Master in Artificial Intelligence (UPC-URV-UB)

Master of Science Thesis

ECOC AND GRAPH CUTS SEGMENTATION OF HUMAN LIMBS

Daniel Sánchez Abril

Advisor/s: Sergio Escalera Guerrero
Dr./Drs. on behalf of the Advisor/s: Miguel Ángel Bautista Martín

31/01/2014

Acknowledgements

The work included in this thesis could not have been performed if not for the assistance, patience, and support of many individuals. I would like to extend my gratitude first and foremost to my thesis advisors Sergio Escalera and Miguel Ángel Bautista for mentoring me over the course of my graduate and postgraduate studies. They have helped me through extremely difficult times over the course of the analysis and the writing of the work and for that I sincerely thank them for their confidence in me. In addition, for their support in both the research and especially the revision process that has lead to this document. Their knowledge and understanding of the written word has allowed me to fully express the concepts behind this research.

This research would not have been possible without the help of Tomás Pérez and Juan Carlos Ortega in several stages of this work.

Finally I would like to extend my deepest gratitude to my friends, specially Marc García for his support to this project which I could never have completed this master thesis.

Abstract

Human multi-limb segmentation in RGB images has attracted a lot of interest in the research community because of the huge amount of possible applications in fields like Human-Computer Interaction, Surveillance, eHealth, or Gaming. Nevertheless, human multi-limb segmentation is a very hard task because of the changes in appearance produced by different points of view, clothing, lighting conditions, occlusions, and number of articulations of the human body. Furthermore, this huge pose variability makes the availability of large annotated datasets difficult. In this work, we introduce the *HuPBA* $8k+$ dataset. The dataset contains more than 8000 labeled frames at pixel precision, including more than 120000 manually labeled samples of 14 different limbs. For completeness, the dataset is also labeled at frame-level with action annotations drawn from an 11 action dictionary which includes both single person actions and person-person interactive actions. Furthermore, we also propose a two-stage approach for the segmentation of human limbs. In a first stage, human limbs are trained using cascades of classifiers to be split in a tree-structure way, which is included in an Error-Correcting Output Codes (ECOC) framework to define a body-like probability map. This map is used to obtain a binary mask of the subject by means of GMM color modelling and GraphCuts theory. In a second stage, we embed a similar tree-structure in an ECOC framework to build a more accurate set of limb-like probability maps within the segmented user mask, that are fed to a multi-label GraphCut procedure to obtain final multi-limb segmentation. The methodology is tested on the novel *HuPBA* $8k+$ dataset, showing performance improvements in comparison to state-of-the-art approaches. In addition, a baseline of standard action recognition methods for the 11 actions categories of the novel dataset is also provided.

Abstract

La segmentación humana multi-extremidad en imágenes RGB ha atraído una gran cantidad de interés en la comunidad científica debido a la enorme cantidad de posibles aplicaciones finales, tales como: Interacción Persona-Ordenador, Vigilancia, eHealth, o juegos interactivos. Sin embargo, la segmentación humana multi-extremidad es una tarea muy difícil debido a los cambios en apariencia producida por diferentes puntos de vista, ropa, condiciones de iluminación, oclusiones, y el número de articulaciones de el cuerpo humano. Además, esta enorme variabilidad en la pose produce la disponibilidad de grandes volúmenes de datos con difíciles anotaciones. En este trabajo, presentamos la base de datos *HuPBA* $8k+$. Esta base de datos está compuesta por más de 8000 imágenes etiquetadas a nivel de píxel, incluyendo más de $120,000$ máscaras etiquetadas manualmente de 14 extremidades diferentes. Para completar, el conjunto de datos es también etiquetado para cada imagen con anotaciones de acciones que en total suma un diccionario de 11 acciones diferentes, las cuales incluyen tanto las acciones de una sola persona y en las que se necesitan dos personas. Por otra parte, también proponemos un enfoque de dos etapas para la segmentación humana multi-extremidad. En una primera etapa, las extremidades del cuerpo son entrenadas utilizando cascadas de clasificadores para ser divididos en una estructura árbol, el cual es incluido en un marco de trabajo llamado Códigos correctores de errores de salida (ECOC) para definir un mapa de probabilidad del cuerpo humano. Este mapa se utiliza para obtener una máscara binaria del sujeto por medio de modelos de mixturas gaussianas (GMM) de color y teoría de corte de grafos. En una segundo etapa, integramos una estructura de árbol similar al marco de trabajo ECOC utilizado anteriormente para construir un conjunto más preciso de mapas de probabilidad de las extremidades dentro de la máscara del sujeto segmentada, que es enviada a un proceso de corte de grafos multi-etiqueta para la obtención final de la segmentación de múltiples extremidades. La metodología se prueba en la nueva base de datos *HuPBA* $8k+$, mostrando mejoras de rendimiento en comparación con las propuestas en el estado del arte. Además, una muestra de métodos estándar de reconocimiento de gestos para 11 gestos es también proporcionado.

Abstract

Segmentació multi-extremitat humana en imatges RGB ha atret una gran quantitat de interès en la comunitat científica a causa de l'enorme quantitat de possibles aplicacions finals com la Interacció Persona- Ordinador, Vigilància , eHealth, o jocs interactius. No obstant això, la segmentació multi-extremitat és una tasca molt difícil a causa dels canvis en aspecte produïda per diferents punts focals, roba, condicions d'iluminació, oclusions, i el nombre d'articulacions de el cos humà. A més, aquesta enorme variabilitat de la postura fa la disponibilitat de grans conjunts de dades amb anotacions difícils. En aquest treball, presentem la base de dades *HuPBA* 8k+. Aquesta base de dades conté més de 8000 imatges etiquetades a nivell de píxel, incloent més de 120.000 màscares manualment etiquetades de 14 extremitats diferents. Per completar, la base de dades és també etiquetada per a cada imatge les acciones a partir d'un diccionari de 11 accions diferents que inclou tant les accions d'una sola persona i de múltiples persones. D'altra banda, també proposem un enfocament de dues etapes per a la segmentació. En una primera etapa, les extremitats humanes estan entranades amb cascades de classificadors per a ser dividits en una estructura d'arbre, que s'inclou en un marc de treball anomenat Codis correctors d'errors de sortida (ECOC) per construir un mapa de probabilitat del cos humà. Aquest mapa s'utilitza per obtenir una màscara binària del subjecte per mitjà de models de mixtures gaussianes (GMM) de color i teoria de tall de grafs. En una segona etapa, integrem una estructura d'arbre similar al ECOC anterior per construir un conjunt més precís de mapes de probabilitat de les extremitats del cos humà dintre de la màscara binària del subjecte, que és enviada a un procés de tall de grafs per obtenir una segmentació final multi-extremitat. La metodologia es prova en la nova base de dades *HuPBA* 8k+, que mostra millores en el rendiment en comparació de les propostes de l'estat de l'art. A més, una mostra de mètodes estàndar de reconeixement de accions per 11 accions es també proporcionada.

Contents

List of Figures

List of Tables

1 Introduction

Human analysis in RGB images is a challenging task because of the high variability of the human body, including the wide range of human poses, lighting conditions, cluttering, clothes, appearance, background, point of view, number of human body limbs, etc. Even so, human analysis in visual data has become one of the more interesting areas of research in Computer Vision and Pattern Recognition because of its capabilities in final applications (i.e. human-computer interaction, surveillance, gaming, eHealth, interactive virtual reality systems, etc.). In this sense, the common pipeline for human body analysis in visual data uses to be defined in a bottom-up fashion. First, the human body limbs are segmented and the body pose is estimated (often with a prior person/background segmentation or person detection step). Then, once the body pose is estimated higher abstraction analysis can be performed. Usually, the following step in the pipeline is action/gesture recognition, since actions can be seen as a set of estimated body poses varying over time.

1.1 State-of-the-art pose estimation

Human limb segmentation or pose estimation in RGB images has been a core problem in the Computer Vision field since its early beginnings. In this particular problem the goal is to provide with a complete segmentation of each of the defined human body parts appearing in an image, discriminating human limbs from each other and from the rest of the image. Usually, human body segmentation is treated in a two-stage fashion. First, a human body part detection step is performed, and then, these human part detections are used as a prior knowledge to be optimized by segmentation/inference strategies in order to obtain the final human-limb segmentation. In literature one can find many works that follow this two-stage scheme. Bourdev et. al. [5] used body part detections in an AND-OR graph to obtain the pose estimation. Vinet et. al [35] proposed to use Conditional Random Fields based on body part detectors to obtain a complete person/background segmentation. Nevertheless, one of the methods that have generated more attraction is the well known pictorial structure for object recognition introduced by Felzenszwalb et al [18]. Some works have applied an adaptation of pictorial structures using a set of joint limb marks to infer spatial probabilities [1, 31, 26, 27]. Later on, an extension was presented by Yang and Ramanan [38, 39] which proposed a discriminatively trained pictorial structure that models the body joints instead of limbs. In contrast, there is also current tendency to use Graph Cuts optimization to segment the human limbs [21] or full person segmentation [29].

1.2 State-of-the-art gesture recognition

The common step after estimating the pose of a subject within the pipeline of human body analysis is analyzing non-verbal communication in terms of actions and/or gestures, which can be interpreted as a set of poses varying over time. In this sense, in order to deal with action/gesture recognition there exists a wide number of methods based on

dynamic programming algorithms for alignment and clustering of temporal series [40, 30]. One of the most common methods for Human Gesture Recognition based on dynamic programming is Dynamic Time Warping (DTW) [28, 22, 30], since it offers a simple yet effective temporal alignment between sequences of different lengths. Other probabilistic methods such as Hidden Markov Models (HMM) or Conditional Random Fields (CRF) have been commonly used in the literature [33]. Some other methods also considered for action/gesture recognition include Neural Networks approaches, Boosting variants, and Random Forest [15, 14].

1.3 Proposal

The Computer Vision community has been lately focusing their efforts on developing methods for both pose estimation and action/gesture recognition. However, one of the main problems is the necessity of public available data sets containing annotations of all the variabilities the methods have to deal with. Substantial effort has been put on designing datasets with different scenarios, people and illumination characteristics. Datasets such as Parse [27], Buffy [19], UIUC People [34], and Pascal VOC [17] are widely used to evaluate different pose estimation and action/gesture recognition methods. However, these public available datasets fail to provide with a sound framework in which to validate pose recovery systems (i.e. the number of samples per limb is small, the labeling is not accurate, there are no interactions of actors, etc.). Given this lack of sound and refined public datasets for human multi-limb segmentation and/or action/gesture recognition, we introduce the *HuPBA* $8k+$ dataset, which to the best of our knowledge is the biggest RGB human-limb labeled dataset. The dataset contains more than 8000 labeled frames at pixel precision and more than 120000 manually labeled samples of 14 different limbs. In addition, the *HuPBA* $8k+$ dataset is also labeled with action annotations drawn from an 11 action dictionary which includes both single person actions and interactive actions (actions performed by more than one person).

