MULTIPLE KERNEL LEARNING FOR FIRST-PERSON ACTIVITY RECOGNITION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS INSTITUTE
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

FATİH ÖZKAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

JUNE 2017

MULTIPLE KERNEL LEARNING FOR FIRST-PERSON ACTIVITY RECOGNITION

Submitted by **FATIH ÖZKAN** in partial fulfillment of the requirements for the degree of **Master of Science in The Department of Information Systems , Middle East Technical University** by,

Prof. Dr. Deniz Zeyrek Bozşahin ______________
Director, **Graduate School Of Informatics**

Prof. Dr. Yasemin Yardımcı Çetin ______________
Head of Department, **Information Systems**

Assoc. Prof. Dr. Alptekin Temizel ______________
Supervisor, **Modeling and Simulation**

Asst. Prof. Dr. Elif Sürer ______________
Co-supervisor, **Modeling and Simulation**

Examining Committee Members

Assoc. Prof. Dr. Altan Koçyiğit ______________
Information Systems, METU

Assoc. Prof. Dr. Alptekin Temizel ______________
Modeling and Simulation, METU

Asst. Prof. Dr. Elif Sürer ______________
Modeling and Simulation, METU

Asst. Prof. Dr. Aykut Erdem ______________
Computer Engineering, Hacettepe University

Asst. Prof. Dr. Erhan Eren ______________
Information Systems, METU

Date: ______________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: FATİH ÖZKAN

Signature :

ABSTRACT

MULTIPLE KERNEL LEARNING FOR FIRST-PERSON ACTIVITY RECOGNITION

Özkan, Fatih
M.Sc., Department of Information Systems
Supervisor : Assoc. Prof. Dr. Alptekin Temizel
Co-Supervisor : Asst. Prof. Dr. Elif Sürer

June 2017, 74 pages

First-person vision applications have recently gained increasing popularity because of advances in wearable camera technologies. In the literature, existing descriptors have been adapted to the first-person videos or new descriptors have been proposed. These descriptors have been used in a single-kernel method which ignores the relative importance of each descriptor. On the other hand, first-person videos have different characteristics as compared to third-person videos which are captured by static cameras. Throughout the first-person video, vast changes occur in some attributes such as illumination or brightness. A significant amount of ego-motion is created because of the movements of the first-person camera wearer. Multiple features are used in order to capture the different changes in video characteristics. Therefore, appropriate feature and kernel selection are needed. In this thesis, local and global motion-related features are used. A data-driven approach is proposed in order to select and combine these features and kernels employed. Feature and kernel selection is performed through AdaBoost algorithm's well-known trials in a probabilistic manner. At training stage, a classifier which shows better performance than other classifiers is determined for each trial. After all trials, classifiers which compose the final classifier are determined. At testing stage, final classifier makes decision for activity labels based on a voting mechanism. Experiments show that the proposed methods outperform the traditional SVM single kernel-based methods in literature in terms of recognition accuracy.

Keywords: multiple kernel learning, kernel boosting, first-person, ego-centric videos, activity recognition

ÖZ

BİRİNCİ ŞAHIS AKTİVİTE TANIMA İÇİN ÇOKLU ÇEKİRDEK ÖĞRENMESİ

Özkan, Fatih
Yüksek Lisans, Bilişim Sistemleri Bölümü
Tez Yöneticisi : Assoc. Prof. Dr. Alptekin Temizel
Ortak Tez Yöneticisi : Asst. Prof. Dr. Elif Sürer

Haziran 2017, 74 sayfa

Birinci-şahıs görü uygulamaları giyilebilen kamera teknolojisindeki ilerlemeler sebebiyle yakın zamanda artan bir rağbet kazandı. Literatürde, birinci- şahıs video'ları için mevcut tanımlayıcılar uyarlanmıştır veya yeni tanımlayıcılar önerilmiştir. Bu tanımlayıcılar, her bir tanımlayıcının göreceli önemini ihmal eden tekli-çekirdek metodunda kullanılır. Öte yandan, birinci-şahıs video'ları sabit kameralarla çekilen üçüncü- şahıs video'larla kıyaslandığında farklı ayırıcı nitelikleri vardır. Birinci şahıs video boyunca, aydınlanma ve parlaklık gibi bazı özelliklerde geniş değişiklikler oluşur. Birinci şahıs kamera ile görüntü alan kişinin hareketleri sebebiyle önemli miktarda öz-hareket oluşur. Çoklu öznitelikler video özelliklerindeki farklı değişiklikleri yakalamak için kullanılması önerilmektedir. Bu sebeple, uygun öznitelik ve çekirdek seçimi gerekir. Bu tezde, lokal ve global harekete ilişkin öznitelikler kullanılır. Bu öznitelikleri ve çekirdekleri seçmek ve bir araya getirmek için veri-güdümlü yaklaşım önerilir. Öznitelik ve çekirdek seçimi olasılık temelli bir yol kullanılarak, AdaBoost algoritmasının bilinen denemeleriyle gerçekleştirilir. Eğitme aşamasında, diğer sınıflandırıcılardan daha iyi bir performans gösteren sınıflandırıcı her deneme için belirlenir. Bütün denemelerden sonra, nihai sınıflandırıcıyı meydana getiren sınıflandırıcılar belirlenir. Test aşamasında, nihai sınıflandırıcı aktivite etiketlerine oylama mekanizmasına dayalı olarak karar verir. Yürütülen deneyler, önerilen metodun literatürdeki geleneksel DVM (Destek Vektör Makineleri - SVM) tekil çekirdek temelli metotlara göre, tanıma doğruluğu bakımından daha üstün olduğunu gösterir.

Anahtar Kelimeler: çoklu çekirdek ögrenmesi, çekirdek takviyesi, birinci- şahıs, öz-hareket videoları, aktivite tanıma

to my loving parents and the memories of my grandparents ...

ACKNOWLEDGMENTS

I would gratefully thank my supervisor Assoc. Prof. Dr. Alptekin Temizel for his continuous support, encouragement, guidance and patience throughout my studies. His supervision and inspiring criticism have been precious for not only my thesis but also for my life. I am indebted to him because of his invaluable advice and motivation which can be found in this thesis.

I would also like to give my special thanks to Assist. Prof. Dr. Elif Sürer for her invaluable advice and feedback. Her guidance helped me throughout my research and writing of this thesis.

I also thank my managers in Innova Inc. and TUBITAK for their support in my studies and allowing me to attend the lectures. I would also like to thank my colleagues at TUBITAK for their help during my graduate studies.

I thank my dear friends for their patience and encouragement during my research. The quality of this thesis has been improved due to their scientific discussions and feedback.

I also give my special thanks to my brother Hüseyin Özkan for his patience and guidance throughout my thesis and life. His invaluable advice and support always keep me going when times are tough.

Finally, I want to give my deepest thanks to my lovely parents who have supported me endlessly throughout my life. They have showed patience and encouraged me during my studies.

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

AdaBoost	Adaptive Boosting
BOW	Bag of Words
DBN	Dynamic Bayesian Network
DC-Int	DogCentric Intersection
DBN	Deep Belief Network
CNN	Convolutional Neural Network
DogC	DogCentric
FPS	Frame per Second
HMM	Hidden Markov Model
HOF	Histogram of Optical Flow
H-Int	Histogram Intersection
JPL	Jet Propulsion Laboratory
JPL-Int	Jet Propulsion Laboratory Intersection
LBP	Local Binary Patterns
LOG-C	Log-Covariance
MKL	Multiple Kernel Learning
NN	Neural Networks
OAA	One-Against-All
OAO	One-Against-One
RFID	Radio-frequency identification
SIFT	Scale-Invariant Feature Transform
STIP	Spatio Temporal Interest Points
SVM	Support Vector Machine

CHAPTER 1

INTRODUCTION

Wearable cameras have become ubiquitous as a consequence of advances in camera and sensor technologies. Hence, a number of devices such as GoPro, Google Glass and Microsoft SenseCam have come into use and received widespread attention. These devices have been broadly employed in several fields such as sport activities and life-logging applications. Wearable cameras which are mostly mounted on head or worn like eyeglasses allow capturing videos from the same viewpoint of the person wearing the camera. Videos which are captured by these cameras are called as *egocentric* or *first-person* videos. First-person videos have brought not only new capabilities but also unique challenges into the computer vision domain. For example, a robot can support a security system by recognizing activities, detecting anomalies or summarizing daily events around itself. The robot can have the capability of automated analysis of first-person videos to perform this task. However, there are also challenges since different characteristics of egocentric videos and camera motion require new approaches. Activity recognition on first-person videos is one of the particular challenges with its new domain specific problems.

Activity recognition is one of the computer vision problems which aims to recognize activities. An activity recognition algorithm determines the types of actions occurred in the videos. If a person runs in a video, then the algorithm is expected to determine this activity accurately. Moreover, it discerns a specified action among other actions occurred in the video. For instance; a dog runs toward a ball and catches it in a video. There are mainly two actions: running and catching. Therefore, it is required to discriminate these actions accurately to recognize both of which separately. Activity recognition performs also this distinction between the actions task.

Activity recognition can be employed in a great number of real-life applications from sport to the health-care systems. Players can benefit from the activity recognition in order to improve their performances. It recognize the activities of the players in a game and can be used to analyze the overall actions of a team. It can also be used for social problems such as elder care. Daily lives of elderly people can be monitored and an assistance service can be provided for them by recognizing their daily activities. It can be employed for also disabled people and patients who have diseases such as senility or rickets. Early detection of unusual behaviours of these persons gains importance for early intervention.

Wearable cameras enable the activity recognition to become more effective in several applications by providing more related and rich information of the user. For example, an elder person walks in a home and performs several activities throughout a day such as watching television, sleeping and eating. If a fixed camera is used in a room, the angle of view becomes constant and user's specific actions cannot be exploited such as head or eye movement. On the other hand, wearable camera gives the information about the movement patterns of a person, interaction with objects and where to look etc. Therefore, activity recognition from first-person videos gains more importance in order to analyze the activities.

Activity recognition is considered as a supervised learning problem which consists of an activity representation model and an activity classification method. It aims to recognize the activities. Although human brain can perform the recognition process easily, it is a complex problem for machine learning classifiers due to different environmental conditions, complex attributes of the subjects, and resolution of the videos among other factors. These can affect the performance of machine learning classifiers. In order to overcome these factors, first of all, they need pre-processing operations such as scaling, noise cleaning or outliers removing operations. After pre-processing, based on the appropriate data, constructing the representation model comes as a second step. Extracting, clustering or combining features are some of the means of the modelling step. After this first phase, based on appropriate data, classifiers make decisions. Hence, we can divide activity recognition process of machine learning classifiers into two phases: Learning the models of data and choosing the classification method. Figure 1.1 shows the learning scheme. In the figure, based on the training dataset, a representation model is built. Then, in order to evaluate the representation model, error rate is measured and based on the error rate, the model parameters are updated. This updating process is called as parameter tuning. "Test data" in the figure is the unseen data on which the prediction is performed. A representation model and a classification method can handle the complex recognition problem by handling the factors aforementioned. Consequently, the performance of an activity recognition system is mainly dependent on the effectiveness of the representation model and the accuracy of the classification method.

In an activity recognition problem, features are the fundamental instruments to construct the most fitting representation model. Features of a video could be different types such as spatial and temporal or local and global. Global features describe an image as a whole whereas local features are computed on specific areas in an image. Spatial features give information about only an image or a frame in a video but temporal features show correlations between the images or frames in a video to capture the dynamic changes. Consequently, each type of feature represents different information of an image. For instance, spatial color information could show object distribution in an image at specific time, while temporal color information could show how the object moves in a specific time interval. Moreover, they could also represent characteristics such as color, motion, frequency value etc. Color, motion or any other feature already give different information whether they are same type.

Feature extraction, as part of the step toward the recognition of an activity, may affect the representation model to a great degree, based on quality, robustness, invariance to scale of the features. In addition, structures of first-person videos are different from third-person videos in terms of data distribution changes, ego-motion and angle of view. Therefore, each first-person video has different type, number, robustness and quality of features according to its viewpoint. Hence, each video type requires a

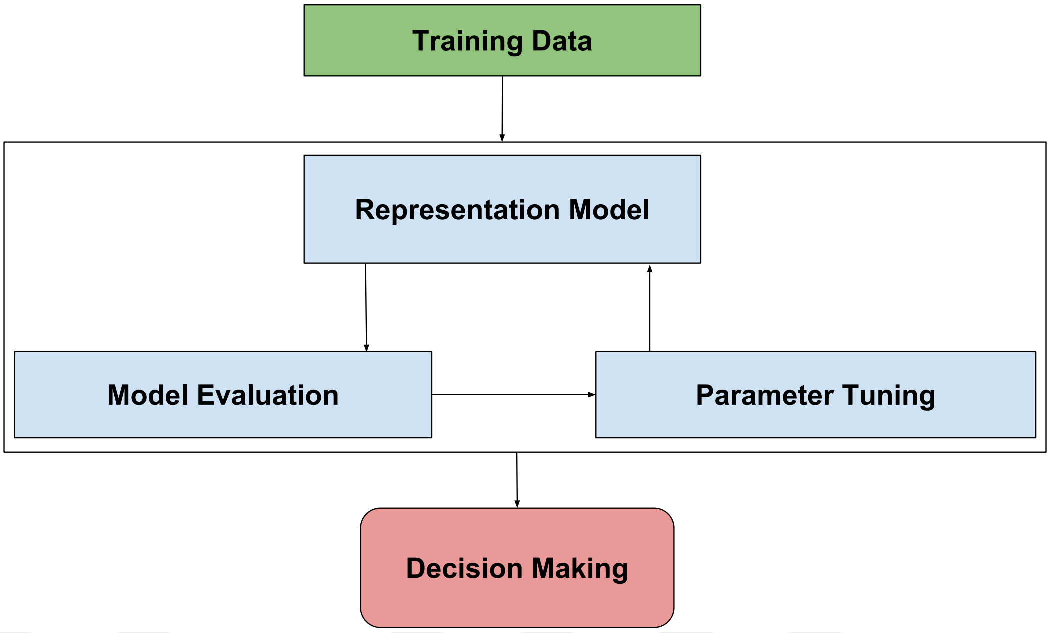


Figure 1.1: A learning process scheme

different representation model. Also, it gains additional importance to represent data in the videos in an effective way and to select an appropriate classification method.

Representation model is composed of extracted features, as mentioned before, which describe an activity uniquely. Learning the model is an essential process before the final stage of decision making. For this purpose, there are several model types such as statistical, hierarchical or histogram-based models. All model types for activity recognition aim to describe an activity and provide the similarity computation between the activities. A model is learned using the features.

In decision making, the purpose is the activity recognition of the unseen video. In order to achieve the decision making, the activity class in the query video is determined based on a dictionary of labelled activity samples in a supervised manner. For instance, if we consider human activities, we are given videos which include human activities and we are expected to determine the activity class of the related video. To ensure the decision making, several classification methods like Hidden Markov Models (HMM), Neural Networks or Support Vector Machine (SVM) are employed.

In all classification methods, a classifier takes a representation model as input, trains the model and make final decision based on the trained model. In this context, extraction and combining features for building the model gain importance again. There are several factors in order to determine the appropriate way to study features. Each feature extracted from the videos could give distinctive information about a different attribute, scene, activity etc. from the video. Furthermore, first-person video characteristics make features more important for the classification since several types of features are used to capture the changing dynamics of egocentric videos. For example, a video that is captured by a static camera contains specific types of activities of the same theme, since the intention is already known before capturing. Thus, third-person videos could generally be processed with only one type of feature. On the other hand, theme of the first-person video is probably changing throughout the capturing since the aim of observer is not specific before capturing. Therefore, attention, objects, persons and activities etc. change along the video. Single type of a feature falls short for capturing the discriminative information from the video. As a result, features could be used alone or together depending on the application, environment or other conditions. Optimal combination of the features could be determined for the best recognition performance. Also, SVM is the preferred approach that is employed in this thesis, with the combinations of different kernel and feature variations. Hence, SVM kernel function and a specific feature that shows good performance could be used together. Also, this gives the opportunity of assigning different weights on specific kernels or features for better recognition performance.

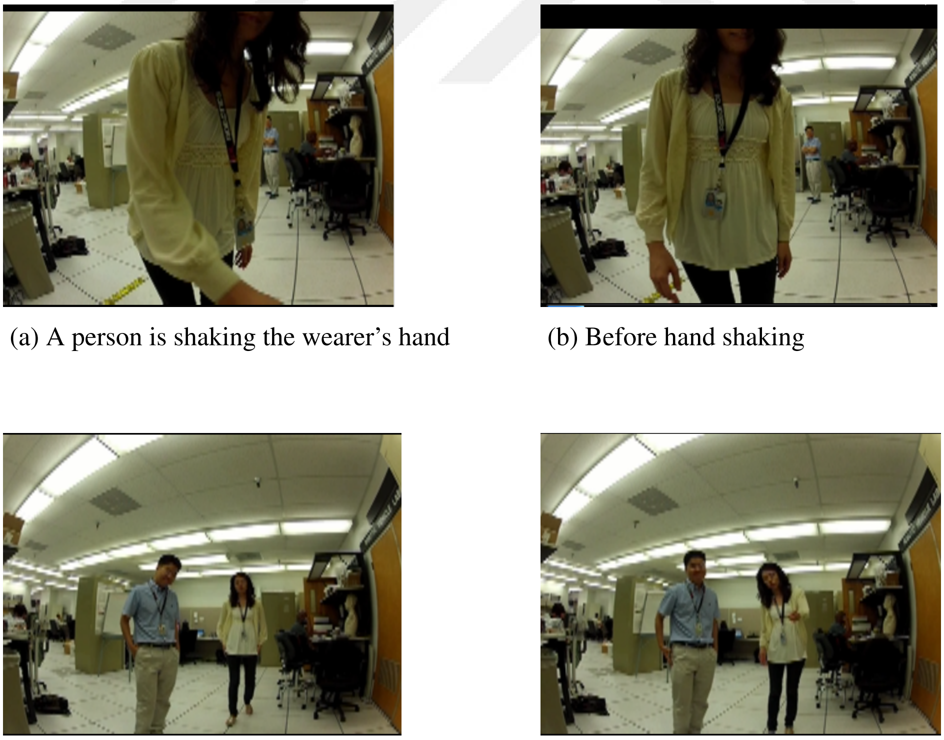
For building representation model and classification steps, deep learning based approaches which exploit many layers for non-linear information transformations can also be employed. They are built hierarchically and each layer processes the outputs of the previous layer. As opposed to traditional machine learning techniques which use and process hand-crafted features, a deep learning based approach processes the data in raw form and transform it through its stack of layers for classification. Thus, from pixels to motifs and objects, there occur several transformations for image or video processing. For instance; low level features are processed and mid-level features are generated in the first-layer. In the second layer, mid-level features are processed and high-level features are computed as an output. Finally, high-level features are used in training step. Thus, each layer transforms its input representation model into a higher-level representation model for the higher level layer. This procedure is called as end-to-end learning. Hand-crafted features are not needed in deep learning approaches for the sake of their end-to-end learning concept. There are two types of deep learning models: unsupervised and supervised. DBN is one of the unsupervised deep learning models, whereas CNN is the supervised model. For example, CNNs perform feature extraction and employ several convolutional operators to create high level features in a hierarchical approach. Thus, representation model learning can be performed by the CNN model.

Deep learning based approaches require considerable amount of data for training. However, the well-known datasets in first-person vision domain such as JPL-Interaction dataset, do not contain such enough data, so that they are not suitable for deep learning based approaches. They also require complex computations and long execution time. Therefore, hand-crafted descriptors and kernel-based approaches are used for first-person activity recognition in this study.

1.1 First-Person Video Characteristics

First-person videos are captured using a wearable camera. Wearable cameras can be head mounted or worn by a person. Therefore, these types of cameras have the same viewpoint with the wearer. Since the camera is carried out by a person, the camera itself is involved in the activities, interactions etc. Consequently, the behaviors of the wearer gain importance. Wearer's activities affect the video's dynamics, events in the videos which makes the wearer also an actor of the video. First-person videos are generally dynamic, as opposed to the third-person videos. Since the actor is involved in the events, the camera creates large amounts and different types of ego-motion such as moving up or down and turning with the activity of the user. Figure 1.2 shows examples of the first-person perspective from the JPL Interaction first-person video dataset [47]. There are two persons who look at and point to the observer in video snapshots in the figure. In this case, the observer to whom the camera is attached interacts with other persons in the video. Since two persons point to the observer from a distance, no ego-motion is visible in the snapshots. There are video snapshots which contain ego-motion in the Figure 1.3. Snapshots in the Figure 1.3 are taken from the videos which belong to the DogCentric Activity Dataset [17].

In these snapshots, there is a person who is walking toward the observer. The person punches the observer and causes a large amount of ego-motion. After punching, the snapshots are blurred due to the high amount of camera motion. This is different from the third-person videos where camera is not affected by the action. First-person videos have rapid changes of motion and illumination, blur, ego-motion. Also, due to different perspective, it is possible to see the different body parts to which the camera is attached. For example, when a camera is attached to the back of a dog, video snapshots contain the constant views of the animal's body parts like in the Figure 1.4. In the example of dog activities, rapid motions of the animal cause sudden changes in video while responding to any trigger like throwing a ball or sound of car horn. Also, first-person videos provide close and small angle of views of objects and object-body interactions. For instance; while a person is writing a note to a paper, the head-mounted camera probably view the person's hand with paper and environment. Hence, the camera



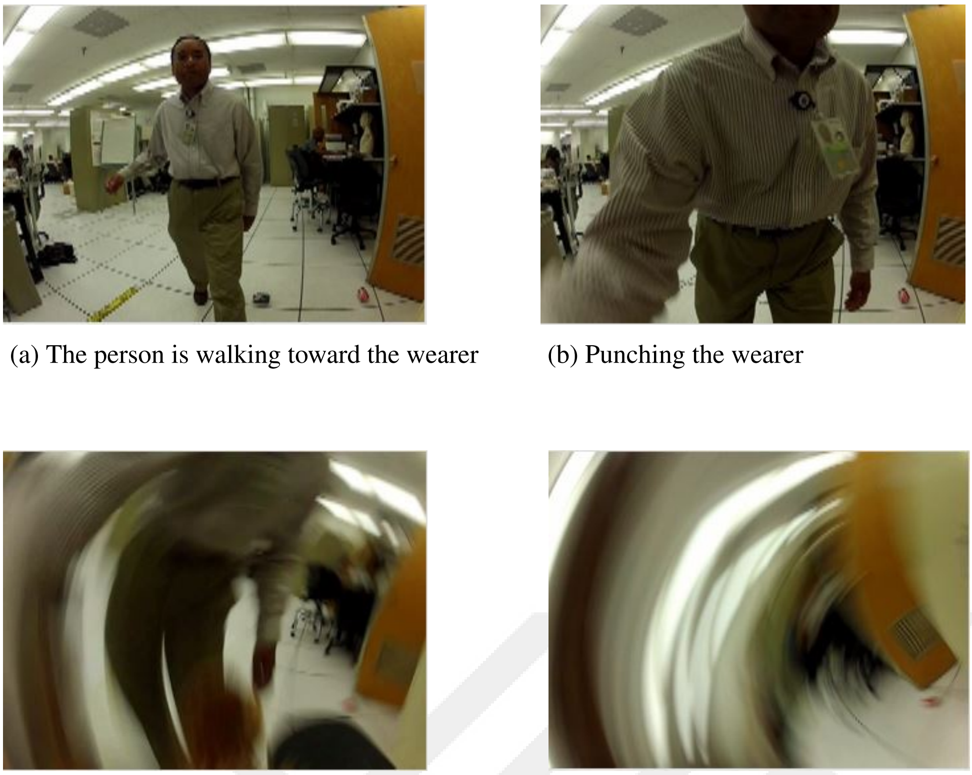
(a) A person is shaking the wearer's hand

(b) Before hand shaking

(c) There are two persons who point to the wearer

(d) Another snapshot of the pointing action

Figure 1.2: Examples of first-person viewpoint. There are two types of activities in the snapshots: Hand-shaking with the wearer and the pointing to the wearer [47].



(a) The person is walking toward the wearer

(b) Punching the wearer

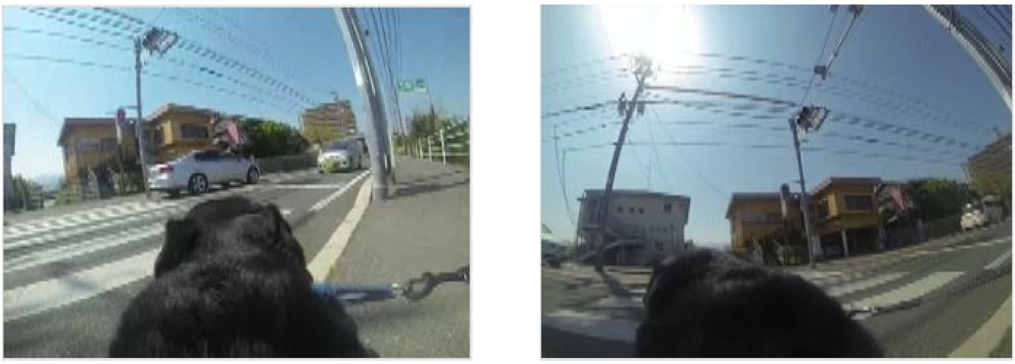
(c) After punching, ego-motion occurs

(d) Ego-motion

Figure 1.3: Examples of first-person ego-motion. Punching activity creates a huge ego-motion.

provide information of related objects or attention as long as the person look at the paper or keep writing. Therefore, this could be specific to first-person videos and used as discriminative information to extract features.

