Master Thesis

Comparing grid-based features and classification settings, in an off-line supervised human action detection and recognition task

Human Media Interaction (University of Twente)
Enschede, The Netherlands

May 25, 2012

Author:
Remco Hijmans

Graduation committee:
dr. R.W. Poppe
dr. D. Reidsma
dr. M. Poel
prof. dr. D.K.J. Heylen

Contents

Appendices

Chapter 1

Introduction

This master thesis describes a final project, where two different grid-based features using different classification settings are compared, in an off-line supervised human action detection and recognition task. The task consists of the detecting and recognizing fitness moves performed in an exercise video, using limited training data acquired from the same video.

The exercise video displays two persons standing side by side performing movements that belong to the XCO[1] workout. The video consist of multiple scenes and each scene has its own specific theme. Furthermore, the persons in each scene always consist of a man and a woman. The first scene "Warming Up" is the only scene that contains some movements that return in most other scenes and therefore this scene is selected as training set for the classifier. Not all movements occur with the same frequency in the training set and therefore only a single occurrence of each movement is selected as template.

The video must first be processed, before any human action detection and/or recognition can be performed. A top-down approach to image processing technique is taken, which means that the persons in the video are first detected and the region of interest per person is selected. These regions of interest are used to extract two features and those are the silhouette and motion (based on optical flow) descriptor. Both features are subdivided using a predetermined grid division, which results in two grid-based features: Histogram of Silhouette (HoS) and Histogram of Flow (HoF). These two grid-based features are evaluated in combination with a number of different grid configurations.

An important step in the supervised human action detection and recognition task is the comparison of each templates with a continuous input stream of features. The chosen metric is a modified version of Dynamic Time Warping, which allows for the comparison of multi-dimensional time series. These acquired distances are the input for the classifier, which tem-

[1] http://www.xco.nl/

porally segments the video and labels one or more action in a segment. This metric and classifier are kept constant for all different settings being evaluated.

The findings of this master thesis can be used to say something about the applicability of both grid-based features: Histogram of Silhouette and Histogram of Flow, using a coarse grid division instead of the more fine-grained divisions chosen in most literature, in an action classification task with only limited training data with varying subjects.

1.1 Document layout

This introduction is followed by Chapter 2, which contains related work on: feature extraction, and action classification. An overview of the entire approach taken for the human action detection and recognition is discussed in Chapter 3 and some details regarding the approach are highlighted further in Chapters 4 and 5. This is followed up by evaluation plan in Chapter 6, which introduces all the variables, research questions, and procedures to answering these research questions. Chapter 7 contains the results of the evaluation including a discussion per research question. This master thesis is finalized in Chapter 8 with the conclusions, followed up by ideas for future research. Remaining chapters are the references list and the appendices.

Chapter 2

Related work

A lot of work has already been done in the field of vision-based human action recognition, as becomes clear when examining the article by Poppe [22], who gives quite an elaborate overview of all kinds of different techniques for action recognition. Poppe separates action recognition into two steps: "image representation" and "action classification". Image representation consists of techniques that extract features of an image or sequence and action classification techniques assign an action label to an image or sequence, according to the result of comparison between the extracted features and reference material. A feature or image representation in this master thesis, is an entity that describes a specific characteristic of a video frame or sequence of frames.

The topics of this chapter match with the two steps mentioned by Poppe. This chapter starts off with section 2.1 "feature extraction", which discusses a small subset of features relevant to the approach taken in this final project. Followed up by section 2.2 "action classification", which discusses subjects regarding temporal segmenting, video indexing, metrics for comparison, and classification.

2.1 Feature extraction

According to Poppe, image representation techniques can be divided into two categories: global and local. The primary difference between the two categories is that global techniques take a top-down approach and a local techniques take a bottom-up approach to processing an image. The text in this section primarily discusses global image representations and starts off with discussing the two features silhouette and motion descriptor in Sections 2.1.1 and 2.1.2. For both features a number of approaches are mentioned for extracting them from video. These two features are discussed, because they are commonly used as the foundation for a number of derivative image representations, of which a few are examined in section 2.1.3.

2.1.1 Silhouette descriptor

A silhouette descriptor is a feature that marks the pixels in an image that belong to a foreground object. The descriptor mostly consists of a binary (black and white) image in which all foreground objects get the same color and the background the other color. This approach requires the construction of a background model, to be able to distinguish between foreground and background pixels. The pixels that contain a value above a certain threshold are considered to be foreground pixels and the pixels scoring below the threshold are considered to be background. This method mostly delivers good result in a recorded scene where the camera set up and background remain constant, and the background colors differ from foreground objects.

The simplest approach to acquiring the foreground pixels is by subtracting a background image from each frame. Such a background image can be extracted from the video itself by selecting a frame containing no foreground objects, or by manually construction a background image. Both approaches require manual interference, which is unacceptable for any full automated task.

The method Least Median of Squares (LMedS) [31] solves this manual interference problem, but adds an additional requirement, LMedS will only work with a sequence of images. LMedS is still not an acceptable approach to most real-life situation, because it requires a rigid scene and a stationary camera. Another automatic background modeling technique is that of Stauffer and Grimson [26], which allows for a complexer background. Their method models each pixel as a mixture of Gaussians (MoG). According to Zhang et.al. [36] the MoG approach does not perform well with dynamic environments and is rather slow. Zivkovic [37] proposes an improvement to the MoG approach, that reduces the processing time and improves the segmentation a little. MoG might be improved, but is still not able to deal with highly dynamic environments.

A more novel approach that is invariant to complex and dynamic environments and varying lighting conditions is the method of Chen et. al. [7]. Their approach first divides the image into a grid of patches with a size of 4x4 pixels and then classifies per patch if it belongs to foreground or background. A sliding window over the patches of previous frames is used for this classification process. By faster updating the blocks that are not specified as foreground, new objects in the scene are absorbed faster into the background, while non-moving foreground objects are still classified as foreground. Another approach that is able to handle dynamic backgrounds, is the approach proposed by Zhang et.al. [36]. The approach of Zhang uses a technique called Spatio-temporal Local binary pattern (STLBP) and the results of their experiment show that it adapts quickly to changes in a dynamic background.

One assumption that is made with all the previous automatic approaches

is that they all assume the camera is stationary. In most video material this assumption does not hold and Sheick et.al. [25] propose a solution to this problem. They solve the problem of a moving camera, by calculating the trajectories of salient points in an image sequence. They classify these trajectories as background or outliers (foreground). An assumption made in their article is that; *"the background is the spatially dominant 'rigid' entity in the image."*. This assumption causes their approach to not work with dynamic backgrounds.

2.1.2 Motion descriptor

A motion descriptor is a feature that uses optical flow to describe 2D motion in a video recording. Optical flow is the concept of projecting 3D velocities of objects, surfaces, and edges onto an imaging surface (see Figure 2.1 for an example). There exist many approaches for acquiring optical flow and these approaches can be categorized into differential, energy-based, and phase-based approaches. Barron et.al. [2] evaluates a large number of these optical flow approaches and compares them on accuracy, reliability, and density of the velocity measurements. The image sequences used in this evaluation were not severely corrupted by spatial or temporal aliasing. Barron concluded that the first-order local differential approach of Lucas and Kanade [20] and the local phase-based approach of Fleet and Jepson [17] delivered the most reliable results. The problem with all the phase-based approaches discussed in the article by Barron et.al. is they all had high computational load. Overall it seems that the local differential approach of Lucas and Kanade outperforms the rest in this particular evaluation.

Figure 2.1: Example of visualized motion descriptor using Lucas-Kanade

A more recent development is the work of Bruhn et.al. [6], who combines the local differential approach, least square fit of Lucas and Kanade [20] with the global differential approach of Horn and Schunck [19]. According to Bruhn, the combined local-global (CLG) approach; *"is highly robust under Gaussian noise while giving dense flow fields."*. An even more recent study by Sun et.al. [27] looked at the Horn-Schunck algorithm, explaining and comparing all different proposed optimizations to the Horn-Schunck algorithm since its introduction. Sun et.al. finds that with certain optimizations, the algorithm is quite competitive to other good performing optical flow approaches, which shows that the Horn-Schunck algorithm has improved much since the evaluation by Barron et.al. [2].

2.1.3 Derivative image representation

This subsection discusses the structure and application of a number of different derivative image representations, which are all based on the previously mentioned: silhouette and motion descriptor. The discussed derivative image representations are all introduced in a template-based supervised human action recognition task. These tasks compare reference material in the form of templates with inputted observations and the best matching template is used to determine the label being assigned to the inputted observation.

A very popular template-based approach, often found as a reference in literature regarding this subject, is the temporal template matching approach proposed by Bobick and Davis [5]. These temporal templates consist of two descriptors: the binary motion-energy image (MEI) and the motion-history image (MHI). Both are built out of a sequence of silhouette descriptors. The MEI indicates where motion occurred and the MHI is a scalar-valued image, where intensity is a function of recency of motion. According to Davis and Bobick, their approach will also work when the video frames are blurred.

Another fairly similar approach to that of Bobick and Davis, is the work of Wang and Suter [30]. They created two high-level descriptors, the average motion energy (AME) and the mean motion shape (MMS) image. According to Wang and Suter; *"They indirectly encode the motion structure and characteristics of an action, and save both storage space and computation complexity."*. Both descriptors describe an entire motion sequence in one single image. The AME averages over the silhouette descriptors for the entire sequence, the MMS is an average of the contour images. The AME is a gray-level image and intensity of a pixel depicts the frequency of the motion that occurred at that pixel. The AME image helps in partially preserving temporal information, by encoding it in the intensity of the image, which is a fairly similar to the motion-history image of Bobick and Davis.

The previously mentioned global approaches take the temporal dimension in consideration. An example of an image representation approach, which is invariant to temporal effects, is the approach by Weinland and Boyer [32]. A major disadvantage of such an approach is the loss of the temporal domain and it becomes impossible to distinguish between inverse movements, such as "sitting down" and "standing up".

A global feature based on the motion descriptor, is the feature used by Efros et.al. [15]. They determine optical flow with the method of Lucas and Kanade [20], which helps them in classifying actions in sequences of images. To improve speed and simplify the comparison, each flow line is first split up in to the horizontal and vertical component (see Figure 2.2) and both components are half-wave rectified, which results in four channels (H+,H-,V+, and V-). The resulting four channels represent a single motion feature, which is used in the comparison process. Using optical flow allows

for more robust comparison, instead of using silhouette based descriptors. With a silhouette-based descriptor a property such as the dimensions of the recorded person could play a role in comparison, which is less the case when using optical flow.

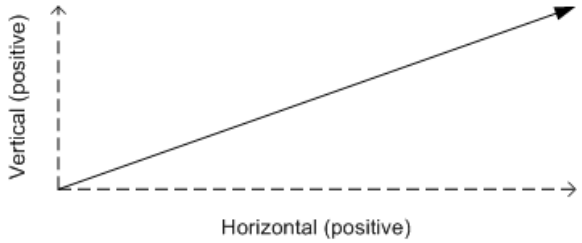


Figure 2.2: Splitting up a flow vector into its H and V component

A grid-based feature is a global feature that splits up a feature extracted for each inputted image frame according to a predefined grid. This results in a multi dimensional global feature, where each grid patch describes a section of the inputted image frame. The work of Danafar and Gheissari [11] and of Tran et.al. [29] are examples of approaches that use grid-based features. Danafar and Gheissari split up an image (so it only contains the recorded person) into three rows and these rows are divided according to predetermined heuristics. These heuristics cause the resulting division to roughly contain the following: the head, the torso and legs. A motion descriptor per patch is used, which is a similar descriptor to the descriptor created in the work of Efros et.al. [15]. Combining these descriptors per patch results in a histogram of flow (HoF) for each image frame. A more elaborate grid-based approach is the approach proposed by Tran et.al. [29]. They divide each frame into 2x2 or 3x3 patches and for each patch two high-level descriptor are constructed, a HoF and a histogram of silhouette (HoS). The histogram of silhouette contains per patch the percentage of pixels that are occupied by the silhouette. These two high-level descriptors are combined and extended with motion context, by performing Principal Component Analysis (PCA) on neighboring frames.

The approach described in this master thesis, is based on the approach that is proposed by Tran et.al. [29]. The exact approach taken in this research, is elaborated in chapter 3 and chapter 4.

2.2 Action classification

The second important step in action recognition, is the classification of video segments. There are numerous approaches to classifying video segments, and they either are supervised or unsupervised. The choice for a suitable classification approach depends on the structure of the data, the functional

requirements, and the availability of training data. A simplified example of a supervised approach is to have a classifier that is trained using a training set, which classifies the movement in a test set using the knowledge gained from the training data. This restricted set of actions makes it difficult to find and classify movements that are not present in this training set. An unsupervised classification approach is not bound to a training set, but these approaches mostly lack the accuracy in comparison to the supervised approaches and they require complex algorithms for detection motion boundaries, when applied to a continuous video stream.

The topic of temporally segmenting a continuous video when dealing with an unsupervised classification approach is discussed in section 2.2.1. Both sections 2.2.2 and 2.2.3 discuss an unsupervised classification approach and both do not require the video to be temporally segmented before classification. These sections are followed up by section 2.2.4, which discusses a supervised classification approach using a single key frame to classify segments containing a movement. This section on key frames is follow up by section 2.2.5, which discusses a number of different metrics for pattern matching that are useful for a supervised human action detection and recognition task. Finally, section 2.2.6 discusses two classification approaches.

2.2.1 Temporal Segmentation

An aspect of action recognition, which often is not elaborated in articles about action recognition, is temporal segmentation. Most approaches presuppose that the video is readily segmented or nothing is mentioned in the article regarding temporal segmentation at all. However, the approach used for segmenting a video greatly influences the results acquired through a classification process. The subject of temporal segmentation therefore deserves an elaborate discussion.