We also extend our work of [11] by proposing a two-stage approach for the segmentation of human limbs. In a first stage, a set of human limbs are normalized by main orientation to be rotation invariant, described using Haar-like features, and trained using cascades of Adaboost classifiers to be split in a tree-structure way. Once the tree-structure is trained, it is included in a ternary Error-Correcting Output Codes (ECOC) framework. This first classification step is applied in a windowing way on a new test image, defining a body-like probability map, which is used as an initialization of a binary Graph Cuts optimization procedure. In the second stage, we embed a similar tree-structured partition of limbs in a ternary ECOC framework and we use Support Vector Machines (SVMs) with HOG descriptors to build a more accurate set of limb-like probability maps within the segmented user binary mask, that are fed to a multi-label GraphCut optimization procedure to obtain the final human multi-limb segmentation. We tested our ECOC-Graph-Cut based approach in the novel *HuPBA* $8k+$ dataset and compared with state-of-the-art pose recovery approaches, obtaining performance improvements in both person/background and multi-limb segmentation steps. For completeness, we also provide with action recognition results as a baseline for the *HuPBA*

$8k+$ dataset. Summarizing, our key contributions are:

- We introduce the *HuPBA* $8k+$ dataset, the largest RGB labeled dataset of human limbs, with more than 120000 manually annotated limbs. The data set also includes frame-level annotation for 11 action/gesture categories.
- We propose a two stage approach based on ECOC and Graph Cuts for the segmentation of human limbs in RGB images.
- The proposed method is compared with state-of-the-art methods for human pose estimation obtaining very satisfying results.
- We provide with a baseline for Action Recognition in the novel dataset.

2 HuPBA 8K+ Dataset

Automatic human limb detection and segmentation, human pose recovery and human behavior analysis are challenging problems in computer vision, not only for the intrinsic complexity of the tasks, but also for the lack of large public and annotated datasets. Usually, public available datasets lack of refined labeling or contain a very reduced number of samples per limb (e.g. *Buffy Stickmen V*3.01, *Leeds Sports* and *Hollywood Human Actions* [19, 23, 24]). In addition, large datasets often use synthetic samples or capture human limbs with sensor technologies such as *MoCap* in very controlled environments [12].

Being aware of this lack of public available datasets for multi-limb human pose detection, segmentation and action/gesture recognition, we present a novel fully limb labeled dataset, the *HuPBA* $8k+$ dataset. This dataset is formed by more than 8000 frames where 14 limbs are labeled at pixel precision[1]. Furthermore, the HuPBA $8k+$ dataset also contains gesture/action annotations for 11 isolated and collaborative action categories. The main characteristics of the dataset are the followings:

1. The images are obtained from 9 videos (RGB sequences) and a total of 14 different actors appear in those 9 sequences. In concrete, each sequence has a main actor (9 in total) which during the video interacts with secondary actors performing a set of different actions.

2. Each video (RGB sequence) was recorded with a mean 15 fps rate.

3. RGB images were stored with resolution 480×360 in BMP file format.

4. For each actor present in an image 14 limbs (if not occluded) were manually tagged: Head, Torso, R-L Upper-arm, R-L Lower-arm, R-L Hand, R-L Upper-leg, R-L Lower-leg, and R-L Foot.

5. Limbs are manually labeled using binary masks and the minimum bounding box containing each subject is defined.

6. The actors appear in a wide range of different poses and performing different actions/gestures.

7. For each video we manually labeled a set of 11 gesture/action categories: Wave, Point, Clap, Crouch, Jump, Walk, Run, Shake Hands, Hug, Kiss, and Fight.

Finally, the easy and challenging aspects of the HuPBA $8k+$ dataset are listed in Table 1.

[1]The whole number of manual labeled limbs exceeds 120000.

Easy
Fixed Camera
Frontal point of view
Full body capture
The main actor is kept within a sequence
Several instances of each gesture/action
Gestures/actions differentiated by an idle pose in most cases
Fixed background across all video sequences
Challenging
Within each sequence:
Gestures/actions execution involve most limbs
Large variability of poses
Some gestures/actions imply the interaction of various actors
Some parts of the body may be occluded
Between sequences:
Variations in clothing, skin color, gender, height and corporal conditions
Some parts of the body may be occluded

Table 1: Easy and challenging aspects of the HuPBA $8k+$ dataset.

2.1 Data Format and Structure

The dataset we introduce is composed by RGB images, labeled limbs (binary masks) and additional information that has a specific structure to distinguish the location of limbs and gestures/actions for each actor. Additionally, for each actor, a pair of structured files are created to store the location of the bounding-boxes for each RGB image and the start-end frames associated to the gestures/actions executed. The folder structure that contains the HuPBA $8k+$ dataset is shown in Fig. 1.

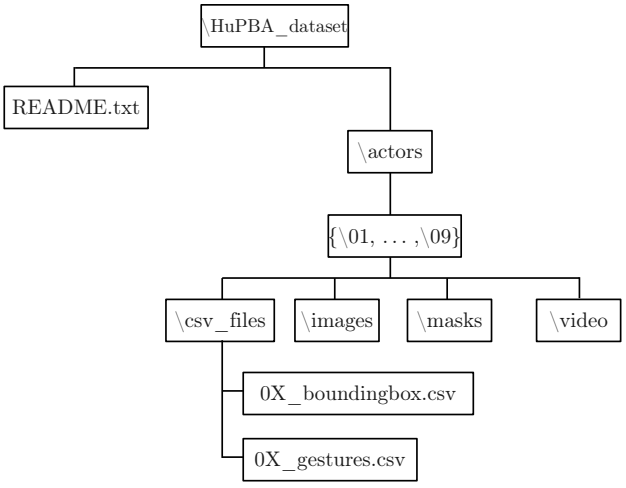


Figure 1: Folders structure.

2.1.1 Folder \images

In this folder, we store the set of frames for a given video sequence. The folder *\images* contains the sequence of RGB images (480×360 pixels). Each image name has the structure *idActor_numberFrame.bmp*, where:

- **idActor**: Numerical identifier of the actor $\{01, 02, ..., 09\}$.
- **numberFrame**: Numerical identifier of the image in the sequence.

2.1.2 Folder \masks

This folder contains the binary masks for each one of the 14 limbs appearing on each frame. In the case of two actors appearing in a frame, there will be an *id* for each one in order to distinguish limbs. Each binary mask name has the structure *idActor_numberFrame_idUser_idLimb.bmp*, where:

- **idActor**: Numerical identifier of the actor $\{01, 02, ..., 09\}$.
- **numberFrame**: Numerical identifier of the image in the sequence.
- **idUser**: Numerical identifier for the actor that appears in the image. Values $\{1, 2, ..., n\}$. In case of appearing two actors: The main actor and another, the main actor is 1, the second is 2, and so on.
- **idLimb**: Numerical identifier of the limb, which are described in Fig. 2.

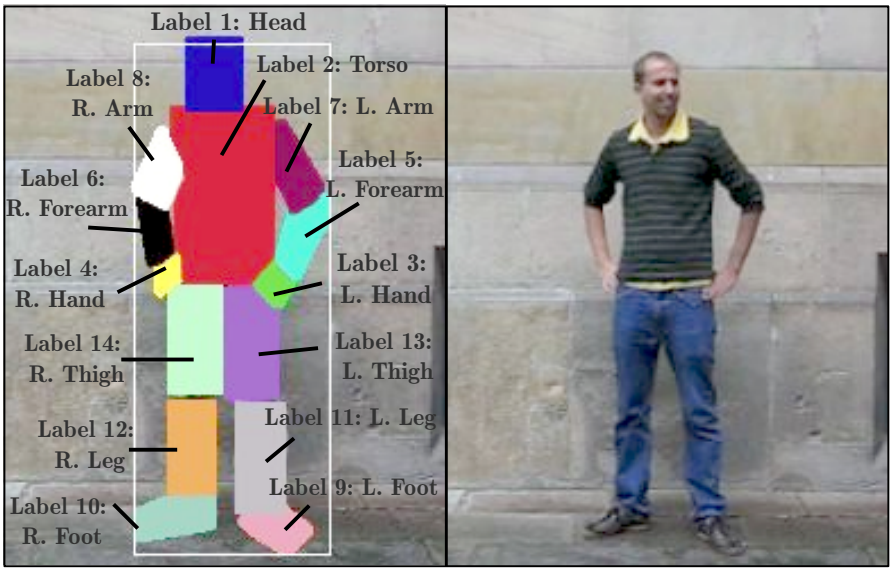


Figure 2: Human-Limb labelling on the HuPBA $8k+$ dataset.

2.1.3 Bounding-boxes

In addition, for each sequence there is a file *0X_boundingbox.csv* located in the directory *\csv_files* that contains the bounding-boxes of all actors that appear in that sequence. That is, for each actor that appears in an image, its bounding-box is given. In the case of two actors appearing in an image, two bounding-boxes will be described, one for each actor, as shown in Fig. 3. The *csv* file contains the following structure:

- **id_user**: Numerical identifier for the actor that appears in the image. Values $\{1, 2, ..., n\}$. In case of appearing two actors: The main actor and another, the main actor is 1 and the second is 2. Thus, there will be two bounding-boxes, one for 1, another for 2, and so on.
- **number_frame**: Numerical identifier of the image in the sequence.
- **x**: Minimum position of X. That is, the leftmost.
- **y**: Minimum position of Y. That is, the uppermost.
- **width**: Width of the bounding-box.
- **height**: Height of the bounding-box.