First-Person videos require different features and representation models because of their different characteristics. First of all, the ego-motion and the distinctive perspective of the videos must be taken into consideration while tackling first-person activity recognition problems. For example, in the activity representation model, the ego-



(a) A snopshot of the dog waits on the road

(b) A Snapshot of the dog while cars are moving

Figure 1.4: Examples of animal first-person animal viewpoint.

motion can be modelled through the use of global features since it affects all the pixels of an image. The other motions or object appearances occurring in the specific region of the camera view can be modelled through local features. A running dog creates a significant amount of ego-motion which affects all the scene and is accepted as global motion. Global motion requires the usage of global features. On the other hand, a car movement on the road which is accepted as local motion and only affects the pixels in a limited region of the image. Local motion requires the usage of local features.

1.2 Advantages and Challenges of First-Person Videos

First-Person videos bring some advantages and challenges as a consequence of unique properties of egocentric videos in comparison with third-person videos.

- Advances in first-person device technologies enable the researchers to study in several fields from security to elderly or disabled people's daily lives. New datasets in new fields are captured and presented to the usage of computer vision community.
- The observer is involved in action or interaction in the first-person videos. Hence, they allow the researcher to investigate the interactions between the objects and the observer deeply.
- It is possible to track the wearer's activities, gaze, hand or observations. Hence, ego-centric videos allow to identify, use and infer the attention information of the wearer. Therefore capturing the main interest points is possible.
- Ego-motion can give the cues of the motion, attention or events in the video. Hence, first-person video allows to infer the type of the activities from the ego-motion.
- There could occur dramatic change in illumination and scene in ego-centric videos. The dramatic changes are another advantage since they make possible to infer the activities from these changes.
- Rapid changes in the first-person videos also provide information to the researcher. For example, a punch action changes the optical flow distribution of the video dramatically, whereas a punch action is only a observation for the third-person camera. As another example, a hand shaking with the observer affects the color based features of the video since the hand shaking occurs in the center of the image because of involving of the observer; but, in third-person viewpoint, a hand shaking action does not affect the color features so much because the action does not necessarily occur in the center of the view.
- A wide range of sensors can be used together with the wearable cameras. Sensors provide different types of information such as voice, heart rate, gaze information ,and so on.

Challenges of first-person videos are also listed below:

- The motion in the video is unpredictable and instant objects or activity changes may occur given that the camera is not static. Blur or external physical effects like a fly, an insect, or the rain may affect the feature distribution and quality of video unexpectedly.

- The content of the videos can differentiate extremely from dataset to dataset. While in a dataset all videos are captured from a sport activity, in another dataset the theme could be daily hospital recordings. In third-person videos, theme is generally known before capturing the video as opposed to ego-centric videos. Therefore, it is not easy to apply existing traditional methods to ego-centric videos.

- In first-person videos, attention can change instantly especially in life-logging or sport activity videos. As a consequence of the changing nature of the attention in the videos, there could not be focusing on any object, person, activity and interaction consistently. This property makes it difficult to extract interesting points from the videos.

- Some first-person videos are captured for real - time decision making which requires real - time video processing that causes new algorithm and software challenges for the researcher.

- Changing illumination is usually seen in first-person videos. For example, a wearer can capture the video indoor in the first part and outdoor in the remainder part. This variation in illumination causes a huge change in distribution of the features.

- Another issue is the privacy problem. A wearable camera can easily capture a video everywhere. The widespread usage of ego-centric videos creates privacy problems. For instance while a person captures a first-person video in a street, other people can also be seen seen in the video. Also, storage of the videos without consent is another problem.

- Storage is not only related to privacy issues, but also creates handling massive amount of data problem. Easily capturing the video makes possible creating a huge amount of data. Therefore, in order to prevent the data loss without decrease in the quality of the video, storage is necessary.

- Since the specific intent of ego-centric videos is not determined before the capture, there occurs uninteresting and repetitive scenes in the video. A wearer who is cycling probably captures the same type and repetitive outdoor scenes. This requires pre-processing or grouping the scenes in order to get rid of the unrelated scenes or processing a huge amount of repetitive data.

- Wearable devices have limited resources such as processing power or storage capabilities. For some applications, such as real-time detection or recognition, real-time processing the data can be a problem due to the limited resources.

- Wearable devices still do not have the same video quality capabilities with the static cameras. Therefore, extracting features from the low-quality videos is another problem as opposed to the third-person videos.

These challenges lead to questioning the suitability of traditional approaches which are currently applied to the third-person videos and motivates the research on specific methods for first-person videos.

1.3 Scope of This Thesis

As the scope of this thesis, we concentrate on activity recognition from first-person videos. Other applications such as object detection, anomaly detection are out of our scope. We study on two datasets, JPL Interaction [47] and DogCentric Dataset [17]. These datasets contain specific types of activities. In [47], there are 7 types of indoor activities including hugging, pointing, and punching. On the other hand, [17] contains 10 types of outdoor activities. These activities are labelled in the training dataset since we use supervised learning.

In first-person videos, activities can be performed in two ways. An action is performed toward a person or it is performed by the person. The datasets which are used in this thesis contain both types of activities. In [47], the actions are performed toward a humanoid model, whereas actions in [17] are performed by a dog itself. In this study, both of them are aimed to recognize.

1.4 Contributions

In this thesis, we propose a new innovative approach for first-person activity recognition based on multi-kernel learning and boosted multi-kernel learning methods. Traditional methods employ equal weighted features for the activity recognition problem regardless of the importance of each feature. On the other hand, we assign different weights to the features based on the their importance and discriminative representation capabilities. For this purpose, we combine the features having distinctive information to use together and in an optimal setting. Thus, instead of exploiting each feature individually, all features take part of the final classifier by contributing to the final decision. Also, each feature is employed with different kernels. Each kernel and feature combination is tried in the training phase and the best combination is determined. The determination of the feature and kernel process is performed optimally by the method, instead of parameter searching and tuning. Standard learning framework which uses traditional feature selection and weighting technique, define rules for the representation model before the classification. Therefore, equal weighing on features and specific kernel usage are constraints that decrease the performance of the classifiers. It can be argued that the performance could be improved by using a combination of different kernels and kernel parameters.

Since the method proposed in this thesis fuses different features and different kernels in an optimized way instead of using pre-determined weights and rules, the dynamic and adaptive structure is the novelty of this study:

- Instead of a single feature, multiple features can be employed

- Instead of a single kernel, multiple kernels with different parameter sets can be employed
- Different kernel and feature combinations which fit the representation model best can be determined
- Assigning weights on classifier and selection of the best kernel-feature combination is performed in an adaptive framework
- Data-driven approach is applied at the training stage

Boosted multiple kernel learning approach[1] described in this thesis has been published in [37].

1.5 Outline

The thesis is organized as follows:

- In Chapter 2, we briefly review some of existing first-person vision methods and make a categorization of the methods regarding application field. Also, activity recognition studies are discussed based on methods employed.
- In Chapter 3, motion based descriptors are discussed. In this context, histogram of optical flow, cuboid and log-covariance descriptors are mentioned. Clustering of the descriptors are also talked.
- Multiple kernel learning and how the AdaBoost technique is integrated with multiple kernel learning are mentioned in Chapter 4. Furthermore, single kernel SVM and SimpleMKL library are introduced.
- In Chapter 5, experiments which are conducted in this thesis are introduced. Multiple kernel learning, boosted multiple kernel learning and single kernel traditional SVM approach performances are compared. The datasets used are mentioned. SVM kernels employed are also talked.
- Chapter 6 concludes this thesis by summarizing the methods mentioned.

[1] Fatih Ozkan et al. “Boosted Multiple Kernel Learning for First-Person Activity Recognition”. In: *2017 25th European Signal Processing Conference (EUSIPCO)*. Accepted for publication. 2017.

CHAPTER 2

FIRST-PERSON VISION

In this chapter, we discuss existing problems in first-person vision.

First-person vision community embrace the wide range of subjects on ego-centric videos. These subjects can be categorized in terms of application fields or methods employed. In the literature, there is a general distinction between studies based on motion-based and object-based methods. These methods differ according to its aim. For interaction or object detection purposes, object-based methods are employed, while motion-based methods are used for action recognition or summarization tasks. Therefore, application fields bring their specific methods in fact. Hence, the studies in the literature can be categorized regarding their application field. Thus, the methods used in that field are also mentioned.

We categorize the studies into six groups of *gesture and hand activity recognition*, *wearable sensors*, *activity recognition*, *eye tracking and gaze detection*, *object detection and recognition* and *life-logging and video summarization*.

After the general categorization, we talk about methods which are applied for the activity recognition that is the research subject of this thesis. Activity recognition studies are divided into three groups of *object based methods*, *motion based methods* and *multi modality based methods*. Figure 2.1 shows the categorization of the studies.

First topics which were focused by researchers in egocentric view domain were hand tracking and hand gesture recognition [53] [54]. Recently, new a few topics were arisen in the community such as activity recognition and eye tracking.

Starner, Schiele, and Pentland (1998) are the pioneers in the egocentric vision studies. They used wearable camera and addressed activity recognition problem from first-person viewpoint. Schiele et al. (1999) proposed a probabilistic algorithm for object recognition using wearable computer. Mayol and Murray in 2005 studied hand activity recognition based on object detection using wearable cameras.

In recent times, first-person vision have regained the attention. Some technological developments were the most important factor of the recent a number of work. Wearable cameras like GoPro, Google Glass have become widespread. Also, these cameras are easy to use. Therefore, some areas have received these devices easily and become convenient for usage of the wearable cameras. In the following subsections, first-person video topics are detailed.

2.0.1 Gesture and Hand Activity Recognition

Human gesture recognition is an important topic in first-person vision. Aims of gesture recognition include the interpreting of gestures and providing human-computer interaction. A user can control a machine by using gestures. Moreover, hand gesture recognition is more feasible than other types of gestures for human-computer interaction since hand gesture is one of the most natural communication tools [34]. Hand tracking and segmentation are the initial steps toward hand gesture recognition. After these steps, there are feature extraction and classification steps. Figure 2.2 shows the hand gesture recognition process scheme.

In [53], Hidden Markov Model based American Sign Language recognition method is proposed. User's hands is tracked by desk mounted and body mounted cameras. This work allows for long, meaningful and different types of sentences to be generated and recognized. Experiments conducted with desk mounted and body mounted cameras show that when body mounted camera used, recognition accuracy is higher than the desk mounted camera. [54] is another study which wearable camera is used for context aware gesture recognition. This is also an augmented reality research that tracks user location and binds virtual data to physical location. Sundaram and Cuevas propose a hand activity recognition based on object manipulation within probabilistic framework using wearable camera [56]. In this paper, authors presents an algorithm which works on low-resolution videos. This paper also benefits from hand-object interaction to recognize activities. In [55], Inside-out hand activity recognition is performed. For this purpose, hand tracking method which provides detecting whether any hand manipulated any object is used. Proposed method handles with blurred images which are caused by gaze directed cameras that are used in this work.

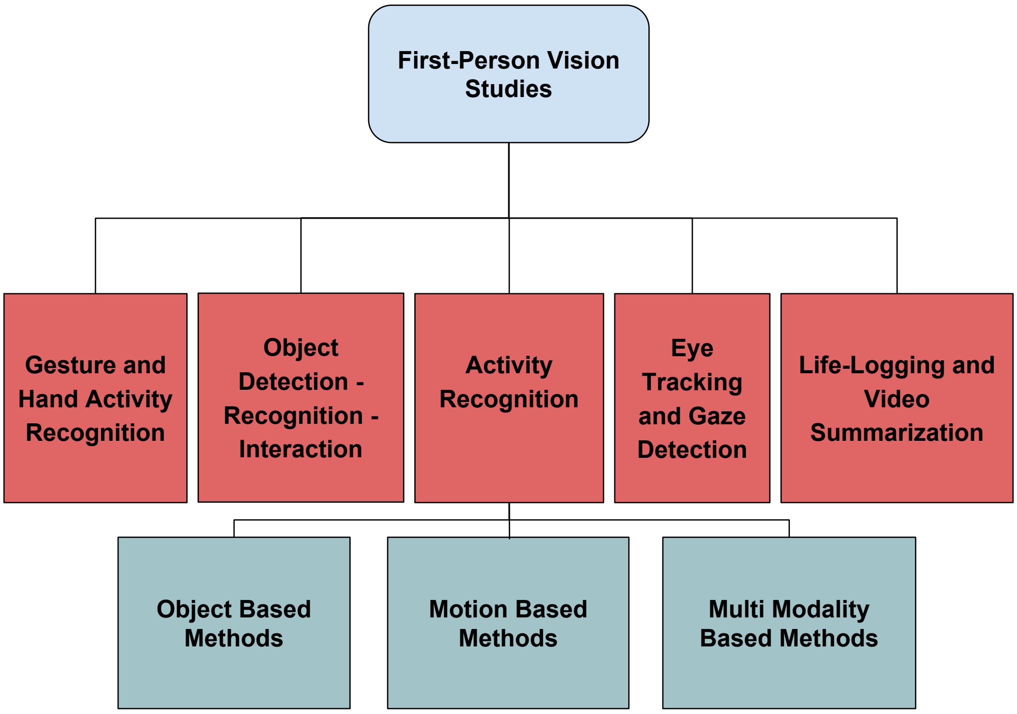

First-Person Vision Studies
Gesture and Hand Activity Recognition
Object Detection - Recognition - Interaction
Activity Recognition
Eye Tracking and Gaze Detection
Life-Logging and Video Summarization
Object Based Methods
Motion Based Methods
Multi Modality Based Methods

Figure 2.1: A categorization of first-person vision studies

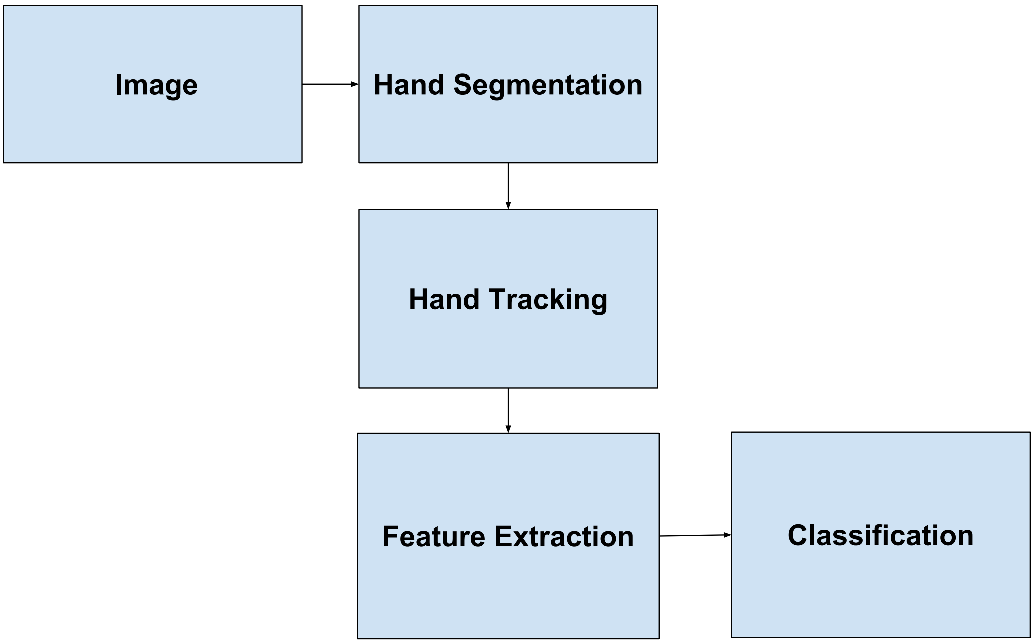


Figure 2.2: A hand gesture recognition process scheme

In [57], low resolution images are processed for activity recognition. The proposed system observes the user to detect whether there is an interaction between the user's hands and objects. If exists, it uses the manipulation information to classify user manipulations. In [67] a model based approach is proposed as opposed to work which use only signal based features. Multiple activities are recognized simultaneously using hand movements to segment the activities.

2.0.2 Activity Recognition

As a course of its nature, an activity which causes color, appearance changes can be handled with appearance based features. On the other hand, an activity which creates motion differences can be dealt with motion-based features. There are also activity types which require an interaction with objects in the environment. Object manipulations can be exploited in order to recognize these activities.

Fathi, Farhadi, and Rehg [11] exploit the object appearance changes in the video which is captured by an egocentric camera. In their study, objects, actions and hands are modelled together instead of independent analysis. Actions are represented with object-hand interactions. So, for example movement information is not used for the recognition task. Activity dataset is composed of daily activities such as meal preparation. In [40], daily living activities like brushing teeth are recognized based on object interactions from first-person viewpoint. Authors propose a method which benefits from the changes in object appearance when anyone interacts with it. Kitani et al. [18] propose an approach for activity recognition in first-person sport videos which uses motion based fatures and composes the feature codebook. Motion in the video is represented globally with histograms. After that, the histograms are clustered in order to compose codebooks. In [41], high-level temporal segmentation of egocentric videos are carried out. After segmentation, hierarchical activities are composed from the whole video. So that video is divided into meaningful parts such as “stationary capturing" or “dynamic capturing", “indoor capturing" etc. Authors argue that this segmentation method is useful for activitiy recognition. Ogaki et al. [36] use the eye movement and ego-motion in order to recognize indoor activities. They benefit from the motion features and combine these two types of motion features to improve the classification accuracy. In [47], motion and appearance based features are extracted from the first-person videos and clustered to compose visual words. Then, features are combined through multi-channel kernels using SVM. In [1] a new multi motion-based feature set which includes motion magnitude, direction and variation information is proposed. It also employs virtual inertial data generated from a video instead of using physical sensors. Inertial data describe the movement of intensity center through specified number of frames. Finally, motion-based and inertial data are combined and used for activity recognition. In [45], a new feature representation which keeps track of changes in descriptor values over time is proposed. Global and local motion descriptors are used. [46] propose human activity prediction from robot-centric viewpoint. [14] propose recognition of multi-type activities by a robot, which occurs sequentially or concurrently. In [17], global and local motion descriptors are employed. These descriptors are combined by using multi-channel kernel. Videos are captured by cameras which are mounted back of the four different dogs. Also [65], [66], [33], [20] are studies which use sensor data like accelerometer, smart phone. Sensors which are used in these work generally sense body movement so that features which are extracted by this sensors are used for activity recognition.

2.0.3 Eye Tracking and Gaze Detection

Eye tracking determines the eye movement and measures the eye activity. Eye activity data is collected through an eye tracker. The eye tracker directs light toward the eye center and the reflections are tracked by the camera inside the eye tracker. The direction of the eyes defines the eye gaze. Eye movement and gaze information may indicate the expression or attention of a person. Eye tracking and gaze detection allow a person communicate with other people and can be used to interact with a machine. So that, for example an elder or disabled person can be helped by such a device that provide the person to communicate with other people via eye movement and gaze detection. Thus, eye tracking and gaze detection are important tools for several computer vision applications. On the other hand, eye tracking and gaze detection cannot be employed easily since it requires a specific tool, eye tracker. Therefore, it can be costly. In addition, it cannot produce reliable and meaningful results. For instance, a person can sometime look at some points unconsciously and it can be difficult to discriminate such gazes. It is also difficult to apply eye tracking to people with glasses. Hence, additional sensors or cameras can be needed in order to provide additional information.

Muir and Conati [31] analyzed students’ attention to a system’s guidance, notifications and hints. Intended for this purpose, students’ eyes are tracked, attentions and reactions to hints which a system shows are measured. In [36], eye tracking is performed for the purpose of activity recognition. In [59], an eye tracking based method is presented for object recognition. Eye tracking is carried out via a head mounted tracker. SIFT features are extracted from the objects and best matched objects are found using these features. Fathi, Li, and Rehg [12] argue that there is a correlation between gaze points and interaction to the object. According to the authors, a person firstly fixed the gaze and after his/her activity comes out. Therefore, gaze behaviour could be exploited in order to recognize daily activities which require hand-eye coordination. In [24], gaze prediction is carried out, while in [64] and [63] attention prediction is performed on first-person videos.

2.0.4 Object Detection and recognition

Object detection find the object of a specific type in an image. It determines the location of a specified object. It is often used for object recognition. Object recognition which can be defined as the identifying an object in a video or image is an important research topic. Human brain immediately can detect and recognize objects in an environment. Unfortunately, it is a more challenging task for machines. Object recognition methods generally use some attributes of objects in the image like color, shape and distance from a specific target. Figure 2.3 gives the information about general process of object detection. In the figure, elephant and box features are extracted. Subsequently, they are detected when there are several objects in the scene. When there exists an object alone in an image, object identity cannot be recognized. For instance, if a ball exists in an image without any other object, the ball cannot be identified. However, if any other objects also exist in the image, then infering meaning from the scene becomes possible. Therefore, object context information is a useful tool for identification of objects. Object context is also used for activity recognition in some cases. [23], [50], [61] are the researches which benefit from object context for activity recognition from third-person viewpoint. In first-person vision, [39], [32], [38] are studies which recognize activities relying on object use which tagged with RFD. Daily activities are determined via RFID data coming from objects based on Dynamic Bayesian Networks. In [5], [4], [8], tracking of objects is performed in first-person viewpoint.

A number of studies also focus on color information in images toward object detection and recognition. For example, in order to discriminate a skin of a person from other objects color histogram is a commonly used tool. Moreiro, Marcenaro, and Carlo extract optical flow and color histograms for the purpose of detection hands in the video [30]. They use super-pixels so as to reduce the computational cost and make the center of the images the reference point of the coordinate system which they use in their studies. In [22], color histogram for human skin detection and super-pixel for cost efficiency are used also like another study [21]. Fathi, Ren, and Rehg [13] propose a method which provides learning objects in a weak supervise manner for first-person videos that include indoor-household activities. For this purpose, a segmentation is performed on the videos that hands, objects and background are divided into different groups. So that, the algorithm can focus only on objects instead of background objects that are not important for the task. [43] also uses segmentation for the object recognition in order to get rid of unnecessary objects in the background. Some of studies focus on object recognition for summarization purpose such as [65]. [59] benefits from eye movement for recognizing objects via an eye-tracker in egocentric videos, while [9] propose a method for discovering objects using appearance based features in egocentric viewpoint.

2.0.5 Life-Logging and Video Summarization

Because of recent developments in wearable camera technologies, people have the ability to capture their daily lives or activities everywhere individually. For example, a person who rides a bicycle can record his/her cycling. Long hours videos are captured in some cases such as sport activities or security systems. When the video record is required to be analyzed, it is not needed to watch the all the video record. Instead, users want to see just necessary or important part of videos. Video summarization provides the important parts of videos by selecting key events, frames, activities, subjects or any necessary things from videos. On the other hand, life-logging is the complete process of capturing all activities that a person performs along a specific time interval such as along a day, a sport activity etc. Life-logging applications include elder or disabled people but video summarization extracts interesting or key points from the video so that helps the analysing the elder or disabled peoples' lives.

In [2], Bai et al. develop a wearable computer, called eButton, which requires multimodal data from the camera and sensors for analysis the people's lifestyle. The

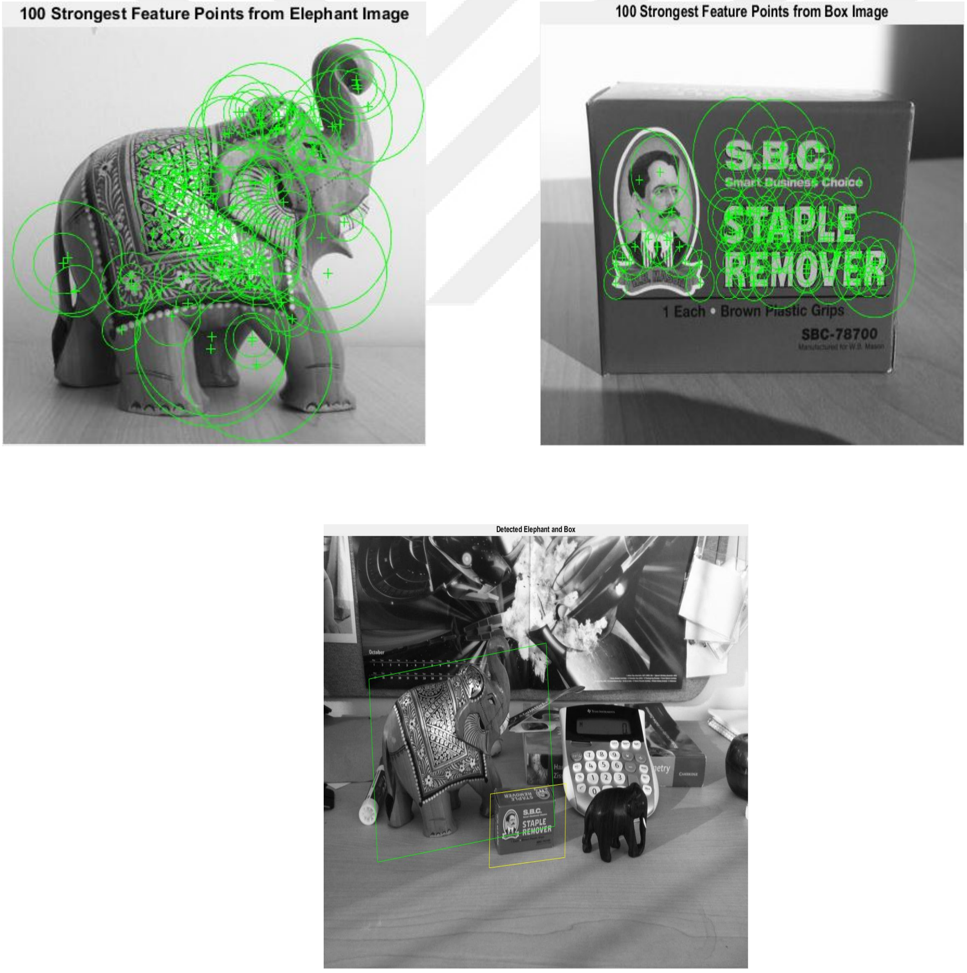

100 Strongest Feature Points from Elephant Image
100 Strongest Feature Points from Box Image
S.B.C.
Smart Business Choice
STAPLE
REMOVER
1 Each • Brown Plastic Grips
SBC-78700
Detected Elephant and Box

Figure 2.3: Object Detection example. Elephant and box features are extracted. They are detected in the whole scene. This figure is a reproduction using Matlab toolbox object detection example code.

computer is integrated with the sensors and camera so that it collects the different type of data and analyze them. The computer can monitor the eating habit or physical activity etc. In summarization applications, studies focus on especially how to select important frames from the whole video. [26] propose a story based method to summarize the video that discovers the story of the video using the influence between subshots of the video as a metric. Essentially, sub-events which lead to another event composed a story together. In [65], again a story based approach is employed for the summarization of videos. The authors Yong Jae Lee, Ghosh, and Grauman concentrate on the interactions between the camera wearer and other people or objects by exploiting the first-person shooting which allows the wearer to involved in the activities or interactions. By this way, people or objects which are labelled as “important" since the wearer interacts with them are used to select key events or frames from the video.