Scene partitioning

The chosen segmentation technique depends heavily on the requirements that are set for the resulting segmentation. A topic often found in literature is the segmentation of a video into scenes. They define a scene as a single uninterrupted camera shot. The scene segmentation approach by Xiong and Lee [33] uses optical flow for determining scene transitions. They accomplish this by determining the dominant camera motions in video shots. To determine this dominant motion, they process the video frames in a similar way as is done with the creation of a HoF (Tran et.al. [29]). However, instead of taking the total sum of each component of each flow line, they calculate the mean and standard deviation in each patch. Their approach uses a sliding window that moves over the entire video and performs binary search within a window to find the scene transition where a significant difference between

the leftmost and rightmost frame of the window is detected.

Motion segmentation

The work by Xiong and Lee [33] could be adapted to segment a video into primitive action segments, which is done by Rui and Anandan [24]. Their approach detects temporal discontinuities in spatial motion patterns and use that to segment a video. Their approach extracts the optical flow and the silhouette descriptor for every frame in the video. The two descriptors are combined with as a result a flow field with only significant flow in the area that is occupied by the silhouette. This flow field is broken down into coefficients, using singular value decomposition (SVD) and those coefficients are analyzed to find discontinuities in their temporal trajectories. These discontinuities are considered to be the boundaries of discrete action primitives. Rui and Anandan achieve with their approach a correlation of 60% with the set of manually selected boundaries. This fairly good result makes this approach an interesting option for the approach taken in this final project.

Ali and Aggarwal [1] temporally segment their motion sequence into discrete action primitives using key-pose frames. They classify each frame into one of the two classes: breakpoint or non-breakpoint. All frames between two breakpoint frames are selected as a single discrete action and this segmented motion sequence is passed on to the human action classifier.

2.2.2 Periodic motion

Cutler and Davis [10] propose an approach that uses the analysis of periodic motion for recognizing actions. Their approach does not require any temporal segmentation to be able to recognize actions. They do object segmentation, track the segmented object, and do a time-frequency analysis on the motion data of the tracked object. This approach works perfectly when the object displays a constant periodic movement, take for example a human that is walking. However, when there are too many variances (non-periodic motion), their approach will not be able to classify the movement.

2.2.3 Event-based video indexing

Zelnik-Manor and Irani [35] propose an approach that is able to find dynamic events, *"without prior knowledge of the types of events, their models, or their temporal extent."* and that does not require any temporal segmentation of a continuous video. This forms the basis for a technique they refer to as event-based video indexing. They describe event-based video indexing as finding all similar occurrences of a preselected action, which is fairly similar to any template matching technique. They construct an empirical distribution, *"where local features at multiple temporal scales are taken as samples of the stochastic process"*, which are used to construct the empirical

distribution. These empirical distributions are used in the comparison process, to find similar segments. They claim that their approach, *"allows for general event-based analysis of video information containing unknown event types."*. Zelnik-Manor and Irani mention that their approach is *"inferior in accuracy to the more sophisticated (but more restricted) parametric models"*, which is unwanted and a good accuracy is preferable. Furthermore, their approach uses local intensity gradients, which is not the most informative feature available.

2.2.4 Key-poses

Supervised human action recognition approaches require reference actions to be available for comparison with the inputted video material. Weinland and Boyer [32] use a set of discriminative key-poses (a single frame), which they match with each inputted observation sequence. The minimum distance for each key-pose to the inputted observation sequence is saved in a vector and serves as input for the classifier. They argue that temporal information is not required for recognizing a movement and they achieve high recognition rates, with only a small set of key-poses. The consequence of taking such an approach is, it becomes impossible to find the exact start and end of an event, when dealing with continuous video material.

2.2.5 Distance algorithms

When a single occurrence of an action is used as template for classifying motion segments, finding similar occurrences in the input sequence can be achieved by calculating the distance between the template and set of input tokens. There are numerous distance algorithm available and each has its own characteristics. A few distance algorithms are discussed in the paragraphs below.

Dynamic Time Warping

The non-linear sequence alignment algorithm called Dynamic Time Warping (DTW) is a distance algorithm, which often is used with success in the field of speech recognition [18], because it is able to take into account different speeds of utterances of the same phrases and words. According to Fang [16], *"It seeks an optimal mapping from the test signal to the template signal, meanwhile allowing a non-linear, monotonic distortion (warping) in the test signal."*. These properties allow for the matching of signals that have variances in temporal intensity and duration. Standard DTW implementations place one restriction on the data, they can only be applied to comparison of 1D time series.

Continuous Dynamic Time Warping

According to Munich and Perora [21], DTW can be improved by allowing matching on continuous curves, instead of discrete samples. They name their improved algorithm, Continuous Dynamic Time Warping (CDTW), with which they acquire smoother results matching handwritten signatures. However, their implementation is three-times slower in comparison with the standard implementation of DTW. Furthermore, the translation of the discrete time-series to a continuous time series is only useful, when the discrete time-series has a low sampling frequency otherwise its better to use normal DTW.

Multi-dimensional Dynamic Time Warping

Dynamic Time Warping may be a very suitable metric for determining the distance between 1D time series. However, according to a number of articles on DTW comparison of n-dimensional time series [3,28], the standard implementation of DTW is not suitable as a metric for comparing n-dimensional time series. Both Holt and Bashir propose a modified multi-dimensional DTW (MD-DTW) algorithm, which should allow for comparison of unimodal n-dimensional time series. The algorithm proposed by Holt is successfully implemented by Mello and Gondra [12], who apply it in a texture image retrieval task. The problem with both proposed MD-DTW algorithms is, they both sum all dimensions for each time step, which results in an 1D signal. This causes both approaches to loose information contained within the dimensionality of the signal being compared.

It is possible to retain the dimensionality of the signal, by altering the function used by a normal DTW that calculates the signal to signal similarities between all data points of the two signals. By considering each data point in a time series to be a point in n-dimensional space and use Euclidean distance to calculate the distance between each point. Donoser et.al. [13] do something similar, they *"use Euclidean distances as measure, where each signal point is defined by the C-dimensional vector containing the similarities to each of the C prototypes."*. The disadvantage of such the modified DTW, is that the computational cost increase in comparison with normal DTW, but this is not an issue with the off-line task.

2.2.6 Classification

The off-line supervised human action detection and recognition task, will need a supervised classification approach for assigning a label to each unclassified feature vector. Two popular approaches for solving supervised classification problems, are: hidden Markov model, and k-nearest neighbor algorithm. Both approaches are discussed in more detail in the text below.

Hidden Markov model (HMM)

The definition of a HMM according to Rabiner and Juang [23] is as follows: *"An HMM is a doubly stochastic process with an underlying stochastic process that is not observable (it is hidden), but can only be observed through another set of stochastic processes that produce the sequence of observed symbols"*. An HMM contains three sets of model parameters, which allow for modeling uncertainty. These parameters can be tuned, too maximize the probability acquired with the model, given an observation sequence. They can either be set manually or by a learning algorithm, such as Baum-Welch [4]. Baum-Welch requires a training set with a fairly large amount of occurrences of each action to successfully (re)estimate the model parameters, so that the model best matches a given test set.

Hidden Markov models have often been used in research for solving supervised classification problems. They either create a single HMM for all actions or a HMM for each action. The two modeling choices require a different approach, when using HMM for classification purposes. A single HMM requires to use of *Viterbi* to determine the path that gives the highest probability. The classifier deduces from this path, which label should be assigned to the observation sequence. With multiple hidden Markov models, the highest probable model for an inputted observation sequence is used for determining the label being assigned to the observation sequence.

An example of a human action recognition task using a HMM as classifier, can be found in the article by Yamato et.al. [34]. Yamato defines a HMM for each action that must be classified and uses the *Forward Algorithm* to determine, which HMM scores the highest probability for a given observation sequence. They maximize the probability of their model by using the automatic learning algorithm of Baum-Welch, for which they allocate a training set that is half of the entire data set. They find that their approach achieves high recognition rates, when training data en test data stars the same person.

K-nearest neighbor (k-NN)

Cover and Hart [9] describe the nearest neighbor decision rule, as assigning a label to an unclassified sample point, based on the nearest previously classified points. Each point can be a n-dimensional vector in n-dimensional feature space. The previously classified points are a set of feature vectors that each have the correct class label assigned to it. Furthermore, the k in k-NN is used to indicate how many nearest neighbors should be taken in consideration, when classifying an unclassified sample point through a majority vote of its neighbors. Using a $k >= 3$ is only useful, when the training set contains multiple occurrences of each class. A $k >= 3$ requires the introduction of a weight function for the majority vote. Dudani [14] discusses a

number of different weight functions, of which he gives the distance-weighted k-nearest neighbor the most attention.

An example of a human action recognition approach using the k-NN as classifier, is the approach by Efros et.al. [15]. They have three different data sets, of which two are classified with $k = 5$ and the last one is classified with $k = 1$. The reason for selecting a certain k for a specific set is not given. Efros seems to get quite good results using the k-NN classifier, considering the input data is of very poor quality. Furthermore, the k-NN decision rule is a fairly simple mechanism, which should make the classification process simpler and still very effective.

Another example of a recognition approach using the k-NN decision rule, is the approach by Corradini [8]. He uses Dynamic Time Warping to match the recorded unclassified feature sequence with the gestures extracted from the training set. The training set does not contain an infinite set of possible gestures and therefore a heuristically determined threshold is used to detect which inputted feature sequences should be classified as unknown. A total of five gestures with each ten occurrences of multiple persons, were selected as the training set. The fact that only five gestures are present in the training set, makes it fairly easy to recognize these movements. Expanding the amount of classes and test data would greatly influence the recognition rate, which also is mentioned by Corradini. Furthermore, his current approach assumes that for each sequences the start and end are known, which is not the case in continuous streaming data and his current approach cannot handle data being inputted in such fashion. However, this approach shows it is possible to use DTW as metric in combination k-NN decision rule and achieve fairly good recognition rates.

Chapter 3

Approach

As is mentioned in the introduction, this master thesis discusses the comparison of two different grid-based features and classification settings, in an off-line supervised human action detection and recognition task. The approach is divided into two phases: feature extraction and action classification. These phases are executed sequentially and the extracted features are used as the input for the action classification phase. The features and the feature extraction algorithm are discussed in section 3.3. The chosen feature extraction approach depends on the video material being used, which is discussed in more detail in section 3.1. The choices regarding action classification are discussed in section 3.4.

3.1 Video material

The video material used for this final project, is a fitness video containing two persons performing an almost identical repertoire of movements that belong to the XCO[1] workout. The video consists of multiple scenes, with each scene having a specific theme. Only the first five scenes are used for this task, because they all are recorded with an almost static background and a similar camera viewpoint. The lighting conditions used in these scenes cause shadows to occur and the skin closest to the light source to become very bright. The background color near those areas are of an almost similar color due to these lighting conditions. All five scenes are given a name according to their theme

ID	Scenes	Left	Right
1	Warming up	Man	Woman 1
2	Lumberjack	Man	Woman 2
3	Twist	Woman 3	Man
4	Shoulder shake	Man	Woman 3
5	Wave	Man	Woman 2

Table 3.1: Instructor in each scene

[1] http://www.xco.nl/

and each scene contains a man and a woman as instructor (see Table 3.1 for an overview). The movements performed in these five scenes are all in a straight-up position and contain almost no occlusion. Each scene starts with the two instructors standing side-by-side in frontal view and most of the motion performed during a scene consists of upper body movements.

3.2 Preprocessing

Each scene consists of two continuous streams of movement made the instructor, which is segmented into discriminative actions. All actions that contain a similar sequence of motion are given the same unique identifier. The goal is to choose a motion sequence as action that reoccurs often and cannot be overlapped by any of the other actions. For example, the instructor is standing still, facing towards the camera, moves his arm in a horizontal movement from the left to the right, and this motion sequence takes about 10 frames. It is possible to have two actions similar in motion, but differing in execution time, to be placed into separate action groups. Each occurrence of an action is recorded in the annotation file, which is considered the ground truth annotation of the video. Table 3.1 shows all sixteen unique identifiers of the annotated actions occurring in the first scene *"Warming up"* and their occurrence in other scenes. The name of each action can be found in Table B.1 (in the appendices).

	Scene 1		Scene 2		Scene 3		Scene 4		Scene 5	
Actions	P = 1	P = 2	P = 1	P = 2	P = 1	P = 2	P = 1	P = 2	P = 1	P = 2
100	7	6								
101	7	6								
200	9	9							8	12
201	9	8							12	12
205	9	9							4	
206	9	9							5	
300	15	15			35	30				
301	22	24			101	73				
302	20	24			100	73				
304	15	14			34	28				
400	4	5								
401	4	3								
402	3	3								
500	12	4								
600	15	14	8	7						
601	15	15	8	8						

Figure 3.1: Amount of occurrences of each action occurring in the warming up.

From the table 3.1 it becomes clear that not every action of the *"Warming up"* occurs in the other four scenes. Furthermore, the amount of occurrences of each action in the first scene itself varies greatly and the fourth scene contains no occurrences of any actions belonging to the first scene. Despite these properties, the first scene is the only scene containing actions that reoccur in the other scenes. The other four scenes have completely no overlap with each other and therefore the video material of the first scene is used as training set and the remaining four scenes as the test set.

3.3 Feature extraction

The feature extraction phase itself is split up into two different parts: image processing and the creation of grid-based features. The primary task of image processing in this approach, is to extract two commonly used features (see sections 2.1.1 and 2.1.2), which are the silhouette and the motion descriptor, from the video material described in the previous section. These two features are used as the basic ingredients for the two grid-based features that are evaluated in this final project.

The silhouette descriptor is extracted from each video frame through background subtraction and all background images are created manually. A reason for not choosing any automated approach, is because the primary focus of this master thesis is on the action classification phase and therefore it is not important how the features are acquired. Furthermore, a problem with all the mentioned automated approaches in chapter 2 is, they do not perform well when a foreground object is relatively stationary in the scenery. Most explored automated approaches require a foreground object to be moving through the scene or a frame to exist were no foreground object is visible, otherwise the stationary parts of the foreground object are considered to be background. Two additional issues with the video material discussed in section 3.1, are the small changes in camera viewpoint and small zooms during the video. This creates a lot of noise in most mentioned automated approaches.