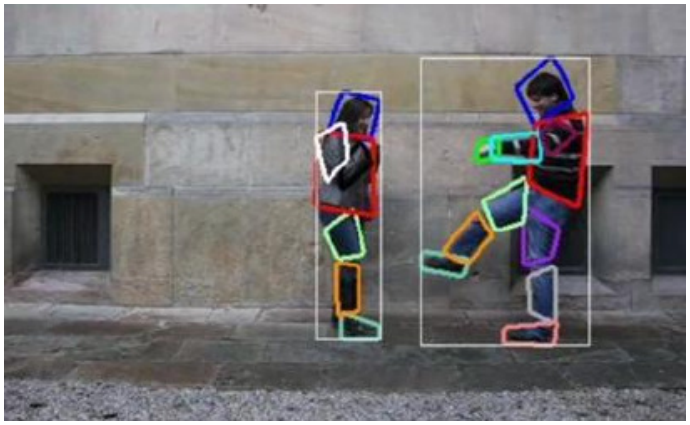


Figure 3: Sample of two bounding-boxes in a frame.

2.1.4 Gestures/Actions

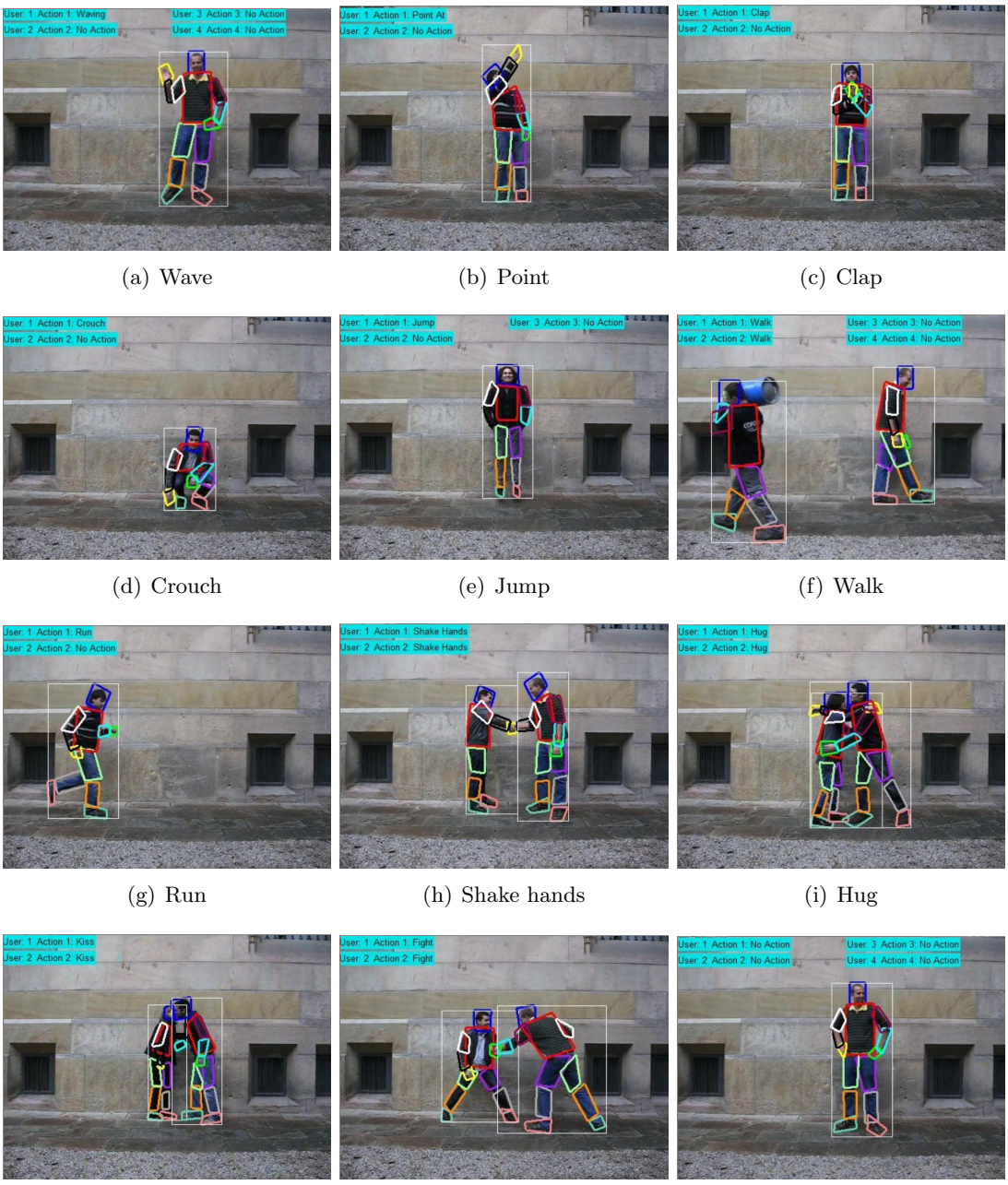
Besides of the human-limb labeling provided on the dataset, we also annotated gestures/actions performed by the actors. The 11 gesture/action categories labeled are the following: Wave, Point, Clap, Crouch, Jump, Walk, Run, Shake Hands, Hug, Kiss and Fight. An example of key frames for the different gesture/action categories are shown in Fig. 4. Each set of gestures/actions performed by an actor is associated to a file *./csv_files/0X_gestures.csv* that contains the following structure:

- **id_user**: Numerical identifier for the actor that appears in the image. Values $\{1, 2, ..., n\}$.
- **label_gesture**: Numerical identifier related to the gesture/action performed. There are gestures/actions that involve just one actor (i.e. walk or run), and others more than one actor (i.e. fight or kiss).
- **start_frame**: The number of image where the gesture/action starts.
- **end_frame**: The number of the image where the gesture/action ends.

Finally, in Table 2 we compare the HuPBA $8k+$ dataset characteristics with some publicly available datasets. These public datasets are chosen taking into account the variability of limbs and gestures/actions. Once can see that the novel dataset offers higher number of annotated limbs at pixel precision in comparison with state-of-the-art public available datasets. In case of gestures/actions, there is more equality in the number of gestures/actions set with the others datasets (i.e. HOLLYWOOD (HW), MMGR13, Human Actions). In contrast, MMGR13 present much more variety of gestures/actions and samples than the proposed dataset.

	HuPBA	PARSE[27]	BUFFY[19]	UIUC people[34]	LEEDS SPORTS[23]	HW[24]	MMGR13[15]	H.Actions[32]	Pascal VOC[17]
Labeling at pixel precision	Yes	No	No	No	No	-	No	No	Yes
Number of limbs	14	10	6	14	14	-	16	-	5
Number of labeled limbs	124 761	3 050	4 488	18 186	28 000	-	27 532 800	-	8 500
Number of frames	8 234	305	748	1 299	2 000	-	1 720 800	-	1 218
Full body	Yes	Yes	No	Yes	Yes	-	Yes	Yes	Yes
Limb annotation	Yes	Yes	Yes	Yes	Yes	No	Yes	No	Yes
Gesture annotation	Yes	No	No	No	No	Yes	Yes	Yes	No
Number of gestures	11	-	-	-	-	8	20	6	-
Number of gesture samples	235	-	-	-	-	430	13 858	600	-

Table 2: Comparison of public dataset characteristics.



(a) Wave (b) Point (c) Clap

(d) Crouch (e) Jump (f) Walk

(g) Run (h) Shake hands (i) Hug

(j) Kiss (k) Fight (l) Idle

Figure 4: Different gesture categories labeled on the HuPBA $8k+$ dataset. Images from (a) to (g) illustrate single actor gestures/actions, and images from (h) to (k) show gestures/actions that required interacting with a secondary actor. Additionally, (l) shows an example of an existing idle gesture/action.

3 ECOC and GraphCut based multi-limb segmentation

In the following subsections we describe the proposed system for automatic segmentation of human limbs. To accomplish this task, we start by defining a framework divided in a two-stage procedure. The first stage, focused on binary person/background segmentation, is split in four main steps: a) Body part learning using cascade of classifiers, b) Tree-structure learning of human limbs, c) ECOC multi-limb detection, and d) Binary GrabCut optimization for foreground extraction. In the second stage, we segment the person/background binary mask into different limb regions. This stage is split in the following four steps: e) Tree-structure body part learning without background, f) ECOC multi-limb detection, g) Limb-like probability map definition, and h) Alpha-beta swap Graph Cuts multi-limb segmentation. The scheme of the proposed system is illustrated in Fig. 5.

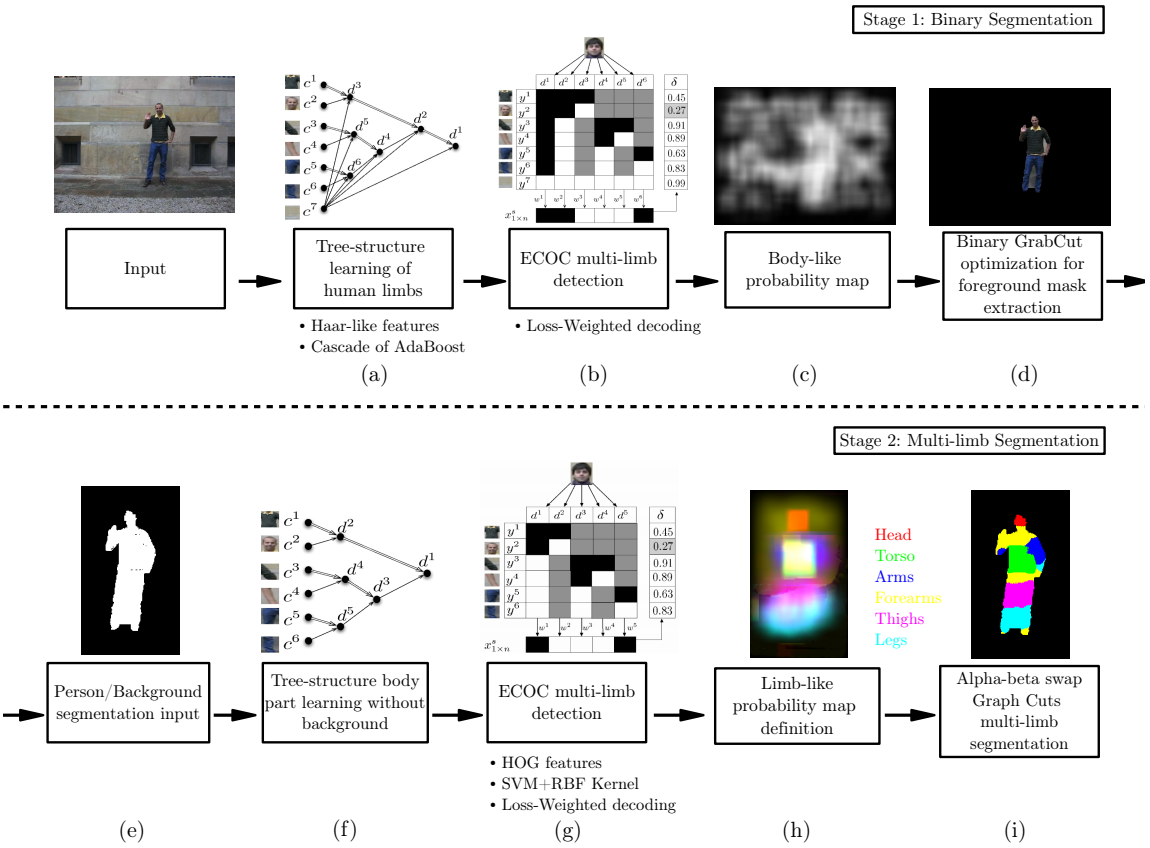


Figure 5: Scheme of the proposed human-limb segmentation method.