2.1 Methods In Activity Recognition Studies

In this section, we make sub-categorization on activity recognition studies since it is also concentration of this thesis. We discuss the limitations and open points for future research in the following sub-sections.

Recently, activity recognition studies have been performing based on three methods generally. One of them is object based methods which define an activity as an object usage pattern which causes the attributes of the objects like appearance changing. Motion based methods which describe an activity according to motion characteristics and the last methods, multi modality based, use the sensors together with visual information that is acquired from the cameras.

2.1.1 Object Based Methods

Object based methods require activities in the videos which contain object or hand usage, since these methods benefit from the appearance change objects or temporal relationship of object interactions semantically etc. This is the limitation of the object based method since some activities do no contain any explicit object usage such as two person talking with each other.

In [11], appearance changes of objects and hands are used for activity recognition from the ego-centric viewpoint. Daily activities such as meal preparation are segmented into sub-actions so that each activity is composed of specific intervals. Superpixels approach that is called for the regions in image in the paper are also employed for the task. Thus, the method infers the objects and hands by modelling the activities based on the object appearances. Pirsiavash and Ramanan [40] propose a method which represents the objects based on objects. Temporal pyramids are used for matching the temporal relationships between the sub-parts of the activities. For example, “making tea" activity firstly requires a filling a cup with water and then boiling water. This temporal relationship is represented in the temporal pyramid. [28] use object manipulation information together with attention information since for example, when a camera wearer looks at a knife without moving hands, an object manipulation does not occur. So, object manipulation could not be enough to infer the activity sometime. In [49], crafted features based on hand and object interactions are used for first-person activity recognition problem. CNNs are employed for the wearer's activities classification task. 4 different datasets are used and classification accuracies indicate the improvement by the proposed method. [27] proposes a twin stream network architecture, one stream for object appearance and one stream for motion information. The authors analyze the contribution of each stream to the recognition performance in the study.

2.1.2 Motion Based Methods

In general, motion based methods use features like optical flow, gradient etc. Most of the daily activities are so complex that just one type of feature cannot give discriminative information. Therefore, two or more features are usually used together. Thus, for example, one feature shows the color change in video while other feature indicates the motion magnitude change. There are more than one way to combine multiple features. Use of a standard learning framework without particular extension on feature selection and weighting implies the use of pre-set rules, such as unweighted sum, which gives equal preference to each feature independent of its classification ability. Use of multichannel kernels is proposed to combine multiple features. Each feature is considered as a separate channel and a pre-defined rule using exponents is utilized to combine them.

Ryoo and Matthies [47] extract motion features from the video in order to perform activity recognition on the first-person videos. In this context, motion features are clustered using K-means algorithm and combined using multi-channel kernel. Also, in the study, activity structures are analyzed and sub-activity features are used together in order to represent the activities effectively. In [1], a number of new multidimensional motion-based descriptors are applied. These global descriptors are used together by concatenating them in a vector in SVM. In [17], dense optical flows, Local Binary Patterns (LBP) are used as global features and cuboid and SpatialTemporal Interest Points (STIP) are used as local features. These features are used in a similar classification setting to recognize animal (dog) activities . [18] uses Dirichlet process mixture model for the motion histograms codebook, after clustering the motion histograms which are acquired by extracting the optical flow features from the videos. Also, [41] perform segmentation to the videos based on motion features while [36] use the eye-movement information for the activity recognition. [34] uses motion pyramid approach, while [45] uses pooled motion features and [45] employ histogram of time-series gradients which are results of motions in the video in a probabilistic approach. Furthermore, in [44], time series pooling is employed to detect short/long term changes of features. HOF and appearance descriptors from CNN are used in the representation model. [58] uses DogCentric activity dataset and features from CNN. In the study, input images are derived from optical flow. Experiments show the effectiveness of the proposed method.

2.1.3 Multi Modality Based Methods

In [51], authors use static and ego-centric cameras for the activity recognition. They define this device environment as a multi-modal approach. Features acquired from different cameras are encoded according to their importance. Thus, the method selects the camera that has the best view of activity and classifies the activities. For activity recognition, 3-axis accelerometer sensor and image sensors are used in [33]. Features which are acquired from these different sensors are the inputs for SVM classifier which decides the labels of activities. In [66], smart glasses, first-person camera and accelerometer re employed for first-person activity recognition in conditional random fields framework. In addition, use of Multi Kernel Learning (MKL) in a multimodal setting to fuse different audio and video features has been proposed for event detection in web videos [35]. It has been shown that MKL performs well even when redundant features are used and it outperforms other popular methods such as wrappers, filters and boosting as well [60]. MKL has been shown to produce promising results during the identification of emergent leaders in meeting scenarios [3]. [16] proposed an activity recognition method within probabilistic framework that infer the data sensor provides and transforms to the activity patterns. In [52] temporal segmentation and recognition of activities are performed via sensors which are worn on body. [67] does not rely on features acquired from signals, but they infer sub-actions of body motion. Their approach is capable of recognizing multiple activities by selecting features that belong to different types of activities. They use body worn sensors, to achieve activity recognition task. In [7], human location detection and activity recognition are performed by observing the users via sensors. Mayol and Murray developed an approach which recognizes hand activities within a probabilistic based framework using wearable sensor. In [6], authors propose a human activity recognition method using accelerometer and a wearable device. Image features are extracted based on optical flow and acceleration data is acquired with the accelerometer and classification is performed using SVM.

CHAPTER 3

MOTION BASED DESCRIPTORS

This chapter focuses on the global and local motion related features which are extracted from the first-person videos. In the previous chapter, various studies for the activity recognition in the literature were examined. The limitations of the aforementioned methods motivated us to solve the activity recognition problem for the first-person videos in a new manner. The previous studies also have paved the way for our new approach.

We employ a model which covers different and challenging aforementioned characteristics and exploits advantages of first-person videos compared to third-person videos. Object interaction-involved methods do not handle activities which cannot be described with object appearances. Also, these methods depend on object detection or recognition and inherit its difficulties. Therefore, the first attribute of the new method must be robustness to color, appearance changes, overlapping in cluttered area and not requiring object usage. Secondly, the new method must not be affected by the ego-motion badly, but exploit it in order to discriminate different aspect of information from videos. Ego-motion is contained in the representation model in this study. Thirdly, not only spatial but also spatio-temporal pattern of a movement must be considered. Thus, the new method does not depend on posture or appearance. Finally, the new method must address not only global or local motion, but both of them.

In this context, global and local descriptors are employed to satisfy the aforementioned concerns for the activity recognition on first-person videos. First of all, global and local descriptors are mentioned. Then, feature clustering is discussed.

3.1 Global Descriptors

In the following sub-sections, global motion-related descriptors, extracted from the first-person videos, are described. Global descriptors capture basic motion information which affects all the scene such as the egomotion. Thus, ego-motion is exploited to discriminate different characteristics of videos. Therefore, camera movement compensation methods are not used to remove ego-motion. In this context, global information is extracted from all pixels in an image that is called as “global feature". After that, global features are employed in a specific representation. The representation could be a histogram or concatenation of the feature vectors.

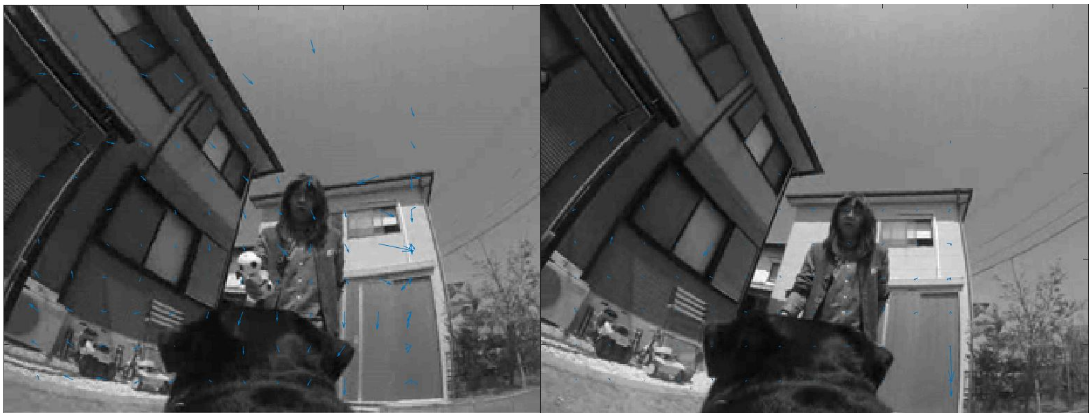


Figure 3.1: Optical flows extracted from two frames of a video

In this study, global motion is represented using two descriptors both based on optical flow information: Histogram of Optical Flow (HOF) [47] and Log Covariance (LogC) [15].

3.1.1 Histogram Of Optical Flow (HOF)

In order to describe global motion which represents the dominant motion fields which have effects throughout a frame, dense optical flow method is used. Optical flows are calculated between every ensuing frames for all the pixels. Then, these optical flows create the global motion flow. In first-person videos, camera movement, ego-motion, usually lasts throughout the video. Besides, it affects the flow fields of all the pixels dramatically or trivially depending on the types and structure of the activities in the video. Thus, global motion gains additional importance for ego-centric videos because of continuous movement of the camera. Figure 3.1 shows the optical flows extracted from a first-person video. A dog with a back mounted camera runs toward a ball which a person throws in sequential images in the figure. While dog runs, optical flow vectors are computed. In the figure, there are small flows since the dog waits before the person throws the ball.

Optical flows are vectors which have magnitude and direction information. Magnitude is the length of the movement (flow) and direction indicates from where the flow starts and until where the movement goes. After calculation of the optical flows of all the pixels, each frame is divided into s-by-s grids so that there occurs $s \times s$ regions in each frame. Also, direction of each flow is represented with eight motion directions. For instance, if the direction of a flow is 10°, then the flow is represented with first motion direction that contains 0°- 45°. Furthermore, a histogram of optical flows is computed in each grid based on the magnitude and the direction of the movement. Thus, there occurs s-by-s-by-8 histograms all over the frame and each optical flow is placed into the related bin of the histogram. Consequently, optical flows are categorized into a number of groups based on direction and location. In the histogram, each optical flow is counted according to its magnitude.

Figure 3.2 gives an example of the histogramming of optical flows. Vertical axis of

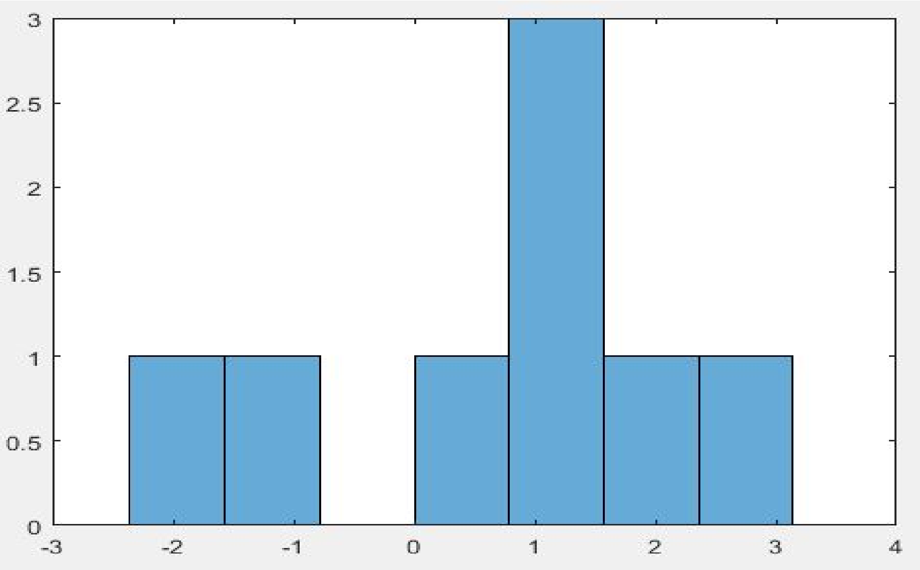


Figure 3.2: An example of optical flow histogramming. Bin values, horizontal axis, show the range of angle from based on Π value. Count, vertical axis, shows the number of optical flows that is angle is within that bin.

the diagram denotes the count of the optical flows and horizontal axis describes the representative angle values based on Π value. For example, number of optical flows which have representative angle value 1 is 3 according to the histogram in the figure. Figure 3.3 shows the counts of first bins of histograms which are generated in each grid in a frame. These counts are acquired by multiplication with magnitude values.

The histograms aforementioned above constitute the descriptors, the histograms of

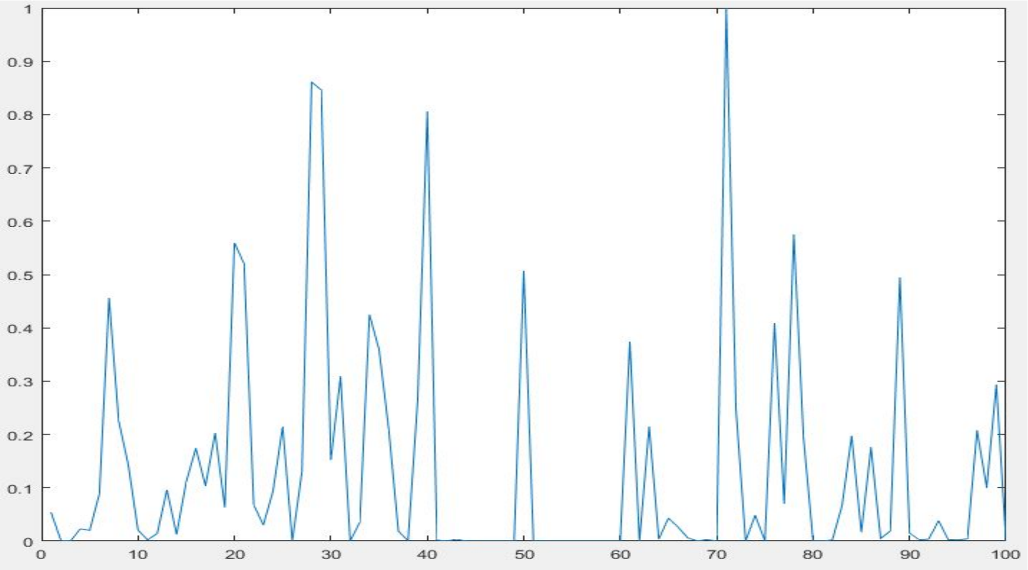


Figure 3.3: Distribution of first bin counts of the histograms of optical flows extracted from a video. Horizontal axis shows the histogram number, and vertical axis shows the count.

optical flow. S values of histogram of optical flow descriptor are discussed in section 5.1.

3.1.2 Log-Covariance (Log-C)

Log-covariance (Log-C) descriptor is originally designed for third-person videos to capture different characteristics of the motion [15]. Videos are divided into L frames length action segments. Then, Log-C is calculated for each action segment separately. For this purpose, at each pixel of each action segment, 12 dimensional optical flow-based motion-related features and intensity-based gradient vectors are extracted. 12 dimensions are listed below:

$$f(x, y, t) = [x, y, t, I_t, u, v, u_t, v_t, Dvr, Vrt, Gt, St]$$

- **Coordinates**: (x, y, t) Horizontal, vertical and temporal axes of a pixel in an action segment from a video.
- **Intensity gradients**: (I_t) First-order partial derivative of intensity gradient of raw video sequences with respect to temporal t direction.
- **Optical flow**: (u, v) Optical flow extracted from the action segment.
- **Optical flow derivatives**: (u_t, v_t) First-order partial derivative of optical flow with respect to temporal t.
- **Divergence**: Dvr The spatial divergence of optical flows computed at each pixel.
- **Vorticity**: Vrt The vorticity of optical flows computed at each pixel.
- **Tensor Invariants of Optical Flow**: (Gt, St) Gradient and strain tensors of optical flow regardless of the coordinate system.

Intensity gradient and optical flow derivatives, divergence, vorticity and tensor invariants of optical flow are computed as follows:

$$I_t = \frac{\partial I(x, y, t)}{\partial t} \tag{3.1}$$

$$u_t = \frac{\partial u(x, y, t)}{\partial t} v_t = \frac{\partial v(x, y, t)}{\partial t} \tag{3.2}$$

Divergence gives us the information of how much a vector field expands or compresses around a point. Therefore, divergence shows how the optical flow field behaves around a pixel regardless of the optical flow magnitude or direction of that pixel. Figure 3.4 shows the positive divergence around the point since all the vectors are away from the black dot. There is optical flow field expanding around the black point whereas there is not any optical flow vector at the exact black point.

$$Dvr(x, y, t) = \frac{\partial u(x, y, t)}{\partial x} + \frac{\partial v(x, y, t)}{\partial y} \tag{3.3}$$

In order to discriminate the circular motion in the image, vorticity is used as an attribute based on optical flow.

$$Vrt(x, y, t) = \frac{\partial v(x, y, t)}{\partial x} - \frac{\partial u(x, y, t)}{\partial y} \tag{3.4}$$

Gradient and strain tensor of optical flow are also computed as follows:

$$\nabla u(x, y, t) = \left(\frac{\partial u(x, y, t)}{\partial x} \frac{\partial u(x, y, t)}{\partial y} \frac{\partial v(x, y, t)}{\partial x} \frac{\partial v(x, y, t)}{\partial y} \right) \tag{3.5}$$

$$St(x, y, t) = 1/2 \times (\nabla u(x, y, t) + \nabla^T u(x, y, t)) \tag{3.6}$$

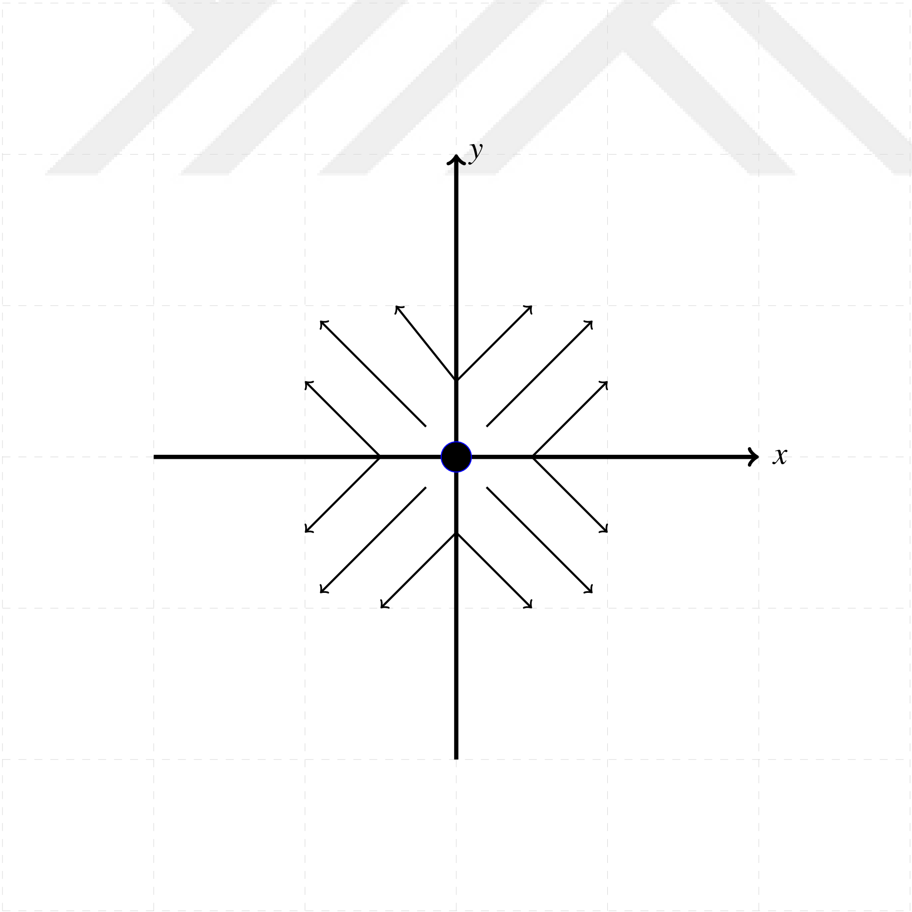


Figure 3.4: An example of positive divergence of vectors

This set of spatio-temporal features represents dynamics of the motion in first-person videos in a more comprehensive way than basic optical flow-based features. Covariance matrix, L, of the feature set is symmetric, so only some of its members are unique. L is $l \times l$ matrix. Unique members of the matrix accepted are the numbers that lie diagonally and under the diagonal. An example of the symmetric matrix:

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 7 & 3 \\ 3 & 7 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}$$

And the example of compact representation of the covariance matrix is mentioned above.

$$\begin{bmatrix} 1 & & \text{sym.} & \\ 2 & 1 & & \\ 3 & 7 & 1 & \\ 4 & 3 & 2 & 1 \end{bmatrix}$$

Therefore, the number of independent values in the matrix are computed as the following:

$$(l^2 + l)/2 \tag{3.7}$$

Then, compact covariance descriptors are created by capturing these features in the covariance matrix since high dimensional feature vectors are not efficient for clustering and classification operations. Euclidean operations cannot be applied into covariance matrix, since it does not lie on Euclidean space. Covariance matrix lies on Manifold space. For example, K-Means clustering is an Euclidean operation. Therefore, the clustering method is not expected to be effective on covariance matrices. For this reason, an appropriate operation to Manifold space can be applied or the covariance matrix can be converted into Euclidean space. We use matrix logarithm [19] operation to convert manifold of covariance matrices into Euclidean space.

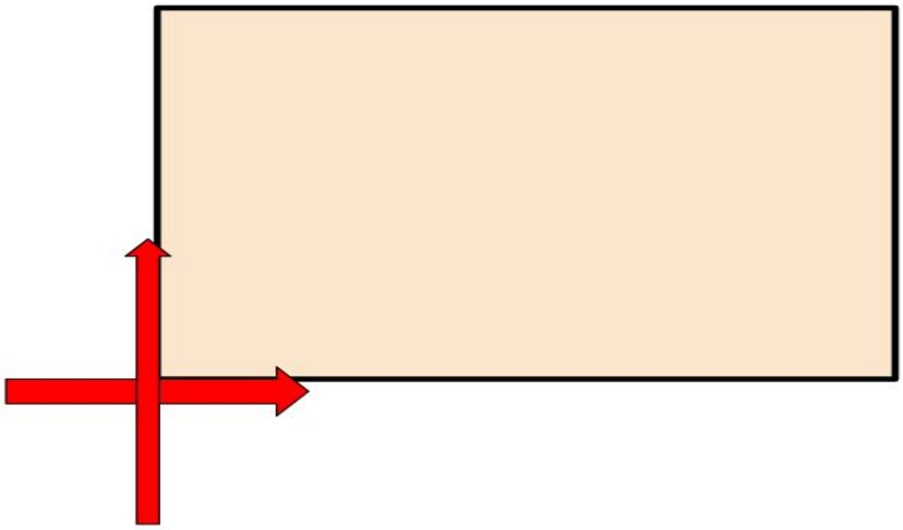


Figure 3.5: A Corner in an image

3.2 Local Descriptors

Local descriptors provide the complementary local information, which is necessary for the recognition of different types of activities. In this context, interest points are detected in an image. The interest points are generally robust to several factors such as changes in appearance or illumination. Then, local regions are determined around the interest points. After local regions are determined, these regions are represented in a specific way, such as histogram.

In this thesis, “*Cuboid*" [10] descriptor is employed as a local descriptor which is mentioned in the following subsection.

3.2.1 Cuboids

Cuboid features are sparse 3D XYT space-time features [10] and they have been used extensively to recognize behaviour in third-person camera perspectives. Sparse feature is a compact representation that compresses high-dimensional features efficiently. Sparse feature representation provides extracting high-level meaningful information from the videos which have low-level pixel values. Sparse space-time features have also been shown to perform well for activity recognition applications [19]. However, as it is shown in [25], sparse features may cause problems for activity recognition problems if they are too rare. Besides, spatio-temporal corners are not often detected in a video, so rarity is also a problem for detecting spatio-temporal corners. In addition, cuboids are based on spatio-temporal corners.

First, spatio-temporal Cuboid feature detector is run in order to detect feature locations.

