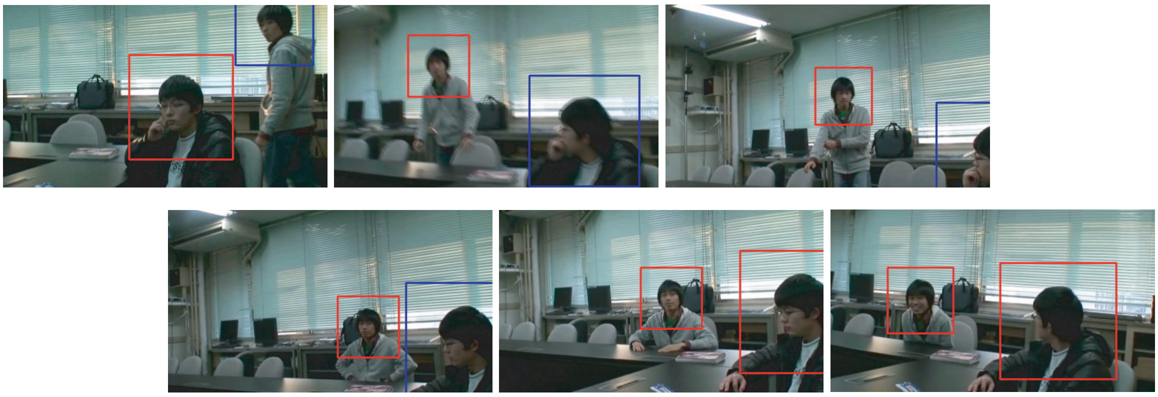


Figure 4.1: Example of intentionally-captured human object (ICHO) surrounded by red rectangles and human object except ICHO (non-ICHO) surrounded by blue rectangles.

camera persons.

In the following sections, we describe ICHO detection. To detect ICHOs, we first detect all human objects in a video frame and classify them into ICHOs/non-ICHOs using features related to the camera person's capture intention. In the next section, we describe human object detection as well as features and an algorithm for ICHO classification. Section 4.3 presents the experimental results including the evaluation of the contributions of the features used for ICHO classification. We give concluding remarks in Section 4.4. This chapter is related to the work published in [71, 72, 73, 74, 75, 76, 77].

4.2 Method for intentionally-captured human object detection

In Fig. 4.1, the camera person intentionally captures the person near the camera (first frame). As his/her capture intention changes, he/she gradually moves the camera to capture the other person so that the ICHO corresponding to the person is arranged around the center of the video frames (second to fourth frames). Finally, he/she intentionally captures both persons (last two frames). From this example, a camera person's capture intention may provoke specific behaviors of the camera person, e.g., following an intentionally-captured person or arranging the ICHO corresponding to the person

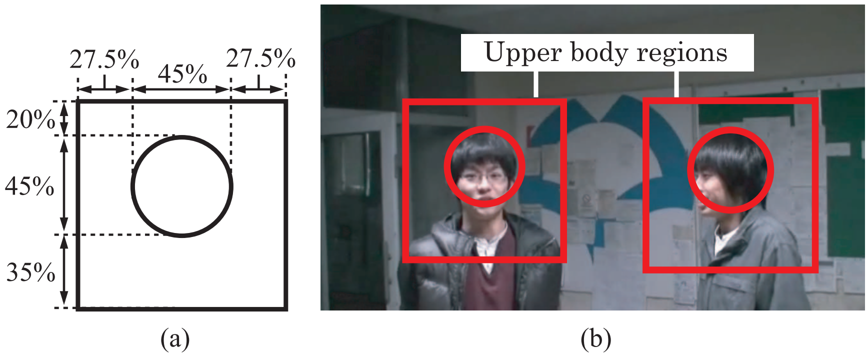


Figure 4.2: (a) Upper body region model and (b) examples of upper body regions. Upper body region is defined as region surrounded by rectangle of upper body region model when it is placed such that circle of the upper body region model surrounds the head of human object.

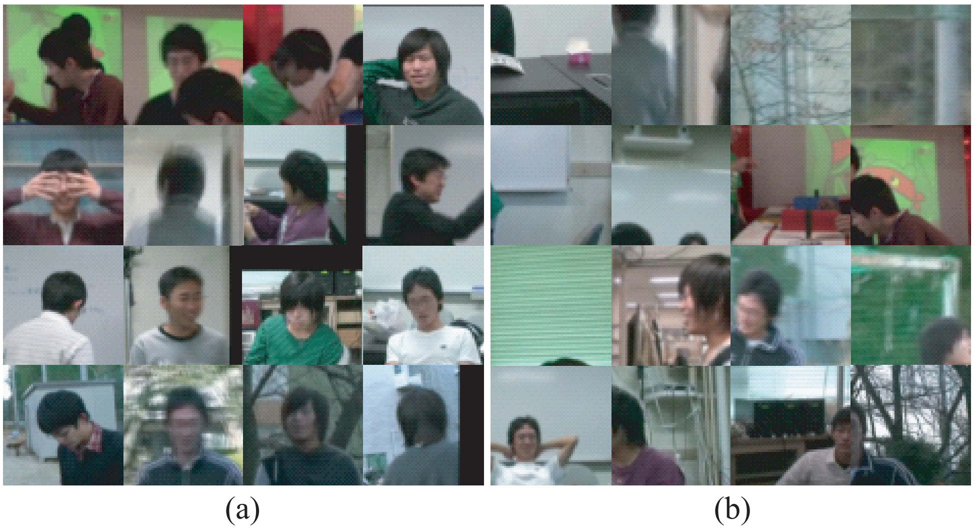


Figure 4.3: (a) Positive and (b) negative samples used to train support vector machine (SVM) for human object detection.

around the center of the frame. Such behaviors are reflected in the camera motion against the motion of each human object. In addition, the camera person captures persons intentionally for a while so that the viewers can comprehend what they are seeing. Therefore, the ICHOs are temporally consistent; i.e., they do not change very frequently.

Based on these observations, ICHOs are detected as follows. We first detect human objects assuming that ICHOs are captured from their upper bodies. For each detected human object, the features related to the camera person's capture intention are extracted. The ICHOs are statistically modeled using these features. Finally, each human object is classified into ICHO/non-ICHO.

4.2.1 Human object detection

The method for ICHO detection assumes that, when a camera person intentionally captures persons, the videos include the upper bodies, defined in Fig. 4.2. Thus, we detect the upper body regions of the human objects using histograms of oriented gradients descriptors (HOG) and a support vector machine (SVM) with a linear kernel [36]. The positive samples used to train the SVM are manually specified and extracted from a video dataset. The negative samples are randomly extracted from the regions in the videos that do not largely overlap with the positive samples. The examples of the positive and negative samples used to train the SVM are shown in Figs. 4.3 (a) and (b). The size of the detection window is 60 × 60 pixels. To detect the upper bodies of the human objects in various sizes, the original frames are scaled for $2^{-\gamma/4}$, where $\gamma = 3, 4, \cdots, 20$.

4.2.2 Feature extraction

As mentioned above, a camera person moves the camera in accordance with the motion of intentionally-captured persons. In addition, he may change the intentionally-captured persons if he finds a more interesting object. According to Elazary et al. and their definition of interesting objects, visual attention can predict such objects [78]. Thus, visual attention affects the process to determine intentionally-captured persons.

Based on this observation, two types of features, i.e., capture intention-related (CI) and visual attention-related (VA), are extracted from each detected human object. These features are used for classifying the human objects into ICHOs or non-ICHOs.

Capture intention-related features

- *Position of human object* (POSX and POSY) is the horizontal and vertical coordinates of the center position of the upper body region, denoted by v^{POSX} and

v^{POSY}.

- *Area of human object* (AREA) is the area of the upper body region denoted by v^{AREA}, which corresponds to the area in the red rectangle in Fig. 4.4 (b).

- *Distance between centers of upper body region and frame* (DDF) is denoted by v^{DDF} and calculated as:

$$v^{\mathrm{DDF}} = \sqrt{(v^{\mathrm{POSX}} - x^{\mathrm{F}})^2 + (v^{\mathrm{POSY}} - y^{\mathrm{F}})^2}, \tag{4.1}$$

  where $(x^{\mathrm{F}}, y^{\mathrm{F}})$ is the center of the frame. DDF is expected to be a useful feature for ICHO classification because camera persons tend to center ICHOs in the frames.

- *Amplitude of camera motion* (ACM) represents how much the camera person moves the camera. To extract this feature, we model camera motion by translation and scaling between two successive video frames. Let $\mathbf{p}_t$ and $\mathbf{p}_{t+1}$ denote two-dimensional column vectors representing an arbitrary point in the t-th frame and its corresponding point in the $(t+1)$-th frame. The camera motion between these frames is modeled as

$$\mathbf{p}_{t+1} = \zeta_t \mathbf{p}_t + \mathbf{c}_t, \tag{4.2}$$

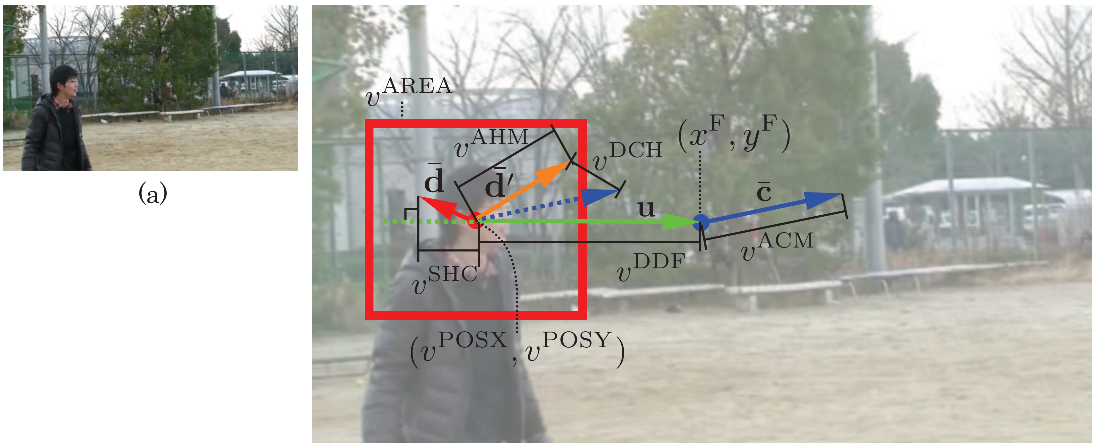
  where $\mathbf{c}_t$ and ζ_t are the translation and the scaling factor. These parameters are obtained by [79]. For the t-th frame, ACM, v^{ACM}, is defined as the amplitude of the weighted average of the translations over $2N_{\mathrm{CI}}$ successive frames centered at the t-th frame indicated by the solid blue arrow in Fig. 4.4 (b). ACM is given by $v^{\mathrm{ACM}} = \|\bar{\mathbf{c}}\|_2$, where $\bar{\mathbf{c}}$ is the average translation obtained by

$$\bar{\mathbf{c}} = \sum_{\tau} \varpi_\tau \mathbf{c}_{t+\tau} \tag{4.3}$$

  and ϖ_τ is the weight given by

$$\varpi_\tau = \frac{N_{\mathrm{CI}} - |\tau|}{\sum_{\tau'}(N_{\mathrm{CI}} - |\tau'|)}. \tag{4.4}$$

  In the above two equations, the summations are calculated for $-N_{\mathrm{CI}}, -N_{\mathrm{CI}} +$



(a)

(b)

Figure 4.4: (a) Example of frame and (b) capture intention-related (CI) features extracted from human object.

$1, \cdots, N_{\mathrm{CI}} - 1$.

- *Amplitude of compensated human object motion* (AHM) represents how much the human object moves indicated by the orange arrow in Fig. 4.4 (b). Using active search [80], the motion of the human object in the t-th frame is obtained by tracking the upper body region in the frame for $2N_{\mathrm{CI}}$ successive frames centered at the t-th frame. Since such human object motion consists of actual human object motion caused by the movement of the corresponding person and the camera motion, we compensate the camera motion. Let $\mathbf{d}_\tau$ denote the human object motion from the $(t+\tau)$-th frame to the $(t+\tau+1)$-th frame. The value of $\mathbf{d}_\tau$ can be erroneous due to tracking errors. Therefore, we introduce weight ϖ'_τ based on similarity λ_τ between the upper body region in the t-th frame and that in the $(t+\tau)$-th frame, defined as the histogram intersection calculated while tracking the upper body region as in [80]:

$$\varpi'_\tau = \frac{\varpi_\tau \varsigma(\lambda_\tau)}{\sum_{\tau'} \varpi_{\tau'} \varsigma(\lambda_{\tau'})}, \tag{4.5}$$

where $\varsigma(\lambda_{\tau'}) = 1/[1+e^{-\phi_1(\lambda_{\tau'}-\phi_2)}]$. ϕ_1 and ϕ_2 are the scaling and the bias, respectively. Using ϖ'_τ, the weighted average of compensated human object motion $\bar{\mathbf{d}}'$

is computed as

$$\bar{\mathbf{d}}' = \sum_{\tau} (\mathbf{d}_\tau + \mathbf{c}_{t+\tau}) \varpi'_\tau. \tag{4.6}$$

This weighted average can alleviate the effect of tracking errors by decreasing the weight when similarity λ_τ is small. AHM, v^{AHM}, is obtained as $v^{\mathrm{AHM}} = \|\bar{\mathbf{d}}'\|_2$.

- *Distance between camera motion and compensated human object motion* (DCH) is the distance between $\bar{\mathbf{d}}'$ and $\bar{\mathbf{c}}$ given by

  $$v^{\mathrm{DCH}} = \|\bar{\mathbf{d}}' - \bar{\mathbf{c}}\|_2. \tag{4.7}$$

  This is the Euclidean distance between the vectors indicated by the orange and dashed blue arrows, which is a translation of the solid blue arrow in Fig. 4.4 (b). A small DCH value implies that the person corresponding to the human object is likely to be followed by the camera person.

- *Similarity between human object motion and vector from center of upper body region to center of frame* (SHC) represents how likely the camera person is to center the human object in the frame. SHC, v^{SHC}, is defined as

  $$v^{\mathrm{SHC}} = \frac{\mathbf{u}^\top \bar{\mathbf{d}}}{\|\mathbf{u}\|_2 \, \|\bar{\mathbf{d}}\|_2}, \tag{4.8}$$

  where $\mathbf{u}$ is the vector from the center of the upper body region to the center of the frame, and $\bar{\mathbf{d}}$ is the human object motion without compensation given as

  $$\bar{\mathbf{d}} = \sum_{\tau} \varpi'_\tau \mathbf{d}_\tau. \tag{4.9}$$

  Vectors $\mathbf{u}$ and $\bar{\mathbf{d}}$ are indicated by the green and red arrows in Fig. 4.4 (b).

We define an eight-dimensional CI feature vector as

$$\mathbf{v}^{\mathrm{CI}} = (v^{\mathrm{POSX}}, v^{\mathrm{POSY}}, v^{\mathrm{AREA}}, v^{\mathrm{DDF}}, v^{\mathrm{ACM}}, v^{\mathrm{AHM}}, v^{\mathrm{DCH}}, v^{\mathrm{SHC}}). \tag{4.10}$$

Visual attention-related features

We employ the bottom-up visual attention model proposed by Itti et al. [66] to extract the VA features from each human object. To construct the saliency map, their model generates seven feature maps: intensity, red-green and blue-yellow opponent colors, and orientations for 0°, 45°, 90°, and 135°. We calculate the average values in the upper body region for the feature maps as well as the resulting saliency map. The average value is denoted by v^{χ}, where $\chi \in \{\text{INT, RG, BY, O0, O45, O90, O135, SAL}\}$, and an eight-dimensional VA feature vector is defined as

$$\mathbf{v}^{\text{VA}} = (v^{\text{INT}}, v^{\text{RG}}, v^{\text{BY}}, v^{\text{O0}}, v^{\text{O45}}, v^{\text{O90}}, v^{\text{O135}}, v^{\text{SAL}}). \tag{4.11}$$

4.2.3 Intentionally-captured human object model

For classifying each human object into ICHO ($y = +1$) or non-ICHO ($y = -1$) where y is a class label, we statistically model the ICHOs based on two SVMs with a radial basis function (RBF): one for the CI feature vectors and the other for the VA feature vectors. Assuming that the CI and VA feature vectors are independent under the condition of given y, these separated feature vectors reduce the dimensionality of the features for each SVM and improve the generalization performance of ICHO classification. We believe that the assumption of conditional independence is reasonable because the CI feature vectors mainly come from the composition of video frames, while the VA feature vectors are mainly derived from intensity, color, and texture. Therefore, the correlation between the CI and VA feature vectors is expected to be small under the condition of given y. The SVMs are trained separately using training data with class labels. The outputs of the trained SVMs for the CI and VA feature vectors are denoted by $g^{\text{CI}}(\cdot)$ and $g^{\text{VA}}(\cdot)$.