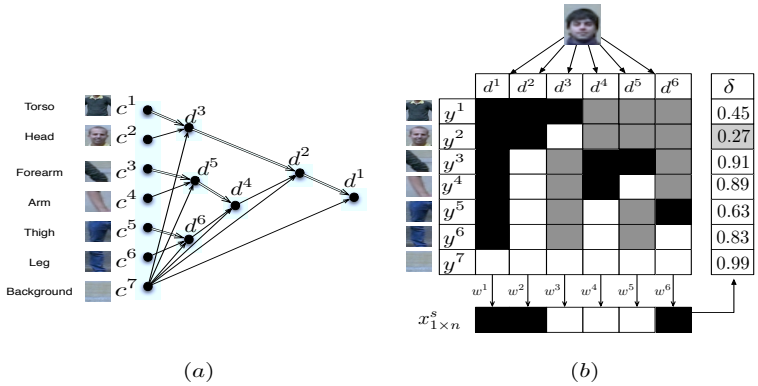
3.1 Body part learning using cascade of classifiers

The core of most human body segmentation methods in the literature relies on body part detectors. In this sense, most part detectors in literature follow a cascade of classifiers architecture [20, 25, 41, 13, 9]. Cascades of classifiers are based on the idea of learning and unbalanced binary problem by using the negative outputs of a classifier d^i as an input for the following classifier d^{i+1}. Particularly, this cascade structure allows any classifier to refine the prediction by reducing the false positive rate at every stage of the cascade. In this sense, we use AdaBoost as the base classifier in our cascade architecture. In addition, in order to make the body part detection rotation invariant, all body parts are rotated to the dominant gradient region orientation. Then, Haar-like features are used to describe the body parts.

Because of its properties, cascade of classifiers are usually trained to split one visual object from the rest of possible objects of an image. This means that the cascade of classifiers learns to detect a certain object (body part in our case), ignoring all other objects (all other body parts). However, if we define our problem as a multi-limb detection procedure, some body parts are similar in appearance, and thus, it makes sense to group them in the same visual category. Because of this reason, we propose to learn a set of cascade of classifiers where a subset of limbs are included in the positive set of a cascade, and the remaining limbs are included as negative instances together with background images in the negative set of the cascade. Applying this grouping for different cascades of classifiers in a tree-structure way and combining them in an Error-Correcting Output Codes (ECOC) framework enables the system to perform multi-limb detection [16].

3.2 Tree-structure learning of human limbs

The first issue to take into account when defining a set of cascades of classifiers is how to define the groups of limbs to be learnt by each individual cascade. For this task, we propose to train a tree-structure cascade of classifiers. This tree-structure defines the set of meta-classes for each dichotomy (cascade of classifiers) taking into account the visual appearance of body parts, which has two purposes. On one hand, we aim to avoid dichotomies in which body parts with different visual appearance belong to the same meta-class. On the other hand, the dichotomies that deal with classes that are difficult to learn (body parts with similar visual appearance) are defined taking into account few classes. An example of the body part tree-structure defined taking into account these issues for a set of 7 body limbs is shown in Fig. 6(a). Notice that classes with similar visual appearance (e.g. upper-arm and lower-arm) are grouped in the same meta-class in most dichotomies. In addition, dichotomies that deal with difficult problems (e.g. d^5) are focused only in the difficult classes, without taking into account all other body parts. In this case, class c^7 denotes the background.



(a) (b)

Figure 6: (a) Tree-structure classifier of body parts, where nodes represent the defined dichotomies. Notice that the single or double lines indicate the meta-class defined. (b) ECOC decoding step, in which a head sample is classified. The coding matrix codifies the tree-structure of (a), where black and white positions are codified as $+1$ and -1, respectively. c, d, y, w, X, and δ correspond to a class category, a dichotomy, a class codeword, a dichotomy weight, a test codeword, and a decoding function, respectively.

3.3 ECOC multi-limb detection

In the ECOC framework, given a set of N classes (body parts) to be learnt, n different bi-partitions (groups of classes or dichotomies) are formed, and n binary problems over the partitions are trained [3]. As a result, a codeword of length n is obtained for each class, where each position (bit) of the code corresponds to a response of a given classifier d (coded by $+1$ or -1 according to their class set membership, or 0 if a particular class is not considered for a given classifier). Arranging the codewords as rows of a matrix, we define a *coding matrix* M, where $M \in \{-1, 0, +1\}^{N \times n}$. During the *decoding* (or testing) process, applying the n binary classifiers, a code x is obtained for each data sample ρ in the test set. This code is compared to the base codewords $(y^i, i \in [1, .., N])$ of each class defined in the matrix M, and the data sample is assigned to the class with the *closest* codeword [16].

The ECOC coding step has been widely tackled in the literature either by predefined or problem-dependent strategies. However, recent works showed that problem-dependent strategies can obtain high performance by focusing on the idiosyncrasies of the problem [2]. Following this fashion, we define a problem dependent coding matrix in order to allow the inclusion of cascade of classifiers and learn the body parts. In particular, we propose to use a predefined *coding* matrix in which each dichotomy is obtained from the body part tree-structure described in previous section. Fig. 6(b) shows the coding matrix codification of the tree-structure in Fig. 6(a).

3.3.1 Loss-weighted decoding using cascade of classifier weights

In the ECOC *decoding* step an image is processed using a windowing method, and then, each image patch, that is, a sample ρ, is described and tested. In this sense, each classifier d outputs a prediction whether ρ belongs to one of the two previously learnt meta-classes. Once the set of predictions $x^{\rho}_{1\times n}$ is obtained, it is compared to the set of codewords of M, using a decoding function $\delta(x^{\rho}, M)$. Thus, the final prediction is the class with the codeword that minimizes $\delta(x^{\rho}, M)$. In [16] the authors proposed a problem-dependent decoding function (distance function that takes into account classifier performances) obtaining very satisfying results. Following this core idea, we use the Loss-Weighted decoding of Equation 1, where M_w is a matrix of weights and L is a loss function ($L(\theta) = \exp^{-\theta}$).

$$\delta_{LW}(x^s, i) = \sum_{j=1}^{n} M_w(i,j) L(y^i_j \cdot d^j(x^s)) \tag{1}$$

In Equation 1, M_w (weight matrix) corresponds to the product of cascade accuracies at each stage. Thus, each column i of M_w is assigned a weight w^i as,

$$w^i = \prod_{j=1}^{k} \frac{TP(d^i_j) + TN(d^i_j)}{TP(d^i_j) + FN(d^i_j) + FP(d^i_j) + TN(d^i_j)}, \tag{2}$$

for a cascade of classifiers of k stages, where d^i_j stands for the i-th cascade and stage j, $j \in [1, .., k]$, and TP, TN, FN, and FP computes the number of true positives, true negatives, false negatives and false positives, respectively. Finally, a body-like probability map $P^{bl} \in [0,1]^{l\times w}$, where l and w are the length and width of I, is build. This map contains, at each position P^{bl}_{ij}, the proportion of body part detections for each pixel over the total number of detections for the whole image. In other words, pixels belonging to the human body will show a higher body-like probability than the pixels belonging to the background. Examples of probability maps obtained from ECOC outputs are shown in Fig. 9(e) and 9(g), respectively. (see also step (c) in Fig. 5).

3.4 Binary GrabCut optimization for foreground mask extraction

GrabCut [21] has been widely used for interactive background/foreground extraction (binary segmentation). Formally, given a color image I, let us consider the array $z = (z_1, ..., z_q, ..., z_Q)$ of Q pixels where $z_i = (R_i, G_i, B_i)$, $i \in [1, ..., Q]$ in RGB space. The segmentation is defined as an array $\boldsymbol{\alpha} = (\alpha_1, ...\alpha_Q)$, $\alpha_i \in \{0, 1\}$, assigning a label to each pixel of the image indicating if it belongs to background or foreground. A trimap T is defined consisting of three regions: T_B, T_F and T_U, each one containing initial background, foreground, and uncertain pixels, respectively. Pixels belonging to T_B and T_F are clamped as background and foreground respectively—which means GrabCut will not be able to modify these labels, whereas those belonging to T_U are actually the ones the algorithm will be able to label. Color information is introduced by GMMs. A full covariance GMM of U components is defined for background pixels ($\alpha_i = 0$), and another

one for foreground pixels ($\alpha_j = 1$), parameterized as follows,

$$\boldsymbol{\theta} = \{\pi(\alpha, u), \mu(\alpha, u), \Sigma(\alpha, u), \alpha \in \{0, 1\}, u = 1..U\}, \tag{3}$$

being π the weights, μ the means and Σ the covariance matrices of the model. We also consider the array $\mathbf{u} = \{u_1, ..., u_i, ...u_Q\}$, $u_i \in \{1, ...U\}$, $i \in [1, ..., Q]$ indicating the component of the background or foreground GMM (according to α_i) the pixel z_i belongs to. The energy function for segmentation E is then,

$$\mathbf{E}(\boldsymbol{\alpha}, \mathbf{u}, \boldsymbol{\theta}, \mathbf{z}) = \mathbf{U}(\boldsymbol{\alpha}, \mathbf{u}, \boldsymbol{\theta}, \mathbf{z}) + \lambda \mathbf{V}(\boldsymbol{\alpha}, \mathbf{z}), \tag{4}$$

where $\mathbf{U}$ is the likelihood potential based on the probabilities $p(\cdot)$ of the GMM,

$$\mathbf{U}(\boldsymbol{\alpha}, \mathbf{u}, \boldsymbol{\theta}, \mathbf{z}) = \sum_i -\log\, p(z_i|\alpha_i, u_i, \boldsymbol{\theta}) - \log\, \pi(\alpha_i, u_i), \tag{5}$$

and $\mathbf{V}$ is a regularizing prior assuming that segmented regions should be coherent in terms of color, taking into account a neighborhood $\mathcal{N}$ around each pixel,

$$\mathbf{V}(\boldsymbol{\alpha}, \mathbf{z}) = \gamma \sum_{\{m,q\} \in \mathcal{N}} [\alpha_q \neq \alpha_m] \exp\, (-\beta \|z_m - z_q\|^2), \tag{6}$$

where weight $\lambda \in \mathbb{R}^+$ specifies the relative importance of the boundary term against the unary term U.