The other feature being extracted per video frame is the motion descriptor. This feature consists of a flow field per frame, which describes the 2D motion in the video. Such a flow field can be determined using an optical flow algorithm and a number of different optical flow algorithms are discussed in subsection 2.1.2. The Horn-Schunck algorithm may have seen many optimizations over the years, but the Lucas-Kanade algorithm is still preferred over Horn-Schunck in this final project. The most important reason is because the standard implementation of Lucas-Kanade is made available in Matlab.

The two grid-based features deduced from the silhouette and motion descriptor are named: histogram of silhouette (HoS) and histogram of flow

(HoF). Both features are created by dividing the silhouette and motion feature up into patches, according to a predefined grid and each patch contains a summarization of the feature data contained within a single patch. The main reason for choosing a histogram representation for both descriptors, is because research by Tran et.al. [29] discussed in subsection 2.1.3 showed that these grid-based features work quite well when used for recognizing human actions in video. Furthermore, the HoS feature is more suitable as generic template, instead of a silhouette descriptor, because it is more robust when comparing two silhouettes that belong to two different persons. The silhouette descriptor takes into account the difference in body shape, which is mostly removed in case of the HoF, by abstracting the silhouette using a grid representation.

3.4 Action detection and classification

The second phase "action classification" extracts actions from the training set and uses those actions to detect and recognize similar segments contained within the test set. The test set consists of a continuous stream of features and the boundaries of each segment are unknown beforehand. A segment is a consecutive sequence of frames and a segment is not allowed to contain any frames that do not have a feature. Probable cause for certain frames to not contain a feature are because the feature extraction phase was unable to extract a usable feature from that video frame, due too much occlusion or background noise. When a segments contains frames missing features, than these segments are not taken into consideration by the action classification phase. A combination of the techniques discussed in section 2.2 is chosen to segment and classify the continuous stream of features.

The chosen combination of techniques for the action classification phase depends largely on the training set. A large variance in duration of each occurrence of each action in the training set could cause problems with certain approaches. It is therefore decided to introduce templates that each are exactly 41 frames long and contain one or more actions. It is possible to define these templates, due to the rhythmic component of the motion performed by the instructor. The templates containing multiple actions are referred to as composite templates (see figure A.1 for an occurrence of template 30) and the others as single action templates (see figure A.4 for an occurrence of template 60). All templates can be grouped into a groups of similar occurrences as with actions. An overview of all different groups of templates is given in Table 3.2 and behind each template the actions and their order per template of that group are displayed. An additional problem that can be deduced from tables 3.1 and 3.2 is that the training set contains an unequal amount of occurrences for both actions and templates, which can vary from only a few to a large amount of occurrences. Training a classifier

Template	Name	Actions	Occurrences
10	Rowing (looking left)	100	7
11	Rowing (looking right)	101	7
20	Tilting 1	200 201	4
21	Tilting 2	201 200	4
25	Fast Tilting 1	205 206 205	3
26	Fast Tilting 2	206 205 206	3
30	Twist	300 304	15
31	Fast Twist 1	301 302 301	8
32	Fast Twist 2	302 301 302	6
40	Load Swing (long)	400	4
41	Load Swing (short,long) 1	401	4
42	Load Swing (short,long) 2	402	3
50	Solid Push (2x)	500	12
60	Diagonal Lumberjack 1	600	15
61	Diagonal Lumberjack 2	601	15

Table 3.2: Templates

on all occurrences could result in a bias towards actions that occur often. It is therefore decided to only select a single occurrence that represents the most average occurrence of each template.

An approach mostly using only a single occurring templates/segment is *video indexing*. The basic idea of *video indexing* is to find all segments in a video that show similarities to a preselected segment. The human action detection and recognition task created in this final project, is a kind of *video indexing* approach. Each template is compared with all possible segments in the continuous stream of features. This comparison delivers a matrix with a distance for each unclassified segment to each template.

The comparison between each template and each unclassified segment is performed with a modified version of the standard Dynamic Time Warping metric, which is introduced in subsection 2.2.5 under paragraph *"Multi-dimensional Dynamic Time Warping"*. An n-dimensional metric is required for comparison, because both grid-based features are n-dimensional. Furthermore, the same modification can be done to the *"Continuous Dynamic Time Warping"* (CDTW) metric, but Munich and Perora [21] mention that their metric is three-times slower in comparison to the standard implementation of DTW. Furthermore, the improvement achieved with CDTW is only minimal, because the sample rate of video is quite high. It is therefore decided not to use n-dimensional CDTW and use the n-dimensional DTW instead.

The best scoring segments per template are selected, merged into a single set and finally any overlap is removed. The overlap is removed by select-

ing the best scoring template for each segment and removing all entries that overlap with better scoring segments. Additional output of the n-dimensional DTW is the warping path, which represents the most optimal alignment between two time-series. The exact starting frame of each action within a template combined with the alignment information contained within the warping path, is used to deduce the starting frames of each action in the selected segment. Each action is labeled separately, which results in a list of starting frames with action labels assigned to them. These classification results are compared with the ground truth annotation of the entire video, which results in a precision and recall score per template.

Chapter 4

Feature extraction

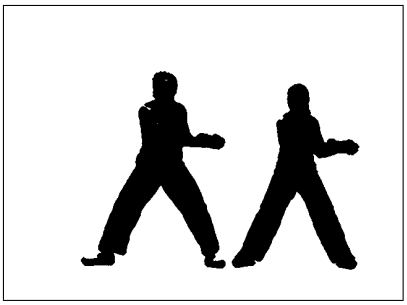
The basis of every action recognition approach, is to extract one or more features that describe the video material in a format, which can be utilized by the action recognition algorithm. These features can either be acquired directly from the video material or based on other features. The features extracted with the approach taken in this research, result into two grid-based features. These two grid-based features are each based on a more common low-level feature, namely the silhouette and the motion descriptor. The acquisition of silhouette and motion descriptors is discussed in sections 4.1 and 4.2. The two features used in the classification process are discussed in section 4.3.

4.1 Silhouette descriptor

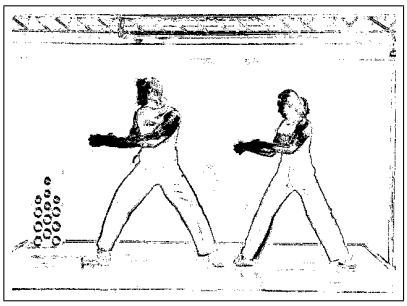
Two of the silhouette extraction approaches discussed in section 2.1 are used for extracting a silhouette descriptor per frame from the video material elaborated in section 3.1. These two silhouette extraction approaches are Mixture of Gaussians (MoG) and background subtraction, both are applied on the two frames shown as sample in figure 4.1. Both samples acquired with MoG show that this method does not perform very well with the video material being used. Furthermore, the large amounts of noise in figure 4.1.3 are caused by small camera movements in preceding video frames. The background subtraction approach delivers much better results, which is because this approach is fine tuned to the inputted video material.

A color-based background subtraction approach is used to acquire the silhouette descriptor. This approach consists of eight steps and starts off with the creation of a background image, which is subtracted from each video frame to determine the pixels belonging to a foreground object. Multiple background images are required, because the camera is occasionally altered during the recording, which results in slightly different background images. The second step consists of the actual subtracting of a background

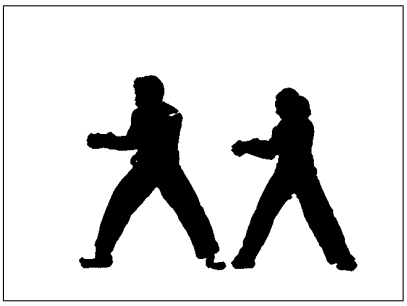
(4.1.1) Sample 1 Mixture of Gaussian



(4.1.2) Sample 1 Background subtraction



(4.1.3) Sample 2 Mixture of Gaussian



(4.1.4) Sample 2 Background subtraction

Figure 4.1: Sample silhouettes extraction results

image from each frame in the video. The subtraction is done per color channel, which gives a better result, instead of using the gray scale version of the frame. The third step divides each subtraction result into predefined rows with various heights and these rows are specified per background image. Each row is converted separately into a binary image, assigning ones to foreground and zeros to background. The decision to classify a pixel as foreground is made according to a predefined threshold for each row. Each row roughly matches with an area that consists of fairly the same color. The reason for subdividing the image into rows is, because foreground pixels are classified according to color difference, which can vary greatly across an image, due to great variance in color in background and foreground. Figure 4.2 displays frame 2000 including the row subdivision used by the feature extraction phase. This figure clearly shows that each row roughly matches with an area that differs greatly from the other areas. The second and third row could have been combined, but the second row is added because the lighting conditions cause all foreground objects to become very light. The threshold for row two is a little lower, than the threshold for row three because the difference between background and foreground is minimal.

Figure 4.2: Frame 2000 of warming up including subdivision

The follow up step consists of merging the binary classified rows into a single binary image. The consecutive steps that follow, are: erosion, small blob removal, and finally dilation. The first two steps are taken to remove most noise still present in the binary image and the third step is taken to fill up any holes caused by eroding the binary image. Figures 4.1.2 and 4.1.4 show two samples acquired with the previous seven steps. The eight and final step, is to find the regions of interest that contain a silhouette of a

foreground object, which in this particular case is a human. These regions of interest are cropped from the binary image and passed on for further processing.

4.2 Motion descriptor

For each video frame a single motion descriptor is determined, which is a flow field determined with an optical flow algorithm. The optical flow algorithm used is the local differential algorithm *Lucas-Kanade* [20], for which an implementation is incorporated in Matlab. This implementation allows for a number of different settings to be configured, the setting *"TemporalGradientFilter"* is set to *"Derivate of Gaussian"*, which means that the image sequence is smoothed using a spatio-temporal Gaussian filter before processing. The reason for smoothing with this filter, is to resolve aliasing occurring in the image sequence. The standard deviation for the spatio-temporal Gaussian filter is set to five. Furthermore, the setting *"DiscardIll-ConditionedEstimates"* is to "true", which means a normal flow estimate is discarded when the constraint equation for the spatio-temporal Gaussian filter is ill-conditioned. The combination of these previously mentioned settings and the default settings, seems to deliver the most suitable result for further processing.

4.3 Grid-based features

The feature extraction process results in two different grid-based features, as mentioned earlier in chapter 3, and these are the HoS, and the HoF. They both split up a selected region of interest into patches according to a predefined grid and extract features representing each single patch and merge it into a single feature representing the entire region of interest. The creation of both grid-based features is discussed in the two subsections that follow.

4.3.1 Histogram of Silhouette

As mentioned in the subsection 2.1.3 of chapter "Related Work", a *"the histogram of silhouette contains per patch the percentage of pixels that are occupied by the silhouette."*. The HoS is constructed by simply dividing each extracted silhouette descriptor into patches according to a predefined grid and for each patch count the total amount of foreground pixels and divide them by the total amount of pixels per patch. These values per patch are taken together, which results in a vector and this vector is considered to be the HoS.

4.3.2 Histogram of Flow

The region of interest selected with the silhouette descriptor is used to select the same region of interest in the flow field. Any flow line that lies outside of the pixels classified as foreground, is considered background movement and is removed by setting the length of the vector to zero. As with the HoS, the entire region of interest is split up into patches according to a predefined grid. All flow lines in a patch are split up into their horizontal and vertical components and are summed together in one of the following categories: horizontal negative, horizontal positive, vertical negative and vertical positive direction. The four categories are chosen, because research by Efros et.al. [15] shows this division is an effective representation of motion made by a human. An additional step is taken to scale down the values within each component, by dividing each channel by the total amount of pixels classified as foreground, which gives the mean motion per patch. These values are taken together for all patches, which results in the HoF feature.

Chapter 5

Action classification

This chapter discusses the action classification phase of the human action detection and recognition task. It starts with the extraction of the templates from the training set discussed in section 5.1 and follows up with a description of a video indexing based approach in section 5.2.

5.1 Templates

The decision has been made to create templates with a fixed size of 41 frames, which are allowed to contain one or more actions. These actions cannot overlap with each other and are required to be completely contained within the boundaries of the template. Actions are chosen to be discriminative and therefore should have no overlap with other actions. The restrictions put on these templates are enforced through the use of an annotation file, which defines the start and end frame of each template and each individual action.

All annotated templates are extracted from the training data and similar templates are grouped together. Two templates are similar, when both contain exactly the same actions in the same order. See Table 3.2 for an overview of all template groups. For each group of templates the most average template is determined. The most average template is selected by calculating the distance between each template within each group, using n-dimensional DTW. These distances are summed up to a single distance for each template and finally selecting the template per group with the lowest summed distance as the most average template.

5.2 Video indexing

The templates acquired in the previous step are compared with a continuous stream of features, which can be temporally segmented before comparison. Various automatic motion boundary detection approaches can be used to

segment this stream, for example the approach proposed by Ali and Aggarwal [1] who use key-poses. Another approach is to combine comparison and temporal segmentation into a single step, by applying a video indexing based approach. With video indexing all similar occurrences of a selected segment are searched for in the rest of the video. This section describes an approach that resembles video indexing.

A sliding window with a fixed size of 41 frames and a step size of one frame is used to select all possible segments in the continuous stream of features. Each segment is compared with each template using the modified n-dimensional DTW, which allows for variety in the temporal and spatial domain to exists between the two n-dimensional time-series. Choosing a fixed window size overcomes issues introduced by variable segment sizes. Figure 5.1 shows a plot with distance values for two thousand segments being compared with template 60. The lower the distance, the better a templates matches with that particular segment.