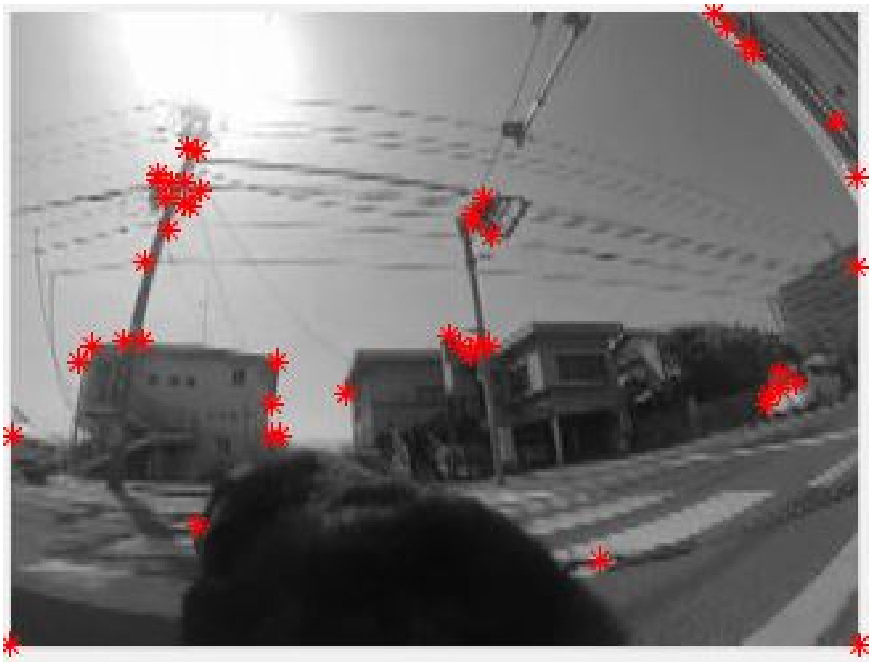


Figure 3.6: Corners detected in the image

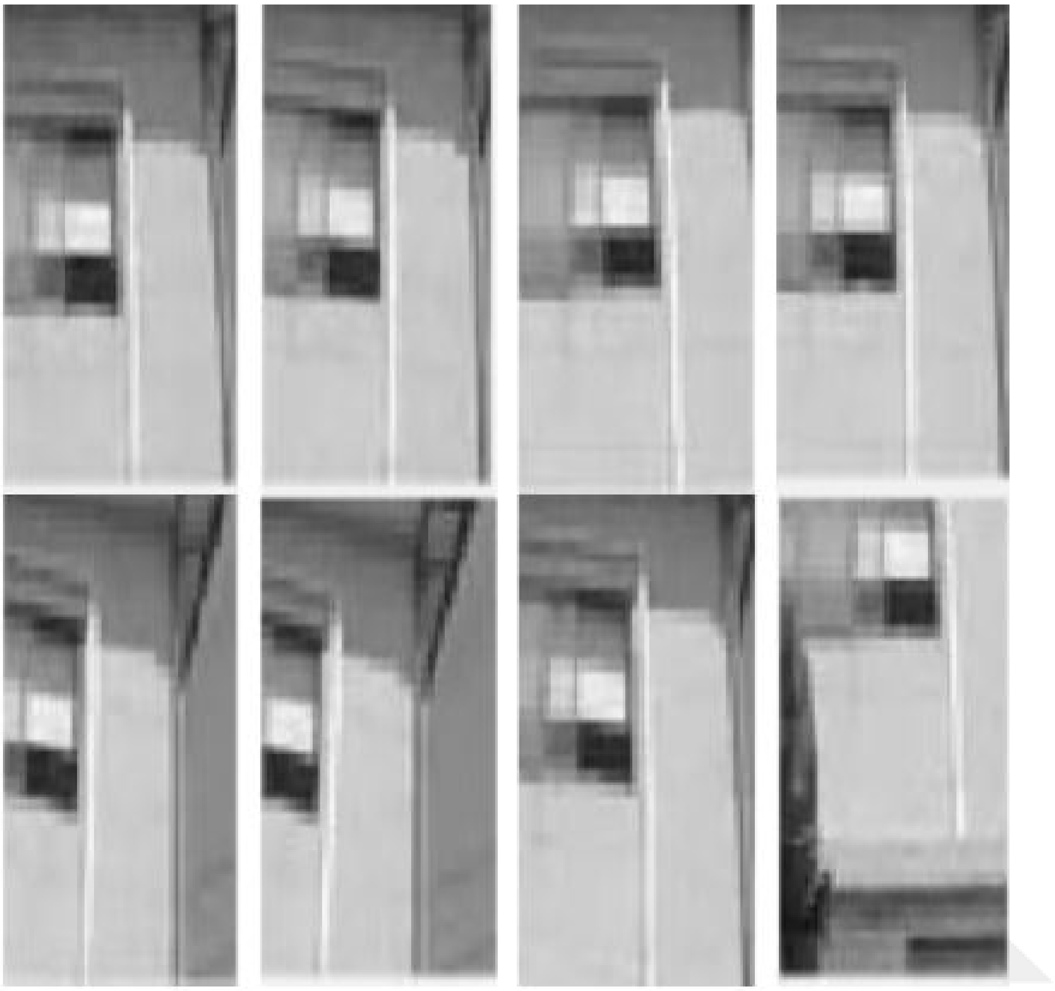


Figure 3.7: A window in frames of a video moving in reverse direction

While the idea is similar to spatial detectors, detection proceeds along the temporal direction *t* in addition to the spatial *x* and *y* directions. Then, at each interest point, spatio-temporally windowed pixel values (i.e. flattened gradient vectors) are calculated to form a Cuboid. The Cuboids are specifically designed for behaviour recognition applications and they aim to detect too many features rather than too few in order to handle challenging conditions.

To be more precise, we elaborate the cuboid feature in this part. Cuboids represent local information in an image. Therefore, the Cuboids operate on interest points which are extracted from images. There are several ways of detecting interest points in an image. One of these ways is based on the corner detection. Since corner detection is the base of Cuboid feature, it is preferred in this thesis rather than other ways of interest points detection.

In an image, corners are the regions that include gradient change in vertical and horizontal direction. Figure 3.5 shows the change of gradient in vertical and horizontal direction in corners. A good interest point is expected to be robust to variations like brightness, and illumination. In addition, interest points should be computed fast and found easily since descriptors are computed around interest points so that, fast computation of interest points support also the fast computation of descriptors. Figure 3.6 shows the corner detected on the image which contains a dog and a number of buildings. Red points indicate that there is a corner there. In the figure, there are dense corners on transmission towers as we expect.

Spatial interest points are found in a spatial plane. For example, corners in an image can be counted as an interest point like in the Figure 3.5. On the other hand, spatio-temporal interest points are not found on a spatial plane since they occur on a spatio-temporal plane. Therefore, a temporal dimension is added to the spatial interest points in order to compose spatio-temporal interest points. Thus, in order to detect a corner, gradient change is searched not only along vertical or horizontal dimension, but also temporal direction throughout a number of frames. For example, an object moving in a street stops and starts to moving reverse, at the moment when the reverse movement starts, a gradient change occurs in temporal dimension.

Cuboids are extracted at each interest point since it is a sparse and local feature. A spatio-temporal window sliding on frames is contained in Cuboid if it contains spatio-temporal corners so that cuboids are computed in some regions of a video that surround an interest point and store local information.

There are various transformations to apply to cuboids like normalizing, the brightness gradient or extracting motion information using optical flow method from the cuboids. After that, local histogramming, global histogramming or just concatenating can be applied to cuboids. Thus, cuboid descriptors are constructed. Figure 3.7 shows 8 selected frames a video. In these frames, the window in the figure moves from left to right, and then starts reverse movement. This is the exact pattern of the movement that cuboid descriptor describes.

Also, in the Figure 3.8 there are 9 sample frames, interest points and cuboids extracted from these frames of the video from JPL-Interaction dataset. Colorful boxes indicate the interest points. They locate at the intersection of the objects. For example; there are pink boxes at the top corner of the desk or red boxes at the intersection of the neck and arm of the man. Images in the middle of the figure show the extracted cuboids. There are 5 sequences of cuboids. In the sequence, a reverse movement is seen. In the first sequence, the dark color region moves to left at first and then moves back to right again. Images at the bottom of the figure are the sample frames of the video.

3.3 Feature Clustering

The motion information of a video by word occurrences is described by using the bag of visual words (BoW) approach. Thus, each video is represented by dictionary of representative feature vectors. Visual word model enables efficiency for representing descriptors in videos. Each frame in videos can contain several important interest points. Therefore, millions of features can be extracted from all videos. It is computationally too complex to perform computations on such millions of features. A dictionary of features provides efficiency in computations. Therefore, instead of employing millions of features, representative visual words are used in the representation model in this study.

Each collection of descriptors is separately clustered into multiple types by K-Means algorithm. K-Means provides partitioning the data, descriptors in this case. Thus, each

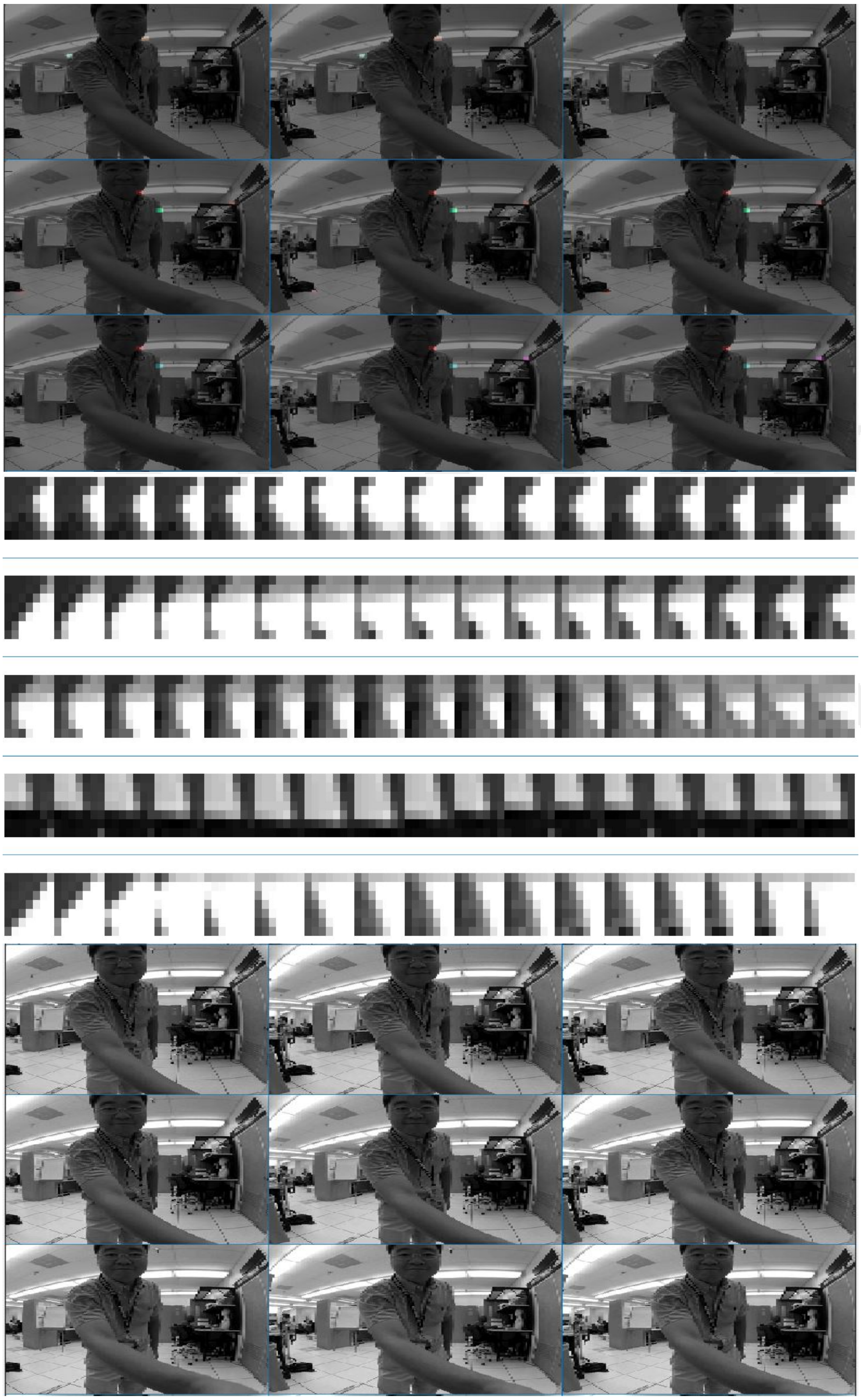


Figure 3.8: Some interest points, cuboids and frames throughout the video, respectively

descriptor is assigned to a visual word and the histograms for each video are computed so that representative visual word histograms of each video are obtained. Since each set of feature descriptors is clustered separately, three histograms are computed for each video. The histogram H_{id} is a w dimensional vector for the ith video obtained using descriptor d and w is the number of visual words. For each video, each descriptor histogram computed is concatenated and final histogram is obtained.

$$H_i = [h_{i1}, h_{i2}, h_{i3}...h_{iw}] \tag{3.8}$$

H_i is the histogram of video v_i, h_{iw} is the number of w^{th} visual word of the i^{th} video.

$$H^i = [H^{d1}H^{d2}H^{d3}] \tag{3.9}$$

H^i is also the concatenated histogram of video v_i. The concatenated histogram is composed of histograms of HOF, Log-C and Cuboid ($H^{d1}H^{d2}H^{d3}$).
K values of visual words model are discussed in section 5.1.
Figure 3.9 shows the clustering process of the features before mentioned. First of all, feature detection is performed. Optical flows are computed for each pixel in all videos. Optical flow features are the base of HOF and Log-C descriptors. HOF is acquired by histogramming locally the optical flows exracted. Log-C is also acquired with the computations of optical flow based features and by concatenating these features. Cuboids are descriptors which applying some transformation techniques to vectors of spartio-temporal interest points. After detection, descriptors are constituted by representing the features as HOF, Log-C and Cuboids. These descriptors are clustered, thus visual words are composed. Finally, as shown in the figure, histograms are computed separately for each descriptor counting the visual words.

Clustering does not store spatial information of descriptors but provides compact, efficient and easy to compute representation. After all, visual words which are the final form of descriptors can be used to train classifiers.

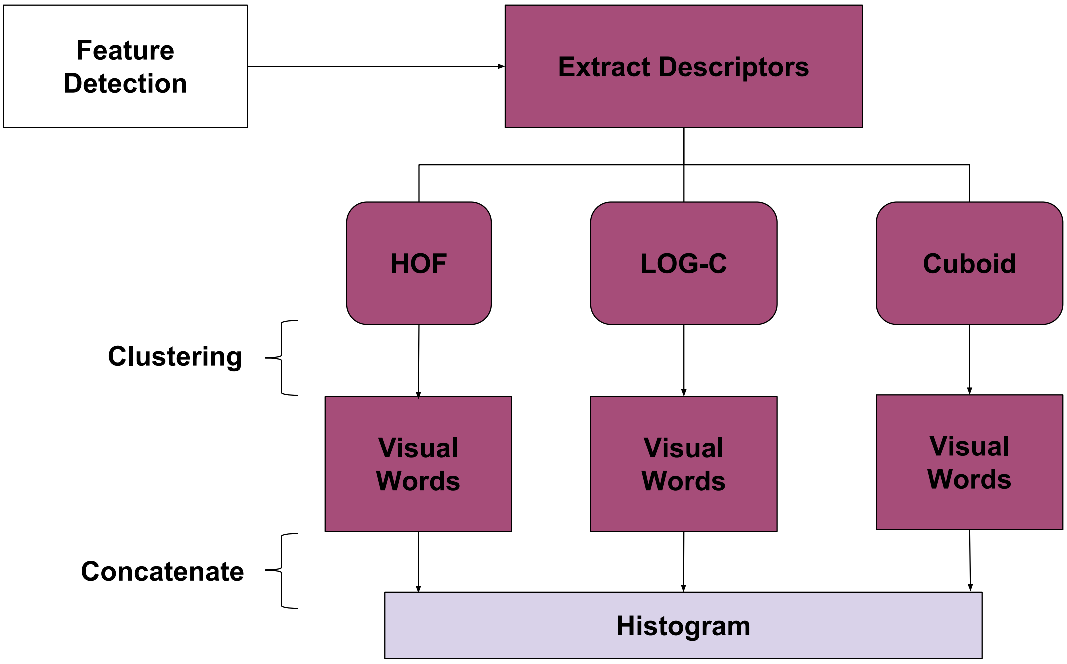


Figure 3.9: Feature clustering process using the HOF, Log-C and Cuboid features

CHAPTER 4

MULTIPLE KERNEL LEARNING

In this chapter, we discuss multi-channel, multiple and boosted multiple kernel learning.

Most of the machine learning algorithms need to transform the features into a higher dimensional space for some operations such as similarity computation between feature vectors. For this purpose, the machine learning algorithms employ kernel methods allow transformation of the vector space to a higher dimensional space. There are both linear and non-linear kernel functions. Linear kernel function separate the features which are linearly separable. Non-linear kernel function transforms the linearly non-separable features into higher dimensional space. A non-linear kernel function computes the similarity between features in higher dimensional space.

SVM is one of the kernel-based learning algorithms. We first start with briefly mentioning about SVM before multiple kernel learning. Then, we present SVM with multi-channel kernels. Finally, we talk about multiple and boosted multiple kernel learning.

4.1 Support Vector Machines (SVM)

SVM is a binary supervised classification algorithm which finds a hyperplane in order to divide the data points into two groups. It is used in various types of problems such as activity recognition, handwritten text recognition and object detection. SVM classifier has a decision boundary, (hyperplane), which separates the linearly separable data points into two parts. The closest data points from two classes to the hyperplane form support vectors. SVM is a maximum margin, which is the distance between closest data points to the hyperplane, classifier due to it finds the maximum margin while separating the data points.

There are a number of ways for applying the SVM to the multi-class classification problems despite the fact that SVM is a binary classification method by its nature. One-against-all and one-against-one approaches are two alternatives. Binary classifiers are used by both alternatives. If there are *N* classes, *N* classifiers are trained for one-against-all technique. Each classifier is trained between class 1 and the rest. Instances belong to the class 1 are labelled as positive and the rest are labelled as negative. On the other hand, in one-against-one approach, there is a binary classifier for each pair of classes. For N classes, $N(N-1)/2$ classifiers are composed. Final decision on test points are made on voting mechanism. Each classifier assigns a label for the test point. Most assigned class label is determined for the test point.

Data points can also be linearly non-separable. A hyperplane cannot be found to separate the data points. In such cases, it is required to map the data into 3D space. For example, an activity recognition problem is too complex in order to apply simple binary linear SVM classifier. There are several images as training data which include different types of activities. In this task, it is needed to transform the data in order to separate them. For this purpose, SVM kernels are employed. A kernel can be simply defined as a similarity function between feature vectors. It transforms data points in 2D space which is not linearly separable into higher dimensional space which makes the data points linearly separable.

SVM has the following advantages:

- Works well on a various types of problems
- Works on problems that are non linearly separable
- Works well on smaller datasets
- It is robust to noise

Following are also disadvantages of SVM:

- SVM uses several kernel parameters. It is required to determine the parameters to classify data points correctly and to prevent the overfitting.
- Appropriate SVM kernels differ according to the problems, so it is also required to select suitable kernel.
- Since it is a binary classifier, it needs pair-wise classifiers for multi-class problems.

In learning problems, the fundamental issue is determining the most appropriate kernel function and its parameters. Kernels are used to create a similarity matrix that is composed of all similarity values between each pair of training instances. The similarity computation with kernel function is done as follows:

$$d(y) = \sum_{j=1}^{N} \alpha_i k(x_j, y) + b \tag{4.1}$$

where x_i is the ith training instance and y is the test instance. k(.,.) is one of the SVM kernel functions such as linear, gaussian.

A kernel-based method works with limited computational cost. On the other hand, selecting the most appropriate kernel for solving a specific problem is challenging. For this purpose, cross-validation is used to determine the kernel and its parameter on a validation dataset. In addition, for multi-class problems, it is required to adapt the kernel methods using approaches such as one-against-all or one-against-one. As a result multi-class problems require running increasing number of SVM classifiers depending on the number of classes. Instead of determining kernels and parameters by trying different kernels on validation set, an algorithm can be used to find the best kernel and parameter combination of best kernel function. Thus, multiple kernels can be used together rather than a single kernel. Multiple kernels can provide more stable and robust similarity computations between training instances. Multiple-kernel approach provides linear separation in higher dimensional space by using multiple kernels.

It is possible to assign equal or different weights to kernels. In the following section, multiple kernel approaches are mentioned.

4.2 Multi-Channel Kernels

Multi-Channel Kernel combines different types of features. Each feature is a separate channel for the kernel. Also, there is a pre-defined rule in order to compute similarities between a pair of features. In [47], a multi-channel kernel is proposed whose function is the following:

$$k(x_i, x_j) = exp(-\sum_{c} D_c(H_a, H_b)) \tag{4.2}$$

$$D_c = \sum_{k=1}^{w}(1 - \frac{min(h_{ak}, h_{bk})}{max(h_{ak}, h_{bk})}) \tag{4.3}$$

where c is the channel of the kernel. For instance; HOF, Log-C and Cuboid are the features given as an input to the Multi-Channel Kernel. In this case, c_1 is the HOF, c_2 is the Log-C and c_3 is the Cuboid feature. H_a is the histogram of the ath video (bag-of-words aforementioned). Each histogram has w dimensions. w is the number of words. h_{ak} is the number of kth word in the the histogram ath video. Hence, according to this kernel function, three features are passed to the kernel as input. After that, for each feature, *channel*, similarity between every H_a and H_b is computed. Thus, non-linear decision boundary, *hyperplane*, is computed to classify the data points.

$$\underbrace{H_a}_{\text{channel 1}} + \underbrace{H_b}_{\text{channel 2}} + \underbrace{H_c}_{\text{channel 3}}$$

In Multi-Channel Kernel SVM, all the features are given equal weights when they are combined in a single vector. Equal weighting ignores the relative importance of each descriptor used in the classification. If, for example, a walking activity is performed in a video, motion-based features are expected to be more discriminatory than color gradient-based features. In such a case, motion-based features should be given more importance than other and a kernel which combines motion-based and color gradient-based features should give more weight to motion-based features.

In spite of equally weighting different features, multi-channel kernel is expected to show better classification accuracy since it makes use of different features together, instead of using a single feature. A single feature can only capture limited types of activities or provide limited information from an activity. On the other hand, multiple features can capture different aspects of information from various types of features. In [47] and [17], when a single feature is used, classification accuracies are less than when multi-channel kernels are employed.

4.2.1 Multi-Channel Kernel Types

In this thesis, four types of kernels are employed for combining different features: Histogram Intersection Kernel, Gaussian Kernel, Modified Histogram Intersection Kernel [47] [17]. Histogram Intersection kernel function is as follows:

$$HK_{int}(X,Y) = \sum_{i=1}^{j} \min x_i, y_i \tag{4.4}$$

where j is the dimension number of X feature vector histogram. x_i is the values of the ith bin of the histogram. Gaussian Kernel function is defined with the equation below:

$$GK(x,x') = \exp(-\frac{\|x - x^2\|}{2\sigma^2}) \tag{4.5}$$

Modified Histogram Intersection Kernel in [47] is as follows:

$$K(X,Y) = exp(-\sum_c D_c(H_a, H_b)) \tag{4.6}$$

$$D_c = \sum_{k=1}^{w}(1 - \frac{min(h_{ak}, h_{bk})}{max(h_{ak}, h_{bk})}) \tag{4.7}$$

Finally, Modified Histogram Intersection Kernel in [17] is defined below:

$$K(X,Y) = exp(-\sum_c D_c(H_a, H_b)) \tag{4.8}$$

$$D_c = 1 - (\frac{\sum_{k=1}^{w} min(h_{ak}, h_{bk})}{\sum_{k=1}^{w} max(h_{ak}, h_{bk})}) \tag{4.9}$$

where H_a is the histogram of the ath video. Each histogram has w dimensions. w is the number of words. h_{ak} is the number of kth word in the the histogram ath video. c

is the channel of the kernel such as HOF. Modified Histogram Intersection kernels are different versions of base Histogram Intersection Kernel. In equation (4.7), instead of summing the minimum values, minimum values of histogram are divided by maximum values of the histogram so that a kind of normalization is performed. On the other hand, in equation (4.9), summation of minimum values are divided by summation of the maximum values of the histogram.

4.3 Multiple Kernel Learning

In this section, we explain Multiple Kernel Learning (MKL), which is employed in this thesis.

As a general practice in vision applications, a predefined parametric kernel is employed and the parameters of the kernel function are specified by cross-validation. In Multi-Channel Kernels, a number of features using a single kernel or multiple kernels are combined with predefined rules. On the other hand, MKL optimizes this fusion operation by the procedure and fuses different features and kernels in an optimal way:

- MKL method selects the best kernel and feature combination. Each kernel represents different type of similarity computation. While traditional SVM approach tries to find the optimum kernel MKL selects the best fitting kernel.
- While each single kernel represents a different similarity, each feature represents the different aspect of information. Therefore, in order to combine different information, MKL combines all these features and kernels.
- A single kernel may have bias towards a feature, but in multi-kernel approach, this bias can be eliminated to some degree.

Weight of each kernel are determined while the model is being trained. Then, these weights are used in the final classifier. As this process is done during training using the training data, it can be called a data-driven feature selection process:

$$\{(x_i, y_i)\}_{i=1}^{L}, x_i = \{(x_{i,1}, x_{i,2}, x_{i,3}, ..., x_{i,M})\}, y_i \epsilon \{1, -1\} \tag{4.10}$$

where $x_{i,m}$ are the feature vectors, m is the number of features and y_i are the class labels. Also, feature vectors may have different dimensions.