The SVM outputs are calibrated to the posterior probabilities by [81] as

$$p(y = +1|\mathbf{v}^{\upsilon}) = \frac{1}{1 + \exp[\vartheta_1^{\upsilon} g^{\upsilon}(\mathbf{v}^{\upsilon}) + \vartheta_2^{\upsilon}]}, \tag{4.12}$$

where $\upsilon \in \{\text{CI}, \text{VA}\}$. Parameters ϑ_1^{υ} and ϑ_2^{υ} are determined by minimizing the crossentropy error function given by

$$-\sum_i \left\{ \psi_i \log p(y_i = +1|\mathbf{v}_i^v) + (1-\psi_i)\log[1 - p(y_i = +1|\mathbf{v}_i^v)] \right\}, \tag{4.13}$$

where

$$\psi_i = \begin{cases} \dfrac{N_+ + 1}{N_+ + 2} & \text{if } y_i = +1 \\ \dfrac{1}{N_- + 2} & \text{otherwise} \end{cases} . \tag{4.14}$$

N_+ and N_- are the numbers of ICHOs and non-ICHOs in the training data.

Assuming the conditional independence of feature vectors given y, we can write the probability of $\mathbf{v}$ given y as

$$p(\mathbf{v}|y) = p(\mathbf{v}^{\text{CI}}|y)\, p(\mathbf{v}^{\text{VA}}|y), \tag{4.15}$$

where $\mathbf{v} = (\mathbf{v}^{\text{CI}}, \mathbf{v}^{\text{VA}})$. Using (4.15) and the Bayesian theorem, the posterior probabilities based on the SVM outputs are combined into the posterior probability of y given $\mathbf{v}$ as

$$\begin{aligned} p(y|\mathbf{v}) &\propto p(\mathbf{v}^{\text{CI}}|y)\, p(\mathbf{v}^{\text{VA}}|y)\, p(y) \qquad (4.16) \\ &\propto \frac{p(y|\mathbf{v}^{\text{CI}})\, p(y|\mathbf{v}^{\text{VA}})}{p(y)}. \qquad (4.17) \end{aligned}$$

Introducing normalizing constant Γ given as

$$\Gamma = \frac{p(y=+1|\mathbf{v}^{\text{CI}})\, p(y=+1|\mathbf{v}^{\text{VA}})}{p(y=+1)} + \frac{p(y=-1|\mathbf{v}^{\text{CI}})\, p(y=-1|\mathbf{v}^{\text{VA}})}{p(y=-1)}, \tag{4.18}$$

we obtain

$$p(y|\mathbf{v}) = \frac{p(y|\mathbf{v}^{\text{CI}})\, p(y|\mathbf{v}^{\text{VA}})}{\Gamma p(y)}. \tag{4.19}$$

Using this posterior probability, the human object can be classified as ICHO if

$$p(y=+1|\mathbf{v}) > TH_{\text{PR}} \tag{4.20}$$

is satisfied where TH_{PR} is a threshold.

4.2.4 Intentionally-captured human object classification incorporating temporal consistency

Since the classification by (4.20) does not consider the temporal consistency of the ICHOs, a human object or even a false positive of the upper body detector that accidentally satisfies (4.20) can be classified into an ICHO. Therefore, we track each human object and classify it into ICHO/non-ICHO based on the tracking results.

Let $H_{t,n}$ and $\mathbf{v}'_{t,n} = (v^{\mathrm{POSX}}_{t,n}, v^{\mathrm{POSY}}_{t,n}, v^{\mathrm{AREA}}_{t,n})$ denote the n-th human object in the t-th frame and the reduced feature vector with its position and area. Assuming that the frame rate of the video is sufficiently high so that the difference of position and area of a human object in successive frames is small, we model the transition from $H_{t-1,k}$ to $H_{t,n}$ by the multivariate Gaussian distribution as

$$p(H_{t,n}|H_{t-1,k}) = \mathcal{N}(\mathbf{v}'_{t,n}|\boldsymbol{\mu}_{\mathrm{TR}}, \Sigma_{\mathrm{TR}}), \tag{4.21}$$

where $\boldsymbol{\mu}_{\mathrm{TR}} = \mathbf{v}'_{t-1,k}$ is the mean and $\Sigma_{\mathrm{TR}} = \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_a^2)$ is the diagonal covariance matrix whose elements are empirically determined as $\sigma_x^2 = \sigma_y^2 = 40$ and $\sigma_a^2 = 0.5 \times v^{\mathrm{AREA}}_{t-1,k}$.

Human object $H_{t,n}$ is tracked by finding a human object sequence $\mathbf{H}_{t,n} = \{H^{\tau}_{t,n}|\tau = 1, \cdots, N_{\mathrm{TR}}\}$ that maximizes joint probability $p(\mathbf{H}_{t,n})$, where N_{TR} is the number of frames to be tracked, $H^{\tau}_{t,n}$ is a human object in $\mathbf{H}_{t,n}$, which is one of the human objects in the $(t+\tau-1)$-th frame, and $H^1_{t,n} = H_{t,n}$. We assume that the joint probability of $\mathbf{H}_{t,n}$ can be factorized as

$$p(\mathbf{H}_{t,n}) = p(H^1_{t,n}) \prod_{\tau=2}^{N_{\mathrm{TR}}} p(H^{\tau}_{t,n}|H^{\tau-1}_{t,n}), \tag{4.22}$$

where $p(H^1_{t,n}) = 1$. The sequence that maximizes the joint probability is given by

$$\mathbf{H}^*_{t,n} = \arg\max_{\mathbf{H}_{t,n}} p(\mathbf{H}_{t,n}). \tag{4.23}$$

```
Classification algorithm
  Initialize $C_{t,n}$ ($t = 1, 2, \cdots, T$, $n = 1, 2, \cdots$) to zero
  for $t = 1$ to $T$ do
    for all $n$ do
      Find $\mathbf{H}^*_{t,n}$ and $V_{t,n}$ by tracking $H_{t,n}$ using (4.24)
      if $p(\mathbf{H}^*_{t,n}) > TH_{\mathrm{TR}}$ and $p(\mathbf{y}_{t,n} = \mathbf{1}|V_{t,n}) > TH_{\mathrm{ALG}}$ then
        Increment all $C_{t,n}$ associated with the human objects in $\mathbf{H}^*_{t,n}$
      end if
    end for
  end for
  for $t = 1$ to $T$ do
    for all $n$ do
      if $C_{t,n} > TH_{\mathrm{C}}$ then
        $\hat{y}_{t,n} \leftarrow +1$
      else
        $\hat{y}_{t,n} \leftarrow -1$
      end if
    end for
  end for
```

Figure 4.5: Classification algorithm.

This is equivalent to

$$\mathbf{H}^*_{t,n} = \arg\max_{\mathbf{H}_{t,n}} \sum_{\tau=2}^{N_{\mathrm{TR}}} \log p(H^{\tau}_{t,n}|H^{\tau-1}_{t,n}), \tag{4.24}$$

which can be maximized by dynamic programming. The tracking of $H_{t,n}$ is judged successful if

$$p(\mathbf{H}^*_{t,n}) > TH_{\mathrm{TR}} \tag{4.25}$$

is satisfied, where TH_{TR} is an empirically determined threshold. This tracking can filter out the false positives of the upper body detector because those that appear in a frame often disappear in the next frame and significantly decrease $p(\mathbf{H}^*_{t,n})$.

Let $\mathbf{y}_{t,n} = \{y^{\tau}_{t,n}|\tau = 1, \cdots, N_{\mathrm{TR}}\}$ and $V_{t,n} = \{\mathbf{v}^{\tau}_{t,n}|\tau = 1, \cdots, N_{\mathrm{TR}}\}$ denote the sequences of the labels and the feature vectors for $\mathbf{H}^*_{t,n}$. Each human object may be

Table 4.1: Values of parameters used in experiments.

N_{CI}	ϕ_1	ϕ_2	TH_{TR}
5	40	0.85	4.25×10^{-18}

classified as ICHO if

$$p(\mathbf{y}_{t,n} = \mathbf{1}|V_{t,n}) = \prod_{\tau=1}^{N_{\mathrm{TR}}} p(y_{t,n}^{\tau} = +1|\mathbf{v}_{t,n}^{\tau}) > TH_{\mathrm{ALG}} \tag{4.26}$$

is satisfied, where $\mathbf{1}$ represents a N_{TR}-dimensional vector whose elements are +1 and TH_{ALG} is a threshold. Otherwise, an optimal sequence for $\mathbf{y}_{t,n}$ can be obtained, for example, by adopting the hidden Markov model. However, classification based on a single tracking result can be erroneous because our tracking method might fail to give the correct sequences. Therefore, we apply the classification algorithm in Fig. 4.5 where T denotes the number of frames in the video. In this algorithm, we count how many times the human object sequences that include $H_{t,n}$ satisfy (4.25) and (4.26) by $C_{t,n}$. We introduce a condition (4.25) because a human object sequence can contain human objects that correspond to multiple persons or false positives of the upper body detector if the tracking of the human object sequence fails. The algorithm classifies $H_{t,n}$ as ICHO if $C_{t,n}$ exceeds the threshold TH_{C}. In this case, classification result $\hat{y}_{t,n}$ is set to +1, which means that $H_{t,n}$ is an ICHO, and otherwise, $\hat{y}_{t,n}$ is set to −1. Hence, in this algorithm, the classification result of a human object is determined based on the results of N_{TR} times tracking, which can differ depending on the starting frame. Therefore, this algorithm can provide the correct classification result even when some tracking results are erroneous.

4.3 Experimental results

Our method for ICHO detection consists of human object detection, feature extraction, and ICHO classification. In our experiments, we evaluate the contributions of the features used for ICHO classification and then compare the ICHO classification performance with several baselines using a video dataset containing 20 videos (VD1). To show the performance of ICHO detection in practical uses, we evaluate the overall performance including human object detection and ICHO classification. We also evaluate the ICHO classification performance and the overall performance using a larger video dataset containing 99 videos (VD2).

The parameter values used throughout our experiments are summarized in Table 4.1. Parameter N_{CI} controls the smoothness of the camera and human object motion, which is usually shaky because of camera shakes. In [82], Matsushita et al. smoothed shaky motion using neighboring 13 frames for video stabilization. According to this work and our observation that ICHO classification does not require extensive smoothing, we set N_{CI} to 5 and consequently used the neighboring 10 frames for smoothing the camera and human object motion. The values of ϕ_1 and ϕ_2 are determined based on our preliminary study that indicated that the active search [80], which is used in the feature extraction, usually succeeds when similarity λ_τ is larger than 0.95, and that it usually fails when λ_τ is smaller than 0.75. The value of TH_{TR} was set empirically.

4.3.1 Contributions of features

We evaluated the contributions of the CI and VA features using VD1, which contains 20 videos consisting of 32,725 frames taken by three camera persons.

These videos have 854×480 pixels with 29.97 frames per second. The human objects were manually specified, and we did not use the upper body detector in Section 4.2.1 to show how efficiently combinations of features discriminate ICHOs from non-ICHOs. The labels representing ICHOs/non-ICHOs were assigned by the camera persons as the ground truth. The number of human objects was 56,067; ICHOs and non-ICHOs were 38,122 and 17,945, respectively. We trained the SVMs for all combinations of features using five-fold cross-validation and calculated the area under the ROC curve (AUC) by applying thresholding to the SVM outputs. Since the SVM training is computationally too expensive if we use all training data, we randomly chose 10,000 samples from the training data for each cross-validation trial and trained the SVMs with them.

Figures 4.6 (a)–(d) show the example results for the CI features when the number of the used features $N_{\mathrm{F}}^{\mathrm{CI}}$ is 1, 2, 6, and 7, respectively. As expected, Fig. 4.6 (a) shows that DDF is the most useful for ICHO classification when only one feature is used. POSX and POSY also give high AUC values, but ACM, AHM, DCH, and SHC give significantly lower values than the others. Figure 4.6 (b) indicates that combinations

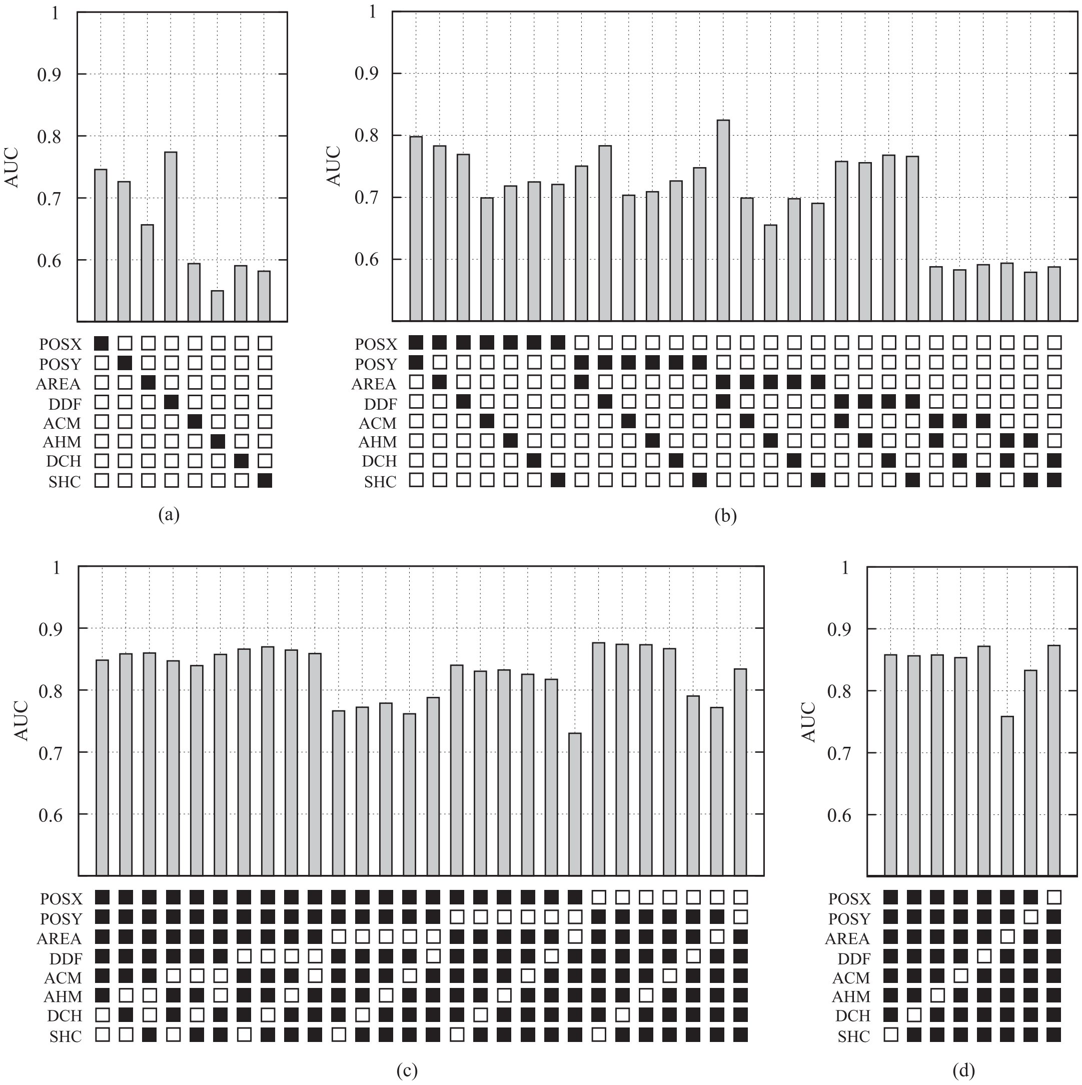


Figure 4.6: Area under ROC curve (AUC) values for combinations of features from capture intention-related (CI) features: number of used features $N_{\mathrm{F}}^{\mathrm{CI}}$ for (a), (b), (c) or (d) is 1, 2, 6, or 7, respectively. ■ and □ represent corresponding features are used and are not used, respectively.

that include POSX, POSY, or DDF give the superior performance. From Fig. 4.6 (d), POSY is more useful than POSX when combined with other features. This is because POSX's distribution for ICHOs is broad since camera persons often arrange human objects near the edges of video frames to capture more than one ICHO. In addition, interestingly, the AUC values of the combinations without AREA are significantly