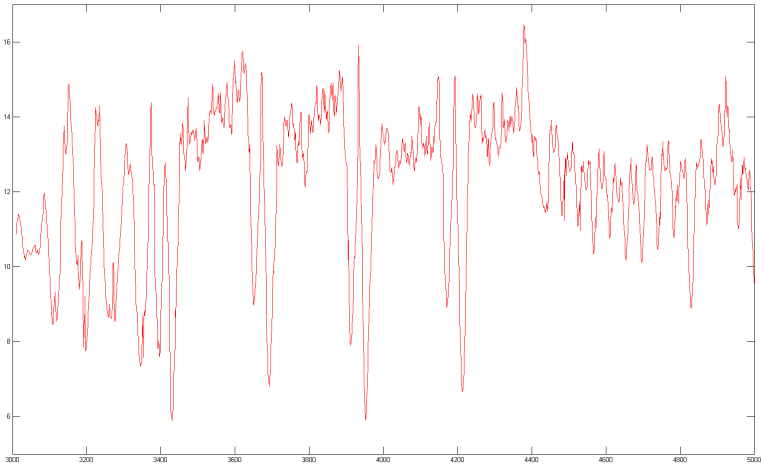
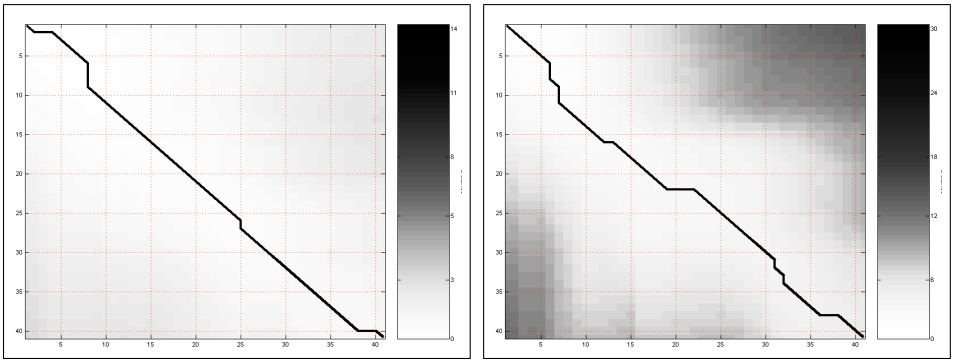


Figure 5.1: Distance of segments starting at frame 3000 till 5000 compared with template 60

Figure 5.1 depicts a wave pattern in which the lowest point in each valley (called local minima) is the best scoring segment, in comparison to segments represented by all other distance values in the same valley. These local minima are acquired by taking the derivative of the signal and finding all points in time where an increase from below zero to above zero occurs. A threshold mechanism is used to only select the segments that are considered to be good matches with a specific template. Each template can have its own threshold value and determining these threshold values is explained in chapter 6. The resulting sets of segments per template are merged into a single

set. This single set can contain overlap, which should be removed, because the actions within a template are discriminative actions. Removing overlap may cause some well matching segments to be removed, but this approach will also remove a lot of partial hits. An overlapping segment is removed, when this segment has a higher distance value than the other segment it overlaps with. The constant size of each segment and template makes this way of comparing for overlap removal possible, because smaller time-series being compared with DTW typically have smaller distance values.



(5.2.1) Warp path example 1 (5.2.2) Warp path example 2

Figure 5.2: Two warping paths outputted by DTW

For each remaining segment the warping path is retrieved, which can be acquired using DTW. The warping path indicates per frame the frame it best matches with. Figure 5.2 shows two examples of two segments being matched with template 20 and 21, which are composite templates. For both templates the exact starting point of each action contained within the template is known. The warping path in combination with the starting frame of an action is used to determine at which frame this action starts in the segment. This is done for all segments, which results in a list of actions and their starting frame. This list of actions including their exact starting frame is the classification result, which is used as input for the evaluation.

Chapter 6

Evaluation

The evaluation discussed in this chapter evaluates different classification settings. This evaluation is a system evaluation, which means that no human participants are required for evaluating the system. The different classification settings for the task are introduced in the form of variables in section 6.1. Section 6.2 gives a short introduction into what information retrieval measures are used to answer all research questions. Each question is stated in section 6.3 including a brief outline on how to find the answer to each question.

6.1 Variables

The are a number of different variables that taken into consideration during this evaluation. Two variables can be deduced from the video material present in the test set, by examining some of the meta data displayed in Table 3.1. From this table it is possible to deduce that each scene contains the same man performing the movements and the other position in each scene is filled up by three different women. Furthermore, the position of each person in the scene is indicated and can either be left or right. The two persons never interchange position and therefore this indication stays valid for the entire scene. The two variables that are deduced from the test data are the scene and the person performing the actions. Two other variables that play a role in this evaluation are actions and templates. All possible options for actions and templates are displayed in Tables 3.1 and 3.2 and these templates can be subdivided into two different groups, they are either a single action template or composite template.

Another variable that plays a role in this evaluation, is the configuration of the grid of a grid-based feature. In total, four different grid configurations are introduced for both grid-based features, to be evaluated on performance in comparison with each other. Each grid configuration specifies how the region of interest must be divided and all four will divide the region of

interest into three rows. Two of these grid configurations have an alternative row division based on a heuristic proposed in the article by Danafar and Gheissari [11], which is: 1/5 (head), 2/5 (body) and 2/5 (legs). This means that for example the top row will have a vertical length of 1/5 of the total height of the region of interest. This heuristic is based on a person standing straight up, but it looses its advantage when any other type of pose or motion is executed, for example stooping or lying on the floor. The other two grid configurations specify that all rows in a grid are to be of an equal height. For the with and without heuristic grids, two different column sizes are chosen, namely three or five columns. These two amounts are chosen because a trade-off is made between specificity and generalization, and the column sizes of three and five are expected to give the best results.

The final variable that plays a role in this evaluation is the type of threshold being applied by the classifier. In total three different thresholds mechanisms are evaluated and these are the following: best scoring 2,5%, single value threshold, and single offset value threshold. The first threshold mechanism is based on the assumption that the lowest 2,5 percent of all local minima segments must be a good matching with a template. This very simple threshold is expected to perform worse than the other thresholds, but is included for comparison. The height of the other two thresholds is found through calibration using a validation set, which consists of the performance of woman 1 in scene 1. A validation set must contain occurrences of all action present in the training set, which is not the case with any of the scenes in the test set. The training set consists of two persons performing a similar repertoire and is therefore split up into a training and validation set. A possible issue caused by the chosen validation set, is that the threshold values are based on a single person, while multiple persons are present in the test set. The second and third threshold can vary for each combination of classification settings (template, grid configuration, and grid type).

The calibration of the second threshold is performed using the distance values and warping paths calculated with DTW for each segment in the validation set. The local minima are determined in the same way as is elaborated in section 5.2 and for each segment (local minimum) the warping path is used to determine if the starting point of each action in a segment matches with that of the ground truth annotation. This process delivers a target list that indicates which segments are considered to be a correct match. An ROC-curve is determined for each possible combination of settings, an example of an ROC-curve is displayed in Figure 6.1. The most optimal threshold value for each combination of classification settings is determined, by calculating the f-measure (is explained in section 6.2) per step in the ROC curve. A threshold is associated with each step in the ROC curve, this threshold is a value between the lowest distance value and the highest distance value of all entries used to determine the ROC curve. The threshold that belongs to the entry with the highest f-score is selected as

the value for the single value threshold. The third threshold mechanism is based on this single value threshold, because its value is the offset between the best scoring segment and the single value threshold.

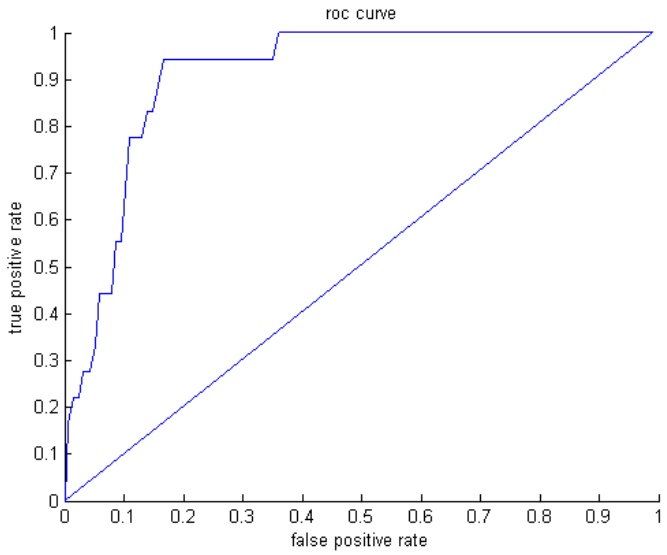


Figure 6.1: ROC-curve for template 60, HoF configured with a 3x3 with heuristic

6.2 Information retrieval measures

Most research questions in this evaluation are answered with the help of three well-known information retrieval performance measures, which are recall, precision, and f-measure. Recall indicates the sensitivity of the classifier, by indicating the percentage of actual classification entries present in the predicted classification. Precision, on the other hand, indicates the percentage of predicted classification entries that are correctly classified. F-measure uses recall and precision to indicate the accuracy of the result acquired with the classifier. Recall and precision base their values on the following variables: true positive, false positive, and false negative. These three variables are determined for each separate combination of classification settings.

The information retrieval step in the evaluation compares all predicted classification entries with the actual classification entries. All predictions that match with the actual classification are counted as true positive. A predicted starting frame of an action is not required to match exactly with the annotation. The annotation of the video is subject to human interpretation, while DTW is much more constant in its findings. It is therefore decided to introduce a margin of three frames (referred to as the annotation window), which boils down to a 7.2 milliseconds margin before and after

each annotation entry to overcome any inaccuracy in the annotation file. A margin of three frames is chosen, because the smallest occurring action is seven frames long and choosing a larger margin would could cause overlap in the margin windows of two consecutive actions.

For each predicted entry that has no corresponding entry in the actual classification, a rest class is introduced and added to the actual classification. These incorrect predicted classification entries are all marked as false positives and the false negatives are all actual classification entries that are not found in the predicted classification. It is possible to create a more informative overview of the actual and predicated classification set, which is called a confusion matrix and is used as a tool for visualizing the actual and prediction classification. The rows are the actual classes and the columns the predicted classes. Each ID belongs to an action and this mapping can be found in table 3.1 and the names of each action can be found in Table B.1.

6.3 Research questions

In total this evaluation tries to answer four research questions. The first two questions have a similar goal and that is to determine two variables for further questions, which are threshold mechanism and grid configuration. The following abbreviations are used often in this section and section that follow, namely TP (true positive), FP (false positive), and FN (false negative).

6.3.1 Question one

Which of the three threshold mechanisms delivers the best result?

Take all TP, FP, and FN calculated for each template and group them per combination of grid configuration, grid type and threshold mechanism. Calculate the f-measure per combination using these three value. Group all calculated f-scores into two groups based on grid type and create a box plot for each. The variables in each box plot are the threshold mechanisms, which result into three boxes that contain four f-scores. The best scoring threshold mechanism is the one with the highest median and if these are equal the highest third quartile. Per box plot (HoS or HoF) the best scoring threshold mechanism is selected and is fixated for all further questions.

6.3.2 Question two

Which of the four different grid configurations can be considered the best choice for the HoS and which one for the HoF?

As with question one, take all TP, FP, and FN calculated for each template and group them per combination of grid configuration and grid type. Calculate per combination recall, precision, and f-measure. Display recall, precision, and f-measure in a table per grid type or a single combined table.

Deduce from this table or tables, which grid configuration is the best choice for a grid type. The best scoring grid configuration per grid-based feature is fixated for all further questions.

6.3.3 Question three

Which grid-based features scores better (HoS or HoF), using only video footage of the male subject?

The first step taken to answer this question is to create an overview of all f-measures per action. This information can be displayed in a table per grid type or a single table containing both grid types. All actions that do not occur in the test data are removed from this overview (see table 3.1), because the precision, recall, and f-scores are all zero. The following step is to examine each scene separately and check per scene the confusion matrix for each grid type and individual classification results. Deduce from this in depth information per scene and the overview the answer to question three.

6.3.4 Question four

How well do both grid-based features perform when another subject is used for testing?

Gather all information for the second person in each scene two till five, which are woman 2 and woman 3 (see Table 3.1). The second person in each scene has a fairly similar repertoire in comparison to the first person (man). Therefore it is possible to compare most of the acquired f-scores per action between the man and one of the two woman. This comparison and the previous information gathered in the other questions should be sufficient to answer question five.

Chapter 7

Results and Discussion

7.1 Threshold selection

Which of the three threshold mechanisms delivers the best result?

The two box plots in Figure 7.1 each represent a grid type and the boxes in both box plots represent the distribution of the f-scores per threshold mechanism. From these two box plots it possible to deduce that the single value (second) threshold mechanism delivers the best results, with both the HoS and the HoF grid-based feature.

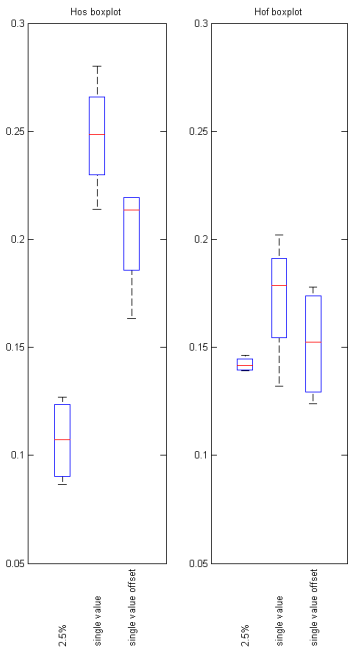


Figure 7.1: Box plot

The 2,5% threshold seems to be the worst scoring threshold of the three thresholds, which was expected beforehand. The 2,5% threshold is based on the heuristic that the best scoring 2,5 percent of all occurrences in the video must be a correct prediction. However, when a video contains no occurrence or too many of a particular action, this threshold will introduce a large number of false positives or negatives. The single value offset threshold is a value that indicates the height of the threshold relative to the best scoring segment. This threshold is based on the heuristic that the best scoring segment must be an occurrence of that action. This threshold fails when no occurrence of an action is present in the video. which results

in a lot of false positives to be included. A disadvantage of the selected threshold is that its height is based on its data set, which means that it must be calibrated every time a completely different data set is introduced.

7.2 Grid configuration selection

Which of the four different grid configurations can be considered the best choice for the HoS and which one for the HoF?