For each feature $m = \{1, 2, ..., M\}$, a kernel function $K_m(x_i, x_j)$ computes the pairwise similarity difference. Thus, we have a total of M kernels:

$$\{K_m\}_{m=1}^{M}$$

In order to optimize the coefficients, learning equation is as follows:

$$f(x) = \sum_{i=1}^{L} \alpha_i \times y_i \times K(x, x_j) + b \tag{4.11}$$

As a result of all kernel computations of each pairwise similarities, a $L \times L$ square kernel matrix is generated. Then, SVM processes this square kernel matrix.

MKL optimizes the kernel weights with the following equations:

$$K(x_i, x_j) = \sum_{m=1}^{M} c_m \times K_m(x_{i,m}, x_{j,m}), c_m \geq 0 \tag{4.12}$$

$$\sum_{m=1}^{M} c_m = 1 \tag{4.13}$$

In MKL approach, not only different features, but also a number of different kernel combinations can be employed. Therefore, for example, HOF feature can be used with different kernels and also a kernel can be used with different parameters. Since each descriptor represents a different aspect, an effective combination of these kernels and features, is expected to show better classification performance. For this purpose, as opposed to equal weighting of multi-channel kernel SVM, MKL assigns different weights to each kernel and feature combination in a self-optimized setting.

In the Figure 4.1, there exists MKL learning scheme where there are kernels from Kernel 1 to k and features from Feature 1 to n. Each kernel and feature combination composes a classifier. For example, when feature 1 is processed with a kernel function, this is called as a classifier. There are classifiers from classifier 1 to c with their weights

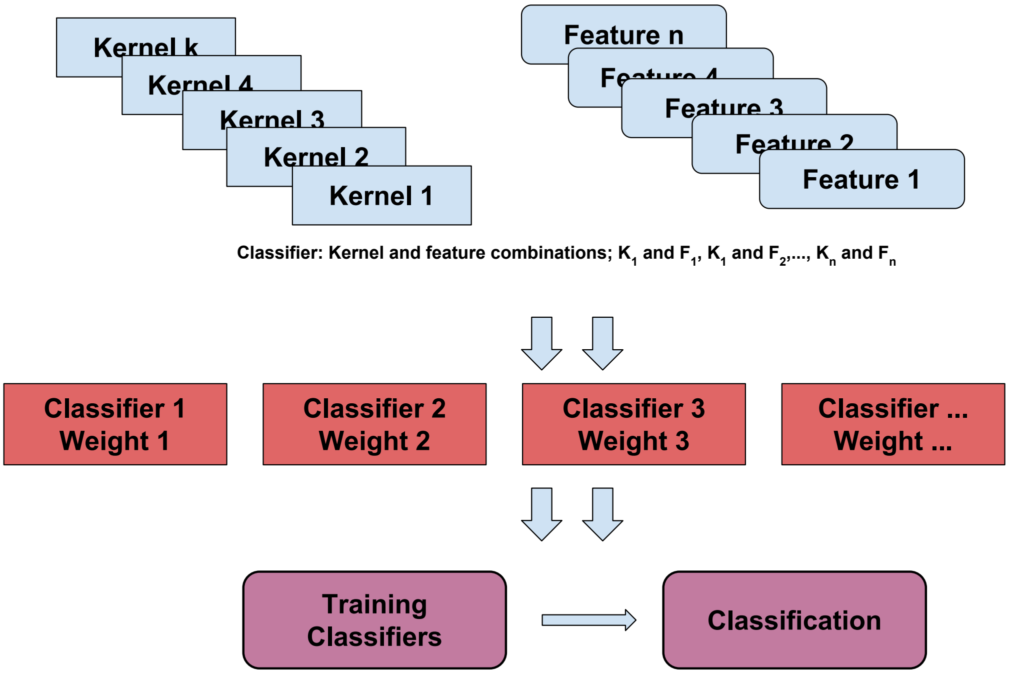


Figure 4.1: Multi kernel learning scheme

from weight 1 to c, classified in such a way. All these classifiers are learned in training. Finally, prediction is performed.

4.3.1 SimpleMKL

In some situations, more flexible representation of the data using more than one SVM kernel can be preferred. Each kernel can employ one type of feature or all sets of features together. In order to combine multiple features and kernels and improve the accuracy, a number of multi kernel learning techniques exist. One of these techniques is SimpleMKL [42].

In order to solve MKL problem, SimpleMKL employs the linear combination of the multiple kernels using gradient descent-based SVM solver iteratively. For the optimization purpose, SimpleMKL uses mixed-norm regularization. It can also be used for regression or multi-class problems. There is an available package of SimpleMKL implementation. It is used in this thesis for the multiple kernel learning implementation.

4.4 Boosted Multiple Kernel Learning

There are a variety of studies in order to improve the optimization process of the multiple kernel learning. Also, the MKL classifiers proposed are extensions to the single kernel classifier. There exist several linear combination of multiple kernels which do not solve complicated patterns. Although it is a new approach that makes the optimization process more efficient, SimpleMKL is not computationally cost effective.

In this thesis, we use boosted Multiple Kernel Learning [62] which exploits the idea of AdaBoost for the first-person videos. Boosted MKL handles the aforementioned limitations of the traditional MKL approaches.

Boosted MKL aims to learn the final strong classifier through training the weak classifiers. In the Algorithm 1, the details of the Boosted MKL is explained. Since the AdaBoost is integrated to the MKL problem, Boosted MKL works through trials. In each trial, each weak classifier is trained to learn the model. Each weak classifier is composed of a kernel with specific parameters and a feature set. Feature set has all features or a subset of features.

For the activity recognition from first-person videos, boosted MKL randomly selects each video according to its probability. Initial probabilities are determined using uniform distribution so that initially each video has equal probability in the first state. Then, at each trial, all classifiers are trained and the winner classifier is determined at the end of the trial based on their performances:

$$e_t = \sum_{l=1}^{L} P_t(i)(c_{t,m}(x_i \neq y_i))$$

Winner classifier of the trial is assigned a weight according to the following equation:

$$w_t = \ln \frac{1 - w_t}{w_t}$$

At the end of the trial, probability of each classifier is updated:

$$P_{t+1}(i) = P_t(i) \times \begin{cases} e^{-w_t}, c_{(x_i)} == y_i \\ e^{w_t}, c_{(x_i)} \neq y_i \end{cases}$$

```
Algorithm 1: Boosted Multiple Kernel Learning Algorithm
Input  : $(x_1, y_1, ..., (x_L, y_L))$ Training set;
         $K_m(x_t, x_n)$ Kernel function;
         $c_n$ $n$th classifier;
         $C$ number of classifiers;
         $M$ number of kernels;
         $T$ trial number;
         Initial set of probabilities:
         $P_1(i) = \frac{1}{L}$, $i$=1..$L$
Output : Kernel weight vector $w_{t,n}$ $n^th$ classifier of trial $t$;
         Output labels of the videos computed based on
 $sign(\sum_{t=1}^{T} w_{t,n=1:K}, c_{t,n=1:C})$
for $t = 1 : T$ do
    Select $n$ videos based on set of probabilities $P_t$;
    for $m = 1 : M$ do
        Train each weak classifier using $K_m$;
        Compute the error based on $P_t$:
        $e_t = \sum_{l=1}^{L} P_t(i)(c_{t,m}(x_i \neq y_i))$
    end
    Select the classifier that gives the minimum error $e_t$
      between errors of all trials:
    $e_t = \min e_{t,m}$
    Update weights of classifiers:
    $w_t = \ln \frac{1 - w_t}{w_t}$
    Update $P_{t+1}(i)$:
    $P_{t+1}(i) = P_t(i) \times \begin{cases} e^{-w_t}, c_{(x_i)} == y_i \\ e^{w_t}, c_{(x_i)} \neq y_i \end{cases}$
end
```

After all T trials, T classifiers and their weights are determined. Thus, final strong classifier is composed of T weak classifiers. In the prediction stage, T classifiers vote a label for each video based on its weight. A final decision is made and the label of the video is determined based on voting.

In the Figure 4.2, classifiers are assigned weights. Weights are summed for the final decision. Finally, the final classifier outputs a label. In addition, Figure 4.3 shows the

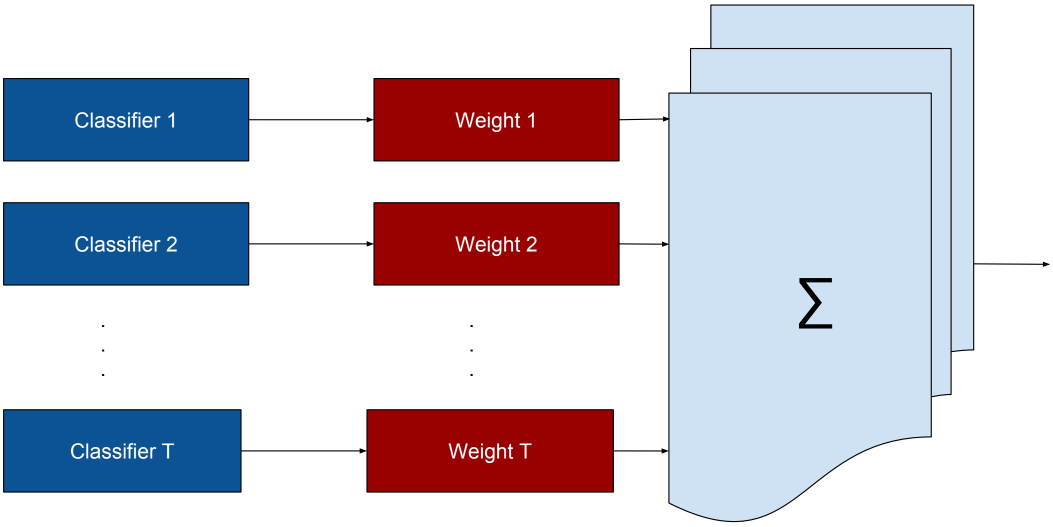


Figure 4.2: Boosted MKL

hyperplanes of weak and final classifiers. Each of the decision boundary of the weak classifiers classifies the data points, but with some misclassification error. On the other hand, final classifier that is composed of weak clasifiers with their weights classifies more accurately the data points since the decision boundary of the final classifier is updated based on the weak classifiers' weights.

After all, MKL and Boosted MKL provide the combination of features and kernels solution to the problem and have the following advantages:

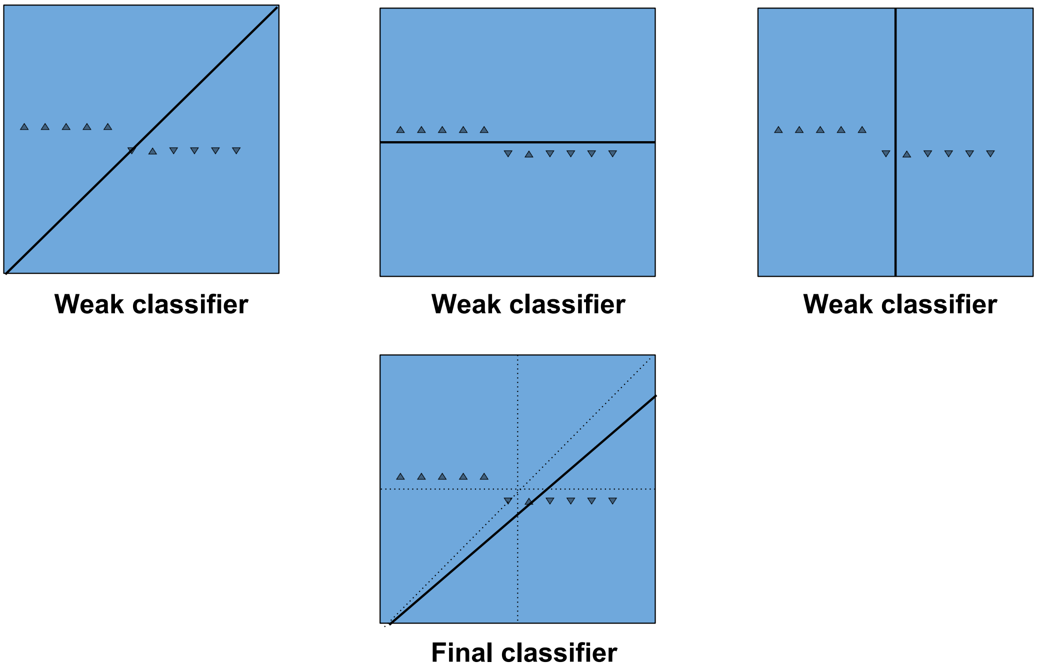


Figure 4.3: Boosted MKL weak classifiers

- Different kernels could be used with all or subset of features.
- Different types of kernels such as Gaussian, Linear and different parameters for the kernels can be employed together.
- Each kernel and feature combination is assigned different weights based on its importance.

CHAPTER 5

EXPERIMENTAL RESULTS

We presented single and multi-channel kernel SVM, MKL and Boosted MKL methods in previous chapters. In addition, we discussed the features which we used in this thesis. We apply *MKL* and *Boosted MKL*, for the first-person activity recognition which allows integrating multiple features in a data-driven adaptive manner as opposed to the previous studies. In this chapter, we present the experimental evaluation of the proposed approaches and compare the results against the other methods in the literature.

5.1 Outcome Measurements

We performed experiments in order to evaluate the single and multi-channel kernel, MKL and Boosted MKL methods [47] [17] based on classification accuracy for first-person activity recognition. In order to apply MKL, SimpleMKL library has been used. Experiments of these four methods were conducted on segmented videos of JPL-Interaction [47] and DogCentric activity [17] datasets.

In all experiments, all descriptor combinations were used for both datasets. HOF, Log-C and Cuboid descriptors were employed individually in each experiment using traditional single kernel approach. In multi-channel kernel, MKL and Boosted MKL experiments, HOF and Cuboid together, and HOF, Log-C and Cuboid together were employed. For both dataset, Gaussian, Histogram Intersection (H-Int) and a modified Histogram Intersection kernels (DC-Int) [17] were used. Also, another modified Histogram Intersection (JPL-Int) [47] kernel was used for JPL-Interaction dataset.

Histogram of optical flow descriptor is constructed in each s-by-s grids of cells in each frame, as previously mentioned. For JPL dataset, each frame is divided into 9-by-9 grids (s-by-s) while constructing descriptors. On the other hand, s value is 3 for DogCentric dataset. S values differentiate in each dataset because of different characteristics of videos such as ego-motion and illumination. In DogCentric dataset, much more ego-motion is seen than JPL dataset. Therefore, in order to prevent the bad effects of noisy data, we used smaller number than JPL dataset for s value. Number of cluster is another varying factor in both dataset. Motion characteristics of dogs affect also cluster number. There occur more optical flow vectors with varying magnitude and direction, illumination changes in DogCentric dataset. Therefore different number of feature structures cause different number of clusters. In order to capture the variations in videos of DogCentric dataset, we assign 350 as K value for clustering, whereas the K value is 150 for JPL dataset.

5.2 JPL-Interaction Dataset Activities

In this section, the activities in the videos of the JPL-Interaction dataset and the experimental setup are discussed in detail.

There are 7 unique activity types and 84 videos in JPL-Interaction dataset. Experiments of each method were repeated for 100 times and the results were averaged. At each iteration, 9 training and 3 testing videos of each activity type were randomly selected. Thus, 84 videos were divided into two groups of training and testing videos: 63 videos for training and remainder 21 videos for testing.

In Figure 5.1, two sample snapshots from each of the 7 unique activity types [47] are shown. These activity types are: Shaking the hand, throwing, waving, hugging, petting, pointing and punching.

Figure 5.1 (a) and (b) show the shaking the hand activity. Shaking activity is composed of two sub-actions. In the first part, a person walks toward the wearer. Then he shakes the wearer's hand. This activity creates a little ego-motion. In the Figure 5.1 (c) and (d) throwing activity occurs. Throwing action happens twice in the video. Again, the ego-motion is observed. Figure 5.1 (e) and (f) shows the waving activity snapshots. The person waves to the wearer. During this activity, nor touching neither ego-motion occurs. In the Figure 5.1 (g) and (h) hugging activity takes place. This activity naturally involves touching between the person and the wearer. The person walks towards the wearer, first shakes and then hugs the wearer. A considerable amount of ego-motion is observed during the activity. Figure 5.1 (i) and (j) show the petting activity. In the first part, the person approaches and holds the wearer. In the second part, the person pets the wearer. Similar actions occurs in both parts and an increasing ego-motion is created from beginning to the end of the activity. Pointing activity is seen in the Figure 5.1 (k) and (l). There are two persons standing at a distance and they point to the wearer. This activity does not contain much motion since the two persons are constant throughout the video. Last two figures show the punching activity. The person walks toward the wearer as in the some previous activities and punches the wearer. There occurs large amounts of ego-motion as a consequence of the punching.

Some of the activities are similar in terms of their characteristics. Shaking, hugging, petting and punching activities are composed of two sub-actions. Commonly, the person walks toward the wearer at first, and then the person performs the particular action. These activities commonly create ego-motion. During these activities, the person doing the action gets close to the wearer and the person takes a large part in the scene. Pointing activity is not similar to the other activity types since there are persons which stand and no movement occurs. It is a difficult activity for the classifier since there are not enough discriminative patterns. Also, petting and hugging activities are similar to some degree. There is a person who dominates the scene. The person touches the wearer in both activities. In addition, the person shakes the wearer using his hands. Hugging activity differs from the other activity near the end of the activity while the person moves his head towards the wearer. Waiving activity is another different activity because no touching occurs. The person stands and waves his hand. During his activity, the person is very close to the wearer. Therefore, it contains some similarity with petting and hugging because of their proximity.

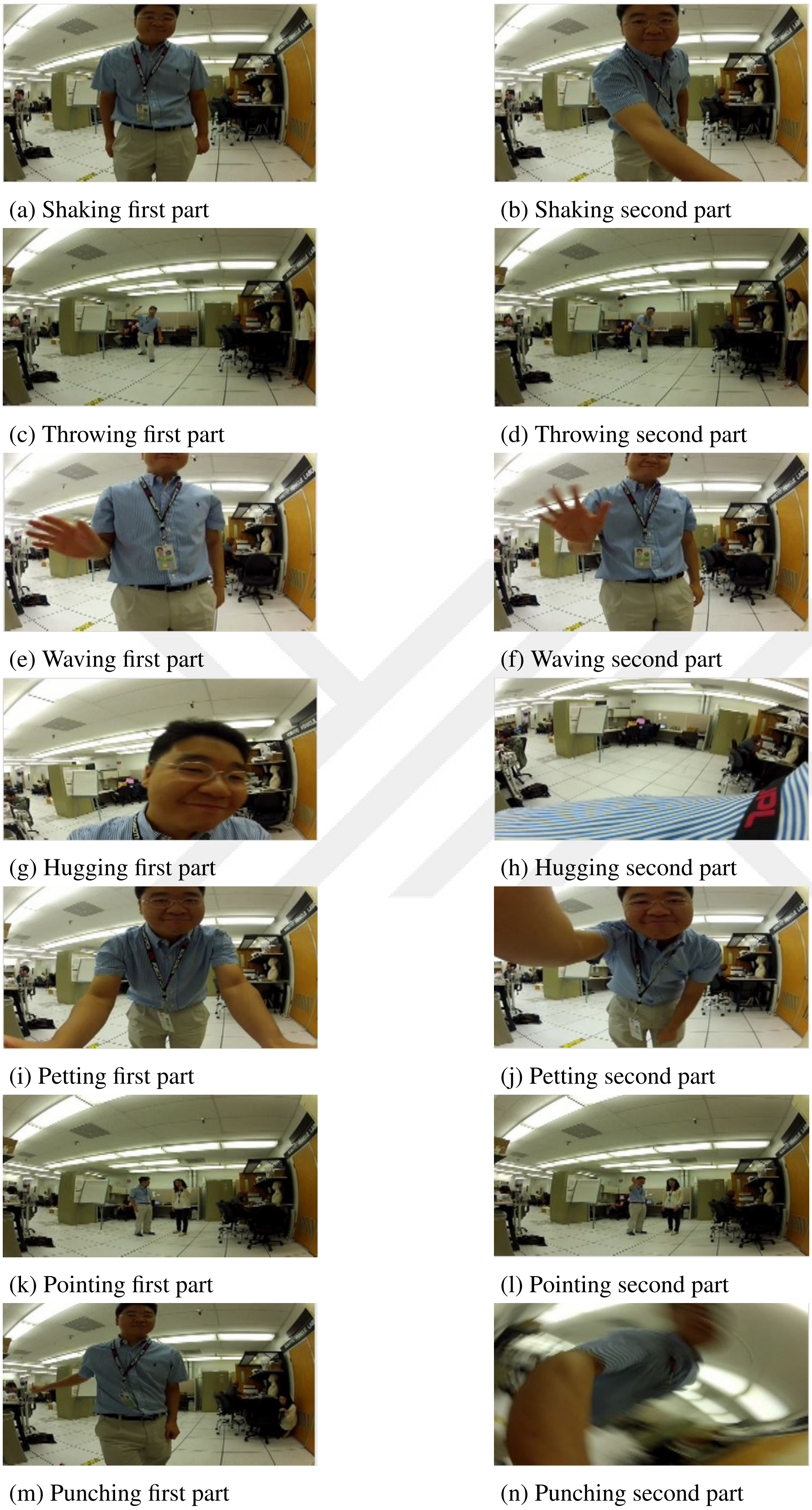
5.3 DogCentric Dataset Activities

In this section, the activities in the videos of the DogCentric activity dataset and the experimental setup are detailed. There are 209 videos in DogCentric activity dataset. 209 videos contain totally 10 unique activities. Experiments of each method were repeated for 100 times and the results were averaged. At each iteration, half of the videos were randomly selected for training and the remainder half of the videos are selected for testing. Thus, for each iteration, the videos were split into two halves of the training and testing.

In the Figure 5.2, there are snapshots of the 10 unique activity types from [17]. 10 unique activity types are: Playing with a ball, walking, sniffing, shaking the body, petting, turning right, turning left, feeding, drinking water and waiting for a car. A snapshot for each activity type is seen in the figure.

A GoPro camera is mounted to the back of each of the 4 dogs. Under different environmental conditions, 4 dogs are took by their owners for walking. Videos are captured indoor, outdoor, near a road with traffic etc.

Figure 5.2 (a) shows the playing with a ball activity. During this activity, the dog runs after the ball and shakes the camera which is attached to its back. There occurs a huge ego-motion. In the Figure 5.2 (b), there is a walk with its owner. They walk on the street. There are a lot of walking videos that some of them contain much more ego-motion since some of dogs are with collar but remainders are not. If a dog has a collar, then it's movement more smooth than the dogs without a collar. In (c), there is a snapshot of the sniffing activity. During this activity, the dog walks for a while. Then, it stops and starts sniffing. Therefore, this type of activity is composed of two parts. In the first part, it is similar to walking and second part contains a different pattern. (d) shows the snapshot of shaking its own body activity. The dog suddenly starts shaking its own body, while it's walking. So, the activity has two parts: walking and shaking. In the first part, there is a consistent and constant type of ego-motion, whereas the second part contains different types and huge ego-motion until the end of the activity. In Figure 5.2 (e), there occurs a petting activity. A person pets the dog, after the dog walks toward the person. This activity is performed indoor and outdoor. In some videos, the dog walks toward the person firstly, but in other videos, the dog stands and a person pets it immediately. (f) and (g) shows the similar activities. A dog looks at the left in the first figure, and looks at the right in the other figure. A person feeds the dog in indoor or outdoor in the Figure 5.2 (h). While feeding, the dog reaches the person out. Especially in outdoor, illumination changes throughout the video since the camera which is attached to the back of the dog shows the sky while the dog reaches out the person. In (i), the dog drinks water from a dish. There are indoor and outdoor capturing of this activity. The activity is simpler than other activities since it's a monotonous activity and contains a little ego-motion. In a traffic,



(a) Shaking first part

(b) Shaking second part

(c) Throwing first part

(d) Throwing second part

(e) Waving first part

(f) Waving second part

(g) Hugging first part

(h) Hugging second part

(i) Petting first part

(j) Petting second part

(k) Pointing first part

(l) Pointing second part

(m) Punching first part

(n) Punching second part

Figure 5.1: Snapshots of each activity types in videos from [47]

the dog waits for a car in order to pass it by in the (j). It's an outdoor activity. The dog does not move much in this type of the activity. Therefore, ego-motion is little during this activity.

Some of the activities are similar in terms of their characteristics. Playing with a ball and shaking activities are similar because of huge ego-motion. While the dog plays with a ball, it runs and naturally shakes itself. This creates vast amount of ego-motion like the shaking activity. The dog shakes off while it walks. In spite of the similarities to some degree, there are certain differences between these activities. For example, playing with a ball activity causes complicated movements, whereas shaking activity causes smooth circular movement.

Looking left, right and waiting for a car activities contain little body movement, but only head movement. The dog moves its head in order to look to right, left and cars. Therefore, the movement pattern and ego-motion is similar during these activities. Drinking and sniffing activities are also similar due to the head movement. The dog stops and looks down while both drinking and sniffing. In addition, walking and sniffing activities are similar to some extent since the dog generally walks before the sniffing. So, it does the same movement before sniffing with the walking activity. Feeding and petting are also similar activities because a person interacts with a dog in both activities. In this context, the dog reaches a person out and stands before the person.