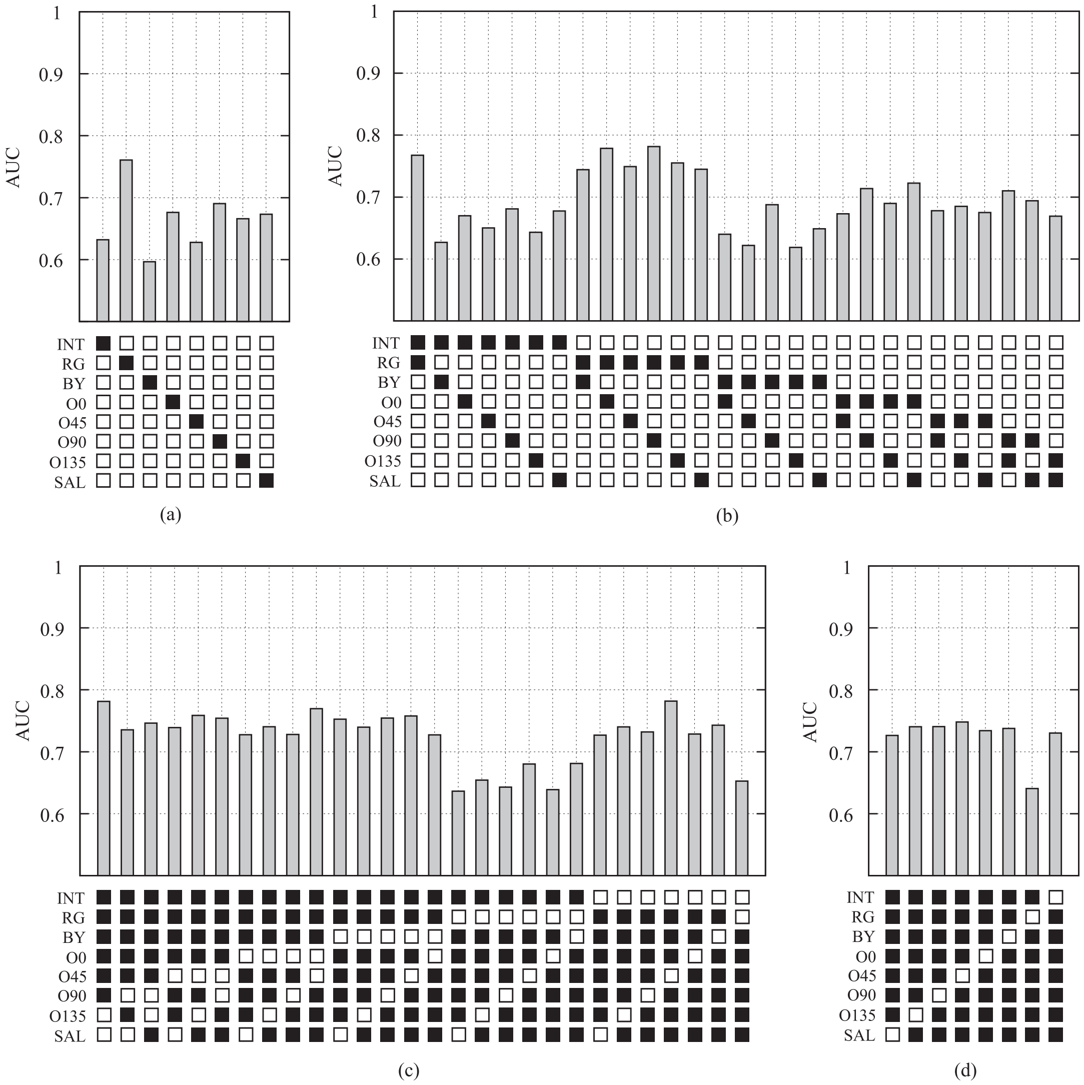


Figure 4.7: Area under ROC curve (AUC) values for combinations of features from visual attention-related (VA) features: number of used features $N_{\mathrm{F}}^{\mathrm{VA}}$ for (a), (b), (c), or (d) is 1, 2, 6, or 7, respectively. ■ and □ represent corresponding features are used and are not used, respectively.

lower (Figs. 4.6 (c) and (d)). Other combinations for $N_{\mathrm{F}}^{\mathrm{CI}} = 3$, 4, or 5 give similar results. The AUC value for $N_{\mathrm{F}}^{\mathrm{CI}} = 8$ (all CI features are used) is 0.86. From these results, the features related to the spatial positions of human objects are beneficial for ICHO classification. AREA is an essential feature when combined with spatial position-related features, although AREA itself is less useful. Furthermore, the contributions

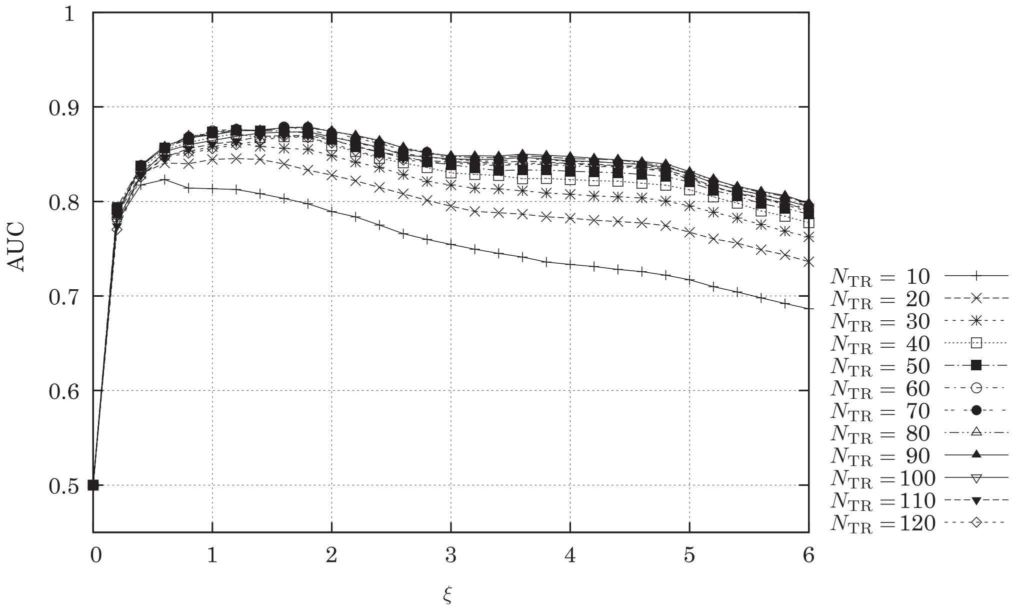


Figure 4.8: Area under ROC curve (AUC) values of intentionally-captured human object (ICHO) classification with ALG for various values of N_{TR} and TH_{ALG}. Horizontal axis is ξ such that $TH_{\mathrm{ALG}} = (2^{-N_{\mathrm{TR}}})^{\xi}$ (see text for details).

of the camera motion related features, i.e., ACM, AHM, DCH, and SHC, are low. Therefore, we conclude that camera persons pay much attention to the positions and sizes of ICHOs, and the CI features related to camera motion cannot well describe the behavior of camera persons very well.

The example results for the VA features are shown in Figs. 4.7 (a)–(d). The number of used features $N_{\mathrm{F}}^{\mathrm{VA}}$ is 1, 2, 6, or 7, respectively. When all VA features are used, the AUC value is 0.73. These results indicate that combinations including RG give good performances. This is the same for all the values of $N_{\mathrm{F}}^{\mathrm{VA}}$. One reason is that ICHOs tend to be captured with their frontal or profile faces, and RG yields larger values when the faces of human objects are visible. These results demonstrate that camera persons are not very affected by visual attention.

4.3.2 Classification performance evaluation

We evaluated the performance of the ICHO classification with the classification algorithm shown in Fig. 4.5 (ALG) using VD1 for various values of parameters N_{TR} and TH_{ALG} in terms of the AUC values obtained by changing TH_{C}. The SVMs for the CI

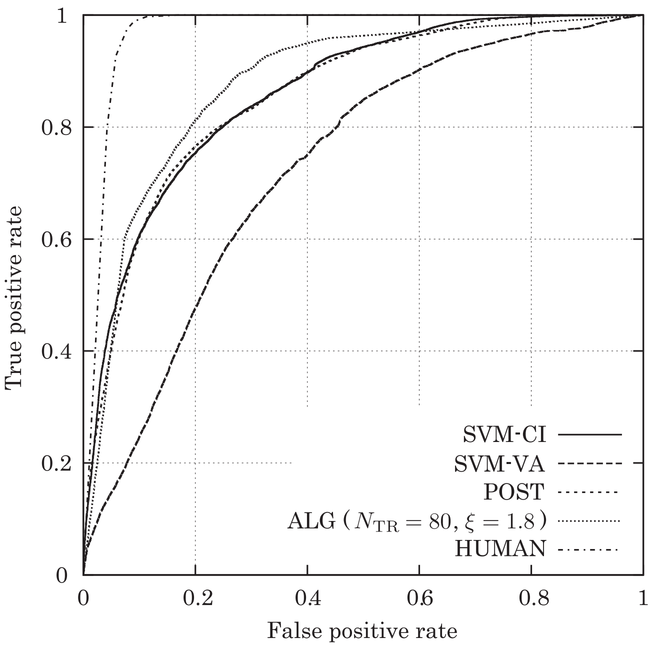


Figure 4.9: ROC curves of SVM-CI, SVM-VA, POST, ALG, and HUMAN over VD1.

and VA features were those trained with $N_{\mathrm{F}}^{\mathrm{CI}} = 8$ and $N_{\mathrm{F}}^{\mathrm{VA}} = 8$ in the previous section. We also compared the ALG performance with several baselines.

The results are shown in Fig. 4.8. We use ξ such that $TH_{\mathrm{ALG}} = (2^{-N_{\mathrm{TR}}})^{\xi}$ for the horizontal axis of Fig. 4.8 instead of actual threshold TH_{ALG} because the probability given by (4.26) largely varies depending on N_{TR}. The maximum value of AUC is 0.879 when $N_{\mathrm{TR}} = 80$ and $\xi = 1.8$. The AUC value increases as N_{TR} increases to 80, but it decreases as N_{TR} increases from 80. A possible reason is that camera persons intentionally capture a person for about 80 frames in most cases.

The ROC curve of ALG is shown in Fig. 4.9. We also show the ROC curves of SVM with all CI features (SVM-CI), the SVM with all VA features (SVM-VA), and the posterior probability given by (4.20) (POST) as baselines. The performance of the human annotators (HUMAN) was also evaluated as a baseline. To this end, we asked six human annotators to separately assign a label that represents ICHO/non-ICHO to each human object. The ROC curve was generated by thresholding the number of human annotators who agreed. The SVM-VA performance is significantly low, and POST fails to improve the performance compared to SVM-CI; ALG actually gives a superior performance. The HUMAN performance is prominently high.

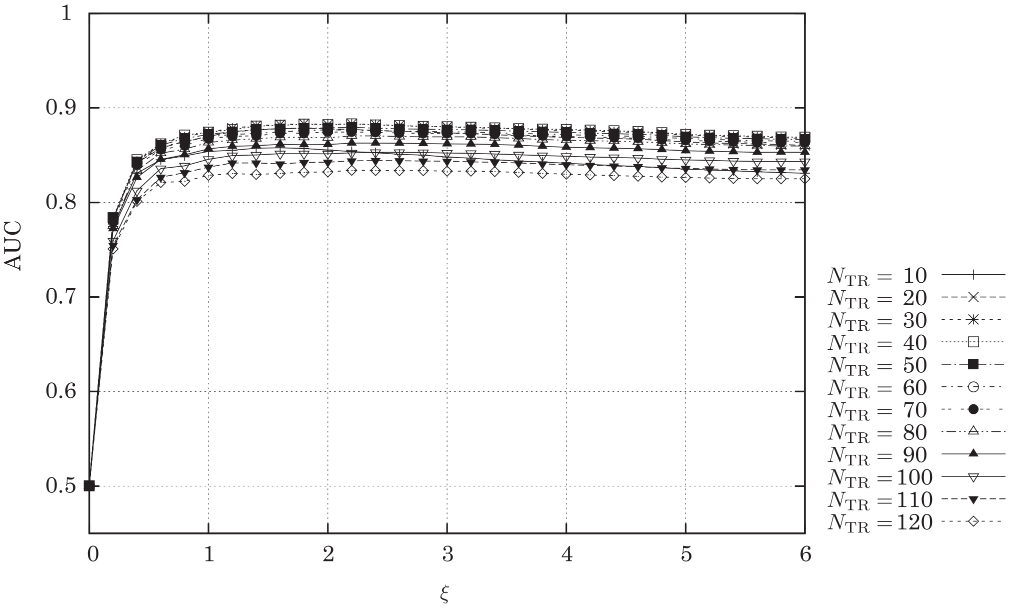


Figure 4.10: Area under ROC curve (AUC) values of intentionally-captured human object (ICHO) detection with ALG for various parameter values using upper body detector.

4.3.3 Overall performance evaluation

To evaluate the overall performance, we applied the upper body detector to the video in video dataset VD1. If a detected human object was close to the position and area of one of the manually specified human objects, the detected human object was judged to have been correctly detected and associated with the manually specified human object. Otherwise, it was judged to be a false positive. The upper body detector correctly detects 56% of the human objects and gives 1.14 false positives per video frame. The correctly detected human objects consist of 68% of ICHOs and 31% of non-ICHOs labeled by the camera persons. This result demonstrates that the upper body detector fails to work well for non-ICHOs because they are too small to be detected or only part of them are captured.

Figure 4.10 shows the AUC values of ICHO detection with ALG for various parameter values. To calculate the AUC values, the ICHO/non-ICHO labels of the manually specified human objects were assigned to the detected human objects associated with them. We only used the detected human objects to which the labels were assigned, and thus the AUC values cannot be compared with those in the previous section. The

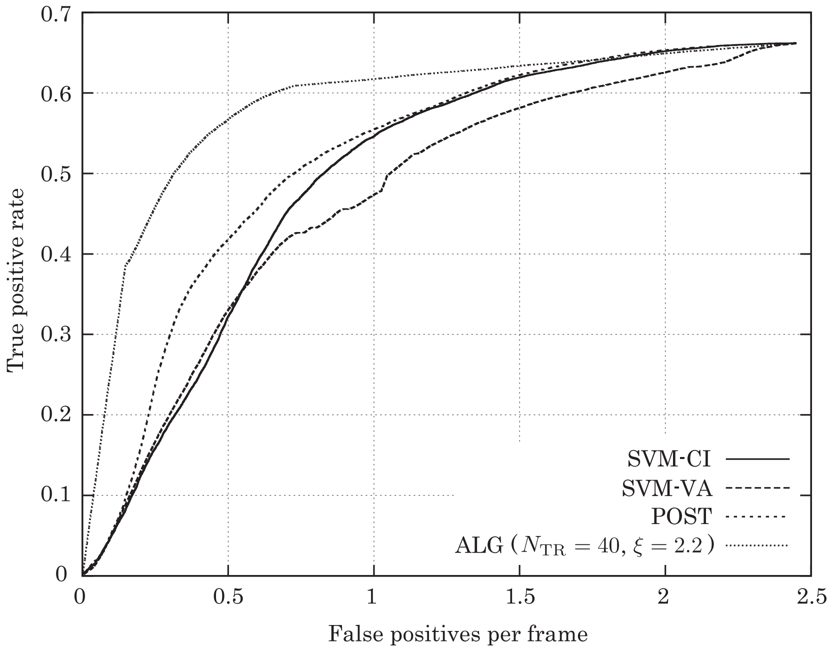


Figure 4.11: False positives per frame versus true positive rate curves of SVM-CI, SVM-VA, POST, and ALG.

maximum value of AUC is 0.88 when $N_{\mathrm{TR}} = 40$ and $\xi = 2.2$. This value of N_{TR} is smaller than that in the previous section. One reason is that the ICHOs, which are not detected by the upper body detector, prevent successful tracking in the classification algorithm when N_{TR} is large. Therefore, N_{TR} should be adjusted based on the performance of the human object detection that is used.