From Table 7.1 it is possible to deduce that both recall and precision increase for HoS, when the amount of columns in the grid increases. Therefore the possibility exists that choosing an even larger grid size in both rows and columns will result in even better results. The possibility exists that an increased grid size could introduce overfitting for the subject "Man", because the feature becomes more specific and detailed. Furthermore, the without heuristic grids seem to be scoring better for HoS, than their with heuristic counter parts. This could point to the fact that the recall and precision decreases when the patch size increases. It is therefore decided to choose the 3x5 without heuristic grid as the grid configuration for the HoS feature for further research questions.

		3x3 WH	3x3 WhH	3x5 WH	3x5 WhH
HoS	recall	0.1746	0.2156	0.2275	0.2421
	precision	0.2762	0.2860	0.2811	0.3321
	f-score	0.2139	0.2459	0.2515	**0.2800**
HoF	recall	0.1773	0.1005	0.1085	0.1151
	precision	0.2351	0.1929	0.4767	0.4163
	f-score	**0.2021**	0.1322	0.1767	0.1803

Table 7.1: Score per grid configuration

The highest f-score with HoF is achieved using 3x3 with heuristic as grid configuration. The values in Table 7.1 show that the precision increases dramatically when increasing the amount of columns. The recall on the other hand scores badly with any of the tested grid configurations and only grid configuration 3x3 with heuristic achieves an acceptable value for recall. The heuristic does not seem to have any influence on the recall, but precision scores better with both with heuristic configurations.

7.3 Comparing both grid-based features (Man)

Which grid-based features scores better (HoS or HoF), using only video footage of the male subject?

The first step consists of comparing recall, precision and f-score per action per grid type. The only actions that are shown in the overview (Table 7.2) of HoS and HoF are actions 200, 201, 205, 206, 300, 301, 302, 304, 600, and 601. The other actions are not included into the analysis, because they do not occur in the test data.

		200	201	205	206	300	301	302	304	600	601
HoS	TP	1	2	4	5	8	39	39	12	4	4
	FP	1	0	0	0	41	85	56	37	4	1
	FN	10	10	0	0	23	96	96	17	4	4
	P	11	12	4	5	31	135	135	29	8	8
	recall	0.0909	0.1667	1.0000	1.0000	0.2581	0.2889	0.2889	0.4138	0.5000	0.5000
	precision	0.5000	1.0000	1.0000	1.0000	0.1633	0.3145	0.4105	0.2449	0.5000	0.8000
	f-score	0.1539	0.2857	1.0000	1.0000	0.2000	0.3012	0.3391	0.3077	0.5000	0.6154
HoF	TP	6	4	2	4	9	29	33	9	6	5
	FP	1	3	0	0	5	28	30	5	20	2
	FN	5	8	2	1	22	106	102	20	2	3
	P	11	12	4	5	31	135	135	29	8	8
	recall	0.5455	0.3333	0.5000	0.8000	0.2903	0.2148	0.2444	0.3103	0.7500	0.6250
	precision	0.8571	0.5714	1.0000	1.0000	0.6429	0.5088	0.5238	0.6429	0.2308	0.7143
	f-score	0.6667	0.4211	0.6667	0.8889	0.4000	0.3021	0.3333	0.4186	0.3529	0.6667

Table 7.2: Overview of male footage per actions for features HoS and HoF

Actions 200 and 201 seem to get recognized better with HoF as feature, but the amount of false positives also increases. As with actions 200 and 201, HoF achieves a higher recall for both actions 600 and 601, but a lower precision. An extremely large amount of false positives is found for action 600, while action 601 has only a few false positives. The threshold calibrated for action 600 is on 10.1265, while the threshold for action 601 is on 7.5182. These two actions are fairly similar, but it seems that badly scoring true positives in the validation set caused the score to increase dramatically. The thresholds for both actions using other grid configurations are much closer together, which could point to a few bad performances marked as true positive in the validation set. Template 30 consists of actions 300 and 304 and both achieve higher f-scores with HoF, because the precision with HoS is very low. Action 302, on the other hand, achieves a slightly higher f-score with HoS, because the high recall makes up for the large amount of false positives.

All TP, FP, and FN shown in Table 7.2 are summed up per grid type and per grid type recall, precision and f-measure are calculated (see Table 7.3 for the results). Purely looking at these values, HoF is the better grid-based feature to use with this video material and this subset of actions. The height of these two f-scores is primarily determined by actions 300, 301, 302 and 304, because they occur more often. These actions also have a large influence on the accumulated f-scores calculated in the previous two questions. To get a better insight into the actual performance per scene, it is decided to examine all scenes separately, before giving a more definitive answer to question 3.

	HoS	HoF
Recall	0.3122	0.2831
Precision	0.3440	0.5323
F-measure	0.3273	0.3696

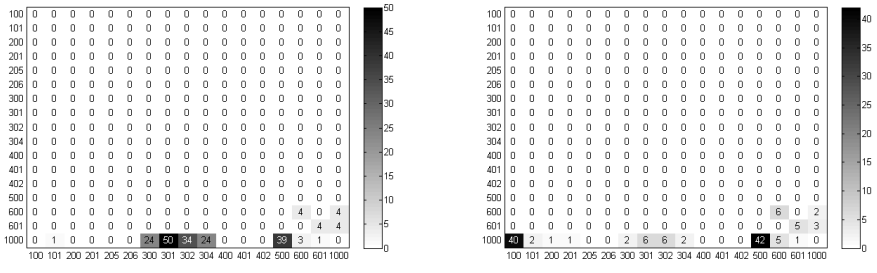
Table 7.3: Scores Man

7.3.1 Scene 2: Lumberjack

First off, two confusion matrices for HoS and HoF (see Figure 7.2) are examined. The rows are the actual classes present in the scene according to annotation and the columns represent the predicted classes by the classifier.

The confusion matrix for the HoS feature contains a large number of false positives in the remainder class, especially actions 300, 301, 302, 304, and 500 seem to score well with a large number of segments in scene 2. Most of the segments marked as false positives for action 500 visually resemble this action, which makes the introduction of a new template necessary to decline the amount of false positives for action 500. To be able to explain the other false positives on actions 300, 301, 302 and 304, a more in-depth examination is required.

Frame 2639, frame of a segment matched with template 30 (consists of actions 300 and 304), is compared with its corresponding frame of template

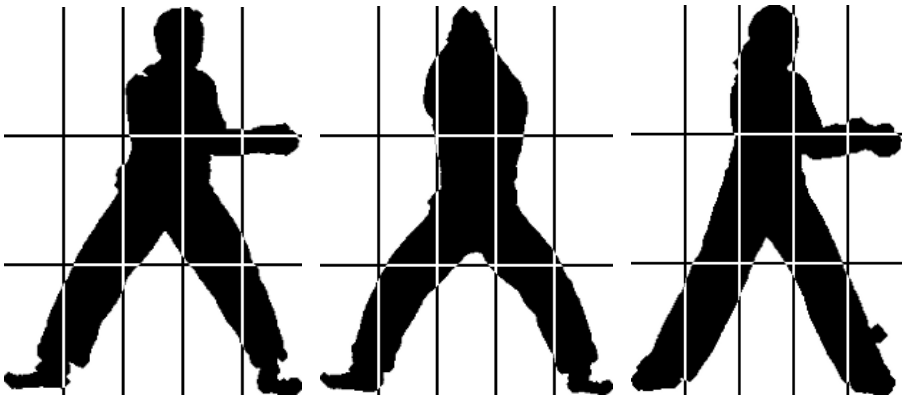


(7.2.1) HoS 3x5, threshold 2, Scene 2 (7.2.2) HoF 3x3(h), threshold 2, Scene 2

Figure 7.2: Confusion matrices scene 2

30 (see first two figures in Figure 7.3). When visually comparing both HoS features, it is clear that both have almost exactly the same shape and pose of the legs and torso. The only major difference between the two, is the arm on the right side and the patch that contains the head. The threshold for template 30 is calibrated at 18.1610, while the DTW distance between template 30 and segment 2636 is 13.6179 and therefore this segment is matched with template 30. This threshold is far above the distance calculated for segment 2636, which could be an indication that this threshold is too general or the template is too general. A possible reason for this fairly high threshold, is because all thresholds are calibrated using footage of a different person (Woman 1). It could be that the difference between the two subjects influences the height of the calibrated threshold to much. Frame 1220 is a frame of a segment marked as a true positive in the validation set. This frame is displayed in Figure 7.3.3 and the other two are displayed in Figures 7.3.1 and 7.3.2. Calculating the euclidean distance using HoS as feature results in a distance of 0.4802 between frame 4153 and 2639 and an euclidean distance of 0.5314 between frame 4153 and 1220. The other subject (Woman 1) clearly causes the euclidean distance to be slightly higher, which can also be deduced from visually inspecting these frames. The slight difference in body shape and pose is present in the entire frame, which causes all patches to differ, instead of only a few patches as with frame 2639.

When examining all silhouette descriptors for all three segments in Figures A.1 to A.3 (in the appendices), a difference per frame and in the overall performance between template 30 and segment 1217 can be observed. Woman 1 seems to have a delay of two/three frames in comparison to template 30, which would be a problem when euclidean distance was used to compare segments with templates, but DTW should be able to warp the segment. It can be concluded that the silhouettes used as template for action 300 and 304 are not discriminative enough and together with the calibrated threshold this template will match easily with a large number of segments



(7.3.1) Template 30 (4153) (7.3.2) Seg. 2636 (2639) (7.3.3) Seg. 1217 (1220)

Figure 7.3: HoS grid-based feature for the fourth frame of three different segments

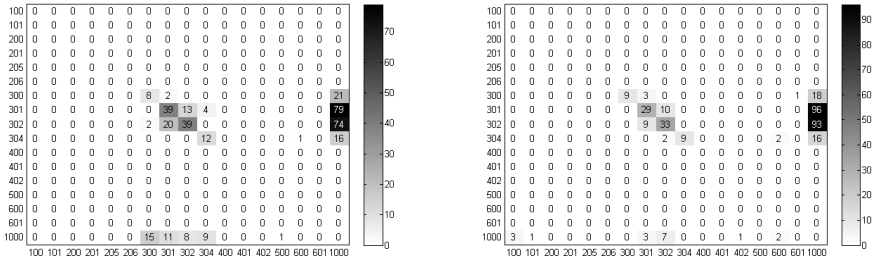
in most scenes. The false positives found for actions 301 and 302 suffer from the same kind of issue, because the same kind of movement is performed at a different speed.

Half of all the occurrences for action 600 and 601 are found using HoS as feature. Both actions occur mostly in pairs of two of the same action and the first entry never seems to get predicted, while the second entry always gets predicted by the classifier. Looking at the video material itself, it shows that the transition from another movement into the first action of each pair has a significant influence on the manner it is performed. Furthermore, comparing the silhouette descriptors of action 600 with a FN (see Figures A.4 and A.5) shows a few differences in the pose: the FN segment seems contain a subject that bends over a little to the left, has his arm closer to his head and misses his right foot in some of the frames. Three of the false positives for action 600 and the single FP for action 601 are all part of a slower version of these two actions.

The confusion matrix (see Figure 7.2.2) also shows a fairly large amount of false positives in the remainder class for actions 100 and 500. Both actions consist mostly of motion in the vertical component and many of the false positives display a similar kind of motion, which is the most probable cause for these segments to be predicted as one these two actions. A portion of the false positives for action 100 does not match with this action, which could point to a threshold that is calibrated too high.

7.3.2 Scene 3: Twist

From the confusion matrices it can be deduced that a large amount of occurrences of actions 300, 301, 302, and 304 are not found with both grid-based features. This scene also has three false positives with HoF and one FP with



(7.4.1) HoS 3x5, threshold 2, Scene 3 (7.4.2) HoF 3x3(h), threshold 2, Scene 3

Figure 7.4: Confusion matrices scene 3

HoS, which are annotated as actions 300 and 304, but predicted to be actions 600 or 601. After inspecting all four false positives no real resemblance can be found in shape or motion between each segment and the predicted actions. This scene is contains a large amount of performances of actions 300, 301, 302 and 304, which increases the chance that a FP matches with an annotated entry instead of the remainder class.

Furthermore, a large number of false positives found with HoS for actions 300 and 304 are caused by a minor mismatch in execution speed between the annotated actions and the template. Only a part of the template is predicted successfully, while the other action(s) just falls one or two frames outside the annotation window. The course of action taken to solve the mismatch in speed is to use the warping path and find the actual starting frame inside a segment. It seems however that no sensible warping path can be deduced by DTW when using HoS as feature, which was also the case with the previous scene.

Every segment that is matched with either template 31 or 32 (containing actions 301 and 302) has at least a single FP. The difference in execution speed between actual and template performance is also present for these templates. Both templates 31 and 32 consist out of three actions, but the big difference in execution speed causes the segments found by the classifier to contain four or five actions. These segments can contain four or five actions, because most occurrences of actions 301 and 302 occur in large consecutive sequences (see Table 7.4 for a small sample). In most cases only two actions are predicted correctly and the sample of the classification below illustrates the cause for most false positives and a few of the false negatives found for actions 301 and 302.