With this energy minimization scheme and given the initial trimap T, the final segmentation is performed using a minimum cut algorithm. However, we propose to omit the classical semiautomatic trimap initialization by an automatic trimap assignment based on the human body probability map $P^{bl} \in [0, 1]^{l \times w}$. In this sense, depending on the probability of each pixel it will be assigned to a certain tag T_B, T_F and T_U.

3.5 Tree-structure body part learning without background

Once the binary person/background segmentation is performed by means of GrabCut (mask shown in Fig. 5(e)), we apply a second procedure in order to split the person mask into a set of human limbs.

For this step, we define a new tree-structure classifier similar to the one described in Section 3.2 without including the background class c^7 shown in Fig. 6(a). An example of the tree-structure body part taking into account the set of 6 body limbs is shown in Fig. 7(a).

3.6 ECOC multi-limb detection

In order to obtain an accurate detection of human limbs within the segmented user mask, we base on HOG descriptor [10] and SVM classifier which have shown to obtain robust results in human estimation scenarios [10, 21, 20]. We extract HOG features for the different body parts (previously normalized to dominant region orientation), and then,

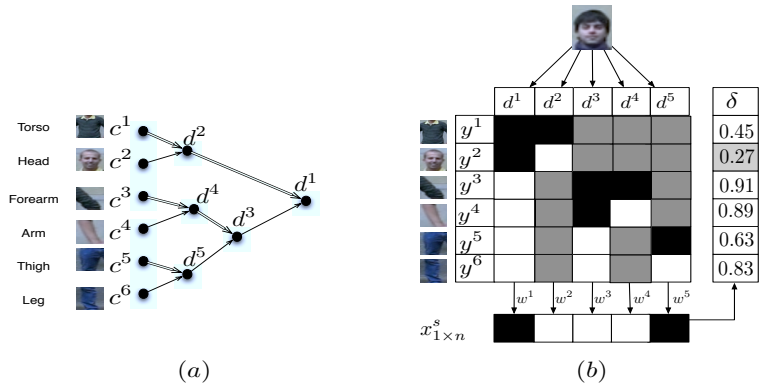


Figure 7: (a) tree-structure classifier of 6 body parts, (b) ECOC decoding step.

SVMs classifiers are trained on that feature space, using a Generalized Gaussian RBF Kernel based on Chi-squared distance [37].

This stage follows a similar pipeline as the one described in Section 3.3. In this sense, each SVM classifier learns a binary partition of human limbs but without taking into account the background class. As shown in Fig. 6(b), we train $n = 6$ SVMs with different binary human-limb partitions.

At the ECOC decoding step, we also use the Loss-Weighted decoding [16] function shown in Equation 1 (an example is shown in Fig 7(b)). In this sense, for each RGB test image corresponding to the binary mask shown in Fig. 5(e), we adopt a sliding window approach and test each patch on our ECOC multi limb recognition system. Then, based on the ECOC output we construct a set of limb-like probability maps. Each map P^c contains, at each position P^c_{ij}, the probability of pixel at the entry (i, j) of belonging to the body part class c, where $c \in \{1, 2, ..., 6\}$. This probability is computed as the proportion of detections at point (i, j) over all detection for class c. Examples of probability maps obtained from ECOC outputs are shown in Fig. 5(h). While Haar-like based on AdaBoost gave us a very accurate and fast initialization of human regions for binary user segmentation, in this second step, HOG-SVM is applied in a reduced region of the image, providing better estimates of human limb locations.

3.7 Alpha-beta swap Graph Cuts multi-limb segmentation

We base our proposal on Graph Cuts theory to tackle our human-limb segmentation problem [7, 21, 29, 6, 8]. In [8], Boykov et. al. developed an algorithm, named α-β swap graph-cut, which is able to cope with the multi-label segmentation problem. The α-β swap graph-cut is an extension of binary graph cuts that performs an iterative procedure where each pair of labels (α_q, α_m), $\{m, q\} \in \{1, 2, ..., 6\}$, are segmented using GraphCuts. This procedure segment all α pixels from β pixels with GraphCuts and the algorithm will update the α-β combination at each iteration until convergence. However, to cope with the multi-label case, an extension of the binary Graph Cuts optimization

framework described in Section 3.4 is needed.

In this sense, $\alpha_i \in \{1, ..., c\}$ and an initial labeling $T \in \{T_1, ..., T_c\}$ is defined by an automatic trimap assignment based on the set of limb-like probability maps $P^c \in [0, 1]^{l \times w}$ defined in previous section. In addition, the coefficient that multiplies the exponential term in Equation 6, $[\alpha_q \neq \alpha_m]$, is changed to $\Omega(c_q, c_m)$, which penalizes relations between pixels z_q and z_m depending on their label assignations and a user-predefined pair-wise cost to each possible combination of labels,

$$\mathbf{V}(\mathbf{c}, \mathbf{z}) = \gamma \sum_{\{m,q\} \in \mathcal{N}} \Omega(c_q, c_m) \exp\left(-\beta \|z_m - z_q\|^2\right). \tag{7}$$

In concrete, in order to introduce prior costs between different labels, $\Omega(c_q, c_m)$ must fulfill some constraints related to spatial coherence between the different labels, taking into account the natural constraints of the human limbs (i.e. head must be closer to torso than legs, arms are nearer to forearms than head, etc.). In particular, we experimentally fixed the penalization function Ω as defined in Table 3.

	Head	**Torso**	**Arms**	**Forearms**	**Thighs**	**Legs**	**Background**
Head	0	20	35	50	70	90	1
Torso	20	0	15	25	40	70	1
Arms	35	15	0	10	60	80	1
Forearms	50	25	10	0	30	60	1
Thighs	70	40	60	30	0	10	1
Legs	90	70	80	60	10	0	1
Background	1	1	1	1	1	1	1

Table 3: Prior cost between each pair of labels.

4 Experimental results

In order to present the experimental results, we first discuss the data, experimental settings, methods and validation protocol.

4.1 Data

We use the proposed HuPBA $8k+$ dataset described in Section 2. We reduced the number of limbs from the 14 available in the dataset to 6, grouping those that are similar by symmetry (right-left) as arms, forearms, thighs and legs. Thus, the set of limbs of our problem is composed by: *head*, *torso*, *forearms*, *arms*, *thighs* and *legs*. Although labeled within the dataset, we did not include hands and feet in our multi-limb segmentation scheme. Finally, in order to train the limb classifiers, ground truth masks are used to normalize all limb regions per dominant orientation, and both Haar-like features and HOG descriptors are computed based on the aspect ratio of each region, making the descriptions scale invariant.

4.2 Methods and experimental settings

In this section we introduce the different methods compared for **binary segmentation**, **multi-limb segmentation** and **action/gesture recognition** tasks. In addition, the experimental settings for these methods are explained.

4.2.1 Binary segmentation methods

As the first stage of our approach computes a binary person/background segmentation, we compare in this step the following methods:

- **P.Detector+GbCut:** The well-known Person Detector of [10] followed by GrabCut segmentation.
- **C.Class+GbCut:** The cascade of classifiers proposed by Viola and Jones [36], training one cascade of classifiers per limb and GrabCut segmentation.
- **ECOC+GbCut:** The proposed ECOC tree-structure body part classifier and automatic GrabCut segmentation for person/background segmentation.

4.2.2 Multi-limb segmentation methods

To evaluate the performance of our proposal for multi-limb segmentation, we compare our strategy with two state-of-the-art methods for multi-limb segmentation:

- **FMP:** This method was proposed by Yang and Ramanan [38, 39] and it is based on Flexible Mixtures-of-Parts (FMP). We compute the average of each set of mixtures for each limb and for each pyramid level in order to obtain the probability maps for each limb category. In order to compute the probability map of the background

category, we subtract 1 with the maximum probability $\in [0, 1]$ of the set of limbs detection at pixel location.

- **IPP:** This method is proposed by Ramanan [27] and it is based on an Iterative Parsing Process (IPP). We use it to extract the limb-like probability maps followed by α-β swap graph-cut multi-limb segmentation. The background category is computed as shown in FMP method.

- **ECOC+GraphCut:** Our proposed human limb segmentation scheme shown in Fig. 5.

4.2.3 Action/gesture recognition methods

In the case of the action recognition task our goal is to provide with a firm baseline of the recognition of the 11 actions categories labeled within the *HuPBA* $8K+$ dataset.In order to do it, we compare performance of the following standard methodologies:

- **Dynamic Time Warping using a random sample**: We use the standard DTW algorithm to recognize the different actions categories in the dataset [30]. In order to compute the cost matrix for each of the gesture/action classes we choose a sample of that category at random.

- **Dynamic Time Warping using the mean sample**: Following the trend in [22], in order to compute the cost matrix we form a mean sample of each one of the action classes. That is, we choose the sample of each category and align all samples with it. Then, once all samples from the same class are aligned (they have the same length) we compute the mean, an example is shown in Fig. 8.

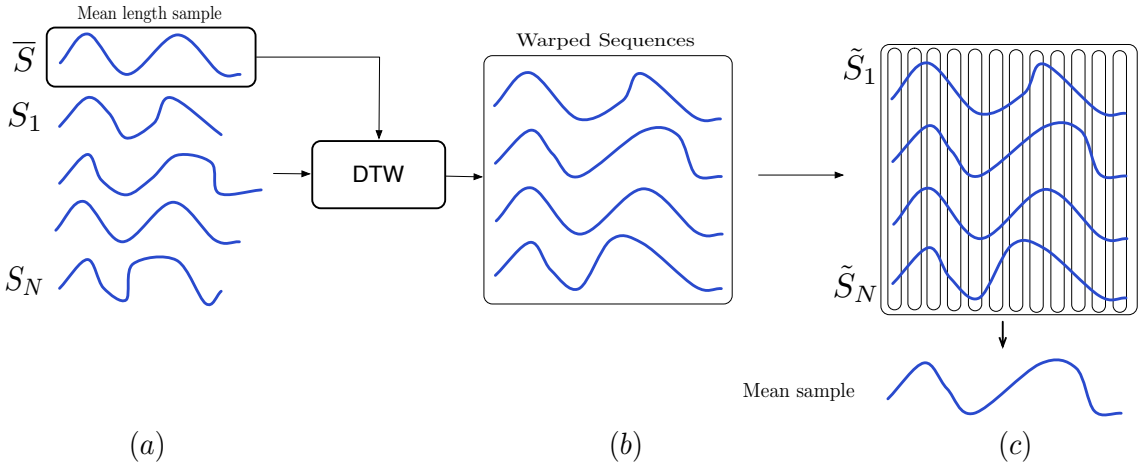


Figure 8: (a) Action samples and selected median length sample. (b) Aligned samples with same length . (c) Computation of the mean sample.