To show the overall performance of ICHO detection, the false positives per frame versus true positive rate curve was generated for ALG (Fig. 4.11), where the false positives per frame and the true positive rate are defined as

$$\begin{aligned} \text{False positives per frame} &= \frac{N'_{\mathrm{FP}}}{T} \\ \text{True positive rate} &= \frac{N'_{\mathrm{TP}}}{N'_{\mathrm{TP}} + N'_{\mathrm{FN}}}. \end{aligned} \tag{4.27}$$

N'_{TP}, N'_{FP}, and N'_{FN} are the numbers of the detected human objects that are correctly classified as ICHOs, those incorrectly classified as ICHOs, and ICHOs incorrectly classified as non-ICHOs or undetected by the upper body detector, respectively. T is the

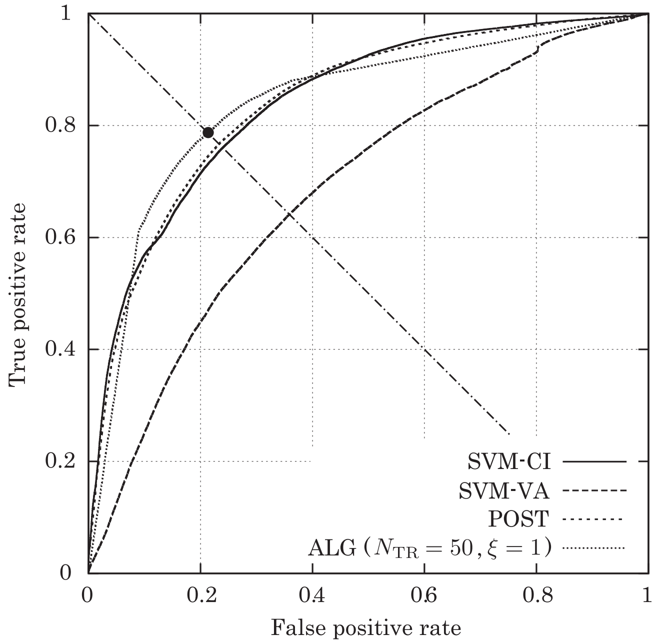


Figure 4.12: ROC curves of SVM-CI, SVM-VA, POST, and ALG over VD2.

number of video frames. Therefore, this curve includes the performance of the upper body detector. We also show the false positives per frame versus true positive rate curves of SVM-CI, SVM-VA, and POST in Fig. 4.11 as baselines. From these curves, the true positive rate for ALG increases much faster than the baselines as the false positives per frame increase, and thus, ALG outperforms the baselines when used with the upper body detector. One of the reasons is as follows. Many false positives of the upper body detector that appear in a frame disappear in the next frame. In this case, the tracking of the false positives in ALG fails and thus such false positives are correctly classified as non-ICHOs. Therefore, the number of false positives of ICHO classification increases slower than the baselines, resulting in the fast increment of the true positive rate.

4.3.4 Classification performance evaluation using larger dataset

We evaluated the ICHO classification performance of ALG as well as baselines using a larger dataset (VD2) that contains 99 YouTube videos. They were selected or excerpted from the original videos based on the following criteria: (a) The video was taken by a camera person with a mobile video camera. (b) It contains human objects. (c) It was not edited; i.e., shot boundaries are not contained in the video. These videos were resized to 854×480. Their frame rate was 29.97 frames per second, and their average length was 40.9 seconds (121,313 frames in total). The 207,539 human objects were manually specified. We again used manually specified human objects instead of the outputs of the upper body detector to demonstrate the classification performance.

Since we did not know which human objects were ICHOs, we asked six human annotators to independently assign ICHO/non-ICHO labels to the human objects as in Section 4.3.2. A human object was judged as ICHO when more than three human annotators agreed. We believe that this procedure is adequate because the accuracy of the human annotators is satisfactory, based on Fig. 4.9. The numbers of ICHOs and non-ICHOs are 123,040 and 84,499. We trained SVMs for the CI and VA features by 33-fold cross-validation. Again, we randomly chose 20,000 samples from the training data for each cross-validation trial and used them for the SVM training to reduce the computational cost.

Figure 4.12 shows the ROC curves of ALG and the baselines, i.e., SVM-CI, SVM-VA, and POST. The parameters for ALG, $N_{\mathrm{TR}} = 50$ and $\xi = 1$, were determined based on the AUC values, as in the previous sections. The AUC values for SVM-CI, SVM-VA, POST, and ALG are 0.842, 0.686, 0.841, and 0.840, respectively. Therefore, the AUC value of ALG does not give the highest value. However, in the sense of the intersection of the ROC curve and the line connecting points $(1, 0)$ and $(0, 1)$ on the ROC space, ALG is the best as indicated in Fig. 4.12. In other words, ALG yields the highest true positive rate while giving the lowest false positive rate among all methods. These curves indicate that the classification performances of ALG and the baselines are almost the same for VD1 and VD2.

4.3.5 Overall performance evaluation using larger dataset

The overall performance of ALG including the upper body detector was also evaluated by 33-fold cross-validation using VD2. We used the trained SVMs and labels that were assigned by the six human annotators in the previous section as the ground truth. The other evaluation settings were the same as in Section 4.3.3.

The upper body detector correctly detected 47% of the 207,539 human objects in

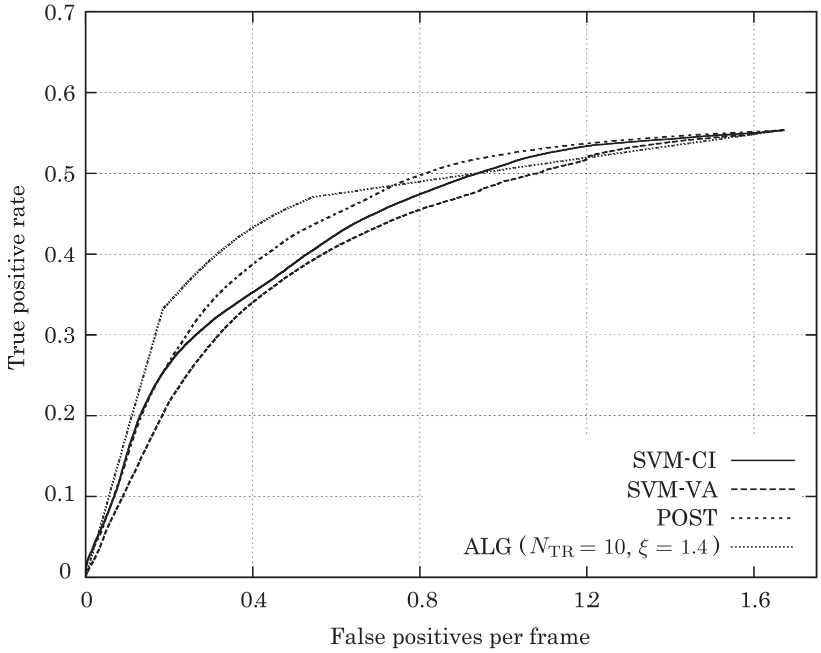


Figure 4.13: False positives per frame versus true positive rate curves of SVM-CI, SVM-VA, POST, and ALG over VD2.

VD2 and gave 1.43 false positives per video frame. The correctly detected human objects consist of 56% of the ICHOs and 25% of the non-ICHOs. The performance of the upper body detector is degraded compared with that in Section 4.3.3 because VD2 contains human objects, e.g., which are only captured with their faces or upside down. ALG gives the highest AUC value when $N_{\mathrm{TR}} = 10$ and $\xi = 1.4$. This value of N_{TR} is much smaller than that in the previous sections because of the low performance of the upper body detector, which makes long-term tracking difficult.

Figure 4.13 shows the false positives per frame versus true positive rate curves of ALG as well as baselines SVM-CI, SVM-VA, and POST. For smaller values of the false positives per frame, ALG gives the best true positive rate. However, the performance improvement by ALG is limited compared with that in Section 4.3.3. This is caused by the small value of N_{TR} due to the low performance of the upper body detector; the small value of N_{TR} is insufficient to model the temporal consistency of ICHOs. Therefore, we must adopt a superior human object detection technique to improve the overall performance of ICHO detection.

4.4 Concluding remarks

In this chapter, we presented a method for ICHO detection that serves as a basis of privacy protection against accidental privacy infringement. This method is potentially applicable to a wide range of applications, such as video summarization [83, 84] and video adaptation [85, 86] as well as privacy protection. To detect ICHOs, we first detect all human objects in a video frame using an upper body detector and classify them into ICHOs/non-ICHOs using capture-intention-and visual-attention-related features.

Our experimental results indicate that the features involving the position of a human object with its area are beneficial cues for ICHO detection. However, visual-attention-related features are not useful. We also experimentally demonstrated that our tracking-based algorithm successfully improved the performance of ICHO classification by exploiting the temporal consistency of ICHOs. Another important result in this chapter is the difficulty of detecting human objects, especially of non-ICHOs. We need to consider this difficulty when designing a system for automatically generating privacy-protected videos.

Chapter 5

Automatic Generation of Privacy-Protected Videos

5.1 Introduction

Mobile video cameras enable us to take videos anywhere including parks and streets. However, such videos may contain persons who are accidentally framed in, which infringes on their privacy. We refer to this privacy infringement as accidental privacy infringement. As mentioned in Section 4.1, regarding accidental privacy infringement, we can reasonably assume that a camera person can obtain permission to capture intentionally-captured persons and to publish the video from them, but not from accidentally-framed-in persons. Our consideration can be summarized as follows:

- The camera person can obtain permission for capturing intentionally-captured persons and publishing the video, or at least can negotiate with them. This is because the number of such persons is usually very small and they are often friends and family members. In this case, the disclosure of the appearance of intentionally-captured persons does not infringe on their privacy.

- It is difficult to obtain permission for capturing and publishing from accidentally-framed-in persons because they are usually passers-by and thus the camera person cannot even negotiate with them. Therefore, the disclosure of the appearance of accidentally-framed-in persons infringes on their privacy.

In this chapter, we present a system that automatically generates privacy-protected videos against accidental privacy infringement. We refer to regions in video frames that correspond to persons as human objects. Based on the above consideration, the human objects corresponding to intentionally-captured persons (ICHOs) can be presented in privacy-protected video. In contrast, the human object except ICHOs (non-ICHOs) should be obscured. We adopt the ICHO detection presented in the previous chapter so that we can selectively obscure only the non-ICHOs. In addition, since ICHOs are essential for the camera person's capture intention, presenting them can maintain the camera person's capture intention.

In the following sections, we describe the results of our preliminary user study for validating the use of ICHO detection with respect to the acceptability of privacy disclosure involving ICHO detection and its ability to preserve capture intentions. We present our system for automatically generating privacy-protected videos in Section 5.3. Section 5.4 presents the experimental results. We finally give concluding remarks in Section 5.5. This chapter is related to the work published in [71, 87, 88, 89, 90].

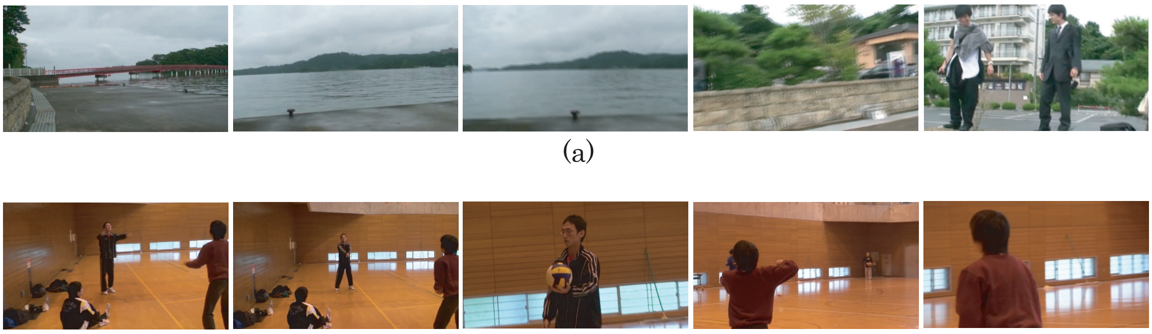
5.2 Preliminary user study

In this section, we present the results of our preliminary study to validate the appropriateness of using ICHO detection for automatically generating privacy-protected videos. Appropriateness was evaluated for the following two aspects:

Acceptability of privacy disclosure: The acceptability of privacy disclosure means whether the privacy disclosure due to the failure of ICHO detection is acceptable. To evaluate this, subjects imagined that they were one of the non-ICHOs in the videos and evaluated whether the privacy disclosure was acceptable.

Adequacy to maintain capture intentions: This means whether visual content essential for the camera persons' capture intentions was sufficiently maintained in the privacy-protected videos. The subjects evaluated whether they felt the video was adequate.

We generated two privacy-protected videos (US1 and US2) from video dataset VD1 that was used in Chapter 4. US1 contains scenery and persons (Fig. 5.1 (a)), and US2



(a)

(b)

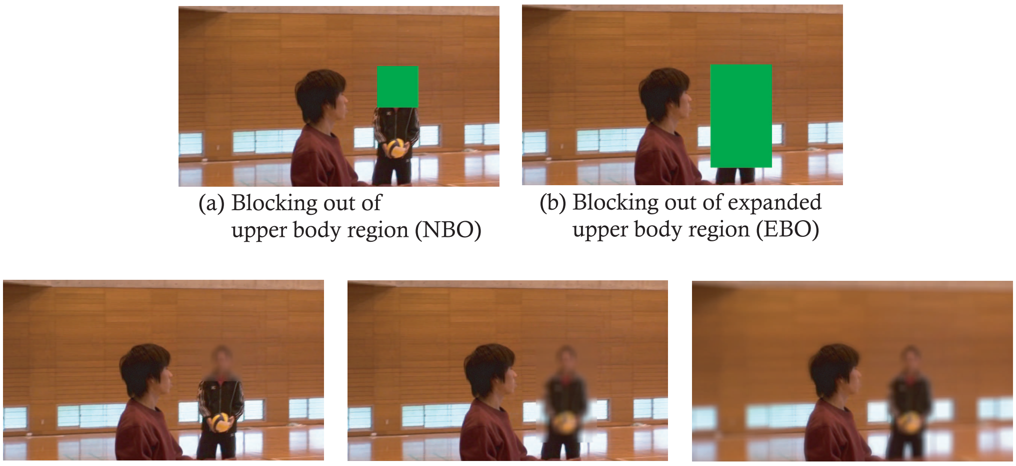
Figure 5.1: Examples of original frames from US1 (a) and US2 (b).

contains persons playing with balls (Fig. 5.1 (b)). Considering the influence of how human objects are obscured, we generated privacy-protected videos by the following obscuring methods:

Blocking out: Blocking out is one of the simplest methods to obscure human objects. The upper body regions of the human objects are blocked out (NBO) (Fig. 5.2 (a)). To obscure a larger part of the human objects, we also blocked out the expanded regions (EBO) (Fig. 5.2 (b)).

Blurring: Blurring obscures the human objects by applying a $K_{\mathrm{B}} \times K_{\mathrm{B}}$ smoothing filter whose elements are $1/{K_{\mathrm{B}}}^2$. In this preliminary user study, we set K_{B} to 10. We adopted blurring on both the upper body regions (NBL) and the expanded regions (EBL) (Figs. 5.2 (c) and (d)). We also adopted a method that blurs regions other than the human objects (OBL) (Fig. 5.2 (e)).