Two segments fall inside the range of the sample from the annotation file (displayed in Table 7.4). The first segment starts at frame 1800 and ends at frame 1840, it is matched with template 31. The second segment starts at frame 1847 and ends at frame 1887, this segment was successfully matched

Start	End	Action
1769	1785	300
1787	1802	304
1817	1826	302
1827	1835	301
1836	1844	302
1845	1853	301
1854	1863	302
1864	1872	301
1873	1881	302
1882	1890	301
1891	1899	302

Table 7.4: Sample annotation of scene 2

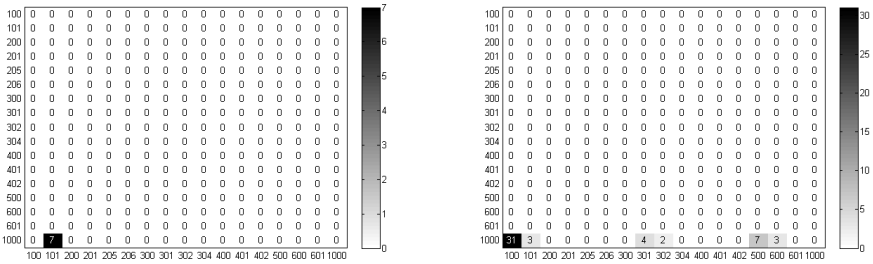
with template 32. Templates 31 and 32 are both composite templates and therefore each action has a different starting point within the segment. The actions for the first segment start at 1800 (301), 1818 (302) and 1826 (301) and for the second segment at 1847 (302), 1864 (301), and 1874 (302). The annotation contains a total of three entries 1817 (302), 1827 (301), and 1836 (302) that fall inside the range of the first segment. The range starts at first frame of the segment minus the annotation window and ends at the last frame of the segment plus the annotation window. The first two annotation entries are predicted by the last two actions in the segment, but no annotation entry can be found for 1800 and therefore is marked as false positive. The annotation entries that fall inside the range for the second segment are: 1845 (301), 1854 (302), 1864 (301), 1873 (302), and 1882 (301). Actions starting at 1864 and 1874 are successfully predicted, while action 302 predicted to start at 1847 is matched with an action 301 that actually starts at 1845. The other two annotation entries 1854 and 1882 have become unreachable for any other segment, because no overlap between segments is allowed. This a general problem when using composite templates that allow overlap, while the classifier does not allow overlap.

Action 302 predicted at 1847 should have been matched with action 302 annotated to start at 1854, but the difference in speed caused it to mismatch. The warping path calculated by the DTW should have solved this false positive, but apparently no usable warping path can be deduced when comparing time-series of HoS features. Furthermore, the overlap between templates in combination with overlapping removal causes many of the false negatives for scene three. The matched templates are not in consecutive sequence, instead gaps are left between matched segment and these gaps are too small to contain an additional segment. However, these gaps span annotated entries, which causes a large number of false positives.

The amount of false positives found with HoF is much less in comparison to the results acquired with HoS as feature. Especially actions 300 and 304 seem to be predicted precisely as they are annotated and in some cases the warping path deduced with DTW is used to solve any issues that occur due to difference in speed between a segment and the template. A usable warping path could not be deduced when using HoS as feature, but with HoF this problem does not seem to exist. Overall it seems that HoF has less false positives, but is not able eliminate all false positives using DTW.

7.3.3 Scene 4: Shoulder shake

Scene 4 is the only scene that does not have any occurrence annotated for any of the templates used by the classifier, which means the classifier should not find any segments matching its classes. The confusion matrix for HoS only displays seven false positives for action 101. HoF on the other hand seems to generate a lot more false positives, especially action 100 seems to match with a lot of segments.



(7.5.1) HoS 3x5, threshold 2, Scene 4 (7.5.2) HoF 3x3(h), threshold 2, Scene 4

Figure 7.5: Confusion matrices scene 4

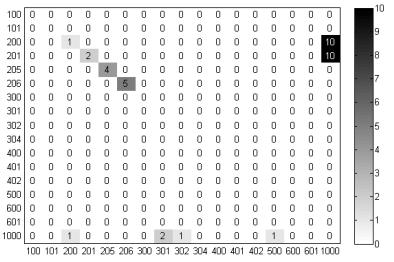
All false positives found using HoS visually resemble the action performed in the template. The performance speed, the camera angle, and position of the hands during the entire performance almost match entirely. A slight difference is visible in the starting position of the hands, instead of behind (left of) the body as with the template, they are right next too the body. This difference seems to be insignificant enough for these segments to match with action 101.

Some of the false positives with HoF for action 100 resemble the movement, but most segment contains movements that do not resemble the template. The high amounts of false positives for action 100 in this scene and the previous ones could indicate to a threshold that is too high. Action 101 is similar to action 100 in motion and differs mostly in orientation. The difference in threshold between these two action is significant (action 100: 9.5543; action 101: 7.8219), which is most likely the cause for the large

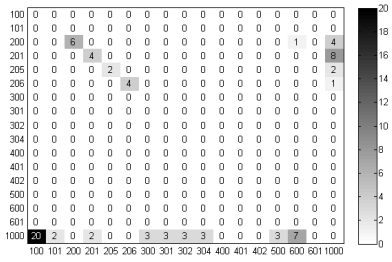
amount of false positives. Furthermore, all false positives found for action 500 contain motion that is very similar to the motion of this action. Most of the other false positives found using HoF do not resemble the action there matched with. Most of these predictions are caused by effects of motion by body parts, such as a feet/leg that wobble on the rhythm of the music or an arm that is used to give instructions.

7.3.4 Scene 5: Wave

Templates 20, 21, 25, and 26 actually occur in scene 5. These templates consist of actions 200, 201, 205, and 206. The exact composition of each template can be found in Table 3.2 and the confusion matrices can be found in Figure 7.6.



(7.6.1) HoS 3x5, threshold 2, Scene 5



(7.6.2) HoF 3x3 (heuristic), threshold 2, Scene 5

Figure 7.6: Confusion matrices scene 5

A large amount of the occurrences for actions 200 and 201 are not predicted by the classifier. After close inspection it is clear that the video material never contains a complete performance of 200 followed by a 201 or the other way around. All occurrences are grouped in three subsequent actions, either – 200,201,200 – or – 201,200,201 – are recorded in the annotation file. However, the first and the last action always stops halfway. In most cases this halfway performance causes the distance value to be above the threshold. The best solution would be modify actions 200 and 201, by letting the new action 200 start halfway the original 200 and end halfway the original 201 and for the new action 201 the other way around. These actions would better fit the test data and increase amount of true positives. Furthermore, it seems as with scene 3 that overall the classifier using HoS as a feature is much stricter than the annotator.

Scene 5 seems to contain a lot of segments that match well with various other actions, when HoF is used as feature. Most false positives are scored on actions 100, 301, 302, 500, and 600. Visual inspection of these false positives shows that action 500 matches with movements which are fairly similar.

Action 600 has a total of twenty false positives for all male footage according to Table 7.2 and seven of these false positives are present in this scene. A movement that resembles template 20, which contains the two actions 200 and 201, seems to be the most common appearance in the segments predicted to be action 600. The biggest difference is that instead of having its hands before his body, they are now above his head. This movement does not resemble action 600 at all, but still gets predicted by the classifier. The reason that these segments are classified as action 600, is elaborated in Section 7.3.5. Furthermore, actions 301 and 302 do not resemble their false positive segments, but most matches are caused by insignificant movements by body parts that wobble.

7.3.5 Action 600

The exact reason for the large amount of false positives for action 600 cannot be deduced from the in depth examination of each scene. Therefore, all predicted occurrences of this action, including their distance value are collected. All these entries are compared with each each other and with the calibrated threshold. This comparison confirms that the calibrated threshold is far above the highest scoring true positive and putting the threshold just above this true positive would result in a total of seven false positives, which is far less than 20 it is currently at. The version of action 600 used as template has only minimal head movement at the start, while almost all entries in the validation set consist of the Woman 1 doing the same movement including a lot of head movements. Action 601, which is similar to action 600, has also got entries in the validation set with the same kind of head movement, but the template used for comparison also contains them. The threshold for action 601 is much lower than the threshold for 600, which indicates the difference in head movement has a significant influence on the calibration process.

7.3.6 Conclusion question three

The easiest way to answer question three is to take the f-scores for HoS and HoF from Table 7.2 and deduce from those scores that HoF performs slightly better with this particular video. However, it is difficult to support this answer, because there are a large number of factors that influence the performance achieved with these two grid-based features. The biggest two factors besides the chosen feature that have a big influence on the classification result, are the chosen template for an action and its calibrated threshold. Choosing a different occurrence of an action as template could cause a dramatic difference in acquired results, especially when these occurrences show differences amongst each other. For example, the threshold calibrated for actions 100, 600 using HoF are too high, because the occurrences in the

calibration set differ too much. Choosing a different occurrence from the training set that matches better with the calibration set would decrease the threshold and this would dramatically decrease the amount of false positives and in turn improve the total f-score acquired with HoF.

Close inspection shows that both features have their advantages over the other feature and with particular actions they outperform the other feature. Two big advantage of using HoF as feature, is that it is more discriminative and is less influenced by body size. The content of a patch with HoS is a simple value that indicates the percentage of the patch occupied by the silhouette, which stays the same for any shape that occupies the same amount of pixels in a patch. The HoF feature on the other hand contains per patch four different values indicating the total amount of optical flow in one of the four directions. The fact that this feature is more discriminative is the reason why the Dynamic Time Warping metric is able to deduce usable warping paths when dealing with comparisons of time-series of HoF features and not with time-series of HoS features.

I would conclude that both features perform almost equally well, because both have their benefits and work well with certain action. When recall is more important than precision, than I would recommend to use HoS as feature. In all other cases I would recommend to use HoF as feature, because it contains more discriminative information.

7.4 Comparing both grid-based features (Woman 2/3)

How well do both grid-based features perform when another subject is used for testing?

There are two different subjects (Woman 2 and 3) and the appearance of these subjects in every scene can be found in Table 3.1. The recall, precision and f-scores for each subject and for both combined are displayed in Table 7.5. The data shows that HoF performs well with Woman 2, while the recall for footage containing Woman 3 is very low. The actions performed by subject Woman 3 are predicted quite well with HoS as feature, but none of the entries for Woman 2 are predicted. This could be caused by a large difference in body shape between the subject Man and the subject Woman 2.

Table 7.6 gives insight how each action scored in comparison with the footage of the subject Man. Actions 205 and 206 are removed from this overview, because both actions do not occur in any of the performances by the female subjects. This is because the repertoire of the both subjects in each scene is not exactly the same. Its clear from the comparison table that actions 200, 201, 600, and 601 give no sensible results with HoS as feature, which could also be concluded from Table 7.5. Remarkable is that action

		Combined	Woman 2	Woman 3
HoS	recall	0.1720	0.0000	0.1917
	precision	0.4140	0.0000	0.5039
	f1-score	0.2430	0.0000	0.2778
HoF	recall	0.0714	0.4359	0.0295
	precision	0.3034	0.2931	0.3226
	f1-score	0.1156	0.3505	0.0541

Table 7.5: Performance of other subjects

304 achieves a higher f1-score with Woman 3 than the Man. A similar situation as with action 600 for HoF occurs with template 30 containing actions 300 and 304. The calibrated threshold lies far above the distance values calculated for the true positives and most false positives lie between the true positives and the threshold. With Woman 3 these false positives disappear because the calculated distance values are much closer to the calibrated threshold and therefore the precision increases and in turn this increases the f1-score.

Using HoF as feature seems to deliver a more constant result, because there are no actions that are not recognized. However, the results for actions performed by Woman 3 are very bad in comparison with the results acquired with HoS as feature. As with HoS there is an action that scores better with footage of the Female, than the subject Man. Both recall (0.4167) and precision (0,8333) are higher with action 201 and visual inspection of individual cases of 201 show that these predicted actions match very well.

As with the footage of subject Man a lot of false positives are found for actions 100, 101, 300, 301, 302, 304 and 500 using HoF as feature (see Figures B.1.2 and B.1.4). The amount of false positives in comparison with footage of the Man is lower, but still significant amounts are found with both subjects (Woman 2 and 3).

It can be concluded that HoF is a grid-based feature that is less influenced by difference in body shape, which plays a very big role with HoS. The introduction of a grid for the silhouette descriptor should have solved this issue, but apparently this feature still suffers a lot from differences in body type. However, both features suffer greatly when trying to classify actions performed by another subject than the reference actions. HoF seems to be a more robust feature and is therefore preferred as feature for comparison between two different subjects.

		200	201	300	301	302	304	600	601
HoS	Man	0.1539	0.2857	0.2000	0.3012	0.3391	0.3077	0.5000	0.6154
	Woman 2/3	0.0000	0.0000	0.0351	0.2439	0.2938	0.4643	0.0000	0.0000
HoF	Man	0.6667	0.4211	0.4000	0.3021	0.3333	0.4186	0.3529	0.6667
	Woman 2/3	0.6667	0.5556	0.0476	0.0375	0.0392	0.1463	0.3636	0.3077

Table 7.6: Comparison f1-scores subject Man with Woman 2 and 3

Chapter 8

Conclusion and Future work

This master thesis describes the comparison of grid-based features and classification settings, in an off-line supervised human action detection and recognition task with a limited training set. The training data for the classifier consists of single occurrences of discriminative templates and each template consists of a movement performed in the XCO video. It can be concluded from the results in section 7 that increasing the grid size for the Histogram of Silhouette feature improves its results, which could be an indication that this feature prefers more fine-grained grid sizes. The best grid configuration for Histogram of Flow was the 3x3 with heuristic grid. This finding indicates that choosing a coarse-grained grid size for Histogram of Flow is a good choice, because with limited training data the feature must not become too specific, otherwise it is impossible to recognize action performed by other subjects. The Histogram of Silhouette seems to perform quite well with motion sequences that are performed by the same subject as is present in the training data, but with other subjects the results are bad. Histogram of Flow on the other hand proves to be a more general feature, because it is less influenced by differences in body shape of subjects. Furthermore, Histogram of Flow features allow for usable warping paths to be deduced when comparing time-series of these features, which seems not be working for Histogram of Silhouette because this feature is less discriminative.

The conclusions that are drawn from the results are greatly influenced by imperfections in the classification task. The height of each threshold, used by the classification task, proved to be very sensitive to small difference in performance between template and occurrences of this template in the validation set. The single occurrence per template is chosen, to avoid biases towards templates that occur more often (see Table 3.2). In some cases this caused a feature to perform worse than expected. For example, the occurrence of action 600 chosen as template slightly differs from the occurrences in the validation set, which results in high threshold and in turn causes many false positives. I would recommend to slightly increase

the amount of occurrences per template in future research. The concept of having a limited training set is kept intact, but increasing the amount of occurrences per template requires the classification approach to be adapted. A possibility is to use a k-Nearest Neighbor classifier, because it is a simple and effective classification approach that is applicable to tasks that only have limited training data.