- **Hidden Markov Model**: We use the standard discrete HMM framework [33]. Each HMM, was trained using the Baum-Welch algorithm, and 3 states were experimentally set for the every action category, using a vocabulary of 50 symbols computed using K-means over the training data features. Final recognition is performed with temporal sliding windows of different wide sizes, based on the training samples length variability.

The computation of the feature vector for training and testing the action recognition approaches is based on the segmentation results of our approach. Given the multi-segmentation of limbs, we computed the feature vector of a frame as the concatenation of the 6 limb-like probability maps, resizing each one of them to a 40×20 pixels region and vectorizing that region. Obtaining a final vector of $d = 40 \cdot 20 \cdot 6 = 4800$ dimensions, which is then reduced to $d = 150$ dimensions using the Random Projection algorithm [4].

4.2.4 Experimental settings

In a preprocessing step, we resized all limb sample to a 32×32 pixels region for computational purposes. Then, we used the standard Cascade of Classifiers based on AdaBoost and Haar-like features [36], and we forced a 0.99 false positive rate and maximum of 0.4 false alarm rate during 8 stages. To detect limbs with trained cascades of classifiers, we applied a sliding window approach with an initial patch size of 32×32 pixels up to 60×60 pixels. As a final part of the first stage, binary Graph Cuts were applied to obtain the binary segmentation where the initialization values of foreground and background were provided to the binary Graph Cut algorithm and tuned via cross-validation.

For the second stage, we set the following parameters for the HOG descriptor: 32×32 window size, 16×16 block size, 8×8 block stride, 8×8 cell size and 8 for number of bins. Then, we trained SVMs with a Generalized Gaussian RBF kernel based on Chi-squared distance, (see Fig.(a) 7). The parameters of the kernel, C and γ were tuned via cross-validation. Finally, the model selection step was done via a leave-one-sequence-out cross-validation. For multi-limb segmentation we used the alpha-beta GraphCut procedure, where we set a 8×8 neighboring grid and tuned the λ parameter of GraphCut using cross-validation.

For the action recognition experiments the cost-threshold and the action/gesture model for both DTW experiments was obtained by cross-validation on training data, using a leave-one-sequence-out procedure. For HMM method, each HMM and its corresponding probability-threshold was obtained by cross-validation on training data, using a leave-one-sequence-out procedure.

4.3 Validation measurement

In order to evaluate the results for the three different tasks: binary segmentation, multi-label segmentation and gesture/action recognition, we use the Jaccard Index of overlapping ($J = \frac{A \bigcap B}{A \bigcup B}$) where A is the ground-truth and B is the corresponding prediction.

4.4 Experimental Results

In this section we show results for the three different tasks: **binary segmentation**, **multi-label segmentation** and **action/gesture recognition**.

4.4.1 Binary segmentation results

In Fig. 9 we can see an example of the person/background segmentation obtained by the compared methodologies. In particular, we can see in Fig. 9(d) how the segmentation obtained by the Person Detector+GbCut method yields a poor result, segmenting dark regions of the image. Furthermore, when comparing Fig. 9(e) and 9(f), the improvement in the body-like probability map obtained by the ECOC+GbCut approach over the cascade class+GbCut method are clearly significant.

Moreover, we can see more results of our proposal in Fig. 10 and Fig. 11 that show relevant results either individual person or two people although there are some regions that are difficult to segment because of their variability in contrast with the fixed background. In addition, some results denote good segmentation in Fig. 12 and Fig. 13 since most background around the person is removed.

In order to evaluate the performance of the compared methodologies, Table 4 shows the mean overlapping obtained on the whole dataset together with the standard deviation. From the results one can see the ECOC+GbCut method outperforms the compared methodologies at least by a 5%. This improvement is the effect of two causes. The former is the Error-Correcting capabilities of the ECOC framework. The latter, is the tree-structure definition of the coding matrix, which allows base classifiers to obtain accurate results.

P.Detector+GbCut	**C.Class+GbCut**	**ECOC+GbCut**
49.60 ± 20.45	58.26 ± 17.31	$\mathbf{61.79 \pm 14.02}$

Table 4: Mean overlapping and standard deviation.

4.4.2 Multi-limb segmentation results

Firstly, we show the priors obtained from HOG descriptors and SVM classifiers for different samples in Fig. 14 and Fig. 15 where some probability maps like head, torso, thighs and legs are more accurate than arms and forearms. In concrete, the forearms probability maps are the less representative for the 6 limb-like probability map categories. Additionally, more results focusing on individual person probability maps are shown in Fig. 16 and Fig. 17 in which we can see more precisely the limb categories more discriminative for each other.

For the Multi-limb segmentation task, we show in Fig. 18, Fig. 19, Fig. 20 and Fig. 21 qualitative results for some samples of the *HuPBA* $8k+$ dataset. When comparing the qualitative results we can see how the FMP method [38, 39] performs worse than its counter parts. In addition, one can se how IPP and our method obtain similar results in most cases. However, the IPP lacks of a good person/background segmentation.

Furthermore, we provide with quantitative results in terms of the Jaccard Index. In Fig. 22 we show the overlapping performance obtained by the different methods, where each plot shows the overlapping for a certain limb. In addition, we use a 'Do not care' value which provides a more flexible interpretation of the results. Consider the ground truth of a certain limb category in an image as a binary image, which pixels take value 1 when those pixels are labeled to belong to such limb. Then, the 'Do not care' value is defined as the number of pixels which are ignored at the limits of each one of the ground truth instances. Thus, by using this approach we can compensate the pessimistic overlap metric in situations when the detection is shifted some pixels. In this sense, we analyze the overlapping performance as a function of a 'Do not care' value that ranges from 0 to 4.

When analyzing quantitative results, we see how our method outperforms the compared methodologies for some limb categories. In particular, for the *head* region both methods obtain similar results, which is intuitive since the method used to detect the head is the well known face detector. Finally, we see how FMP method is in almost all cases obtaining the worst performance. As shown in Fig. 22(g), for the mean overlapping considering all the segmented limbs our method outperforms the rest of approaches up to 3 pixels of "Do not care" evaluation.

4.4.3 Action recognition results

In this section we show some samples according to the gesture categories in Fig. 23, Fig. 24 and Fig. 25 and quantitative results obtained by the different gesture recognition methods in terms of the Jaccard Index. Furthermore, to allow a deeper analysis of the proposed methodologies, in our evaluations we use a 'Do not care' value which provides a more flexible interpretation of the results. Consider the ground truth of a certain action category in a video sequence as a binary vector, which activates when a sample of such category is observed in the sequence. Then, the 'Do not care' value is defined as the number of bits (frames) which are ignored at the limits of each one of the ground truth instances. Thus, by using this approach we can compensate the pessimistic overlap metric in situations when the detection is shifted some frames. The Jaccard Index as a function of the 'Do not care' value for the 11 action categories and the mean Jaccard Index among action categories are shown in Fig. 26.

When analyzing quantitative results we see how the DTW Mean methods outperforms for most action categories the standard DTW Random and HMM methods. In addition, when computing the mean Jaccard Index among all gesture categories the DTW Mean approach also ranks first, obtaining a mean Jaccard Index of 0.20. This good result is due to the use of information from all action samples which encodes the intra-class variability of the gesture categories. Finally, we can see how in all cases Hidden Markov Model achieves the lowest performance.

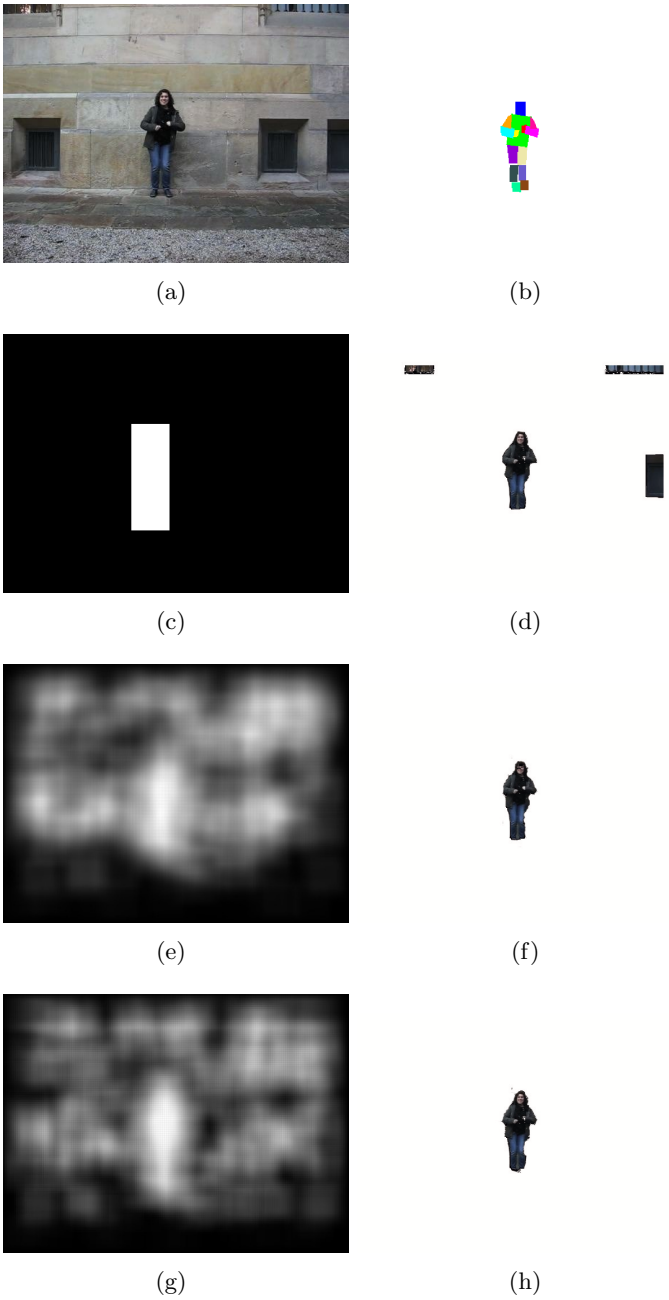


Figure 9: (a) Original RGB image. (b) Multi-limb ground truth. (c) Probability map obtained by the Person Detector method. (d) Person/background segmentation of the Person Detector+GbCut approach. (e) Probability map yielded by the cascade class. method. (f) Person/background segmentation of the cascade class method. (g) Probability map obtained from the ECOC method. (h) RGB segmentation obtained by the ECOC+GbCut approach.