In this preliminary user study, to demonstrate the potential performance of ICHO detection when the detection of human objects is perfect, ICHO classification with ALG ($N_{\mathrm{TR}} = 80$, $\xi = 1.8$, and $TH_{\mathrm{C}} = 80$) was applied to the human objects manually specified by human annotators and the upper body detector was not applied to the videos. The other parameter values were identical to those in Section 4.3.2. With this parameter setting, the true positive rates of ICHO classification for US1 and US2 were 0.73 and 0.85. The false positive rates for US1 and US2 were 0.15 and 0.27. For comparison, we also evaluated the cases where the ground truth of non-ICHOs was used (GT), and where the same number of human objects as the actual non-ICHOs were randomly chosen as non-ICHOs (RND). We asked 11 subjects to assign scores



(a) Blocking out of upper body region (NBO)

(b) Blocking out of expanded upper body region (EBO)

(c) Blurring of upper body region (NBL)

(d) Blurring of expanded upper body region (EBL)

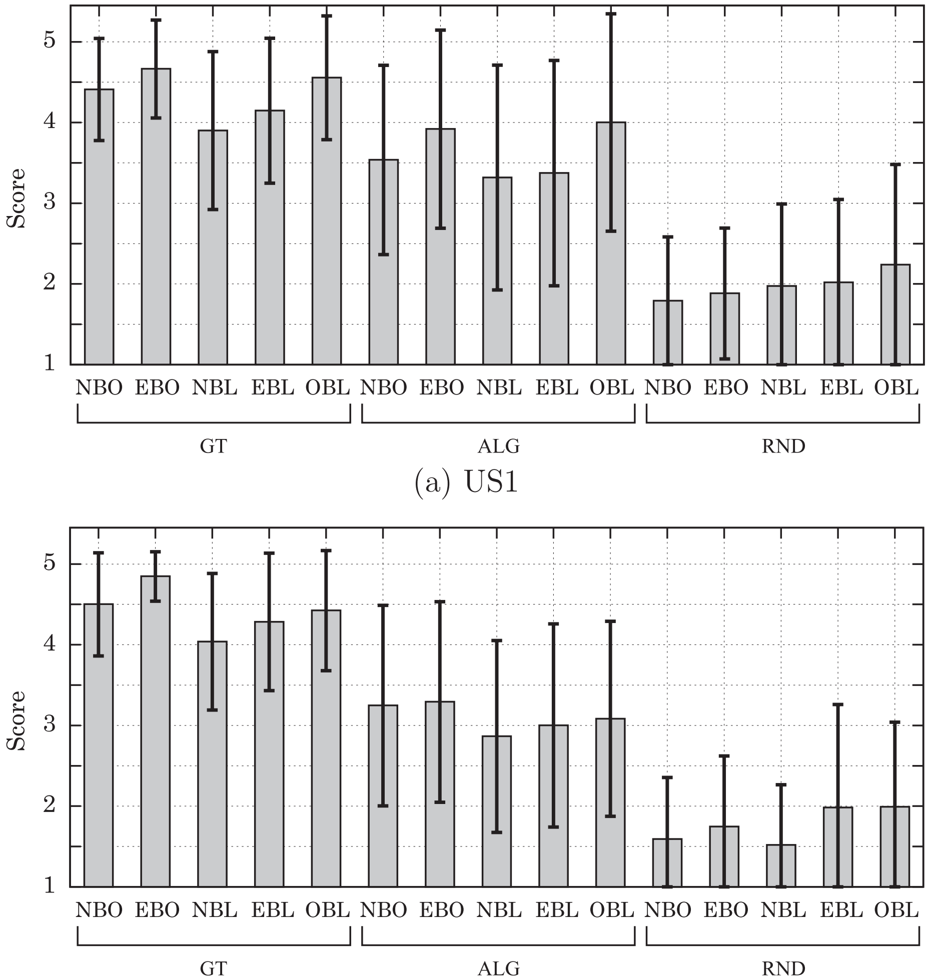
(e) Blurring of regions other than human object to be presented (OBL)

Figure 5.2: Methods for obscuring human objects except intentionally-captured human objects (non-ICHOs).

from 1 (bad) to 5 (excellent). For adequacy, the original videos were presented as a reference for a score of 5. For acceptability, they were presented as a reference for a score of 1.

The means and standard deviations of the acceptability scores for US1 and US2 are shown in Figs. 5.3 (a) and (b). For both US1 and US2, the ALG scores are lower than those of GT. The NBO and EBO scores are higher than those of NBL and EBL. Furthermore, EBO and EBL scores are higher than those of NBO and NBL. These results suggest that the main factors that degrade acceptability are the false positives of ICHO detection and the disclosure of appearance, which is not obscured by the adopted method, such as clothes for NBO and hair color for NBL. For both US1 and US2, the standard deviations for ALG are larger than those for GT because some subjects assigned very low scores and others assigned relatively high scores to the videos generated by ALG when the non-ICHOs were not obscured in some frames due to classification errors. Therefore, TH_{C} should be selected to reduce the false positive rate.

Figures 5.4 (a) and (b) show the means and standard deviations of the adequacy scores for US1 and US2. Significantly large NBO, EBO, NBL, and EBL scores for GT justify using ICHO detection with respect to adequacy. However, although most sub-



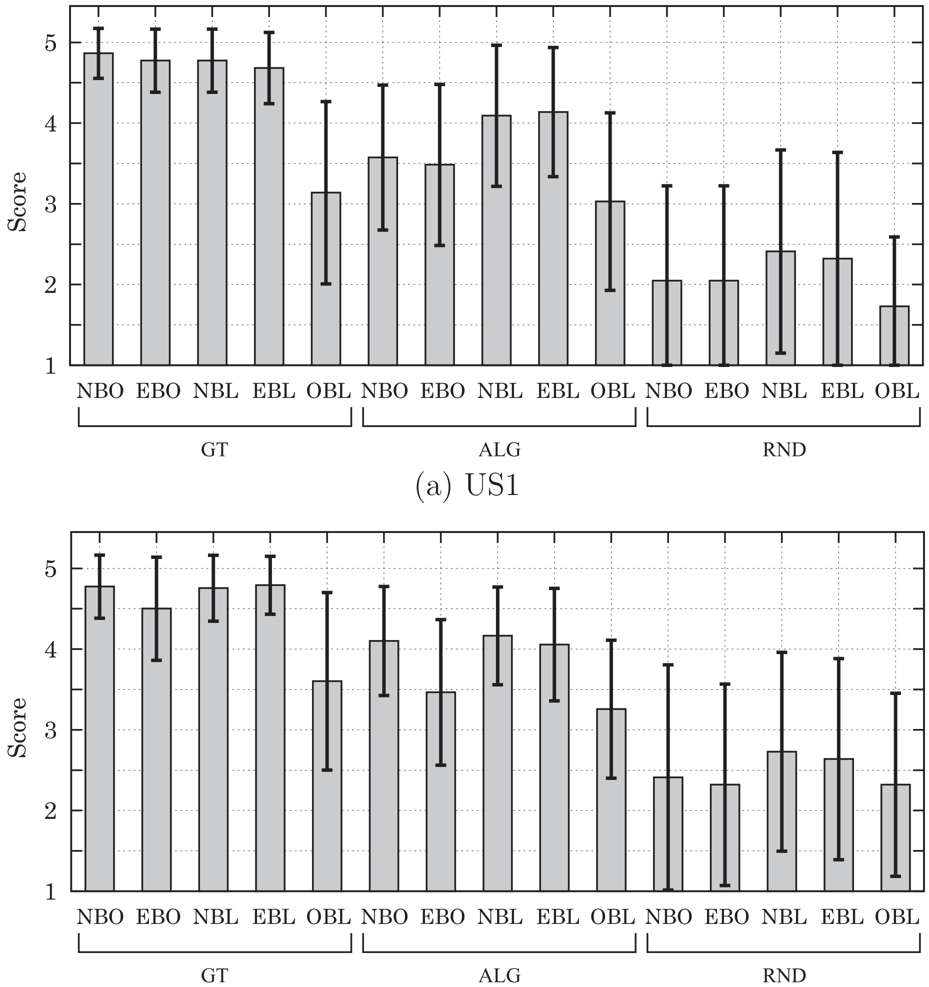
(a) US1

(b) US2

Figure 5.3: Means and standard deviations of acceptability of privacy disclosure for US1 (a) and US2 (b).

jects gave positive responses, the ALG scores are degraded compared to GT, especially for NBO and EBO. This can be caused by the false negatives of ICHO classification, which obscure the ICHOs. Even for GT, OBL gives low scores especially for US1 because US1 contains scenery, which is obscured by OBL.

In summary, ALG is not as good as GT, although it is vastly superior to RND in all cases. Such degradation of ALG can be enhanced if we use the upper body detector to find human objects. Therefore, we need to improve the overall performance of ICHO detection. From the GT results, most subjects are satisfied with the privacy protection



(a) US1

(b) US2

Figure 5.4: Means and standard deviations of adequacy for US1 (a) and US2 (b).

with respect to adequacy. These results also indicate that the subjects prefer EBO to obscure appearance, which means that most pixels in non-ICHOs should be removed.

5.3 System for automatically generating privacy-protected videos

Figure 5.5 shows an overview of our system for automatically generating privacy-protected videos. It consists of background estimation and ICHO detection, extrac-

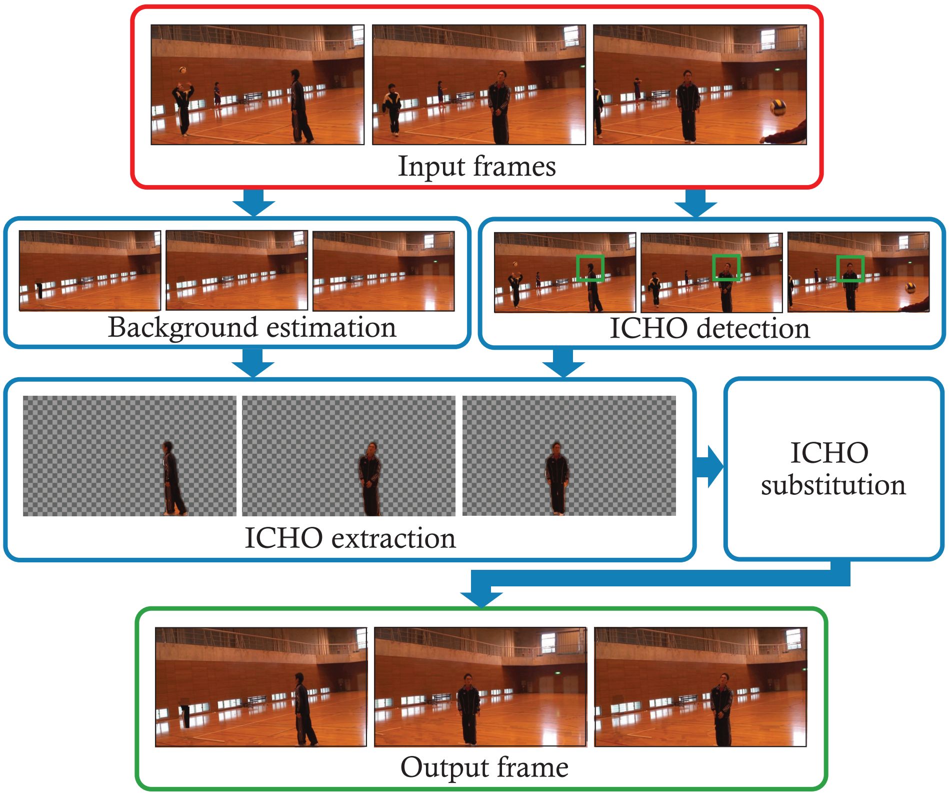


Figure 5.5: Overview of our system using intentionally-captured human object (ICHO) detection and background estimation.

tion, and substitution. First, we estimate the background pixels of a frame using other frames in the video. ICHOs are then detected and extracted. Finally, the background pixels are substituted with ICHOs. This is a novel paradigm for privacy protection. Most existing systems for privacy protection introduced in Chapter 2 first detect human objects to be obscured and then obscure them. On the other hand, in our system, the problem of detecting human objects to be obscured is converted into the problem of detecting human objects to be presented. That is, our system detects the ICHOs that are presented in privacy-protected videos.

This paradigm potentially overcomes the problems of the automatic generation of privacy-protected videos that are revealed in the experimental results in Section 4.3.3 and our preliminary user study in Section 5.2. The problems can be summarized as follows:

(a) The disclosure of the appearance, even a part of the non-ICHOs, degrades the acceptability of privacy disclosure.

(b) The detection of human objects, especially of non-ICHOs, is difficult because they can be too small or only their parts are captured. This difficulty results in the disclosure of non-ICHOs.

In our system, it is not necessary to detect non-ICHOs because the regions other than ICHOs are replaced with estimated background pixels. Since our method for background estimation can estimate the background as long as the non-ICHOs are moving regardless of the difficulty in detecting non-ICHOs, problems (a) and (b) can be partly overcome.

Since we use the ICHO detection presented in the previous chapter, in the following sections, we describe background estimation and ICHO extraction and substitution.

5.3.1 Background estimation

Assuming that non-ICHOs are moving objects, we adopt background estimation to obscure them. Although some background estimation methods for a small number of images have been realized [91, 92] using the graph cuts algorithm [93, 94, 95] and they have been experimentally proven to be practical, it is computationally infeasible to apply them to videos because videos contain too many frames. However, in videos, many frames resemble each other, and such frames only slightly contribute to the background estimation because the background pixels that are occluded by moving objects in these frames are almost the same. Therefore, to make the background estimation computationally feasible, we cluster the frames using a similarity measure based on numbers of correspondent points in pairs of images and extract the most representative frame from each cluster. For background estimation, we only use representative frames and the target frame for which the background is estimated.

First, we extract the SURF features [96] from each frame to find the correspondent points. A SURF feature consists of salient point $\mathbf{p}$ in a frame and its feature. We can find correspondent points by choosing a pair of points on two different frames such that the distance between their features is small. Since exactly finding all correspondent points for all pairs of images is computationally expensive, we use approximate nearest neighbors [97] instead. Let Q_t and $Q_{t'}$ denote the sets of all SURF features for the

t-and t'-th frames, and $ANN_{t,t'}$ be the set of SURF features in the t-th frame for which approximate nearest neighbors are found in the t'-th frame.

Since $ANN_{t,t'}$ contains outliers that are not actual nearest neighbors, we refine $ANN_{t,t'}$ using RANSAC [98]. Assuming the planarity of the scene, RANSAC calculates homography matrix $H_{t',t}$ from the t'-th to the t-th frames using $ANN_{t,t'}$. More specifically, $H_{t',t}$ is a 3×3 matrix that projects point $\mathbf{p}_{t'}$ in the t'-th frame to its correspondent point $\mathbf{p}_t$ in the t-th frame such that the projection error $\mathbf{p}_t - H_{t',t}\mathbf{p}_{t'}$ is small for all correspondent points in $ANN_{t,t'}$, where $\mathbf{p}_t$ and $\mathbf{p}_{t'}$ are represented in the homogeneous coordinates. RANSAC iteratively estimates $H_{t',t}$ to improve the estimation accuracy while excluding outliers that give large projection error. The set of SURF features excluding the outliers is denoted by $A\tilde{N}N_{t,t'}$.

To cluster the frames based on $A\tilde{N}N_{t,t'}$, we adopt affinity propagation [99] because it finds the most representative frame for each cluster during clustering. Similarity measure $sim_{t,t'}$ between the t-and t'-th frames, which is used in affinity propagation, is given by

$$sim_{t,t'} = \frac{|A\tilde{N}N_{t,t'}|}{\max(|Q_t|, |Q_{t'}|)}, \tag{5.1}$$

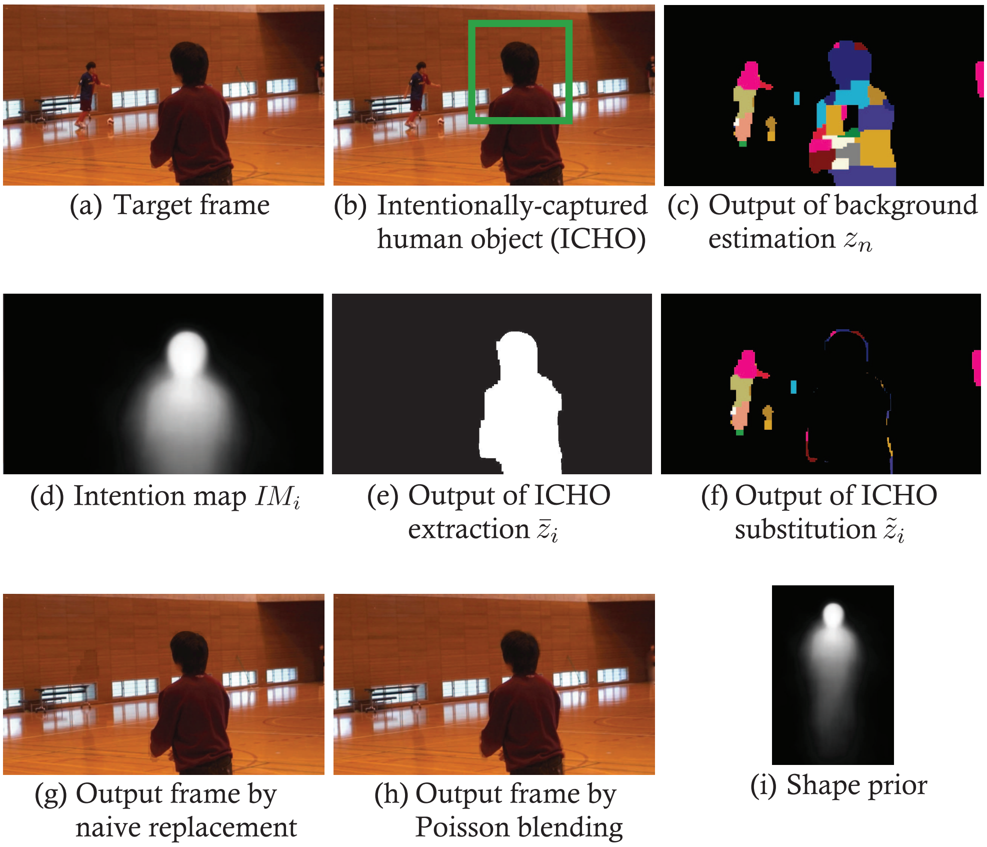
where $|X|$ represents the number of points in set X. This similarity measure describes well how similar the frames are under the assumption of scene planarity. To determine how likely the t-th frame is to be a representative frame, affinity propagation requires a preference value, denoted by $pref_t$, for each value of t. We use the median of the similarity measure based on [99] for all t. The set of the representative frames obtained as the result of affinity propagation is denoted by R.