In future research it may be an idea to examine the effect of a wide range of grid configurations for both features and see what these different grid sizes have as effect on different kinds of classification tasks with different training set sizes. For example increasing the amount of patches for the Histogram of Silhouette feature may help to increase the recall and precision with the action detection and recognition task discussed in this master thesis. However, increasing the grid size too much for HoS may cause some actions to become overfitted for the subject similar to the one present in the training data. With classification tasks that use large training sets, such as a classifier using a Hidden Markov model, may benefit from choosing a small grid size for HoF. Another interesting expansion of the work done in this final project, is to examine the performance of a feature that combines the HoS and HoF. Tran et.al. [29] do something similar in their research, by combining the silhouette and motion information as channels of each patch into a single grid-based feature. They subdivide each patch using a radial division, which basically boils down to additional subdivision of an already subdivided region interest, but another kind of division is used. It could be interesting to examine the difference between grids using with and without radial division.

With the current approach a window with a fixed size is slided over each scene and the scene is segmented according to the distance scores acquired for each template. Instead the scene could also be temporally segmented using key frames as is done by Ali and Aggarwal [1] or using the approach by Rui and Anandan [24]. It would be recommended to take such an approach in the future for segmenting the scene, because with the current approach a large number of false negatives are caused by gaps introduced due to overlap in the templates. Furthermore, it will become easier to compare slower movements with faster movements and the other way around and may also diminish the need for equally sized templates. An alternative approach is to (partially) allow overlap to exists, which should improve the results, because the mismatch between template and classifier policy on overlap does not exist anymore. Furthermore, I would recommend for future research to use multiple publicly available video sets, instead of video material selected from a private collection. This increases the comparability of the approach to other published approaches, which is not the case in this final project.

Bibliography

[1] A. Ali and J. Aggarwal. Segmentation and Recognition of Continuous Human Activity. In *IEEE Workshop on Detection and Recognition of Events in Video*, pages 28–35, 2001.

[2] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, 12:43–77, 1994.

[3] M. Bashir and J. Kempf. Reduced Dynamic Time Warping for Handwriting Recognition Based on Multidimensional Time Series of a Novel Pen Device. In *World Academy of Science, Engineering and Technology*, volume 45, pages 382–388, 2008.

[4] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[5] A. F. Bobick and J. W. Davis. The Recognition of Human Movement Using Temporal Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:257–267, 2001.

[6] A. Bruhn, J. Weickert, and C. Schnörr. Lucas-Kanade meets Horn-Schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61:211–231, 2005.

[7] X. Chen, Z. He, D. Anderson, J. Keller, and M. Skubic. Adaptive Silhouette Extraction and Human Tracking in Complex and Dynamic Environments. In *IEEE International Conference on Image Processing, 2006*, pages 561–564, October 2006.

[8] A. Corradini. Dynamic time warping for off-line recognition of a small gesture vocabulary. In *IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 2001.*, pages 82–89, 2001.

[9] T. M. Cover and P. E. Hart. Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, 13(1):21–27, January 1967.

[10] R. Cutler and L. S. Davis. Robust real-time periodic motion detection, analysis, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):781–796, Augustus 2000.

[11] S. Danafar and N. Gheissari. Action Recognition for Surveillance Applications Using Optic Flow and SVM. In Y. Yagi, S. Kang, I. Kweon, and H. Zha, editors, *Computer Vision ACCV 2007*, volume 4844 of *Lecture Notes in Computer Science*, pages 457–466. Springer Berlin / Heidelberg, 2007.

[12] R. F. de Mello and I. Gondra. Multi-Dimensional Dynamic Time Warping for Image Texture Similarity. In *Proceedings of the 19th Brazilian Symposium on Artificial Intelligence: Advances in Artificial Intelligence*, SBIA '08, pages 23–32, Berlin, Heidelberg, 2008. Springer-Verlag.

[13] M. Donoser, H. Riemenschneider, and H. Bischof. Shape Prototype Signatures for Action Recognition. *International Conference on Pattern Recognition*, 0:1796–1799, 2010.

[14] S. A. Dudani. The Distance-Weighted k-Nearest-Neighbor Rule. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-6(4):325–327, April 1976.

[15] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing Action at a Distance. In *IEEE International Conference on Computer Vision*, pages 726–733, Nice, France, 2003.

[16] C. Fang. From Dynamic Time Warping (DTW) to Hidden Markov Model (HMM). Technical report, University of Cincinnati, 2009.

[17] D. J. Fleet and A. D. Jepson. Computation of Component Image Velocity from Local Phase Information. *International Journal of Computer Vision*, 5:77–104, September 1990.

[18] B. Gold and N. Morgan. *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1999.

[19] B. K. P. Horn and B. G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

[20] B. D. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2*, pages 674–679, 1981.

[21] M. E. Munich and P. Perona. Continuous Dynamic Time Warping for translation-invariant curve alignment with applications to signature verification. In *Proceedings of the 1999 7th IEEE International Conference on Computer Vision (ICCV'99)*, pages 108–115, 1999.

[22] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990, 2010.

[23] L. Rabiner and B.-H. Juang. An introduction to hidden Markov models. *ASSP Magazine, IEEE*, 3(1):4–16, January 1986.

[24] Y. Rui and P. Anandan. Segmenting Visual Actions Based on Spatio-Temporal Motion Patterns. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 111–118, 2000.

[25] Y. Sheikh, O. Javed, and T. Kanade. Background Subtraction for Freely Moving Cameras. In *IEEE 12th International Conference on Computer Vision, 2009*, pages 1219–1225, 2009.

[26] C. Stauffer and W. E. L. Grimson. Adaptive Background Mixture Models for Real-Time Tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999.*, volume 2, pages 637–663, 1999.

[27] D. Sun, S. Roth, and M. J. Black. Secrets of Optical Flow Estimation and Their Principles. In *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2439, June 2010.

[28] G. A. ten Holt, M. J. Reinders, and E. A. Hendriks. Multi-Dimensional Dynamic Time Warping for Gesture Recognition. In *Thirteenth annual conference of the Advanced School for Computing and Imaging*, June 2007.

[29] D. Tran, A. Sorokin, and D. Forsyth. Human Activity Recognition with Metric Learning. In *Proceedings of the 10th European Conference on Computer Vision: Part I*, ECCV '08, pages 548–561, 2008.

[30] L. Wang and D. Suter. Informative Shape Representations for Human Action Recognition. In *18th International Conference on Pattern Recognition, 2006. ICPR 2006.*, volume 2, pages 1266–1269, 2006.

[31] L. Wang, T. Tan, H. Ning, and W. Hu. Silhouette Analysis-Based Gait Recognition for Human Identification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25:1505–1518, December 2003.

[32] D. Weinland and E. Boyer. Action Recognition using Exemplar-based Embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pages 1–7, Anchorage, United States, 2008. IEEE Computer Society.

[33] W. Xiong and J. Chung-Mong Lee. Efficient Scene Change Detection and Camera Motion Annotation for Video Classification. *Computer Vision and Image Understanding*, 71(2):166–181, 1998.

[34] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden Markov model. In *Conference on Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92., 1992 IEEE Computer Society*, pages 379–385, June 1992.

[35] L. Zelnik-Manor and M. Irani. Event-Based Analysis of Video. In *Computer Vision and Pattern Recognition, 2001.*, pages 123–130, 2001.

[36] S. Zhang, H. Yao, and S. Liu. Dynamic Background Modeling and Subtraction using Spatio-temporal Local Binary Patterns. In *IEEE International Conference on Image Processing, 2008. ICIP 2008. 15th*, pages 1556–1559, October 2008.

[37] Z. Zivkovic. Improved Adaptive Gaussian Mixture Model for Background Subtraction. *International Conference on Pattern Recognition*, 2:28–31, 2004.

Appendices

Appendix A

Silhouette descriptor samples

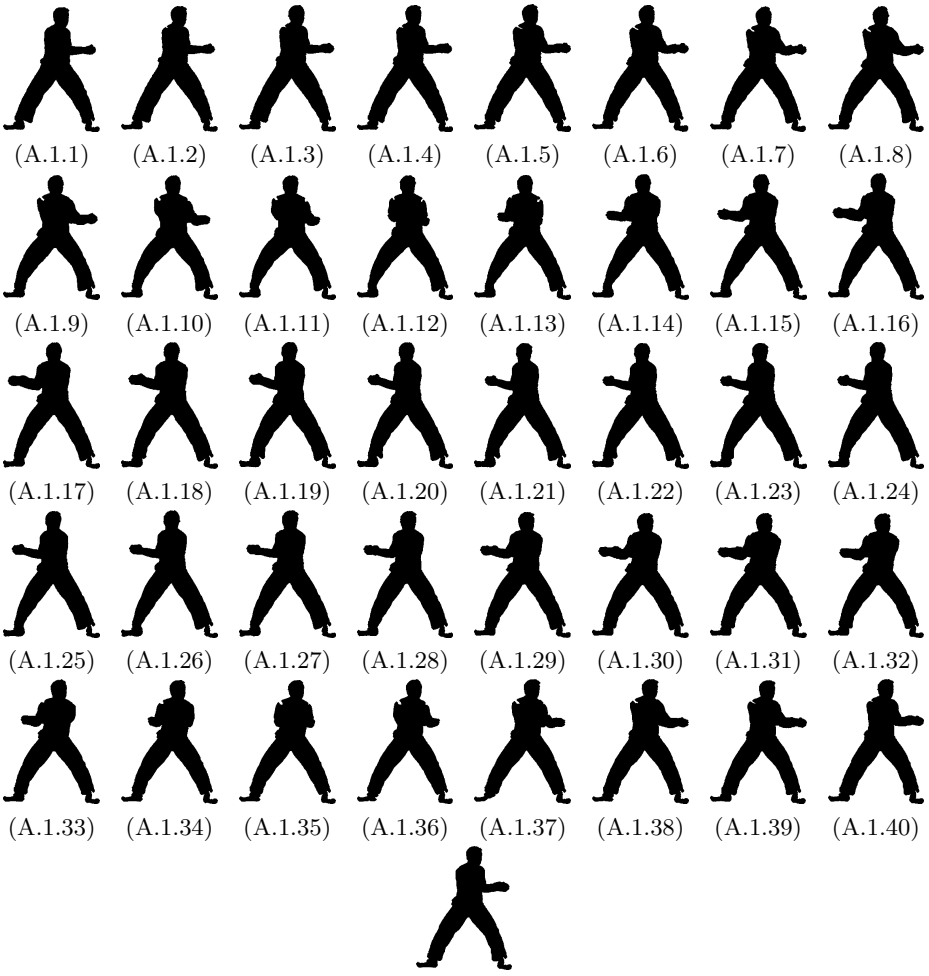


Figure A.1: Scene: 1, Person: Man, Template 30 (Actions 300, 304)