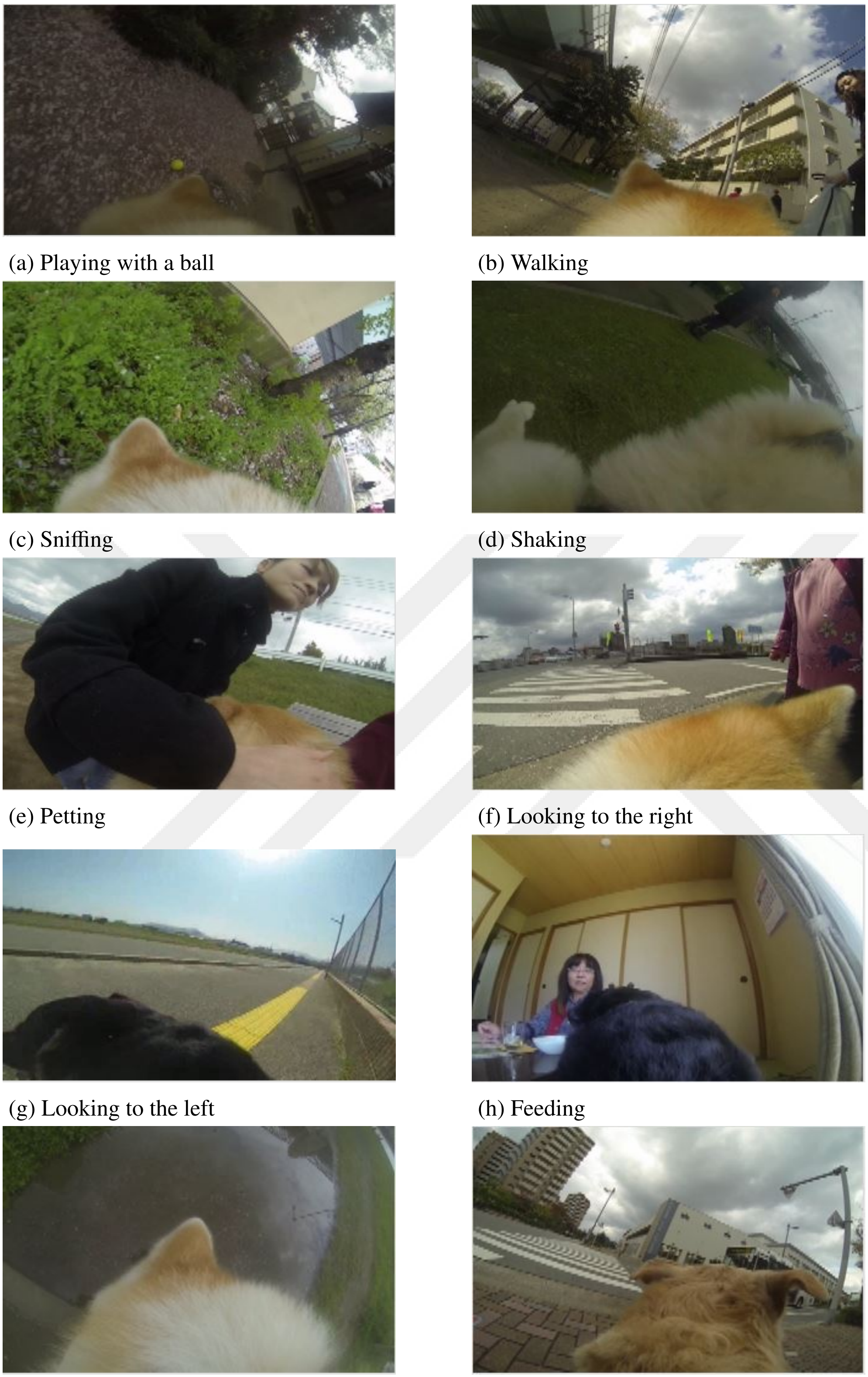
5.4 Video Properties

Videos in JPL-Interaction dataset have a resolution of 320 x 240 at 30 frame per second (fps). This dataset include friendly, hostile and neutral interactions with the observer. Friendly activities are "*petting*", "*hugging*", "*waving*" and "*shaking hands*". Neutral activity is the "*pointing to the observer*". "*Punching*" and "*throwing*" are the hostile activities. These videos are captured by a head mounted camera as shown in the Figure 5.3 (a).

DogCentric activity dataset contain videos with 320 x 240 resolution at 48 fps. A GoPro camera is attached to the back of the dog as in the Figure 5.3(b). There are both indoor and outdoor videos. Videos are recorded in a traffic, along a river or in a park. 4 different dogs having different owners take part in the videos. Each of the dogs performs the same activity but the environment changes in each activity.

5.5 Comparison of Two Datasets

Video characteristics in these two datasets are different. In JPL-Interaction dataset, the camera is attached to the head of the fixed humanoid model as in the Figure 5.3(a). Persons in the videos, interact with the humanoid model. Though the viewpoint of the camera is similar to a human, the camera does not move since the observer is stationary, unless a person or an object interacts with the observer. Therefore, these videos have less ego-motion and less dynamic characteristics relative to the Docentric



(a) Playing with a ball

(b) Walking

(c) Sniffing

(d) Shaking

(e) Petting

(f) Looking to the right

(g) Looking to the left

(h) Feeding

(i) Drinking

(j) Waiting for a car

Figure 5.2: Snapshots of each activity types in videos from [17]

activity dataset. Cameras are attached to the back of a dog in DogCentric dataset. In these videos, dogs are already moving as opposed to the humanoid model. In addition, dogs as a course of their nature are more brisk than humans. Their movement are rough and juddery whereas movement of a human is smoother. Also, dogs could move and react to something suddenly. Consequently, the nature of the movements of a dog bring several additional difficulties. Furthermore, DogCentric dataset contain not only indoor but also outdoor activities as opposed to JPL dataset. So, there are much more illumination, color etc. changes in the videos that are captured by the cameras on the dogs.

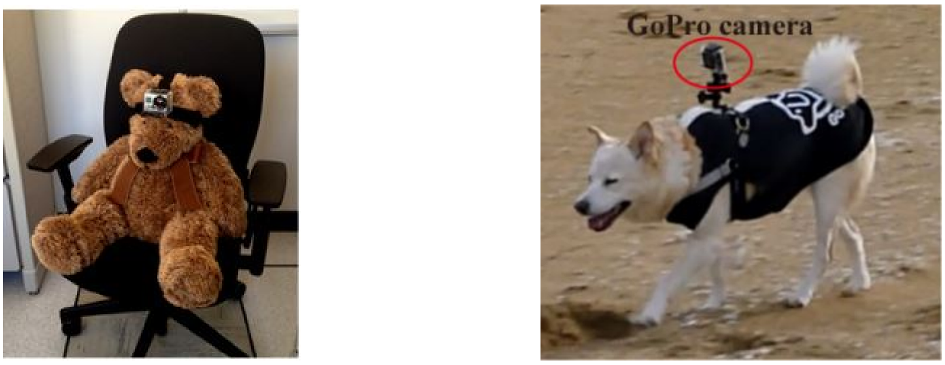
In JPL dataset, the observer is involved in the events implicitly through a person interacting with the observer. On the other hand, dogs are directly involved in the events themselves. For example, they play with a ball or walk towards a person. Also, since the camera is attached to the back of the dog, the view includes some parts of the dog's body, but no part of the humanoid model is visible in the videos.

In DogCentric dataset, there are 4 subjects whereas a single subject is used in JPL-Interaction dataset. Since a single humanoid model is employed in the first dataset, this does not bring any additional variance in videos. On the other hand, since there are four different dogs in the second dataset, movement patterns and body shapes of the dogs change among different videos.

In the Table 5.1, number of occurrences of specific activities are shown. Since there are 4 dogs, activities are firstly categorized according to the dogs. Number of samples for each activity type is different for different dogs. On the other hand, for JPL-Interaction dataset, there are 12 samples for each activity type.

5.6 Discussion of the Results

In the following subsection we discuss the results of experiments which are performed in JPL and DogCentric datasets.



(a) JPL-Interaction Observer Setup. This figure is taken from [47].

(b) Dogcentric Observer Setup. This figure is taken from [17].

Figure 5.3: Observer setups

5.6.1 JPL-Interaction Dataset Results

Figure 5.4 shows the confusion matrices obtained using different base features and their combinations using DC-Int kernel in JPL-Interaction dataset. Confusion matrices in Figure 5.4 (a) to (c) belong to base features, HOF, LogC and Cuboid respectively. Inspection of these matrices reveal that the base features complement each other. In Figure 5.4 (a), HOF feature shows good performance (above 80%) for “hug", “wave", “point" and “punch" activities, whereas Log-C has classification accuracy above 80% for “shake", “wave", “point" and “punch" activities. Also, Cuboid feature’s performance is higher than 80% for “point", “punch" and “throw" activities. Log-C feature is more stable than other features based on the accuracy results since it has higher than 70% performance for all activities but there are two activities for Cuboid and three activities for HOF below 70% accuracy. False positive rates are also differentiate for each feature. With HOF feature, false positive rate of “pet" activity is 0%, but this rate is 7.7% and 4.7% with Log-C and Cuboid features respectively. “Punch" activity is the common for Log-C and Cuboid features according to false positive rate (0%) but HOF feature shows high false positive rate (34.3%) for this activity. Log-C feature gives at the most 14.6% false positive rate for features which it outputs high classification accuracies. On the other hand, HOF gives at least 19% false positive rate for features which it performs good performance. Also, Cuboid 47.7% false alarm rate for the “throw" activity whereas it shows 80% accuracy for the same activity.

As shown in Figure 5.4 (a), (b) and (c), each feature is distinctive for a set of activities. Each feature perform well or poorly for different set of activities. Therefore, combinations of these features are expected to perform better than single features. Figure 5.4 (d) and (e) verify this expectation. When HOF and Cuboid features combination is used with Multi-Channel Kernel, it shows accuracy higher than 80% for “shake", “hug", “point", “punch" and “throw" activities. For example, HOF and Cuboid features perform lower than 80% individually whereas the result is 84% when multi-channel

Table 5.1: Activity Tables

(a) Activity numbers in [17]

Activity	Dog 1	Dog 2	Dog 3	Dog 4	Total
Ball	6	5	3	0	14
Walk	7	1	14	4	26
Sniff	5	2	2	1	10
Shake	7	3	8	7	25
Pet	8	4	3	6	21
Look right	7	2	4	5	18
Look left	8	4	8	5	25
Feed	8	2	3	5	18
Drink	8	7	7	5	27
Car	7	4	7	7	25
All	71	34	59	45	209

(b) Activity numbers in [47]

Activity	Total
Shake	12
Throw	12
Wave	12
Hug	12
Pet	12
Point	12
Punch	12
All	84

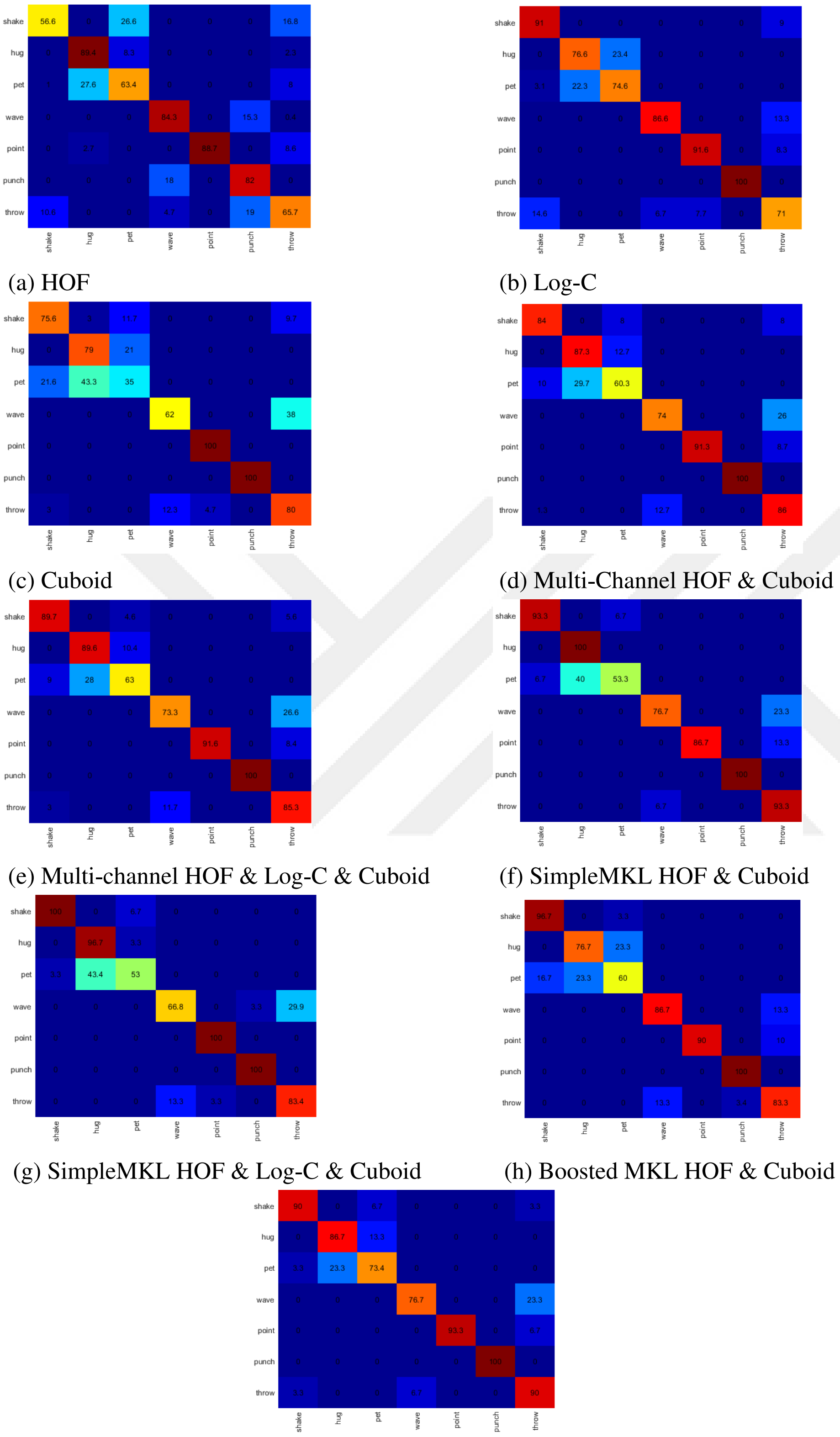
kernel is employed. Combined feature set also performs better than individual features for “pet" and “throw" activity. When HOF, Log-C and Cuboid features are used together, results of the combination is better than both HOF & Cuboid combination and individual features. Only for “wave" and “throw" activities HOF & Log-C & Cuboid feature set performs worse than HOF & Cuboid feature set. The results are 73.3% and 85.3% with 3 features and 74% and 86% with 2 features for "wave" and "throw" activities respectively.

Figure 5.4 (f), (g) show SimpleMKL results using HOF & Cuboid and HOF & LogC & Cuboid repectively. Boosted MKL results for the same feature sets are shown in the Figure 5.4 (h) and (i). Multi-channel kernel performs better than individual features but it assigns equal weight for each feature regardless of its importance. This rudimentary weighting mechanism prevent exploiting features well. Best accuracy results are acquired when one of the MKL methods are employed except for “pet" activity. For 4 of all activities, SimpleMKL HOF & Cuboid feature set performs higher than 93% whereas SimpleMKL HOF & Log-C and & Cuboid feature set performs higher than 95% accuracy for the same activities. Boosted MKL shows highest performance in respect to the overall performance. When Boosted MKL is employed, higher increases in accuracy values are observed than SimpleMKL results. For instance, SimpleMKL shows lower than 70% accuracy for “pet" and “wave" activities whereas Boosted MKL shows higher than 70% accuracy for all activities.

None of the methods perform well for “pet" activity. Both petting and hugging activities contain touch and close interaction between the person and the humanoid model. This creates a difficulty for the classifiers and prevents them to be successful in recognizing this particular activity.

In Figure 5.5, the confusion matrices of the base and combined features using JPL-Int kernel on JPL dataset are seen. According to these matrices, Multi-Channel kernel performance is better than using a single kernel. When multi-channel kernel is employed, recognition accuracies of 5 activity types are higher than 80% with HOF & Cuboid feature combination, whereas accuracy value of only 1 activity type is higher than 80% with HOF feature. For “punch" activity, Log-C and Cuboid features’ accuracies are 100% and 99% respectively. In addition, both of Log-C and Cuboid features show the same performance for “wave" activity. On the other hand, with multi-channel kernel, when all three features are combined it performs worse than two feature combination. For “shake", “hug" and “wave", accuracy values of HOF & Cuboid combination are higher than HOF & Cuboid & Log-C combination. On the other hand, for “punch" activity, accuracy value of 2 or 3 feature combination with multi-channel kernel is 100 %.

As seen in Figure 5.6, Log-C performs better than Cuboid. Also, accuracy values of Cuboid are higher than accuracy values of HOF feature. Furthermore, performance of multi-channel HOF & Log-C & Cuboid feature set is higher than performance of multi-channel HOF & Cuboid feature set. For “hug" and “point" activities, 2 feature multi-channel kernel performs better (93.7% and 99.3%) than 3 feature multi-channel kernel (69.3% and 93%). For all remainder activities, 3 feature set with multi-channel kernel works better. For some specific activities, there are some differences in performance of kernels. When JPL-Int kernel is employed, 3 multi-channel feature combination performs 69% and 2 multi-channel feature combination performs 80% for



(a) HOF

(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

(e) Multi-channel HOF & Log-C & Cuboid

(f) SimpleMKL HOF & Cuboid

(g) SimpleMKL HOF & Log-C & Cuboid

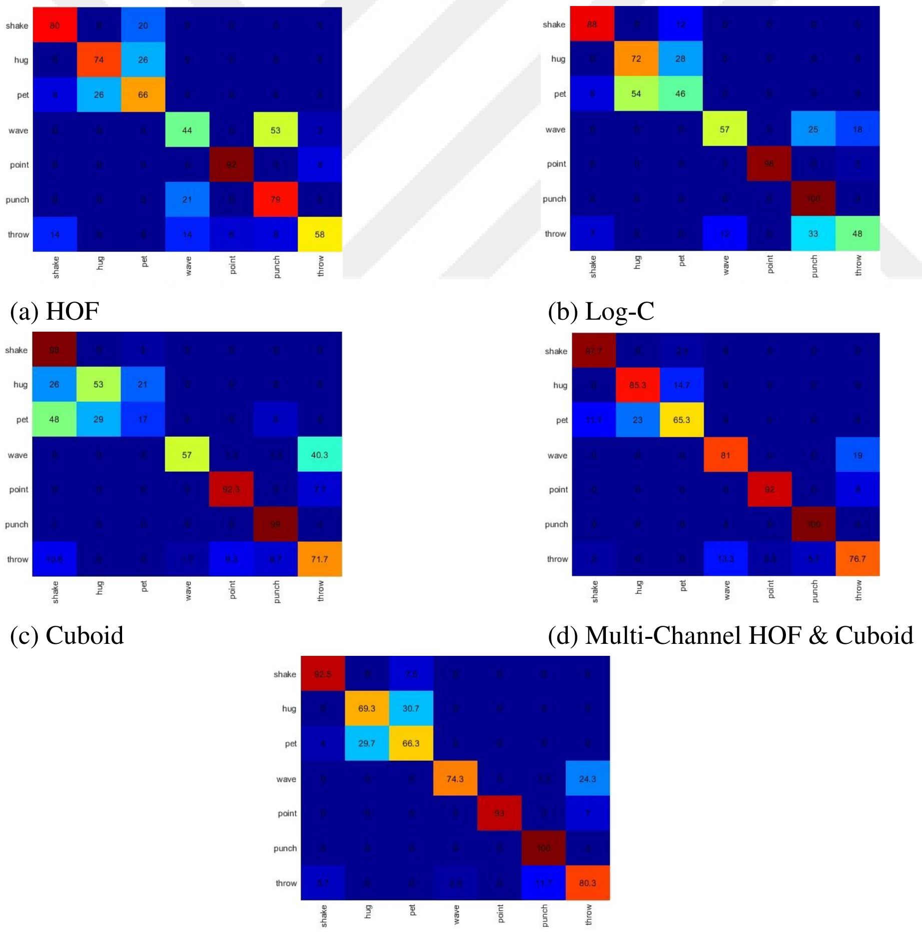
(h) Boosted MKL HOF & Cuboid

(i) Boosted MKL HOF & Log-C & Cuboid

Figure 5.4: The confusion matrices of the base and combined features using DC-Int kernel on JPL dataset, SimpleMKL and Boosted MKL

“hug" activity, whereas these values are 95% and 80.7% with Histogram Intersection kernel. On the other hand, for “pet" activity, JPL-Int kernel’s performance is better than Histogram Intersection kernel. All confusion matrices show us that JPL-Int kernel can handle high amounts of ego-motion better than Histogram Intersection kernel. Accuracy values of JPL-Int kernel for the activities which involve ego-motion such as “punch", “shake" and “throw" are higher than Histogram Intersection kernel according to Figure 5.6 (d), (e) and 5.5 (d), (e). In addition, when there is a close interaction between the person and the humanoid model, such as “pet" and “hug" activities, Histogram Intersection kernel performs better than JPL-Int kernel.

Figure 5.7 shows that, for “wave" and “point" activities, all single features and feature combinations perform well (higher than 75% except for “wave" activity with Log-C feature). Except for 3 multi-channel feature combination, accuracy values of “pet" activity with each feature are lower than 60%. “Point" activity is the activity that persons in the video do not much move and only cuboid feature perform 100% accuracy. Although “punch" activity creates great amount of ego-motion, HOF feature shows the worst performance because huge and sudden ego-motion causes large dispersion of optical flow vectors.



(a) HOF

(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

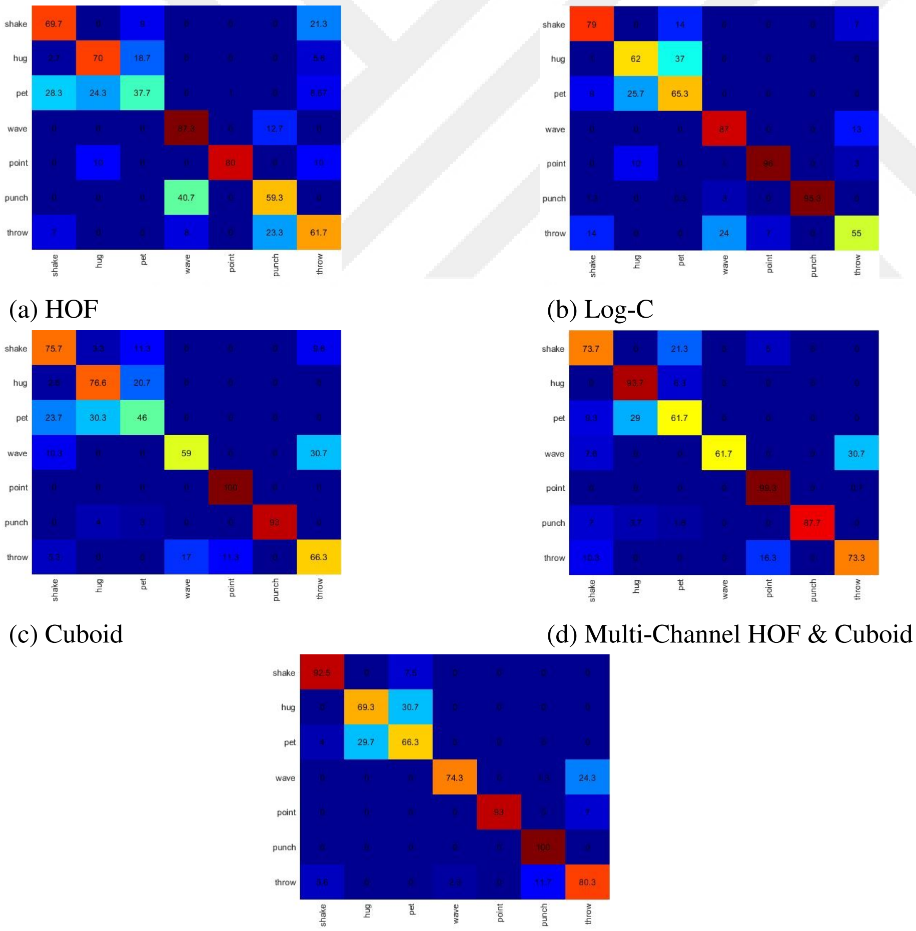
(e) Multi-channel HOF & Log-C & Cuboid

Figure 5.5: The confusion matrices of the base and combined features using JPL-Int kernel on JPL dataset

5.6.2 DogCentric Activity Dataset Results

Base features and feature combinations discriminate different activities on DogCentric activity dataset as similar to JPL-Interaction dataset. Figure 5.8 shows the confusion matrices obtained using different base features and their combinations using DC-Int kernel in DogCentric activity dataset. Figure 5.8 (b) shows that, Log-C feature fails to discriminate “pet" and “feed" activities. In both of these two activities, the dog looks at and interacts with a person. On the other hand, HOF feature is more successful than Log-C feature. An interesting finding is that all base features consistently perform better in classifying “turn left" activity than “turn right" activity. According to the Figure 5.8, “drink" activity cannot be distinguished from other activities. With base or multi-channel kernels, none of the accuracy values for “drink" activity is higher than 12%. During this activity, the dog does not move much and only drinks water from the same viewpoint which makes it difficult for classifiers to find any discriminative information. For the most of the activities, Cuboid feature performs better than other features but it is consistently worse for the “playing with the ball" activity.