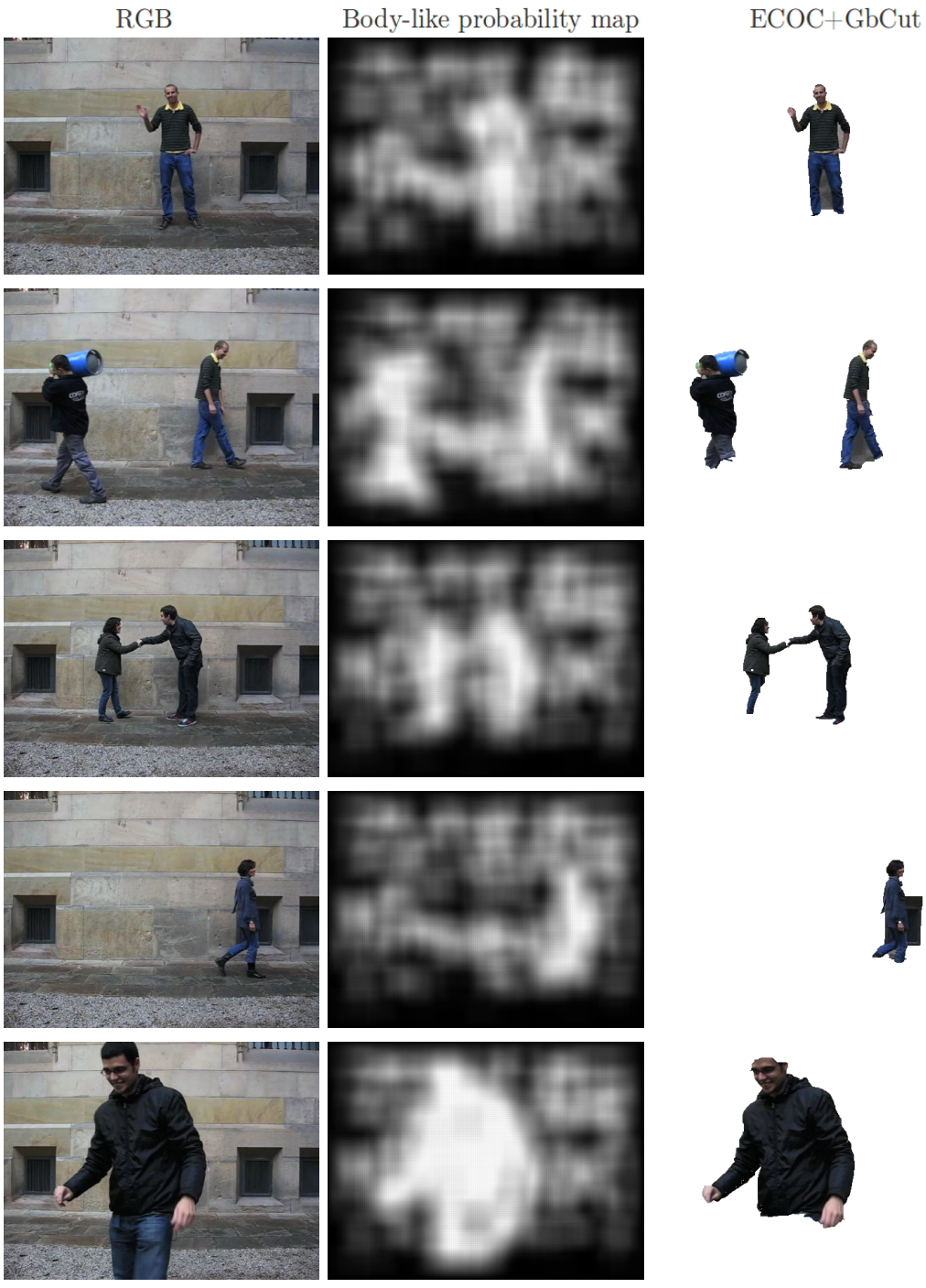


Figure 10: Binary segmentation results of our proposal. From left to right, the columns show the original RGB images, probability maps and ECOC+GbCut approach.

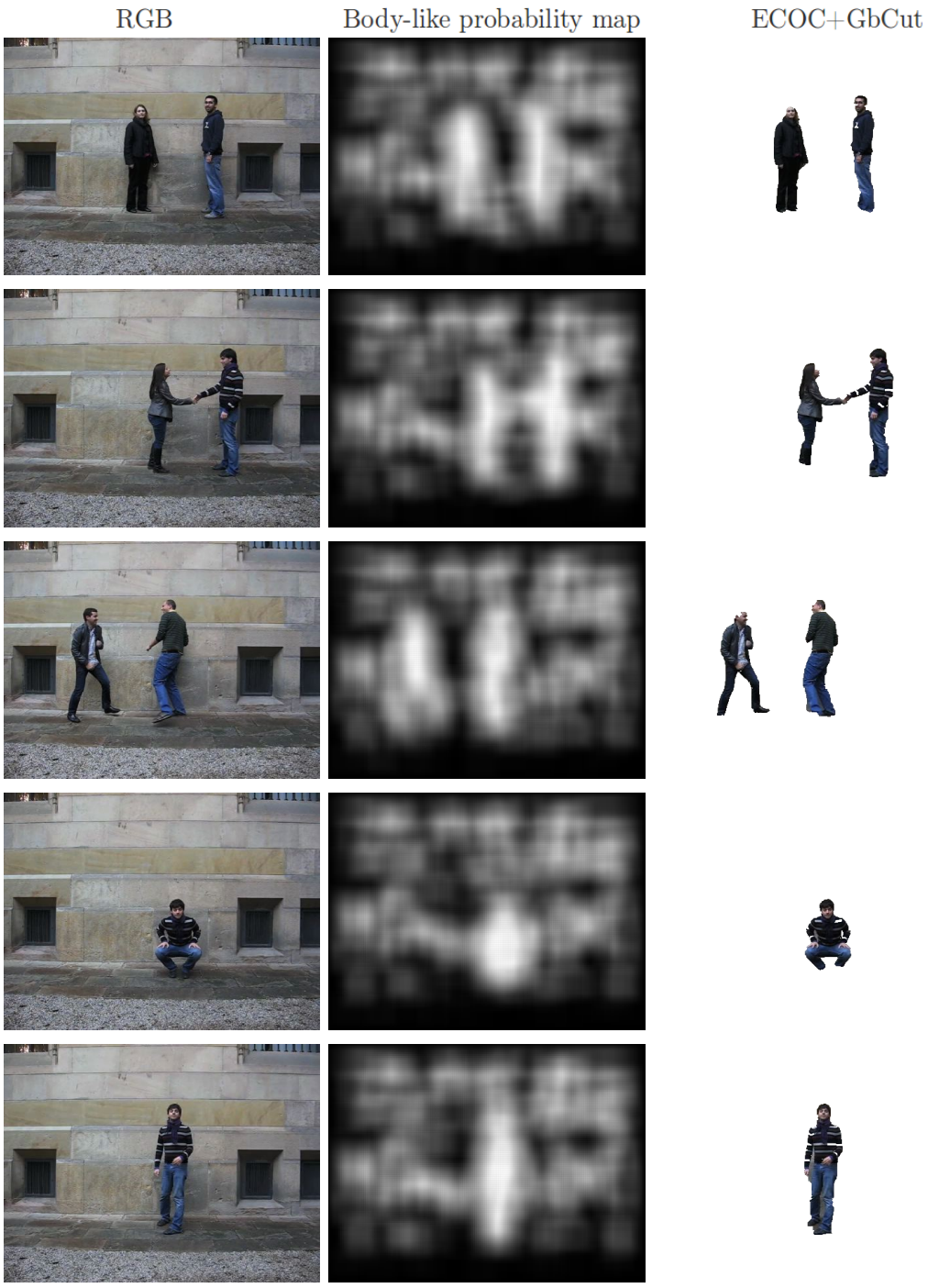


Figure 11: Binary segmentation results of our proposal. From left to right, the columns show the original RGB images, probability maps and ECOC+GbCut approach.