Next, we estimate the background pixels of a target frame from R using a method proposed by Kim et al. [92], based on the graph cuts algorithm [95, 94, 93]. For target frame I_t, we transform $I_{t'} \in R$ by $H_{t',t}$. The transformed frame is denoted by $H_{t',t}(I_{t'})$. The set consisting of target frame I_t and the transformed frames is denoted by

$$R_t = \{I_t\} \cup \{H_{t',t}(I_{t'}) \,|\, I_{t'} \in R\}. \tag{5.2}$$

Each frame in R_t is divided into square grids consisting of 5×5 pixels.

Let $\mathcal{L}$ be the set of indices of target frame I_t and the frames in R, i.e., $\mathcal{L} = \{t\} \cup \{t'|I_{t'} \in R\}$. The most likely frame to be the background for the n-th grid of



(a) Target frame

(b) Intentionally-captured human object (ICHO)

(c) Output of background estimation z_n

(d) Intention map IM_i

(e) Output of ICHO extraction $\bar{z}_i$

(f) Output of ICHO substitution $\tilde{z}_i$

(g) Output frame by naive replacement

(h) Output frame by Poisson blending

(i) Shape prior

Figure 5.6: Example outputs of each component of our system.

the target frame is denoted by z_n, which takes an index in $\mathcal{L}$. That is, $z_n = t'$ means that $I_{t'}$ is most likely to be the background for the n-th grid of the target frame. The background is estimated by finding z_n that minimizes the energy given by

$$E^{\mathrm{BE}} = \sum_n F_n^{\mathrm{BE}}(z_n) + \sum_{(n,k)\in AG} G_{n,k}^{\mathrm{BE}}(z_n, z_k). \tag{5.3}$$

In this equation, $F_n^{\mathrm{BE}}(z_n)$ is the data term that represents how likely the n-th grid of the frame associated with z_n is to be the background, and $G_{n,k}^{\mathrm{BE}}(z_n, z_k)$ is the smooth term to confirm the continuity among adjacent grids where AG is the set of all the adjacent grids.

Data term $F_n^{\mathrm{BE}}(z_n)$ is defined as follows:

$$F_n^{\mathrm{BE}}(z_n) = \epsilon_1 \sum_{z'\in\mathcal{L}} \max[d_{\Omega_n}(z_n, z'), \epsilon_3] + \epsilon_2 \delta(t, z_n), \tag{5.4}$$

where ϵ_1 and ϵ_2 are constants to determine the contribution of each term. Function $d_{\Omega_n}(z_n, z')$ is defined as the distance between frames J and J' in R_t that correspond to indices z_n and z' in the n-th grid given by

$$d_{\Omega_n}(z_n, z') = \frac{1}{|\Omega_n|} \sum_{i \in \Omega_n} \left(\frac{J(i)}{\bar{J}_{\Omega_n}} - \frac{J'(i)}{\bar{J'}_{\Omega_n}} \right), \tag{5.5}$$

where Ω_n is the set of pixels in the n-th grid, $\bar{J}_{\Omega_n}$ and $\bar{J'}_{\Omega_n}$ are the averages of the pixel values in Ω_n of J and J', respectively. $J(i)$ and $J'(i)$ are the pixel values of the i-th pixel in J and J'. Value ϵ_3 is a lower bound of the distance and is determined based on [92]. The first term of (5.4) becomes small when the n-th grid in the frame associated with z_n resembles those in other frames in R_t. Therefore, minimizing this term is analogous to choosing the mode of the n-th grid over the frames in R_t. Function δ is defined as follows:

$$\delta(t, z_n) = \begin{cases} 1 & \text{if } t = z_n \\ 0 & \text{otherwise} \end{cases}. \tag{5.6}$$

This term represents a preference for I_t. If the target frame is as likely to be the background as some other frames, this term with large ϵ_2 encourages us to use the target frame as the background for reducing the temporal discontinuity between successive frames.

Smooth term $G^{\mathrm{BE}}_{n,k}(z_n, z_k)$ is given by

$$G^{\mathrm{BE}}_{n,k}(z_n, z_k) = \theta_1 d_{\Omega_n \cup \Omega_k}(z_n, z_k) + \theta_2 \delta(z_n, z_k), \tag{5.7}$$

where θ_1 and θ_2 are constants to determine the contribution of each term. The first term penalizes the discontinuity when the pixel values in the frames selected for the adjacent grids largely differ, and the second term penalizes the assignment of different frames to adjacent grids.

Figure 5.6 (c) shows an example of background estimation applied to the target frames in Fig. 5.6 (a). Black represents the regions where $z_n = t$, i.e., the pixels from the target frame are copied to these regions, and the other colors represent $z_n \neq t$, i.e., the pixels of the frame corresponding to z_n are copied.

5.3.2 Intentionally-captured human object extraction

To extract ICHOs, we again use the graph cuts algorithm. Although in background estimation, it is applied to grids consisting of 5 × 5 pixels, we apply it to the pixels in ICHO extraction to maintain the detailed shapes of ICHOs. Label $\bar{z}_i = 0$ represents that the i-th pixel belongs to an ICHO and $\bar{z}_i = 1$ otherwise.

Using the output of ICHO detection and a shape prior (Figs. 5.6 (b) and (i)), intention map $IM_i \in [0, 1]$ (Fig. 5.6 (d)) is generated with which we extract the ICHOs. Energy function E^{IE} to be minimized is defined as

$$E^{\mathrm{IE}} = \sum_i F_i^{\mathrm{IE}}(\bar{z}_i) + \sum_{(i,j)\in AP} G_{i,j}^{\mathrm{IE}}(\bar{z}_i, \bar{z}_j), \tag{5.8}$$

where AP is the set of all adjacent pixels.

To define the data and the smooth terms, we make the following four assumptions: (i) An ICHO is around a region with larger values of IM_i. (ii) An ICHO is in the region with $z_n \neq t$ because it is a moving object, and thus the background is selected from the representative frames but not from the target frame. (iii) The boundary of an ICHO gives significant discontinuities in pixel values. (iv) The pixel values are continuous for the regions except the boundary of an ICHO. Based on these assumptions, we define data term $F_i^{\mathrm{IE}}(\bar{z}_i)$ as

$$F_i^{\mathrm{IE}}(\bar{z}_i) = \begin{cases} \nu_1 \delta(z_n, t) + \nu_2(1 - IM_i) & \text{for } \bar{z}_i = 0 \\ \nu_1[1 - \delta(z_n, t)] + \nu_2 IM_i & \text{otherwise} \end{cases}, \tag{5.9}$$

where ν_1 and ν_2 determine the contributions of each term; and z_n is the frame index for Ω_n in which the i-th pixel is included. In this equation, the terms involving ν_1 and ν_2 are based on assumptions (i) and (ii).

Based on assumptions (iii) and (iv), smooth term $G_{i,j}^{\mathrm{IE}}(\bar{z}_i, \bar{z}_j)$ penalizes similar pixel values for adjacent pixels when the labels are different as

$$G_{i,j}^{\mathrm{IE}}(\bar{z}_i, \bar{z}_j) = \begin{cases} 0 & \text{for } \bar{z}_i = \bar{z}_j \\ \exp(-\frac{\Upsilon^2}{\varrho_1}) + \varrho_2 & \text{otherwise} \end{cases}, \tag{5.10}$$

where $\Upsilon = I_t(i) - I_t(j)$ and $I_t(i)$ is the pixel value of the i-th pixel in the target frame I_t; ϱ_1 and ϱ_2 are constants to determine the contribution of each term. The term involving ϱ_1 penalizes the similar pixel values for different labels, while the term involving ϱ_2 penalizes different labels assigned to adjacent pixels.

Figure 5.6 (e) shows an example output of ICHO extraction. The white region represents $\bar{z}_i = 0$, and the black region represents $\bar{z}_i = 1$. In this example, the ICHO is accurately extracted.

5.3.3 Intentionally-captured human object substitution

The background pixels determined by z_n are substituted with ICHOs based on $\bar{z}_i$. When $\bar{z}_i = 0$, which means that the i-th pixel is in an ICHO, the pixel of the target frame should be used so that the ICHO is presented in the privacy-protected video. Therefore, label $\tilde{z}_i$, which determines the frame whose pixel is used for the output frame, is obtained by

$$\tilde{z}_i = \begin{cases} t & \text{if } \bar{z}_i = 1 \\ z_n & \text{otherwise} \end{cases}, \tag{5.11}$$

where z_n is the frame index for Ω_n in which the i-th pixel is included. An example of $\tilde{z}_i$ is shown in Fig. 5.6 (f).

The output frame can be obtained by replacing the pixels of I_t with the corresponding pixel of the frame in R_t according to $\tilde{z}_i$. However, this naive replacement introduces discontinuity to the boundaries of different labels caused by, e.g., an illumination change (Fig. 5.6 (g)). Therefore, we adopt Poisson blending [100], which replaces (A) a region in an image with (B) a region in another image without introducing discontinuity on the boundary between the images. In Poisson blending, the pixel values in the replaced region are determined such that the gradients on the boundary of (A) and those in (B) are preserved. Due to Poisson blending, the pixel values on the boundary of (A) propagate in the replaced region, and thus the discontinuity can be alleviated (Fig. 5.6 (h)).

5.4 Experimental results

To quantitatively evaluate our system, we adopted the following two measures:

Removal rate (*RR*): Our system removes non-ICHOs. *RR* measures how many pixels are removed that belong to non-ICHOs. We deem a pixel is removed if the pixel in the output frame comes from one of the representative frames but not from the target frame. In other words, the i-th pixel is judged to have been removed if $\tilde{z}_i \neq t$.

Preservation rate (*PR*): *PR* measures how many pixels are preserved that belong to ICHOs. The i-th pixel is judged to have been preserved when the pixel in the output frame comes from the target frame, i.e., $\tilde{z}_i = t$.

Let Ω_{NH} and Ω_{IH} denote the sets of pixels belonging to ICHOs and non-ICHOs (Fig. 5.7 (a)) and Ω_O be the set of pixels that satisfy $\tilde{z}_i = t$ (black region in Fig. 5.7 (b)). *RR* and *PR* are given by

$$\mathit{RR} = \frac{|\Omega_{NH} \cap \bar{\Omega}_O|}{|\Omega_{NH}|} \tag{5.12}$$

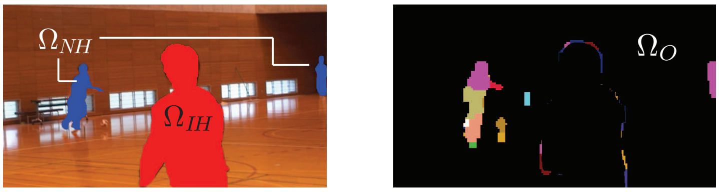
$$\mathit{PR} = \frac{|\Omega_{IH} \cap \Omega_O|}{|\Omega_{IH}|}, \tag{5.13}$$

where $\bar{\Omega}_O$ is the complementary set of Ω_O.

We applied our system to three videos (EV1, EV2, and EV3) excerpted from the videos in VD1. The size of the frames is 854 × 480 pixels and the frame rate is 29.97 frames per second: EV1 captures a scene with an almost stationary ICHO and a moving non-ICHO. EV2 is a scene with moving ICHOs and stationary non-ICHOs. EV3 captures a moving ICHO and moving non-ICHOs. The average duration of the videos are 12 seconds. To demonstrate the potential applicability of our system, we used ICHOs that were manually specified by the camera persons who captured the videos instead of the outputs of ICHO detection.

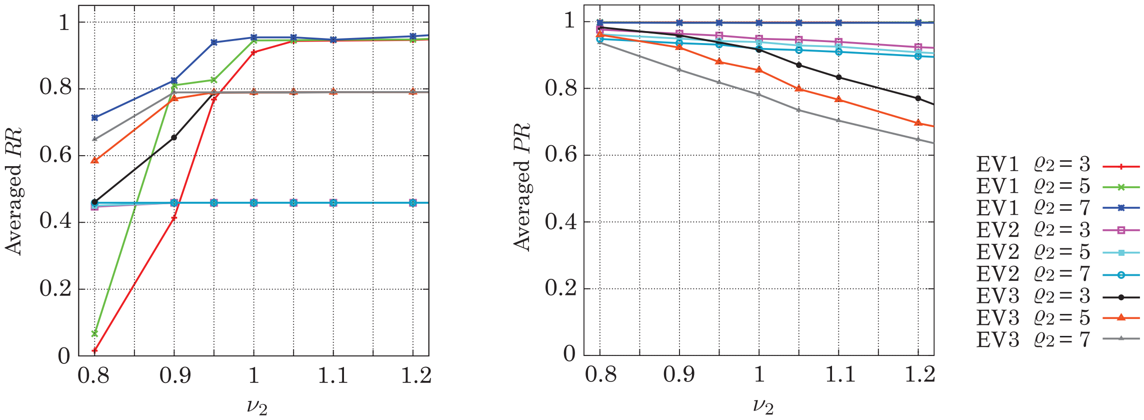
Although our system has many parameters for background estimation and ICHO extraction, our experimental results indicated that their influence was small except ν_2 and ϱ_2. Therefore, we show the results when the values of the parameters for background estimation were $\epsilon_1 = 1$, $\epsilon_2 = 0.15$, $\theta_1 = 1$, and $\theta_2 = 0.025$; those for ICHO extraction were $\nu_1 = 5$ and $\varrho_1 = 100$. For various parameter values of ν_2 and ϱ_2, the average values of *RR* and *PR* were calculated for each video.

The results are shown in Figs. 5.8 (a) and (b). The horizontal axes are ν_2. From Fig. 5.8 (a), our system successfully removes non-ICHOs for EV1 when ν_2 is large, but it



(a) (b)

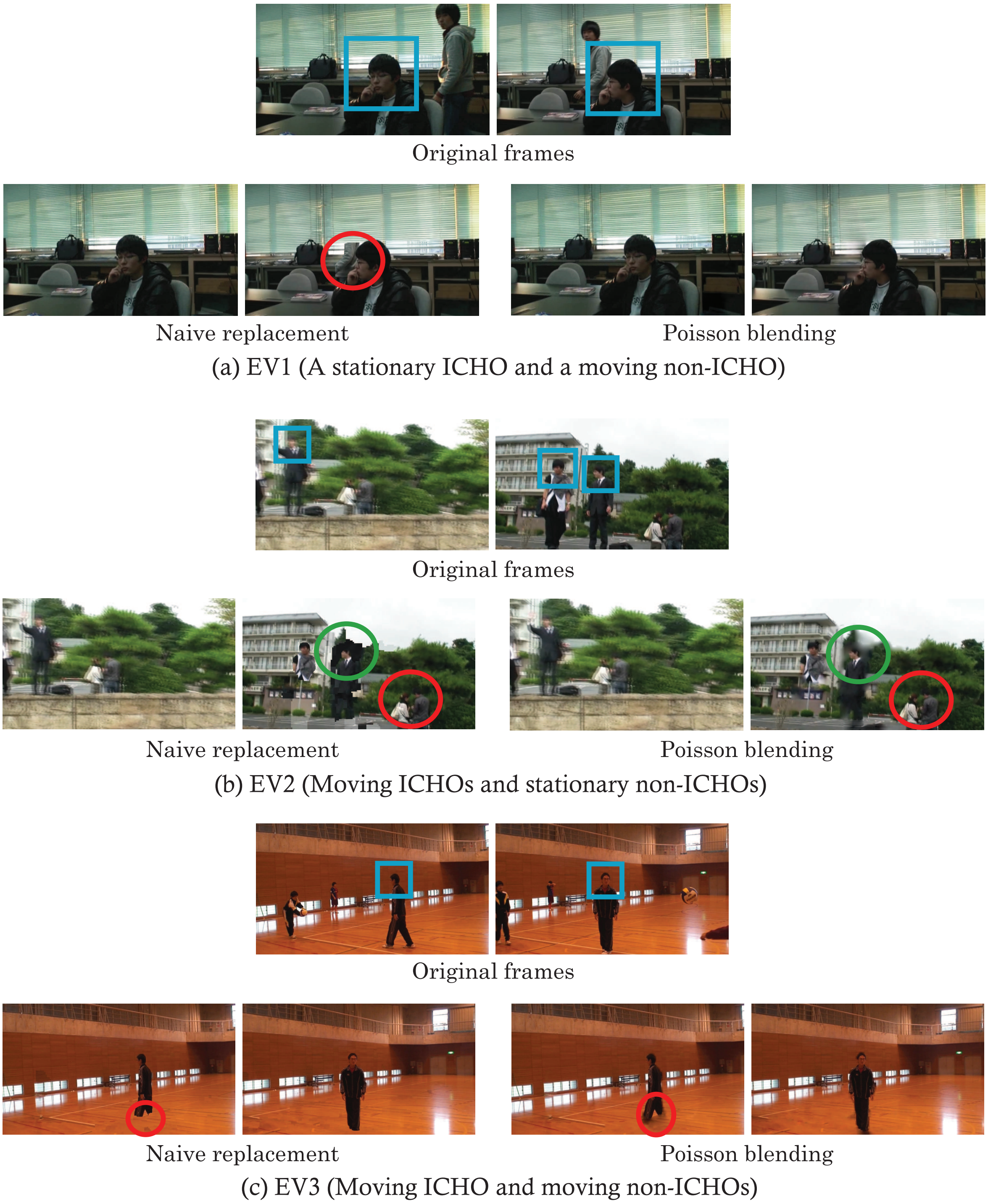
Figure 5.7: Definitions of (a) Ω_{IH}, Ω_{NH}, and (b) Ω_O.