The results also show that Multi-Channel Kernel performs better than single kernel



(a) HOF

(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

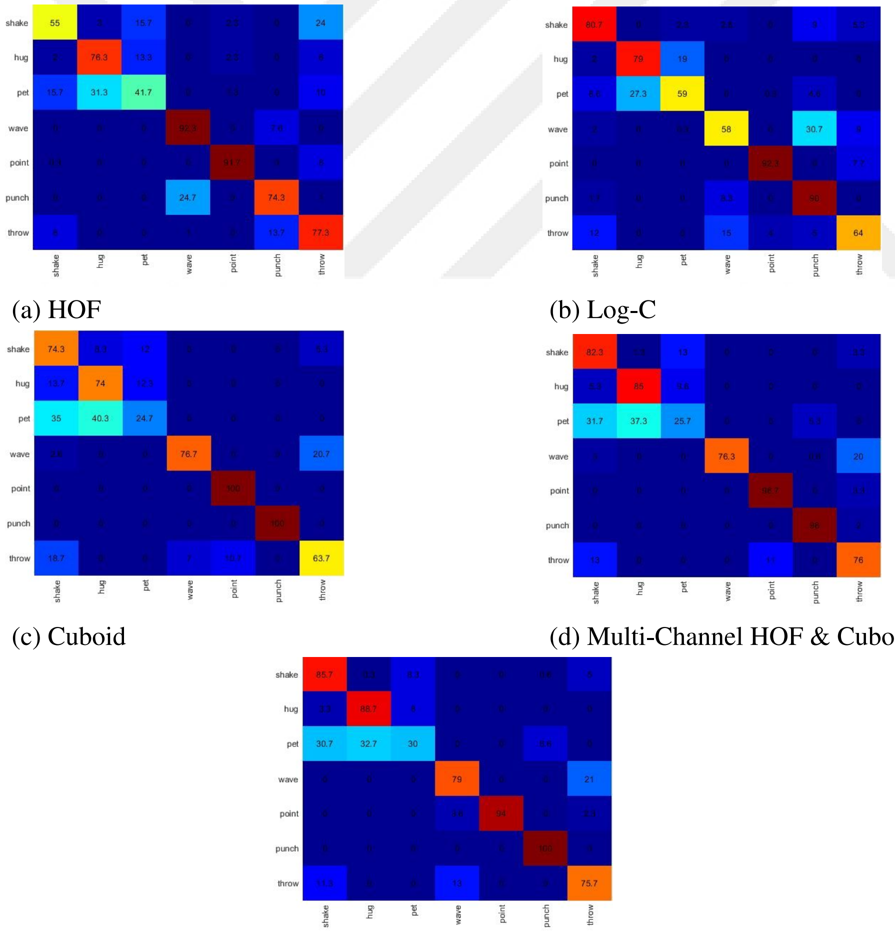
(e) Multi-channel HOF & Log-C & Cuboid

Figure 5.6: The confusion matrices of the base and combined features using Histogram Intersection kernel on JPL dataset

approach for the most of the activities. For “walk", “shake", “sniff" and “turn left" activities. Accuracy values of multi-channel kernel with 2 features is better than accuracy values of all single features. It works better in classifying “pet" activity except for Cuboid, “turn right" and “feed" activities except for HOF feature. On the other hand, multi-channel kernel with 3 features gives higher accuracies than 2 features except for “pet" activity.

SimpleMKL improves the classification performance significantly, according to Figure 5.8 (f) and (g). For example, the accuracy value of “playing with ball" activity is 85.4% with SimpleMKL HOF & Cuboid feature set and 79.7% with SimpleMKL HOF & Log-C & Cuboid feature set, whereas the highest accuracy value of the other approaches is 43.1%. Also, for “feed", “turn left", “turn right" and “pet" activities SimpleMKL methods are better than other single and multi-channel approaches. Although general performance improvement of SimpleMKL methods, the accuracy values of “shake", “sniff" and “walk" activities, which have smooth and slow movement, decrease with SimpleMKL methods.

In the Figure 5.8 it is seen that combining multiple features improves accuracies.



(a) HOF

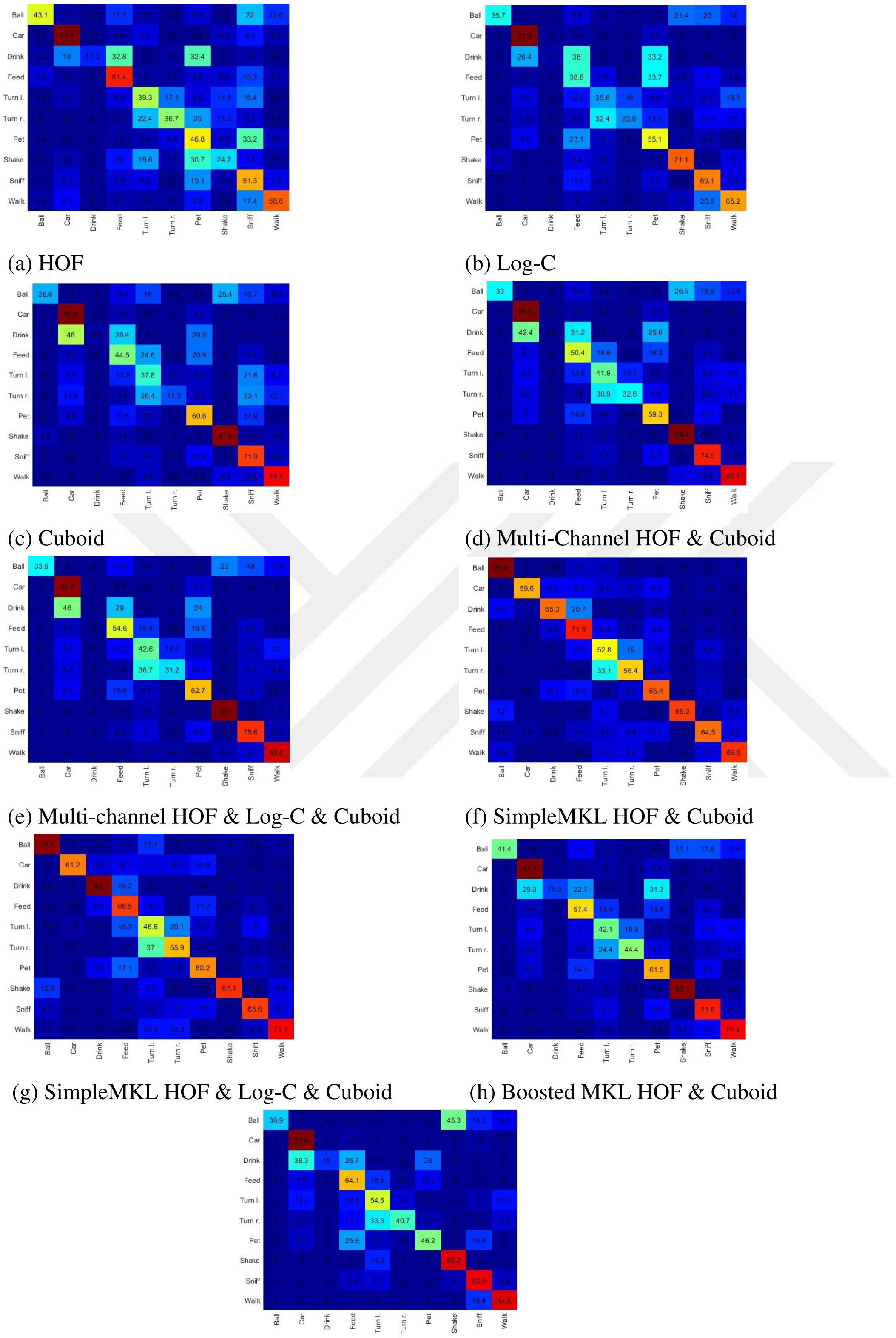
(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

(e) Multi-channel HOF & Log-C & Cuboid

Figure 5.7: The confusion matrices of the base and combined features using Gaussian kernel on JPL dataset



(a) HOF

(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

(e) Multi-channel HOF & Log-C & Cuboid

(f) SimpleMKL HOF & Cuboid

(g) SimpleMKL HOF & Log-C & Cuboid

(h) Boosted MKL HOF & Cuboid

(i) Boosted MKL HOF & Log-C & Cuboid

Figure 5.8: The confusion matrices of the base and combined features using DC-Int kernel, SimpleMKL and Boosted MKL on DogCentric activity dataset

Among the three different methods of combining features multiple kernel learning and SimpleMKL approaches are more accurate than multi-channel kernel approach. When SimpleMKL and Boosted MKL approaches are compared, SimpleMKL gives more accurate results than Boosted MKL approach. “Playing with a ball" is an activity that SimpleMKL classifies more accurate noticeably than Boosted MKL. This fact is also seen for “drink" activity. For some specific activities, effects of local and global features in describing the activities are more distinct. For example “sniff" activity. During this activity the dog moves and creates global motion continuously but several objects such as collar, trees and other persons creates local motion. Therefore, both of local feature (cuboid) and global features (HOF and Log-C) are useful in order to get accurate results. Consequently, for these specific activities, combining the local and global features is more important. For “sniff" activity, cuboid gets higher values than global features. Furthermore, if these features are combined in a multi-channel kernel, then the results are improved by about 4%. If these features are combined in a Boosted MKL approach, then the results are improved almost 8% according to multi-channel approach. On the other hand, SimpleMKL is not so successful in combining the features in order to recognize the “sniff" activity. Cuboid is also the best feature (78%) in order to discriminate the “walk" activity. The worst is HOF (56%) and the second is Log-C (65%).

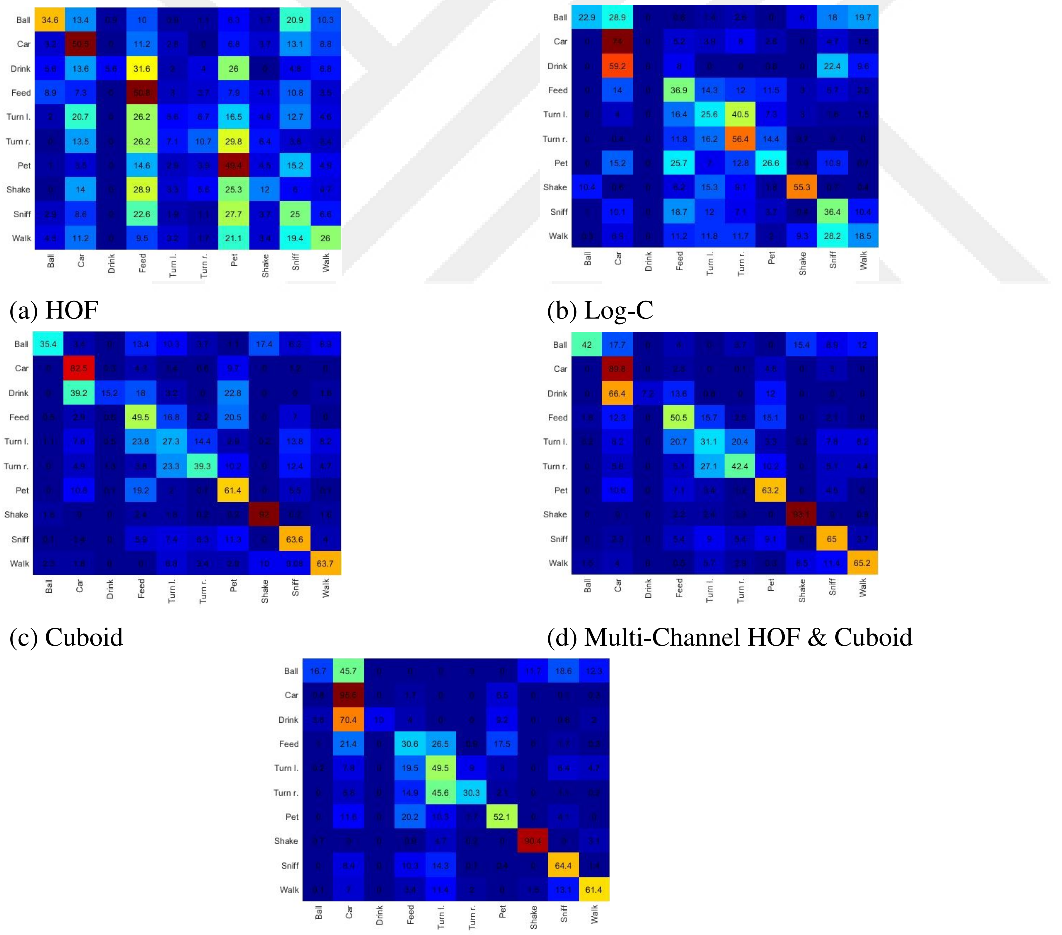
However, Boosted MKL is not the best approach this time, but multi-channel kernel performs better than other approaches. This trend is opposite for “turn left" and “turn right" activities. Global motion features classify these activities more accurately than local feature. This activity has very characteristic motion pattern. The dog only turn its head to left or right. This movement always generates the same pattern and local information is very little. MKL approaches are more effective combining these features than multi-channel kernel approach as opposed to “walk" activity. “Waiting for car" activity also has both local and global information. Cars moving on the road create local motion and the dog creates the global motion itself. Therefore, both global and local features are successful while describing the activity. Also, the most accurate results are seen with Boosted 3 feature MKL approach.

The Figure 5.9 shows the confusion matrices of Gaussian kernel experiments. Gaussian kernel results lead to similar observations to the ones obtained using DC-Int kernel. For “shake" activity, Cuboid is still significantly more accurate than global features. Multi-channel also improves this result by about 1%. For also “sniff" and “walk" activities, HOF and Log-C is even worse than DC-Int kernel whereas Cuboid is the best feature in order to classify these activities in spite of its performance decrease according to DC-Int kernel. Log-C is still more successful than Cuboid feature for recognizing the “turn right" activity. On the other hand, global descriptors are no more successful than Cuboid feature for “turn left" activity. According to the confusion matrices of Figure 5.9, Gaussian kernel is not as successful as DC-Int kernel for classifying the activities when global motion is dominant.

The Figure 5.10 shows the confusion matrices using Histogram Intersecton kernel. For instance, Cuboid descriptor performs still better than global descriptors for “shake", “sniff" and “walk" activities. Also, the multi-channel approach with 3 features still gives more accurate results than single feature cases. For “pet" activity that has local and global motion, multi-channel kernel brings significant improvement. Histogram Intersection kernel and DC-Int kernel show similar performance for “playing with a ball" activity. Global features are better than Cuboid for recognizing the activity with both kernels. On the other hand multi-channel approach does not bring any improvement for this activity. For both “turn left" and “turn right" activities, the best feature is HOF. When multi-channel approach is employed, “turn left" activity is classified more accurately. However, the multi-channel approach does not show the expected performance for “turn right" activity.

Among all single and multi-channel kernel approaches, Histogram Intersection kernel is the most successful approach, according to the confusion matrices above. On the other hand, Boosted MKL and SimpleMKL perform better than Histogram Intersection Kernel which is employed in multi-channel or single-kernel approaches.

Table 5.2 and 5.3 show the most successful approaches on both dataset. In these tables, overall performance of the approaches is seen. According to the Table 5.2, none of single kernel approaches are among the most successful approaches on DogCentric activity dataset. In addition, more than half of the most successful approaches are MKL methods and only 4 of them are not MKL. All of the approaches employ multiple feature together. None of the single kernels is the most successful method for any of



(a) HOF

(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

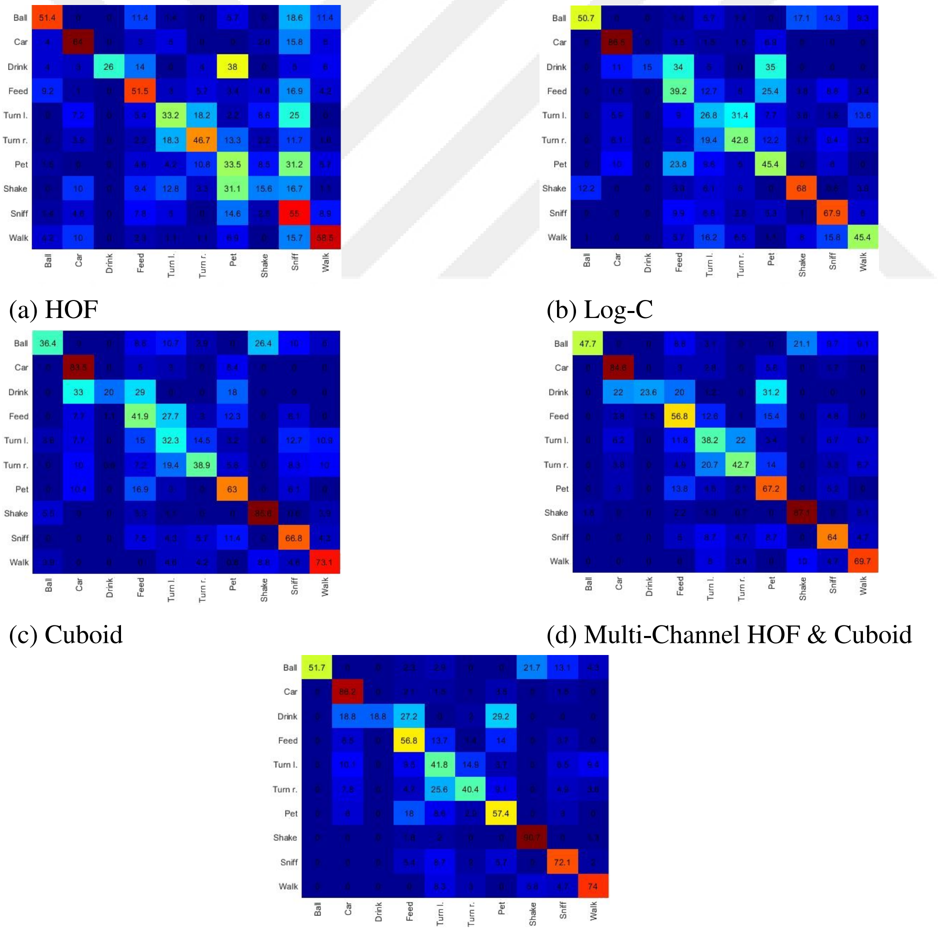
(e) Multi-channel HOF & Log-C & Cuboid

Figure 5.9: The confusion matrices of the base and combined features using Gaussian kernel on DogCentric activity dataset

Table 5.2: Most successful feature and kernels on DogCentric activity dataset

Activity	Kernel	Number of Features	Accuracy (%)
Ball	SimpleMKL	2	85.4
Walk	Multi-Channel	3	85.6
Sniff	Boosted	3	83.3
Shake	Multi-Channel	2	93.1
Pet	Multi-Channel	2	67.2
Look right	SimpleMKL	2	56.4
Look left	Boosted	3	54.5
Feed	SimpleMKL	2	71.9
Drink	SimpleMKL	3	82
Car	Multi-Channel	3	95.6

the activities. SimpleMKL performs better than Boosted MKL according to the table.



(a) HOF

(b) Log-C

(c) Cuboid

(d) Multi-Channel HOF & Cuboid

(e) Multi-channel HOF & Log-C & Cuboid

Figure 5.10: The confusion matrices of the base and combined features using Histogram Intersection kernel on DogCentric activity dataset

Table 5.3: Most successful feature and kernels on JPL-Interaction dataset

Activity	**Kernel**	**Number of Features**	**Accuracy (%)**
Shake	SimpleMKL	3	100
Hug	SimpleMKL	2	100
Pet	Cuboid	1	74.6
Wave	HOF	1	92.3
Point	Cuboid, SimpleMKL	1,3	100
Punch	Cuboid, Multi-Channel, SimpleMKL	1,3,3	100
Throw	SimpleMKL	2	93.3

Table 5.3 shows that MKL methods are not so dominant on JPL-Interaction dataset. First of all, 100% performance is seen for more than half of the activities. In the table, accuracy values of multiple feature combinations are written for "point" and "punch" activities since both single and multiple kernel approaches perform 100%. There does not occur a distinctive performance between approaches for these two activity types. For "shake", "hug" and "throw" activities, SimpleMKL shows the best performance. On the other hand Cuboid and HOF single features are successful for "pet" and "wave" activities respectively. Boosted MKL methods are not seen among the most successful approaches on JPL-Interaction dataset.

5.6.3 Overall Classification Accuracies

Table 5.4 shows classification accuracies for JPL-Interaction Dataset using different kernel types. According to this table, DC-Int performs better than other kernels for individual features on JPL-Interaction dataset. When three features are combined, DC-Int performs also the best. For HOF & Cuboid feature set, JPL-Int shows the best performance. Among individual features, accuracy values of Log-C feature are the highest with all kernel types. When features are combined, performance increases significantly. In Table 5.6, there are also accuracy values of multiple kernel learning approach. MKL approaches are more successful than traditional methods except for Boosted MKL with 2 features. For JPL-Interaction dataset, the best result is achieved by using Boosted MKL method with 3 feature set. According to the Table 5.5, Cuboid feature outperforms all other features on DogCentric activity dataset. DC-Int kernel mostly outperforms other kernels with all features with the exception of Cuboid feature, where it performs slighly worse (0.5% difference) than H-Int kernel. Furthermore,

Table 5.4: Classification accuracies for JPL-Interaction dataset

	Kernel Types (%)			
Features	***JPL-Int***	***DC-Int***	***H-Int***	***Gaussian***
HOF	70.3	**76**	66.4	72.6
Log-C	72.7	**84.2**	77	74.7
Cuboid	69.5	**75.7**	73.8	73.3
HOF & Cuboid	**85.4**	82.9	78.7	77.1
HOF & Log-C & Cuboid	82.2	**84.6**	79.0	79

Table 5.5: Classification accuracies for DogCentric activity dataset

	Kernel Types (%)		
Features	***DC-Int***	***H-Int***	***Gaussian***
HOF	**48**	45.6	29.8
Log-C	**52.6**	51	37.2
Cuboid	57	**57.5**	56.4
HOF & Cuboid	60.4	**61.2**	58.8
HOF & Log-C & Cuboid	**62.4**	62.3	54.4

multi-channel with three features is better than 2 features based on accuracy values except for Gaussian kernel. When three features are combined with DC-Int multi-channel kernel, it shows the best performance, 62.4% classification accuracy. In the Table 5.6, the accuracy values of MKL approaches on DogCentric activity dataset are shown. In Boosted MKL, three feature set is more successful than two feature set significantly. On the other hand, In SimpleMKL approach, there is not significant difference between classification performances of two feature and three feature sets. In addition, MKL approaches outperform the conventional methods according to Table 5.5 and Table 5.6.

Table 5.7 shows the results of experiments based on number of trial. Boosted MKL works through trials in which each weak classifier is trained to learn the model. At the end of each trial, a best classifier which is also known as weak classifier is selected. Final classifier is composed of weak classifiers. When number of trial becomes 200, there occurs decrease in accuracy values for Boosted MKL approach with regardless of feature numbers on DogCentric activity dataset. On the other hand, this fact is not true on JPL-Interaction dataset since the accuracy value increases from 82.7% to 84.6% with 2 features approach. Furthermore, there is not a accuracy trend depending on number of trial according to the table. For instance; when the number of trial is 10 and 20 the accuracy value becomes 64.1% and 64.3% respectively. So, there occurs 0.2% increment in the accuracy value with 3 feature approach on DogCentric activity dataset. However, when the number of trial becomes 50, the same trend does not seen in the table. On JPL-Interaction dataset, the highest accuracy value (84.6%) is observed when 200 trials are performed with 2 feature combination. This fact is not true for 3 feature combination experiments. With 3 feature combination experiments,

Table 5.6: Accuracy results on JPL and DogCentric datasets

	Accuracy (%)	
Approaches	***DogCentric dataset***	***JPL dataset***
Ryoo et al. [17] [47]	60.5	84.4
Abebe et al. (RMF features) [1]	61	86.0
SimpleMKL (2 features)	**64.9**	86.1
SimpleMKL (3 features)	64.8	85.7
Boosted MKL (2 features)	62.8	82.7
Boosted MKL (3 features)	**64.9**	**87.4**

Table 5.7: Accuracy results on JPL and DogCentric datasets

		Accuracy (%)	
Number of trials	**Kernel type**	***DogCentric dataset***	***JPL dataset***
10	Boosted MKL (2 feature)	62.5	82.3
20	Boosted MKL (2 feature)	62	82.6
50	Boosted MKL (2 feature)	62.7	83.1
100	Boosted MKL (2 feature)	**62.8**	82.7
200	Boosted MKL (2 feature)	61.7	**84.6**
10	Boosted MKL (3 feature)	64.1	87.2
20	Boosted MKL (3 feature)	64.3	**87.6**
50	Boosted MKL (3 feature)	63.6	87.1
100	Boosted MKL (3 feature)	**64.9**	87.4
200	Boosted MKL (3 feature)	63.9	86.1

the highest accuracy is 87.6% and seen when 20 trials are performed.

After all, the accuracy matrices and accuracy tables show that multiple kernel learning methods are superior to traditional single and multi-channel kernel approaches on both datasets. SimpleMKL and Boosted MKL methods give similar results. Whereas accuracy values of Boosted MKL are higher than rest on JPL-Interaction dataset, they give the same accuracy values on DogCentric activity dataset. Boosted MKL is the most successful approach for combining all features since it's highest accuracy value is acquired with 3 features. On the other hand, SimpleMKL gives similar results with 2 or 3 features. It seems most accurate performances are acquired with multiple features rather than single feature.