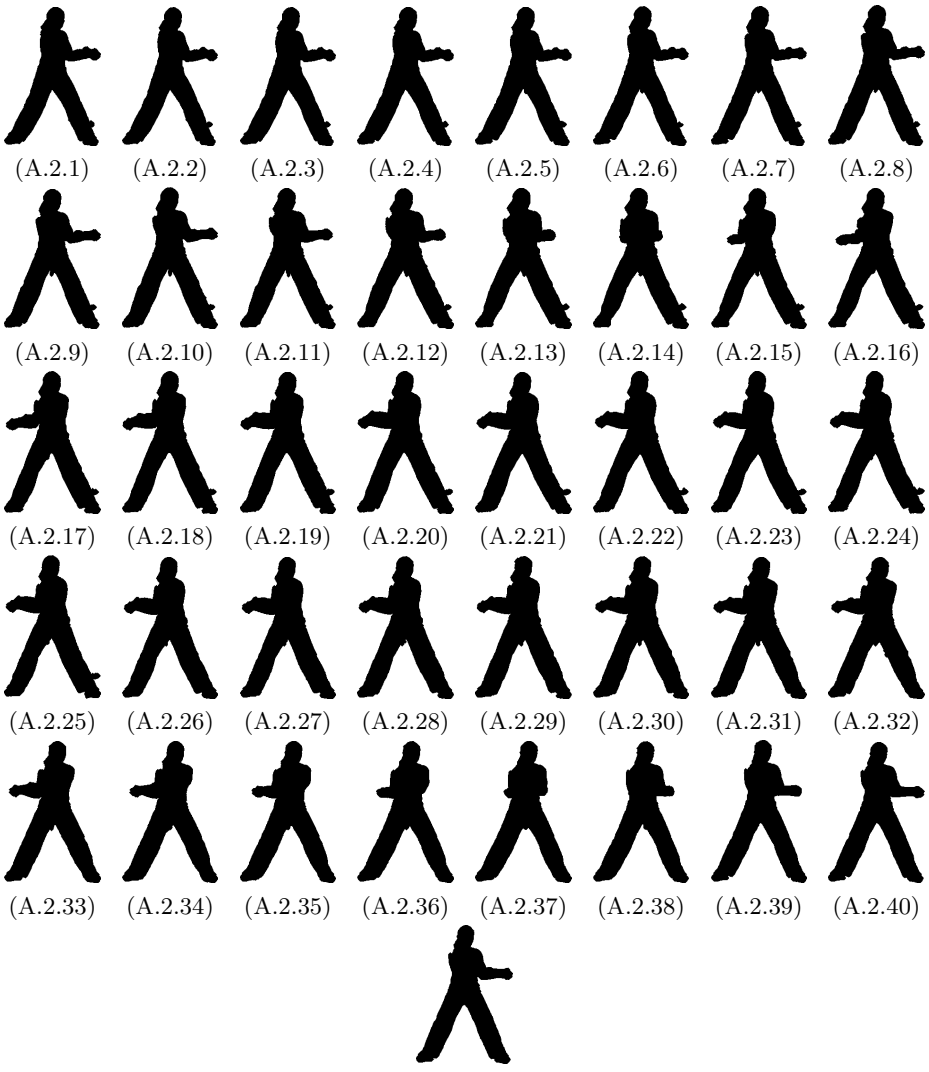
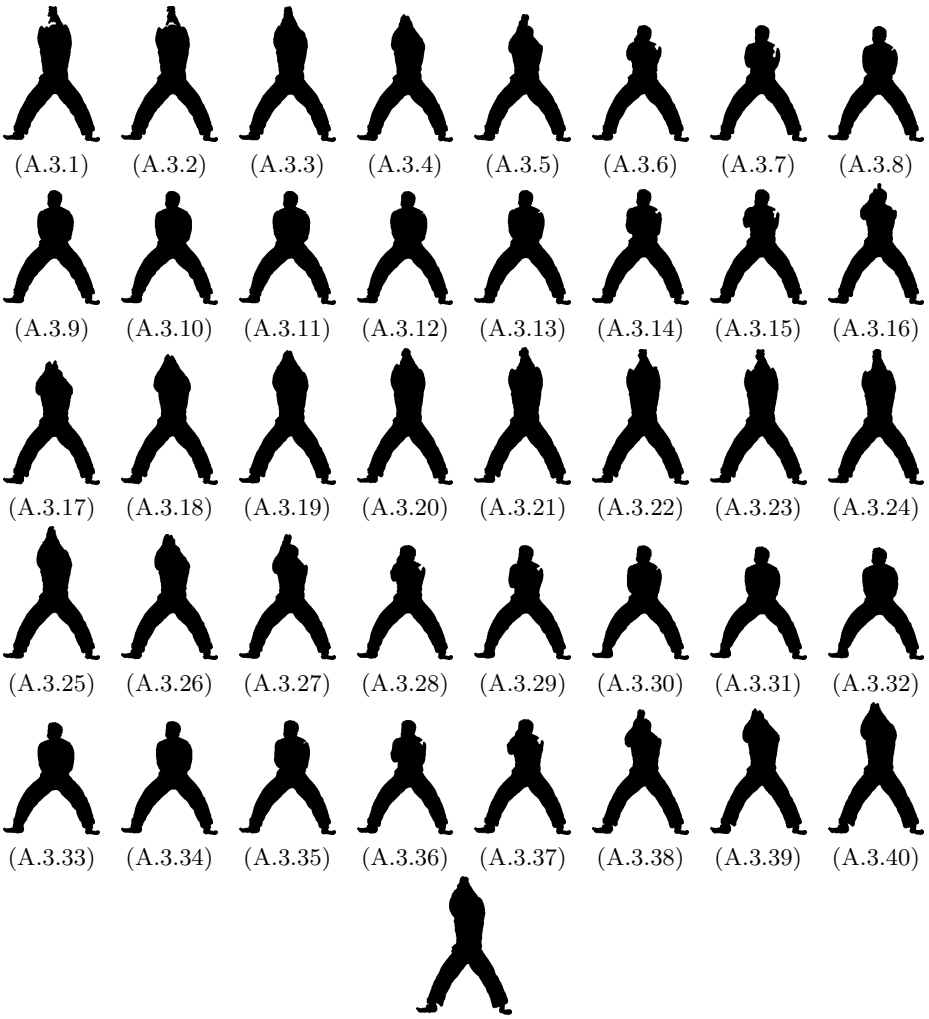


Figure A.2: Scene: 1, Person: Woman 1, Segment 1217



(A.3.1) (A.3.2) (A.3.3) (A.3.4) (A.3.5) (A.3.6) (A.3.7) (A.3.8)

(A.3.9) (A.3.10) (A.3.11) (A.3.12) (A.3.13) (A.3.14) (A.3.15) (A.3.16)

(A.3.17) (A.3.18) (A.3.19) (A.3.20) (A.3.21) (A.3.22) (A.3.23) (A.3.24)

(A.3.25) (A.3.26) (A.3.27) (A.3.28) (A.3.29) (A.3.30) (A.3.31) (A.3.32)

(A.3.33) (A.3.34) (A.3.35) (A.3.36) (A.3.37) (A.3.38) (A.3.39) (A.3.40)

(A.3.41)

Figure A.3: Scene: 2, Person: Man, Segment 2636

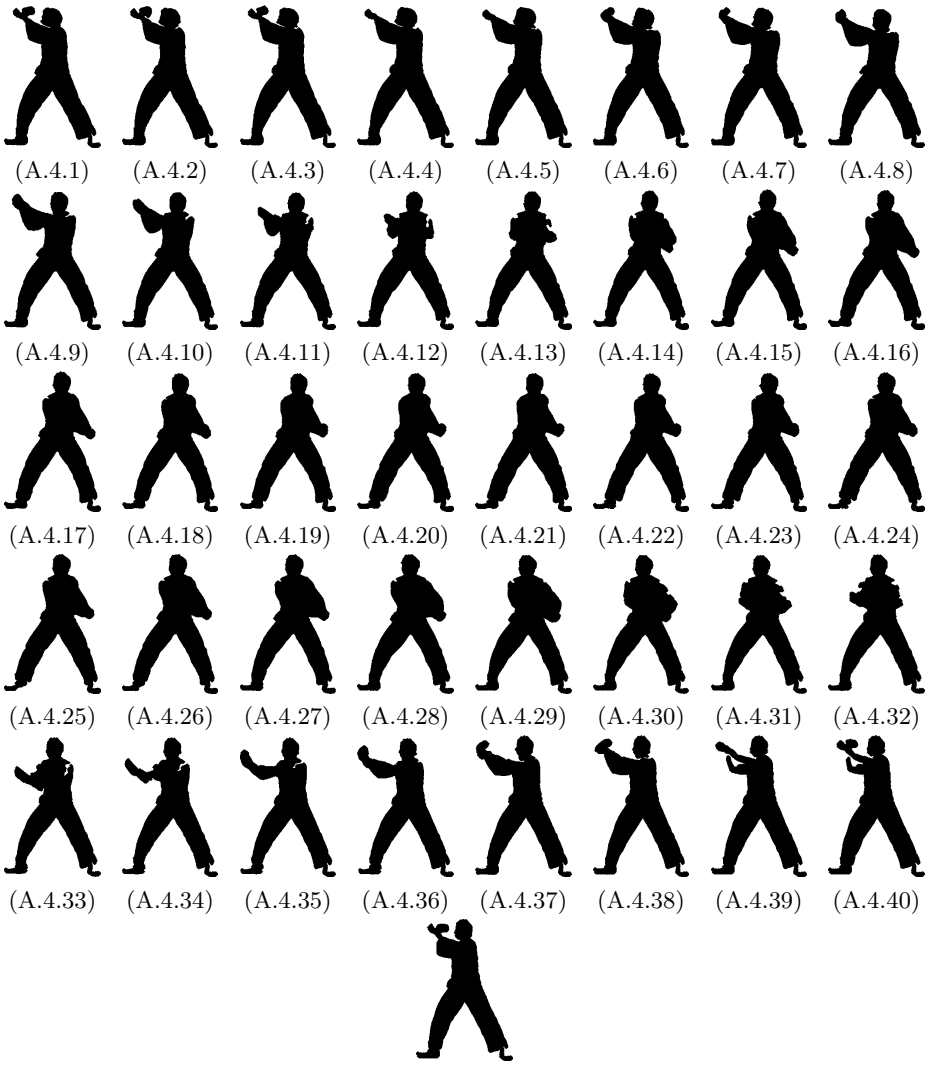
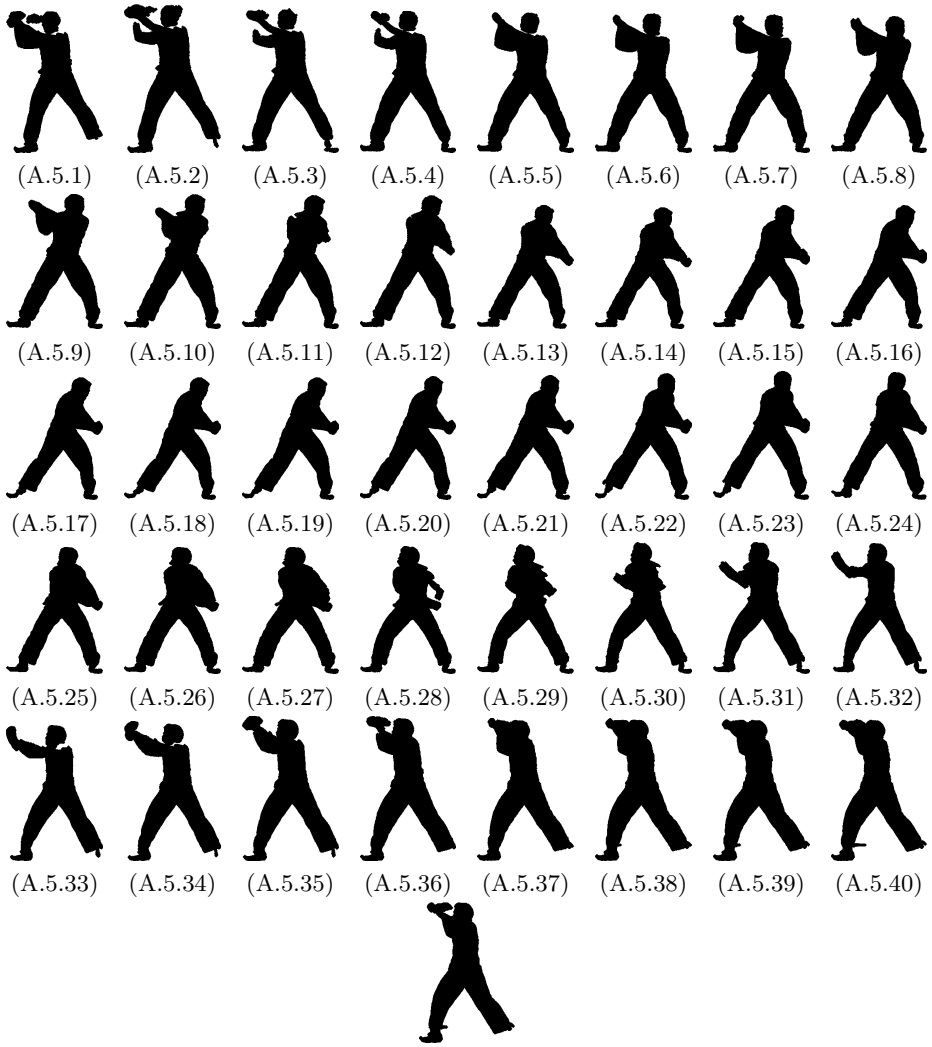


Figure A.4: Scene: 1, Person: Man, Template 60 (Action 600)



(A.5.1) (A.5.2) (A.5.3) (A.5.4) (A.5.5) (A.5.6) (A.5.7) (A.5.8)

(A.5.9) (A.5.10) (A.5.11) (A.5.12) (A.5.13) (A.5.14) (A.5.15) (A.5.16)

(A.5.17) (A.5.18) (A.5.19) (A.5.20) (A.5.21) (A.5.22) (A.5.23) (A.5.24)

(A.5.25) (A.5.26) (A.5.27) (A.5.28) (A.5.29) (A.5.30) (A.5.31) (A.5.32)

(A.5.33) (A.5.34) (A.5.35) (A.5.36) (A.5.37) (A.5.38) (A.5.39) (A.5.40)

(A.5.41)

Figure A.5: Scene: 2, Person: Man, Segment 3913

Appendix B

Confusion matrices

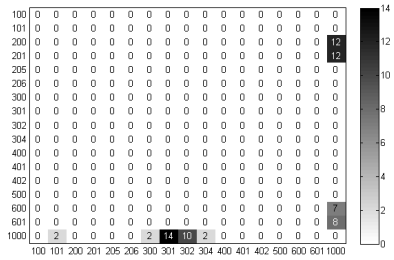
Table B.1 contains all the names that belong to each action.

ID	Name
100	Rowing (Looking left)
101	Rowing (Looking right)
200	Kantellen (L->R)
201	Kantellen (R->L)
205	Kantellen Snel (R->L)
206	Kantellen Snel (L->R)
300	Twist (R->L)
301	Twist Snel (R->L)
302	Twist Snel (L->R)
304	Twist (L->R)
400	Load swing (L,R)
401	Load swing (short and starts Left)
402	Load swing (short and starts Right)
500	Solid push (2x)
600	Lumberjack (Top L -> Bottom R and Bottom R -> Top L)
601	Lumberjack (Top R -> Bottom L and Bottom L -> Top R)
1000	Remainder class

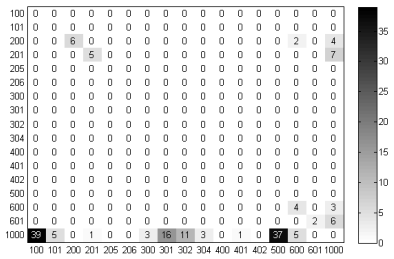
Table B.1: Description for each ID in the confusion matrix

B.1 Subjects Woman 2/3

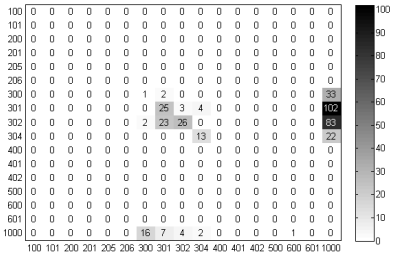
The confusion matrices in Figure B.1.4 for HoS are generated for combination of classification setting: 3x5 without heuristic grid and single value threshold. For HoF the 3x3 with heuristic grid is used.



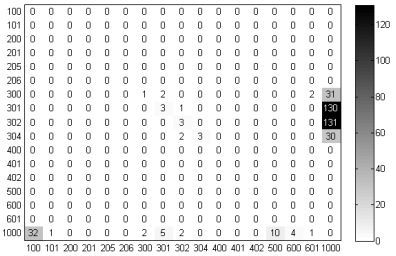
(B.1.1) Woman 2 (HoS)



(B.1.2) Woman 2 (HoF)



(B.1.3) Woman 3 (HoS)



(B.1.4) Woman 3 (HoF)