(a) (b)

Figure 5.8: Averaged removal rate and (b) averaged preservation rate.

fails for small values of ν_2. The value of RR remains unchanged for EV2, but improves as ν_2 increases until 0.9 for EV3. This result indicates that two factors determine RR. The first is ν_2, which controls the contribution of intention map IM_i. For a small value of ν_2, the intention map is mostly discarded and the graph cuts algorithm judges that target frame I_t is most likely to be the background for most pixels as EV1 and EV3. This inclination is enhanced by small ϱ_2 because it relatively increases the effect of ν_2. The second factor is the failure of the background estimation caused by stationary non-ICHOs, which results in a low RR for EV2. This is irrelevant to ν_2 and ϱ_2. The second factor is critical, and we need to leverage other techniques for background estimation. For example, Chen et al. [91] adopted an image inpainting technique for background estimation. The image inpainting originally recovers corrupted regions in images specified by users. Chen et al. proposed to automatically find regions where the background estimation failed, and recover the regions as corrupted regions using



(a) EV1 (A stationary ICHO and a moving non-ICHO)

(b) EV2 (Moving ICHOs and stationary non-ICHOs)

(c) EV3 (Moving ICHO and moving non-ICHOs)

Figure 5.9: Example of output frames. For each video, whether intentionally-captured human objects (ICHOs) and human objects except ICHOs (non-ICHOs) are moving or stationary is indicated.

the image inpainting technique.

From Fig. 5.8 (b), most of the ICHOs in EV1 are successfully preserved regardless of ν_2 and ϱ_2. For EV2 and EV3, *PR* slightly decreases due to the increment of the parameter value, because larger ν_2 tightly constrains the shapes of the extracted regions based on the intention map and leads to failure to extract the complete shapes of the ICHOs. Larger ϱ_2 prefers smoother boundaries between ICHOs and other regions, which results in failure to extract the protruded parts of ICHOs, e.g., legs. Adopting a more appropriate shape prior of the human objects can partly overcome these problems.

Figure 5.9 shows example frames of (a) EV1, (b) EV2, and (c) EV3 when $\nu_2 = 1$ and $\varrho_2 = 5$. In the original frames, the ICHOs are surrounded by blue squares. In EV1, the pixels of the non-ICHO are disclosed for naive replacement as indicated by the red circle. This disclosure is alleviated by Poisson blending because it changes the pixel values in the non-ICHO so that they can resemble to the surrounding region. In EV2, the non-ICHOs are disclosed for naive replacement and Poisson blending, as indicated by red circles. In addition, significant visual artifact is introduced, as indicated in the green circles. In EV3, our system fails to extract the complete shape of the ICHO, as indicated by red circles.

The disclosure of the non-ICHO in EV1 and the failure of ICHO extraction in EV3 are caused by the shape prior that does not model a human object's individual shape very well. The problem in EV2 stems from the failure of background estimation. Our method for background estimation fails to estimate the background pixels around the non-ICHOs because they are stationary and the background pixels do not appear in the representative frames. Visual artifact is also caused by background estimation, which incorrectly estimates the background pixels as indicated by green circles in Fig. 5.9. Incorrect estimation of the background pixels is caused by the failure to estimate the homography matrices. Adopting an image inpainting technique can solve this problem as mentioned above.

5.5 Concluding remarks

In this chapter, for privacy protection against accidental privacy infringement, we presented a system that automatically generates privacy-protected videos. Our system superbly estimates background pixels and substitutes them with intentionally-captured human objects (ICHOs). Therefore, detecting human objects except ICHOs (non-ICHOs), whose detection is usually harder than ICHOs, is not necessary for our system. In addition, our system protects the privacy of persons corresponding to non-ICHOs while preserving the camera person's capture intention by presenting ICHOs.

Although our experimental results are encouraging because the ICHOs are correctly preserved in most cases, there are limitations regarding background estimation and the shape prior of the human objects used for ICHO extraction. Background estimation fails to estimate the background pixels if the non-ICHOs are stationary. In this case, we need to leverage another technique such as image inpainting [91]. We also need to improve the shape prior so that it can more appropriately represent the shapes of human objects.

Chapter 6

Conclusion

In this dissertation, we discussed the problem of copyright and privacy infringement, which reflects the deep penetration of mobile video cameras and video sharing services. Among various types of copyright and privacy infringement, we focused on in-theater movie piracy and accidental privacy infringement, and described multimedia signal processing-based approaches as countermeasures.

For in-theater movie piracy where a pirate captures a movie shown in a theater with a mobile video camera, we presented pirate position estimation to help identify pirates. We embed watermarks into movie soundtracks, and a maximum likelihood-based position estimator finds the pirate position based on them. For accidental privacy infringement where a video taken with a mobile video camera by a camera person infringes on the privacy of accidentally-framed-in persons, we described the generation of privacy-protected videos in which only the accidentally-framed-in persons are obscured using intentionally-captured human object (ICHO) detection and background estimation. The contributions of this dissertation are summarized as follows:

- Pirate position estimation for in-theater movie piracy can estimate the pirate position in environments that have at least three loudspeakers. The average estimation error is 0.44 m, which almost corresponds to the seat intervals in a theater. In addition, our subjective evaluation indicates that the watermarks hardly degrade the acoustic quality of movie soundtracks. Note that this is the world-first application of a digital audio watermarking technique for position estimation in large spaces, indicating that digital watermarking techniques can cover a wide range of applications.

- For accidental privacy infringement, we take a unique approach where the human objects to be obscured are adaptively determined by ICHO detection based on camera motion caused by the camera person and the human object motion. Compared with a conventional approach where human objects are detected and all are obscured, our approach maintains a camera person's capture intention, which is essential for video taken with a mobile video camera. ICHO detection can find 57% of ICHOs in videos when the false positives per frame is 0.5, and background estimation successfully obscured human objects except the ICHOs (non-ICHOs) when they are moving. This work might be a milestone toward privacy protection against accidental privacy infringement.

For the problems of in-theater movie piracy and accidental privacy infringement, established countermeasures have not been realized so far. Against these problems, we introduced novel approaches to protect copyright and privacy, as mentioned above. In addition, we achieve these results without modifying existing mobile video cameras or such environments as theaters due to multimedia signal processing techniques. Therefore, the copyright and privacy protection presented in this dissertation can be easily deployed without enormous initial costs.

Concerning the future directions of copyright and privacy protection, in this dissertation, we handled two specific problems included in copyright and privacy infringement. However, addressing other types is also important. For copyright protection, although we only focused on in-theater movie piracy, capturing live performances, for example, also infringes on copyright. The difficulty of protecting live performances is that such techniques as encryption, digital watermarking, and fingerprinting cannot be used. For this problem, sonic watermarking [101] is useful since it can embed a watermark into sonic waves. Another interesting approach is to establish a fingerprinting technique that is even applicable to captured live performances. For privacy protection, the problem where the camera person intentionally captures persons without permission should be addressed. This problem is challenging because we cannot make any assumption as we did in Chapters 4 and 5. Therefore, we may need a technique that uses special devices to notify the presence of persons, as in [23].

Finally, the core ideas in this dissertation, the position estimator based on digital audio watermarking and ICHO detection, are potentially applicable to other applications. For example, the position estimator can be used for such location-based services as location-based advertising (e.g. [102]) and indoor navigation (e.g. [103]). Also, ICHO detection is applicable to video summarization (e.g. [84]), video adaptation (e.g. [86]), and so forth. We believe that this research work will contribute to the development of novel technologies for these applications as well as copyright and privacy protection.

Bibliography

[1] Monthly Consumer Confidence Survey covering all of Japan (April 2010, in Japanese). Cabinet Office of Japan. [Online]. Available: http://www.esri.cao.go.jp/jp/stat/shouhi/2010/1003honbun.pdf

[2] Mobile Cellular Subscriptions. International Telecommunication Union. [Online]. Available: http://www.itu.int/ITU-D/ict/statistics/material/excel/2010/MobileCellularSubscriptions00-10.xls

[3] Gartner Says Nearly 50 Percent of Worldwide Mobile Phones Will Have a Camera in 2006 and 81 percent by 2010. Gartner Inc. [Online]. Available: http://www.gartner.com/it/page.jsp?id=498310

[4] Understanding Copyright and Related Rights. World Intellectual Property Organization. [Online]. Available: http://www.wipo.int/export/sites/www/freepublications/en/intproperty/909/wipo/_pub/_909.pdf

[5] J. Kang, "Information privacy in cyberspace transactions," *Stanford Law Review*, Vol. 50, pp. 1193–1294, April 1998.

[6] J. Haitsma and T. Kalker, "A watermarking scheme for digital cinema," in *Proc. 2001 IEEE International Conference on Image Processing*, Vol. 2, pp. 487–489, October 2001.

[7] P. Nguyen, R. Balter, N. Montfort, and S. Baudry, "Registration methods for non blind watermark detection in digital cinema applications," in *Proc. SPIE Security and Watermarking of Multimedia Contents V*, Vol. 5020, pp. 553–562, June 2003.

[8] I. Kitahara, K. Kogure, and N. Hagita, “Stealth vision for protecting privacy,” in *Proc. 17th Internatinal Conference on Pattern Recognition*, Vol. 4, pp. 404–407, August 2004.

[9] J. Chaudhari, S. S. Cheung, and M. V. Venkatesh, “Privacy protection for life-log video,” in *Proc. 2007 IEEE Workshop on Signal Processing Applications for Public Security and Forensics*, 5 pages, April 2007.

[10] T. Yamada, S. Gohshi, and I. Echizen, “IR hiding: a method to prevent video re-shooting by exploiting differences between human perceptions and recording device characteristics,” in *Proc. 9th International Workshop on Digital Watermarking*, pp. 280–292, October 2010.

[11] Content Scrambling System. DVD Copy Control Association. [Online]. Available: http://www.dvdcca.org/

[12] Content Protection for Recordable Media. 4C Entity. [Online]. Available: http://www.4Centity.com/

[13] E. T. Lin, A. M. Eskicioglu, R. L. Lagendijk, and E. J. Delp, “Advances in digital video content protection,” *Proceedings of the IEEE*, Vol. 93, No. 1, pp. 171–183, January 2005.

[14] Announcing the Advanced Encryption Standard (AES). National Institute of Standards and Technology. [Online]. Available: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[15] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, “Secure spread spectrum watermarking for multimedia,” *IEEE Trans. Image Processing*, Vol. 6, No. 12, pp. 1673–1687, December 1997.

[16] A. Ramalingam and S. Krishnan, “Gaussian mixture modeling of short-time Fourier transform features for audio fingerprinting,” *IEEE Trans. Information Forensics and Security*, Vol. 1, No. 4, pp. 457–463, December 2006.

[17] A. Joly, O. Buisson, and C. Frélicot, “Content-based copy retrieval using distortion-based probabilistic similarity search,” *IEEE Trans. Multimedia*, Vol. 9, No. 2, pp. 293–306, February 2007.

[18] S. Wei, Y. Zhao, C. Zhu, C. Xu, and Z. Zhu, "Frame fusion for video copy detection," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 21, No. 1, pp. 15–28, January 2011.

[19] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Processing*, Vol. 66, No. 3, pp. 283–301, May 1998.

[20] B. Chupeau, A. Massoudi, and F. Lefèbvre, "In-theater piracy: finding where the pirate was," in *Proc. SPIE Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, Vol. 6819, 10 pages, January 2008.

[21] K. Muneishi and M. Iwakiri, "Unauthorized camcorder position estimation based on geometrical distortion analysis in illegally redistributed image," in *Proc. Symposium on Cryptography and Information Security*, 3F2-3, 6 pages, January 2010 (in Japanese).

[22] M.-J. Lee, L.-S. Kim, and H.-K. Lee, "Digital cinema watermarking for estimating the position of pirate," *IEEE Trans. Multimedia*, Vol. 12, No. 7, pp. 605–621, November 2010.

[23] J. A. Halderman, B. Waters, and E. W. Felten, "Privacy management for portable recording devices," in *Proc. 2004 ACM Workshop on Privacy in the Electronic Society*, pp. 16–24, October 2004.

[24] J. Brassil, "Technical challenges in location-aware video surveillance privacy," in *Protecting Privacy in Video Surveillance*, pp. 91–113, Springer Verlag, 2009.

[25] S. Park and M. M. Trivedi, "A track-based human movement analysis and privacy protection system adaptive to environment contexts," in *Proc. IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance*, pp. 171–176, September 2005.

[26] A. Chattopadhyay and T. E. Bault, "Privacycam: a privacy preserving camera using uclinux on blackfin dsp," in *Proc. 2007 IEEE Conference on Computer Vision and Pattern Recognition*, 8 pages, June 2007.

[27] G. Li, Y. Ito, X. Yu, N. Nitta, and N. Babaguchi, "Recoverable privacy protection for video content distribution," *EURASIP Journal on Information Security*, Vol. 2009, 11 pages, 2009.

[28] A. Frome, G. Cheung, A. Abdulkader, M. Zennaro, B. Wu, A. Bissacco, H. Adam, H. Neven, and L. Vincent, “Large-scale privacy protection in google street view,” in *Proc. Twelfth IEEE International Conference on Computer Vision*, pp. 2373–2380, September 2009.

[29] A. Flores and S. Belongie, “Removing pedestrians from google street view images,” in *Proc. IEEE International Workshop on Mobile Vision*, pp. 53–58, June 2010.

[30] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, “Pfinder: real-time tracking of the human body,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 780–785, July 1997.

[31] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real-time tracking,” in *Proc. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 246–252, June 1999.

[32] D.-S. Lee, “Effective Gaussian mixture learning for video background subtraction,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 27, No. 5, pp. 827–832, May 2005.

[33] A. M. Elgammal, D. Harwood, and L. S. Davis, “Non-parametric model for background subtraction,” in *Proc. 6th European Conference on Computer Vision*, pp. 751–767, July 2000.

[34] M. Enzweiler and D. M. Gavrilla, “Monocular pedestrian detection: survey and experiments,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 31, No. 12, pp. 2179–2195, December 2009.

[35] D. M. Gavrila, “The visual analysis of human movement: a survey,” *Computer Vision and Image Understanding*, Vol. 73, No. 1, pp. 82–98, January 1999.

[36] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proc. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 886–893, June 2005.

[37] P. Viola and M. J. Jones, “Robust real-time face detection,” *International Journal of Computer Vision*, Vol. 57, No. 2, pp. 137–154, May 2004.

[38] J. Wickramasuriya, M. Datt, S. Mehrotra, and N. Venkatasubramanian, “Privacy protecting data collection in media spaces,” in *Proc. ACM International Conference on Multimedia 2004*, pp. 48–53, October 2004.

[39] W. Zhang, S. S. Cheung, and M. Chen, “Hiding privacy information in video surveillance system,” in *2006 IEEE International Conference on Image Processing*, pp. 68–71, September 2006.

[40] X. Yu, K. Chinomi, T. Koshimizu, N. Nitta, Y. Ito, and N. Babaguchi, “Privacy protecting visual processing for secure video wurveillance,” in *Proc. 15th IEEE International Conference on Image Processing*, pp. 1672–1675, October 2008.

[41] S. Tansuriyavong and S. Hanaki, “Privacy protection by concealing persons in circumstantioal video image,” in *Proc. 2001 Workshop on Perceptive User Interfaces*, 4 pages, 2001.

[42] P. N. Belhumeur, J. P. Haspanha, and D. J. Kriegman, “Eigenfaces vs. fisherfaces: recognition using class specific linear projection,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 711–720, July 1997.

[43] J. Yim, C. Park, J. Joo, and S. Jeong, “Extended kalman filter for wireless LAN based indoor positioning,” *Decision Support Systems*, Vol. 45, No. 4, pp. 960–971, November 2008.