Table 5.8 shows the weights of kernel and feature combinations which are employed in Boosted MKL approach on DogCentric dataset. When DC-Int kernel is used, the highest weights are obtained. When HOF descriptor is used with DC-Int kernel, the weight becomes 0.19. On the other hand the weight of three descriptors with DC-Int kernel is 0.21. The weight of classifier with three descriptor is similar to classification accuracies of the multi-channel kernel with three descriptor approach since the best accuracies are obtained with combined three descriptor set in multi-channel kernel. The weight of individual HOF descriptor is the second highest value among all descriptor weights with DC-Int kernel. However, when HOF descriptor is used in DC-Int single

Table 5.8: Weight of each feature and kernel combination on DogCentric dataset with Boosted MKL

Kernel	**HOF**	**Log-C**	**Cuboid**	**HOF, Cuboid**	**HOF, Log-C, Cuboid**	**Total Weight**
Gaussian	-	0.05	0.04	0.1	0.08	0.27
H-Int	0.03	0.04	-	0.05	0.07	0.19
DC-Int	0.19	0.09	0.05	-	0.21	0.54
Total Weight	0.22	0.18	0.09	0.15	0.36	

Table 5.9: Weight Of each feature and kernel combination on JPL dataset with Boosted MKL

Kernel	**HOF**	**Log-C**	**Cuboid**	**HOF, Cuboid**	**HOF, Log-C, Cuboid**	**Total Weight**
JPL-Int	-	0.03	0.02	0.1	0.04	0.19
Gaussian	-	0.01	0.01	0.05	0.02	0.09
H-Int	-	-	-	0.01	0.06	0.07
DC-Int	0.05	0.22	0.03	0.25	0.1	0.65
Total Weight	0.05	0.26	0.06	0.41	0.22	

kernel, it does not give high classification accuracy. When features are combined with H-Int kernel, their weights are lower than other two kernels although the classification accuracies of H-Int kernel are greater than Gaussian kernel. However, the weights of classifiers with H-Int kernel are lower than weights of other classifiers with Gaussian kernel.

There are weights of classifiers which are described as feature and kernel combinations in this study in the Table 5.9 on JPL dataset.

When two descriptors are combined using DC-Int kernel, the highest weight is assigned to them among all descriptor and kernel combinations. On the other hand, the weight of three descriptors set with DC-Int kernel is greater than two descriptors combination with DC-Int kernel. However, two descriptors are combined with multi-channel kernel, the accuracy is lower than the accuracy of three descriptor set with multi-channel kernel. Log-C descriptor gives 84.2% when it is employed individually so it gives better results than other individual descriptors. Therefore, the weight of Log-C with DC-Int kernel is high. When H-Int kernel is employed with single or multi-channel kernel approach, its classification accuracies are lower than other multi-channel kernel approaches. Hence, multiple descriptors with H-Int kernel are assigned lower weights than other descriptor combinations. If HOF and Cuboid descriptors are combined with JPL-Int kernel then they are assigned high weight (0.1) and their classification accuracy are better than others.

5.6.4 Computational Evaluation

In this subsection, training time of multiple kernel learning and boosted multiple kernel learning approaches are analyzed and compared. As a multiple kernel learning method, SimpleMKL tooblox is adopted. Number of kernels used are the same for both approaches. Parameters of all kernels used in both methods are also the same.

Table 5.10: Training time evaluation results on DogCentric dataset

Approaches	**Training time (second)**		
	36 kernels	***72 kernels***	***100 kernels***
SimpleMKL	549.8	940.9	1462.7
Boosted Multiple Kernel Learning	1200	2199.1	3294.9

Number of kernels are 36, 72 and 100 respectively. Also, number of iteration is 10 for both methods. Furthermore, varying number of trials (10, 50, 100) is used for boosted multiple kernel learning method with constant kernel number 36. Training time analysis experiments are conducted only on DogCentric activity dataset. According to the results on Table 5.10, Boosted MKL approach is slower than SimpleMKL for all kernel numbers. When 36 kernels are employed, SimpleMKL is more than 2 times faster than Boosted MKL. This trend is similar for also experiments with 72 and 100 kernels. Table 5.11 shows that when number of trials increase, the training time of Boosted MKL also increases. When the number of trial is 10, Boosted MKL lasts 15 times shorter than the number of trial is 100. According to these experiments, Boosted MKL shows worse performance than SimpleMKL method in terms of training time.

Table 5.11: Trainin time evaluation results of Boosted MKL on DogCentric dataset

Number of trials	**Training Time (second)**
10	1200
50	8949
100	17550

CHAPTER 6

CONCLUSION

In this thesis, first-person activity recognition is performed with multiple kernels rather than traditional single kernel SVM. Features, which are employed in this study, represent different motion types and image characteristics. Instead of using a single type of information, fusing different features and different types of kernels provide more robust and accurate classification performance. Multiple kernel learning perform this fusing operation and allows employing best discriminatory features and kernels together. Also, different weights are assigned to these combinations based on their performance.

Three different approaches namely multi-channel kernels, multiple kernel learning (MKL) and Boosted MKL have been investigated.

Multi-channel kernels allow fusing different features which represent different types of motion but assigns equal weights to the features. This way of fusing ignores the relative importance of the features. If global motion is dominant in a video, assigning different weights to global and local features is more reasonable than assigning equal weight to both features. However, multi-channel kernel approach assigns equal weights to these features.

As opposed to multi-channel kernels, multiple kernel learning method assigns different weights to the kernels based on their relative importance. If global motion is dominant in a video, then MKL assigns more weight to a global motion feature than a local motion feature. In this thesis SimpleMKL framework is used in order to apply MKL to the first-person videos. MKL selects the most appropriate kernels and features in a data-driven approach during training. Adaptive structure of MKL approach ensures fusing different features and kernels in an optimized way rather than using pre-determined rules.

Boosted MKL integrates AdaBoost approach with MKL and selection of the features and kernels are achieved through AdaBoost trials. After all trials, a final classifier is composed of weak classifiers of each trial. Final classifier makes its decision based on a voting mechanism.

According to our experiments on Dogcentric activity and JPL-Interaction datasets, MKL outperform other methods in the literature in terms of classification accuracy. MKL approaches achieve state-of-the-art recognition accuracy values as it can integrate different types of information from videos compared to the traditional methods. In order to combine multiple features, MKL provides a robust and flexible framework. Therefore, other types of information can be integrated easily using the framework in the future. On the other hand, accuracy values of SimpleMKL are slightly behind to MKL approaches' whereas it shows better performance than single and multi-channel kernel methods.

MKL methods tries several kernel and feature combinations iteratively. In the future, the computational cost of this method can be improved using parallel programming techniques. In addition, instead of using pre-determined set of kernels, the approach can be updated in order to select the kernel and feature combinations heuristically. Also, different source of information such as audio features and virtual inertial data can be used.

REFERENCES

[1] Girmaw Abebe, Andrea Cavallaro, and Xavier Parra. “Robust multi-dimensional motion features for first-person vision activity recognition”. In: *Computer Vision and Image Understanding* 149 (2016), pp. 229–248. ISSN: 10773142. DOI: 10.1016/j.cviu.2015.10.015.

[2] Yicheng Bai, Chengliu Li, Yaofeng Yue, Wenyan Jia, Jie Li, Zhi-Hong Mao, and Mingui Sun. “Designing a Wearable Computer for Lifestyle Evaluation.” In: *IEEE Northeast Bioengineering Conference*. NIH Public Access, 2012, pp. 93–94. DOI: 10.1109/NEBC.2012.6206978.

[3] Cigdem Beyan, Francesca Capozzi, Cristina Becchio, and Vittorio Murino. “Identification of emergent leaders in a meeting scenario using multiple kernel learning”. In: *Proceedings of the 2nd Workshop on Advancements in Social Signal Processing for Multimodal Interaction - ASSP4MI ’16*. New York, New York, USA: ACM Press, 2016, pp. 3–10. ISBN: 9781450345576. DOI: 10.1145/3005467.3005469.

[4] Gary R. Bradski. “Real time face and object tracking as a component of a perceptual\nuser interface”. In: *Proceedings Fourth IEEE Workshop on Applications of Computer Vision. WACV’98 (Cat. No.98EX201)* (1998), pp. 14–19. ISSN: 09600760. DOI: 10.1109/ACV.1998.732882.

[5] R. O. Castle, D. J. Gawley, G. Klein, and D. W. Murray. “Towards simultaneous recognition, localization and mapping for hand-held and wearable cameras”. In: *Proceedings - IEEE International Conference on Robotics and Automation*. April. 2007, pp. 4102–4107. ISBN: 1424406021. DOI: 10.1109/ROBOT.2007.364109.

[6] Yongwon Cho, Yunyoung Nam, Yoo-Joo Choi, and We-Duke Cho. “Smart-Buckle”. In: *Proceedings of the 2nd International Workshop on Systems and Networking Support for Health Care and Assisted Living Environments - Health-Net ’08*. 2008, p. 1. ISBN: 9781605581996. DOI: 10.1145/1515747.1515757.

[7] B. Clarkson, K. Mase, and A. Pentland. “Recognizing user context via wearable sensors”. In: *Digest of Papers. Fourth International Symposium on Wearable Computers*. 2000, pp. 1–7. ISBN: 0-7695-0795-6. DOI: 10.1109/ISWC.2000.888467.

[8] Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. “MonoSLAM: Real-time single camera SLAM”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.6 (2007), pp. 1052–1067. ISSN: 01628828. DOI: 10.1109/TPAMI.2007.1049.

[9] Andrew Calway Dima Damen, Teesid Leelasawassuk, Osian Haines and Walterio Mayol-Cuevas. “You-Do , I-Learn : Discovering Task Relevant Objects and their Modes of Interaction from Multi-User Egocentric Video”. In: *Proceedings of the British Machine Vision Conference 2014*. 2014, pp. 1–13.

[10] Piotr Dollár, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. “Behavior recognition via sparse spatio-temporal features”. In: *Proceedings - 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, VS-PETS*. Vol. 2005. 2005, pp. 65–72. ISBN: 0780394240. DOI: 10.1109/VSPETS.2005.1570899.

[11] Alireza Fathi, Ali Farhadi, and James M. Rehg. “Understanding egocentric activities”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2011, pp. 407–414. ISBN: 9781457711015. DOI: 10.1109/ICCV.2011.6126269.

[12] Alireza Fathi, Yin Li, and James M. Rehg. “Learning to recognize daily actions using gaze”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 7572 LNCS.PART 1 (2012), pp. 314–327. ISSN: 03029743. DOI: 10.1007/978-3-642-33718-5_23.

[13] Alireza Fathi, Xiaofeng Ren, and James M. Rehg. “Learning to recognize objects in egocentric activities”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2011, pp. 3281–3288. ISBN: 9781457703942. DOI: 10.1109/CVPR.2011.5995444.

[14] Ilaria Gori, J. K. Aggarwal, Larry Matthies, and M. S. Ryoo. “Multitype Activity Recognition in Robot-Centric Scenarios”. In: *IEEE Robotics and Automation Letters* 1.1 (2016), pp. 593–600. ISSN: 2377-3766. DOI: 10.1109/LRA.2016.2525002.

[15] Kai Guo, Prakash Ishwar, and Janusz Konrad. “Action recognition from video using feature covariance matrices.” In: *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society* 22.6 (2013), pp. 2479–94. ISSN: 1941-0042. DOI: 10.1109/TIP.2013.2252622.

[16] Tâm Huynh, Mario Fritz, and Bernt Schiele. “Discovery of activity patterns using topic models”. In: *Proceedings of the 10th international conference on Ubiquitous computing*. 2008, pp. 10–19. ISBN: 978-1-60558-136-1. DOI: 10.1145/1409635.1409638.

[17] Yumi Iwashita, Asamichi Takamine, Ryo Kurazume, and M.S. Ryoo. “First-Person Animal Activity Recognition from Egocentric Videos”. In: *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 4310–4315. ISBN: 978-1-4799-5209-0. DOI: 10.1109/ICPR.2014.739.

[18] Kris M. Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. “Fast unsupervised ego-action learning for first-person sports videos”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. June. 2011, pp. 3241–3248. ISBN: 9781457703942. DOI: 10.1109/CVPR.2011.5995406.

[19] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. “Learning realistic human actions from movies”. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8. ISBN: 978-1-4244-2242-5. DOI: `10.1109/CVPR.2008.4587756`.

[20] Oscar D. Lara and Miguel A. Labrador. “A Survey on Human Activity Recognition using Wearable Sensors”. In: *IEEE Communications Surveys & Tutorials* 15.3 (2013), pp. 1192–1209. ISSN: 1553-877X. DOI: `10.1109/SURV.2012.110112.00192`.

[21] Cheng Li and Kris M. Kitani. “Model recommendation with virtual probes for egocentric hand detection”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2013, pp. 2624–2631. ISBN: 9781479928392. DOI: `10.1109/ICCV.2013.326`.

[22] Cheng Li and Kris M. Kitani. “Pixel-level hand detection in ego-centric videos”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2013, pp. 3570–3577. ISBN: 978-0-7695-4989-7. DOI: `10.1109/CVPR.2013.458`.

[23] Li Jia Li and Li Fei-Fei. “What, where and who? Classifying events by scene and object recognition”. In: *Proceedings of the IEEE International Conference on Computer Vision* (2007). ISSN: 1550-5499. DOI: `10.1109/ICCV.2007.4408872`.

[24] Yin Li, Alireza Fathi, and James M. Rehg. “Learning to predict gaze in egocentric video”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 1. 2013, pp. 3216–3223. ISBN: 9781479928392. DOI: `10.1109/ICCV.2013.399`. arXiv: `1505.04868`.

[25] David G Lowe. “Distinctive image features from scale invariant keypoints”. In: *Int’l Journal of Computer Vision* 60 (2004), pp. 91–11020042. ISSN: 0920-5691. DOI: `http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94`. arXiv: `0112017 [cs]`.

[26] Zheng Lu and Kristen Grauman. “Story-Driven Summarization for Egocentric Video”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 2714–2721. ISBN: 978-0-7695-4989-7. DOI: `10.1109/CVPR.2013.350`.

[27] M. Ma, H. Fan, and K. M. Kitani. “Going Deeper into First-Person Activity Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 1894–1903.

[28] Kenji Matsuo, Kentaro Yamada, Satoshi Ueno, and Sei Naito. “An attention-based activity recognition for egocentric video”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 2014, pp. 565–570. ISBN: 9781479943098. DOI: `10.1109/CVPRW.2014.87`.

[29] W. W. Mayol and D. W. Murray. “Wearable hand activity recognition for event summarization”. In: *Proceedings - International Symposium on Wearable*

Computers, ISWC. Vol. 2005. 2005, pp. 122–129. ISBN: 0769524192. DOI: 10.1109/ISWC.2005.57.

[30] Pietro Moreiro, Lucio Marcenaro, and Regazzoni Carlo. “Hand detection in First Person Vision”. In: *18th International Conference on Information Fusion*. Istanbul: IEEE, 2015, pp. 280–286. ISBN: 9780996452717.

[31] Mary Muir and Cristina Conati. “Understanding Student Attention to Adaptive Hints with Eye-Tracking”. In: (2012), pp. 148–160. DOI: 10.1007/978-3-642-28509-7_15.

[32] Usman Naeem and Queen Mary. “Recognising Activities of Daily Life Using Hierarchical Plans”. In: October 2007. 2007. ISBN: 978-3-540-88792-8. DOI: 10.1007/978-3-540-88793-5.

[33] Yunyoung Nam, Seungmin Rho, and Chulung Lee. “Physical activity recognition using multiple sensors embedded in a wearable device”. In: *ACM Transactions on Embedded Computing Systems* 12.2 (2013), pp. 1–14. ISSN: 15399087. DOI: 10.1145/2423636.2423644.

[34] Sanath Narayan, Mohan S. Kankanhalli, and Kalpathi R. Ramakrishnan. “Action and interaction recognition in first-person videos”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 2014, pp. 526–532. ISBN: 9781479943098. DOI: 10.1109/CVPRW.2014.82.

[35] P. Natarajan, Shuang Wu, S. Vitaladevuni, Xiaodan Zhuang, S. Tsakalidis, Unsang Park, R. Prasad, and P. Natarajan. “Multimodal feature fusion for robust event detection in web videos”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1298–1305. ISBN: 978-1-4673-1228-8. DOI: 10.1109/CVPR.2012.6247814.

[36] Keisuke Ogaki, Kris M. Kitani, Yusuke Sugano, and Yoichi Sato. “Coupling eye-motion and ego-motion features for first-person activity recognition”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. June. 2012, pp. 1–7. ISBN: 9781467316118. DOI: 10.1109/CVPRW.2012.6239188.

[37] Fatih Ozkan, Mehmet Ali Arabaci, Elif Surer, and Alptekin Temizel. “Boosted Multiple Kernel Learning for First-Person Activity Recognition”. In: *2017 25th European Signal Processing Conference (EUSIPCO)*. Accepted for publication. 2017.

[38] Donald J. Patterson, Dieter Fox, Henry Kautz, and Matthai Philipose. “Fine-grained activity recognition by aggregating abstract object usage”. In: *Proceedings - International Symposium on Wearable Computers, ISWC*. Vol. 2005. 2005, pp. 44–51. ISBN: 0769524192. DOI: 10.1109/ISWC.2005.22.

[39] M Philipose, Kp Fishkin, M Perkowitz, Dj Patterson, D Hahnel, D Fox, and H Kautz. “Inferring ADLs from interactions with Objects”. In: *IEEE Pervasive Computing* 3.4 (2004), pp. 50–57.

[40] Hamed Pirsiavash and Deva Ramanan. “Detecting Activities of Daily Living in First-person Camera Views”. In: *2012 IEEE Conference on Computer Vision*

and Pattern Recognition. IEEE, 2012, pp. 2847–2854. ISBN: 978-1-4673-1228-8. DOI: 10.1109/CVPR.2012.6248010.

[41] Y. Poleg, C. Arora, and S. Peleg. “Temporal Segmentation of Egocentric Videos”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 2537–2544. DOI: 10.1109/CVPR.2014.325.

[42] Alain Rakotomamonjy, Francis Bach, Stephane Canu, and Yves Grandvalet. “SimpleMKL”. In: *Journal of Machine Learning Research* 9 (2008), pp. 2491–2521.

[43] Xiaofeng Ren and Chunhui Gu. “Figure-Ground Segmentation Improves Handled Object Recognition in Egocentric Video”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2010.

[44] M. S. Ryoo, B. Rothrock, and L. Matthies. “Pooled motion features for first-person videos”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 896–904.

[45] M. S. Ryoo, Brandon Rothrock, and Larry Matthies. “Pooled motion features for first-person videos”. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol. 07-12-June. Figure 1. 2015, pp. 896–904. ISBN: 9781467369640. DOI: 10.1109/CVPR.2015.7298691. arXiv: 1412.6505.

[46] M. S. Ryoo, Thomas J. Fuchs, Lu Xia, J.K. Aggarwal, and Larry Matthies. “Robot-Centric Activity Prediction from First-Person Videos”. In: *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction - HRI ’15*. New York, New York, USA: ACM Press, 2015, pp. 295–302. ISBN: 9781450328838. DOI: 10.1145/2696454.2696462.

[47] Michael S. Ryoo and Larry Matthies. “First-person activity recognition: What are they doing to me?” In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. June. 2013, pp. 2730–2737. ISBN: 1063-6919 VO -. DOI: 10.1109/CVPR.2013.352.

[48] Bernt Schiele, Nuria Oliver, Tony Jebara, and Alex Pentland. “An interactive computer vision system DyPERS: Dynamic personal enhanced reality system”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 1542 (1999), pp. 51–65. ISSN: 16113349. DOI: 10.1007/3-540-49256-9_4.

[49] S. Singh, C. Arora, and C. V. Jawahar. “First Person Action Recognition Using Deep Learned Descriptors”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2620–2628.

[50] Jeffrey Mark Siskind, Allan Jepson, and Jeffrey Mark Siskind. “Computational perception of scene dynamics”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 1065.2 (1996), pp. 528–539. ISSN: 16113349. DOI: 10.1007/3-540-61123-1_167.

[51] Bilge Soran, Ali Farhadi, and Linda Shapiro. “Action recognition in the presence of one egocentric and multiple static cameras”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9007 (2015), pp. 178–193. ISSN: 16113349. DOI: 10.1007/978-3-319-16814-2_12.

[52] Ekaterina H. Spriggs, Fernando De La Torre, and Martial Hebert. “Temporal segmentation and activity classification from first-person sensing”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009*. 2009, pp. 17–24. ISBN: 9781424439911. DOI: 10.1109/CVPR.2009.5204354.

[53] T Starner, B Schiele, and A Pentland. “Visual contextual awareness in wearable computing”. In: *Digest of Papers Second International Symposium on Wearable Computers Cat No98EX215*. 1998, pp. 50–57. ISBN: 0-8186-9074-7. DOI: 10.1109/ISWC.1998.729529.

[54] T. Starner, J. Weaver, and A. Pentland. “Real-time American sign language recognition using desk and wearable computer based video”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.12 (1998), pp. 1371–1375. ISSN: 0162-8828. DOI: 10.1109/34.735811.

[55] Li Sun, Ulrich Klank, and Michael Beetz. “EYEWATCHME- 3D hand and object tracking for inside out activity analysis”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009*. 2009, pp. 9–16. ISBN: 9781424439911. DOI: 10.1109/CVPR.2009.5204358.

[56] S. Sundaram and W.W.M. Cuevas. “High level activity recognition using low resolution wearable vision”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 2009, pp. 25–32. ISBN: 978-1-4244-3994-2. DOI: 10.1109/CVPRW.2009.5204355.

[57] Sudeep Sundaram and Walterio W. Mayol Cuevas. “High Level Activity Recognition using Low Resolution Wearable Vision”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 2009, pp. 25–32. ISBN: 978-1-4244-3994-2. DOI: 10.1109/CVPRW.2009.5204355.

[58] A. Takamine, Y. Iwashita, and R. Kurazume. “First-person activity recognition with C3D features from optical flow images”. In: *2015 IEEE/SICE International Symposium on System Integration (SII)*. 2015, pp. 619–622.

[59] Takumi Toyama, Thomas Kieninger, Faisal Shafait, and Andreas Dengel. “Gaze guided object recognition using a head-mounted eye tracker”. In: *ETRA ’12 Proceedings of the Symposium on Eye Tracking Research and Applications*. Vol. 1. 212. New York, New York, USA: ACM Press, 2012, pp. 91–98. ISBN: 9781450312219. DOI: 10.1145/2168556.2168570.

[60] Manik Varma and Bodla Rakesh Babu. “More generality in efficient multiple kernel learning”. In: *Proceedings of the 26th Annual International Conference on Machine Learning - ICML ’09*. New York, New York, USA: ACM Press, 2009, pp. 1–8. ISBN: 9781605585161. DOI: 10.1145/1553374.1553510.

[61] Jianxin Wu, Adebola Osuntogun, Tanzeem Choudhury, Matthai Philipose, and James M. Rehg. “A Scalable Approach to Activity Recognition based on Object Use”. In: *2007 IEEE 11th International Conference on Computer Vision* (2007), pp. 1–8. ISSN: 1550-5499. DOI: 10.1109/ICCV.2007.4408865.

[62] H. Xia and S. C. H. Hoi. “MKBoost: A Framework of Multiple Kernel Boosting”. In: *IEEE Transactions on Knowledge and Data Engineering* 25.7 (2013), pp. 1574–1586. ISSN: 1041-4347. DOI: 10.1109/TKDE.2012.89.

[63] Kentaro Yamada, Yusuke Sugano, Takahiro Okabe, Yoichi Sato, Akihiro Sugimoto, and Kazuo Hiraki. “Attention Prediction in Egocentric Video Using Motion and Visual Saliency”. In: *Proceedings of the 5th Pacific Rim conference on Advances in Image and Video Technology - Volume Part I*. Springer-Verlag, 2011, pp. 277–288. ISBN: 978-3-642-25366-9. DOI: 10.1007/978-3-642-25367-6_25.

[64] Kentaro Yamada, Yusuke Sugano, Takahiro Okabe, Yoichi Sato, Akihiro Sugimoto, and Kazuo Hiraki. “Can Saliency Map Models Predict Human Egocentric Visual Attention?” In: *Computer Vision – ACCV 2010 Workshops: ACCV 2010 International Workshops, Queenstown, New Zealand, November 8-9, 2010, Revised Selected Papers, Part I*. Ed. by Reinhard Koch and Fay Huang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 420–429. ISBN: 978-3-642-22822-3. DOI: 10.1007/978-3-642-22822-3_42.

[65] Yong Jae Lee, J. Ghosh, and K. Grauman. “Discovering important people and objects for egocentric video summarization”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1346–1353. ISBN: 978-1-4673-1228-8. DOI: 10.1109/CVPR.2012.6247820.

[66] K Zhan, S Faux, and F Ramos. “Multi-scale conditional random fields for first-person activity recognition”. In: *2014 12th IEEE International Conference on Pervasive Computing and Communications, PerCom 2014*. 2014, pp. 51–59. DOI: 10.1109/PerCom.2014.6813944.

[67] Andreas Zinnen, Christian Wojek, and Bernt Schiele. “Multi activity recognition based on bodymodel-derived primitives”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5561 LNCS (2009), pp. 1–18. ISSN: 03029743. DOI: 10.1007/978-3-642-01721-6_1.

TEZ FOTOKOPİ İZİN FORMU

ENSTİTÜ:

Fen Bilimleri Enstitusu

Sosyal Bilimler Enstitüsü

UygulamalıMatematik Enstitüsü

Enformatik Enstitüsü

Deniz Bilimleri Enstitüsü

YAZARIN

Soyadı . . . .

Adı . . . .

Bölümü . . . .

TEZİN ADI

. . . .
. . . .

TEZİN TÜRÜ.............: Yuksek Lisans □ Doktora □

1. Tezimin tamamı dünya çapında erişime açılsın ve kaynak gösterilmek şartıyla tezimin bir kısmı veya tamamının fotokopisi alınsın.

2. Tezimin tamamı yalnızca Ortadoğu Teknik Üniversitesi kullanıcılarının erişimine açılsın. (Bu seçenekle tezinizin fotokopisi yada elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.)

3. Tezim 1 yıl süreyle erişime kapalı olsun. (Bu seçenekle tezinizin fotokopisi yada elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.)

Yazarın imzası .... Tarih ....