[44] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, “LANDMARC: indoor location sensing using active RFID,” *Wireless Networks*, Vol. 10, No. 6, pp. 701–710, November 2004.

[45] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, “The Cricket location-support system,” in *Proc. Sixth Annual International Conference on Mobile Computing and Networking*, pp. 32–43, August 2000.

[46] 2005 US Piracy Fact Sheet. Motion Picture Association of America. [Online]. Available: http://www.mpaa.org/USPiracyFactSheet.pdf

[47] Anti-Piracy Fact Sheet Asia-Pacific Region. Motion Picture Association. [Online]. Available: http://www.mpaa.org/AsiaPacificPiracyFactSheet.pdf

[48] Y. Nakashima, R. Kaneto, and N. Babaguchi, “Indoor positioning system using digital audio watermarking,” *IEICE Transactions on Information and Systems*, Vol. E94-D, No. 11, pp. 2201–2211, November 2011.

[49] Y. Nakashima, R. Tachibana, and N. Babaguchi, “Watermarked movie soundtrack finds the position of the camcorder in a theater,” *IEEE Transactions on Multimedia*, Vol. 11, No. 3, pp. 443–454, April 2009.

[50] R. Kaneto, Y. Nakashima, and N. Babaguchi, “Real-time user position estimation in indoor environments using digital watermarking for audio signals,” in *Proc. 2010 International Conference on Pattern Recognition*, pp. 97–100, August 2010.

[51] Y. Nakashima, R. Tachibana, and N. Babaguchi, “Maximum-likelihood estimation of recording position based on audio watermarking,” in *Proc. Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 255–258, November 2007.

[52] Y. Nakashima, R. Tachibana, M. Nishimura, and N. Babaguchi, “Determining recording location based on synchronization positions of audio watermarking,” in *Proc. 2007 International Conference on Acoustics, Speech, and Signal Processing*, pp. II–253–II–256, April 2007.

[53] Y. Nakashima, R. Tachibana, M. Nishimura, and N. Babaguchi, “Estimation of recording location using audio watermarking,” in *Proc. ACM Multimedia and Security Workshop 2006*, pp. 108–113, September 2006.

[54] R. Kaneto, Y. Nakashima, and N. Babaguchi, “Recording position estimation in indoor environment using digital watermarking for audio signal,” in *Proc. 9th Multimedia Information Hiding*, DS-3-1, pp. S–15–S–16, March 2010 (in Japanese).

[55] R. Kaneto, Y. Nakashima, and N. Babaguchi, “Position estimation using detect strength of digital watermarking for audio signal,” in *Proc. 6th Multimedia Information Hiding*, DS-3-10, pp. S–37–S–38, March 2009 (in Japanese).

[56] Y. Nakashima, R. Kaneto, R. Tachibana, and N. Babaguchi, “Maximum-likelihood estimation of recording position based on synchronization position of

audio watermarking," in *Proc. Second Multimedia Information Hiding*, MIH02-09, pp. 45–50, November 2007 (in Japanese).

[57] Y. Nakashima, R. Tachibana, M. Nishimura, and N. Babaguchi, "Recording device localization using multiple audio watermark," in *Proc. ASJ Autumn Meeting 2006*, 2-1-9, pp. 458–459, September 2006 (in Japanese).

[58] R. Tachibana, S. Shimizu, S. Kobayashi, and T. Nakamura, "An audio watermarking method using a two-dimensional psuedo-random array," *Signal Processing*, Vol. 82, pp. 1455–1469, 2002.

[59] M. D. Swanson, B. Zhu, A. H. Tewfik, and L. Boney, "Robust audio watermarking using perceptual masking," *Signal Processing*, Vol. 66, pp. 337–335, 1998.

[60] D. Kirovski and H. S. Malvar, "Spread-spectrum watermarking of audio signals," *IEEE Trans. Signal Processing*, Vol. 51, No. 4, pp. 1020–1033, April 2003.

[61] *Information technology–Coding of moving pictures and associated audio for digital storage media up to about 1.5Mbits/s—part 3: Audio*, ISO/IEC Std. 11 172-1:1993, 1993.

[62] *Method for the subjective assessment of intermediate quality levels of coding systems*, ITU Std. BS. 1534, 2003.

[63] Y. Suzuki, F. Asano, H.-Y. Kim, and T. Sone, "An optimum computer-generated pulse signal suitable for the measurement of very long impulse responses," *The Journal of the Acoustical Society of America*, Vol. 97, No. 2, pp. 1119–1123, February 1995.

[64] T. Mei, X.-S. Hua, H.-Q. Zhou, and S. Li, "Modeling and mining of users' capture intention for home video," *IEEE Trans. Multimedia*, Vol. 9, No. 1, pp. 66–77, January 2007.

[65] D. Chen, Y. Chang, R. Yan, and J. Yang, "Tools for protecting the privacy of specific individuals in video," *EURASIP Journal on Advances in Signal Processing*, Vol. 2007, 9 pages, January 2007.

[66] L. Itti, C. Koch, and E. Niebur, "A model of saliency based visual attention for rapid scene analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1254–1259, November 1998.

[67] Y.-F. Ma and H.-J. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *Proc. ACM International Conference on Multimedia 2003*, pp. 374–381, November 2003.

[68] L. Itti and P. Baldi, "A principled approach to detecting surprising events in videos," in *Proc. 2005 IEEE Computer Society Conference on Computer and Vision Pattern Recognition*, pp. 631–637, June 2005.

[69] L. Itti and P. Baldi, "Bayesian surprise attracts human attention," *Vision Research*, Vol. 49, No. 10, pp. 1295–1306, June 2009.

[70] Y. Hu, D. Rajan, and L.-T. Chia, "Attention-from-motion: A factorization approach for detecting attention objects in motion," *Computer Vision and Image Understanding*, Vol. 113, No. 3, pp. 319–331, March 2009.

[71] Y. Nakashima, N. Babaguchi, and J. Fan, "Intended human object detection for automatically protecting privacy in mobile video surveillance," *Multimedia Systems*, 17 pages, DOI: 10.1007/s00530-011-0244-y, 2011 (Online published, printed version in press).

[72] Y. Nakashima and N. Babaguchi, "Extracting intentionally captured regions using point trajectories," in *Proc. ACM International Conference on Multimedia 2011*, pp. 1417–1420, November 2011.

[73] H. Uegaki, Y. Nakashima, and N. Babaguchi, "Discriminating intended human objects in consumer videos," in *Proc. 2010 International Conference on Pattern Recognition*, pp. 4380–4383, August 2010.

[74] Y. Nakashima, N. Babaguchi, and J. Fan, "Detecting intended human objects in human-captured videos," in *Proc. 2010 IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 8 pages, June 2010.

[75] H. Uegaki, Y. Nakashima, and N. Babaguchi, "Inference of intentionally captured regions based on camera motion and visual features," in *Proc. Meeting on Image Recognition and Understanding 2011*, IS4-54, pp. 1645–1652, July 2011 (in Japanese).

[76] Y. Nakashima, H. Uegaki, and N. Babaguchi, "Detecting human subjects the cameraman intended to capture in video," in *Proc. 2010 IEICE General Conference*, D-12-41, p. 152, March 2010 (in Japanese).

[77] H. Uegaki, Y. Nakashima, and N. Babaguchi, "Inferring camcorder user's intended subject of persons based on visual feature," in *Proc. Forum on Information Technology 2009*, K-046, pp. 639–642, September 2009 (in Japanese).

[78] L. Elazary and L. Itti, "Interesting objects are visually salient," *Journal of Vision*, Vol. 8, No. 3, 15 pages, March 2008.

[79] F. Dufaux and J. Konrad, "Efficient, robust, and fast global motion estimation for video coding," *IEEE Trans. Image Processing*, Vol. 9, No. 3, pp. 497–501, Mar. 2000.

[80] H. Murase and V. V. Vinod, "Fast visual search using focused color matching — active search," *Systems and Computers in Japan*, Vol. 31, No. 9, pp. 81–88, July 2000.

[81] J. C. Platt, "Probabilities for SV machines," in *Advances in Large Margin Classifiers.* MIT Press, March 1999.

[82] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum, "Full-frame video stabilization with motion inpainting," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 28, No. 7, pp. 1150–1163, July 2006.

[83] Y.-F. Ma, X.-S. Hua, L. Lu, and H. J. Zhang, "A generic framework of user attention model and its application in video summarization," *IEEE Trans. Multimedia*, Vol. 7, No. 5, pp. 907–919, October 2005.

[84] T. Wang, T. Mei, X.-S. Hua, X.-L. Liu, and H.-Q. Zhou, "Video collage: a novel presentation of video sequence," in *Proc. 2007 IEEE International Conference on Multimedia and Expo*, pp. 1479–1482, July 2007.

[85] H.-J. Zhang, X.-S. Hua, and L. Lu, "AVE—automated home video editing," in *Proc. ACM International Conference on Multimedia 2003*, pp. 490–497, November 2003.

[86] X.-S. Hua, L. Lu, and H.-J. Zhang, "Optimization-based automated home video editing system," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 14, No. 5, pp. 572–583, May 2004.

[87] Y. Nakashima, N. Babaguchi, and J. Fan, "Automatic generation of privacy-protected videos using background estimation," in *Proc. 2011 IEEE International Conference on Multimedia and Expo*, 6 pages, July 2011.

[88] Y. Nakashima, N. Babaguchi, and J. Fan, "Automatically protecting privacy in consumer generated videos using intended human object detector," in *Proc. ACM International Conference on Multimedia 2010*, pp. 1135–1138, October 2010.

[89] T. Takehara, Y. Nakashima, N. Nitta, and N. Babaguchi, "Digital diorama: sensing-based real-world visualization," in *Proc. 13th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp. 663–672, June 2010.

[90] T. Takehara, Y. Nakashima, N. Nitta, and N. Babaguchi, "Digital diorama: real-time adaptive visualization of public spaces," in *Proc. First International Conference on Security Camera Network, Privacy Protection and Community Safety*, 2 pages, October 2009.

[91] X. Chen, Y. Shen, and Y. H. Yang, "Background estimation using graph cuts and inpainting," in *Proc. Graphics Interface*, pp. 97–103, May 2010.

[92] D.-W. Kim and K.-S. Hong, "Practical background estimation for mosaic blending with patch-based Markov random fields," *Pattern Recognition*, Vol. 41, No. 7, pp. 2145–2155, July 2008.

[93] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 11, pp. 1222–1239, November 2001.

[94] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 26, No. 2, pp. 147–159, February 2004.

[95] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 26, No. 9, pp. 1124–1137, September 2004.

[96] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," *Computer Vision and Image Understanding*, Vol. 110, No. 3, pp. 346–359, June 2008.

[97] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Proc. Fourth International Conference on Computer Vision Theory and Application*, pp. 331–340, February 2009.

[98] M. A. Fischlaer and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395, June 1981.

[99] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, Vol. 315, pp. 972–976, February 2007.

[100] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graphics (Proc. ACM SIGGRAPH 2003)*, Vol. 22, No. 3, pp. 313–318, July 2003.

[101] R. Tachibana, "Sonic watermarking," *EURASIP Journal on Applied Signal Processing*, Vol. 13, pp. 1955–1964, January 2004.

[102] L. Aalto, N. Göthelin, J. Korhonen, and T. Ojala, "Bluetooth and wap push based location-aware mobile advertising system," in *Proc. Second International Conference on Mobile Systems, Applications, and Services*, pp. 49–58, June 2004.

[103] A. Butz, J. Baus, A. Krüger, and M. Lohse, "A hybrid indoor navigation system," in *Proc. 2006 International Conference on Intelligent User Interfaces*, pp. 25–32, January 2006.

Publications

A. Journal Papers

1. Y. Nakashima, N. Babaguchi, and J. Fan, "Intended human object detection for automatically protecting privacy in mobile video surveillance," *Multimedia Systems*, 17 pages, DOI: 10.1007/s00530-011-0244-y, 2011 (Online published, printed version in press).
2. Y. Nakashima, R. Kaneto, and N. Babaguchi, "Indoor positioning system using digital audio watermarking," *IEICE Trans. on Information and Systems*, Vol. E94-D, No. 11, pp. 2201–2211, November 2011.
3. Y. Nakashima, R. Tachibana, and N. Babaguchi, "Watermarked movie soundtrack finds the position of the camcorder in a theater," *IEEE Trans. on Multimedia*, Vol. 11, No. 3, pp. 443–454, April 2009.

B. International Conference Papers (refereed)

1. Y. Nakashima and N. Babaguchi, "Extracting intentionally captured regions using point trajectories," *Proc. ACM International Conference on Multimedia 2011*, pp. 1417–1420, November 2011.
2. Y. Nakashima, N. Babaguchi, and J. Fan, "Automatic generation of privacy-protected videos using background estimation," *Proc. 2011 IEEE International Conference on Multimedia and Expo*, 6 pages, July 2011.
3. Y. Nakashima, N. Babaguchi, and J. Fan, "Automatically protecting privacy in consumer generated videos using intended human object detector," *Proc.*

ACM International Conference on Multimedia 2010, pp. 1135–1138, October 2010.

4. R. Kaneto, Y. Nakashima, and N. Babaguchi, "Real-time user position estimation in indoor environments using digital watermarking for audio signals," *Proc. 2010 International Conference on Pattern Recognition*, pp. 97–100, August 2010.

5. H. Uegaki, Y. Nakashima, and N. Babaguchi, "Discriminating intended human objects in consumer videos," *Proc. 2010 International Conference on Pattern Recognition*, pp. 4380–4383, August 2010.

6. Y. Nakashima, N. Babaguchi, and J. Fan, "Detecting intended human objects in human-captured videos," *Proc. 2010 Conference on Computer Vision and Pattern Recognition Workshop*, 8 pages, June 2010.

7. T. Takehara, Y. Nakashima, N. Nitta, and N. Babaguchi, "Digital diorama: sensing-based real-world visualization," *Proc. 13th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp. 663–672, June 2010.

8. T. Takehara, Y. Nakashima, N. Nitta, and N. Babaguchi, "Digital diorama: real-time adaptive visualization of public spaces," *Proc. First International Conference on Security Camera Network, Privacy Protection and Community Safety*, 2 pages, October 2009.

9. Y. Nakashima, R. Tachibana, and N. Babaguchi, "Maximum-likelihood estimation of recording position based on audio watermarking," *Proc. Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 255–258, November 2007.

10. Y. Nakashima, R. Tachibana, M. Nishimura, and N. Babaguchi, "Determining recording location based on synchronization positions of audio watermarking," *Proc. 2007 International Conference on Acoustics, Speech, and Signal Processing*, pp. II-253–II-256, April 2007.

11. Y. Nakashima, R. Tachibana, M. Nishimura, and N. Babaguchi, "Estimation of recording location using audio watermarking," *Proc. ACM Multimedia*

and Security Workshop 2006, pp. 108–113, September 2006.

C. Domestic Conferences

1. H. Uegaki, Y. Nakashima, and N. Babaguchi, “Inference of intentionally captured regions based on camera motion and visual features,” *Proc. Meeting on Image Recognition and Understanding 2011*, IS4-54, pp. 1645–1652, July 2011 (in Japanese).
2. Y. Nakashima, H. Uegaki, and N. Babaguchi, “Detecting human subjects the cameraman intended to capture in video,” *Proc. 2010 IEICE General Conference*, D-12-41, p. 152, March 2010 (in Japanese).
3. R. Kaneto, Y. Nakashima, and N. Babaguchi, “Recording position estimation in indoor environment using digital watermarking for audio signal,” *Proc. 9th Multimedia Information Hiding*, DS-3-1, pp. S-15–S-16, March 2010 (in Japanese).
4. H. Uegaki, Y. Nakashima, and N. Babaguchi, “Inferring camcorder user’s intended subject of persons based on visual feature,” *Proc. Forum on Information Technology 2009*, K-046, pp. 639–642, September 2009 (in Japanese).
5. R. Kaneto, Y. Nakashima, and N. Babaguchi, “Position estimation using detect strength of digital watermarking for audio signal,” *Proc. 6th Multimedia Information Hiding*, DS-3-10, pp. S-37–S-38, March 2009.
6. Y. Nakashima, R. Kaneto, R. Tachibana, and N. Babaguchi, “Maximum-likelihood estimation of recording position based on synchronization position of audio watermarking,” *Proc. Second Multimedia Information Hiding*, MIH02-09, pp. 45–50, November 2007 (in Japanese).
7. Y. Nakashima, R. Tachibana, M. Nishimura, and N. Babaguchi, “Recording device localization using multiple audio watermark,” *Proc. ASJ Autumn Meeting 2006*, 2-1-9, pp. 458–459, September 2006 (in Japanese).