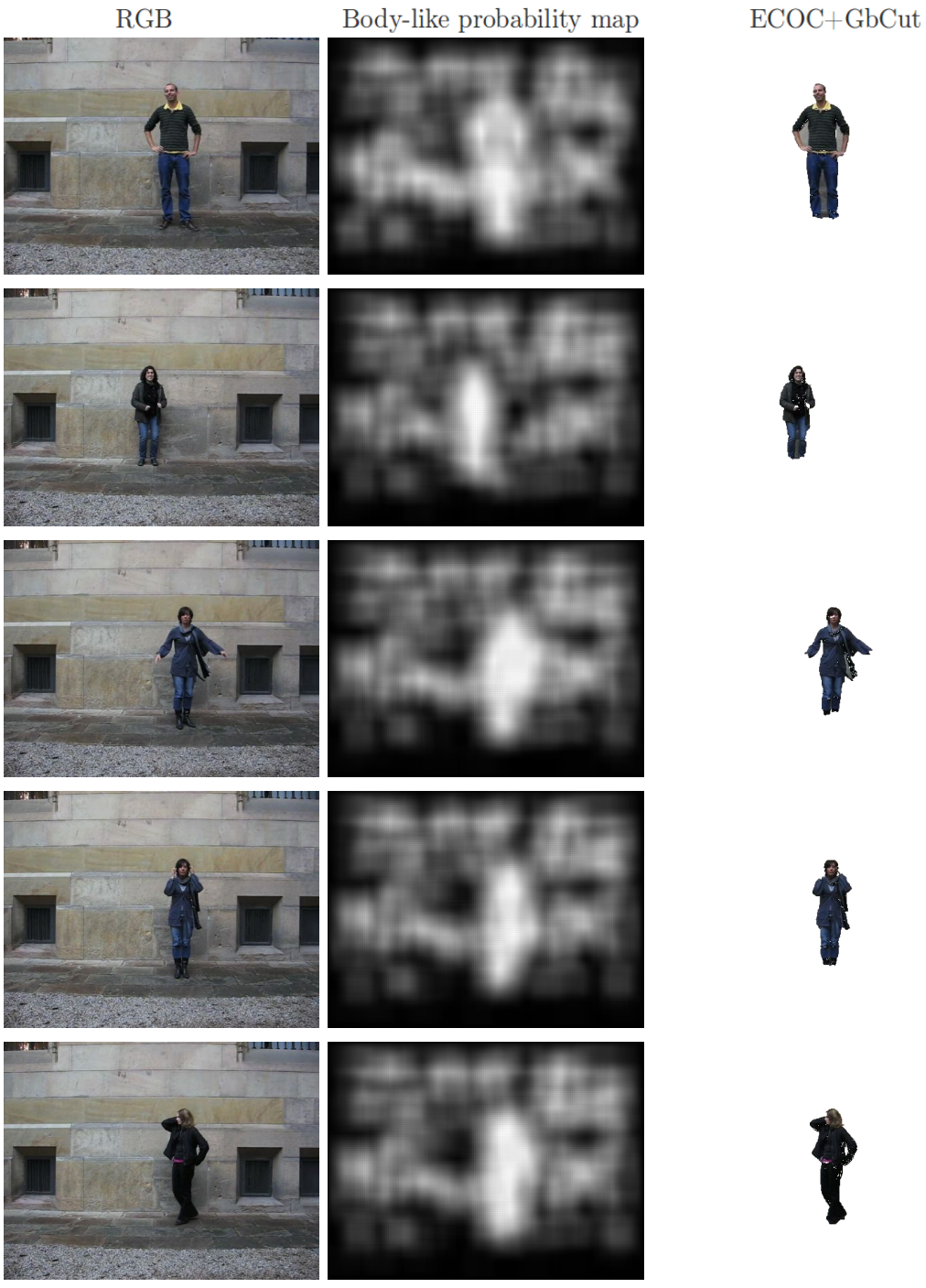


Figure 12: Binary segmentation results of our proposal. From left to right, the columns show the original RGB images, probability maps and ECOC+GbCut approach.

Figure 13: Binary segmentation results of our proposal. From left to right, the columns show the original RGB images, probability maps and ECOC+GbCut approach.

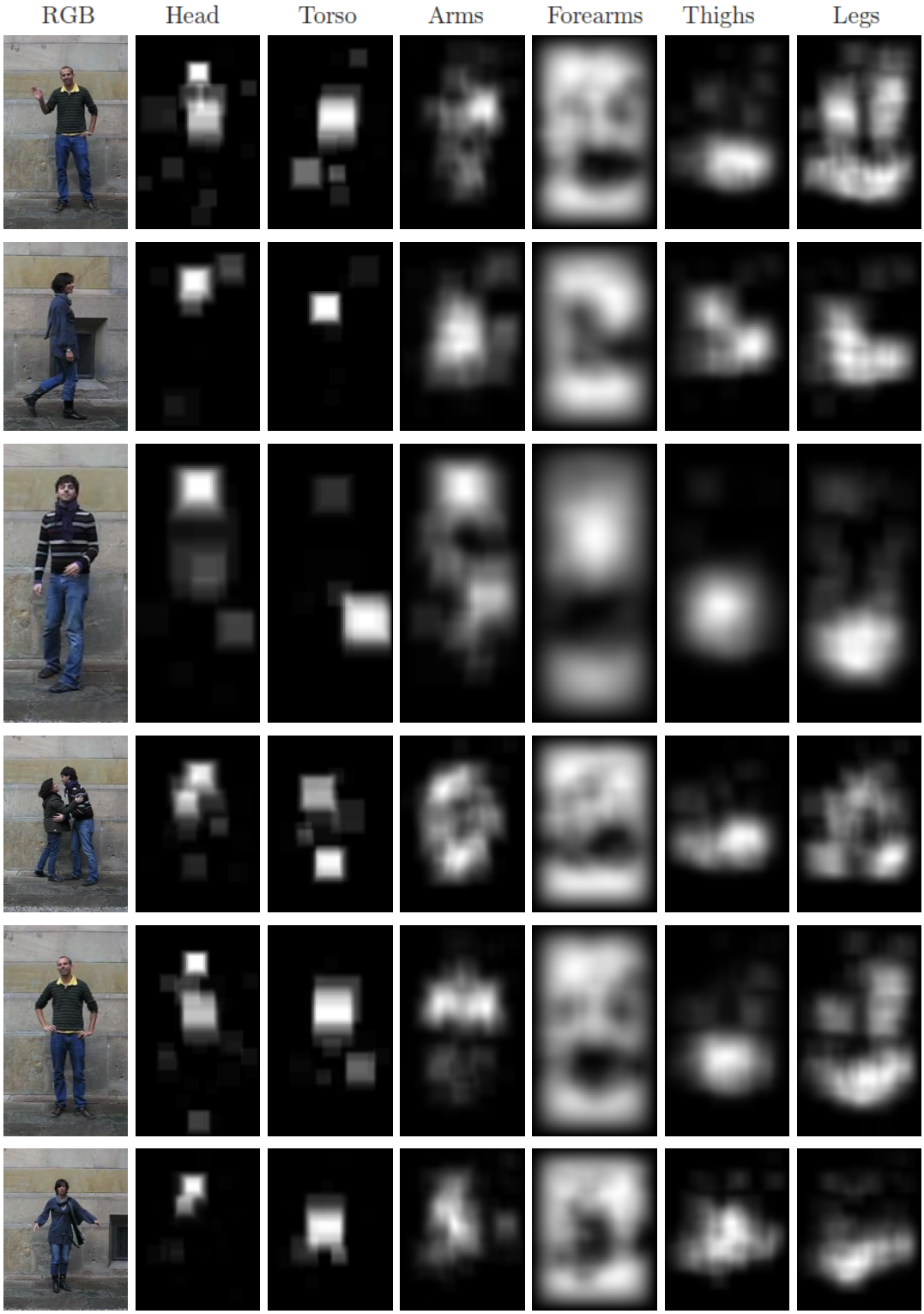


Figure 14: Body-like probability maps obtained by applying HOG descriptors and SVM classifiers. From left to right, the columns represent show the RGB image, head, torso, arms, forearms, thighs and legs.

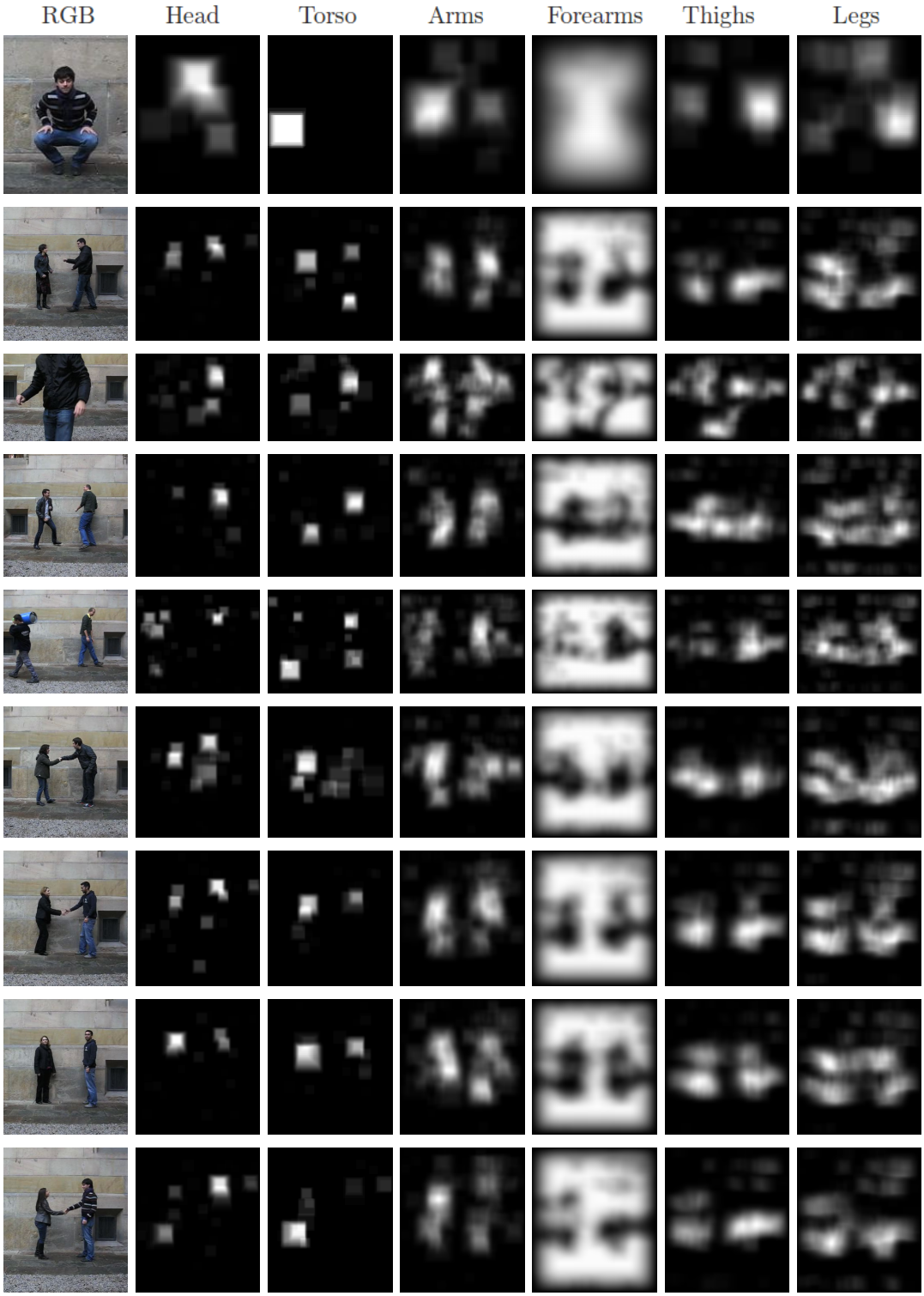


Figure 15: Body-like probability maps obtained by applying HOG descriptors and SVM classifiers. From left to right, the columns represent show the RGB image, head, torso, arms, forearms, thighs and legs.

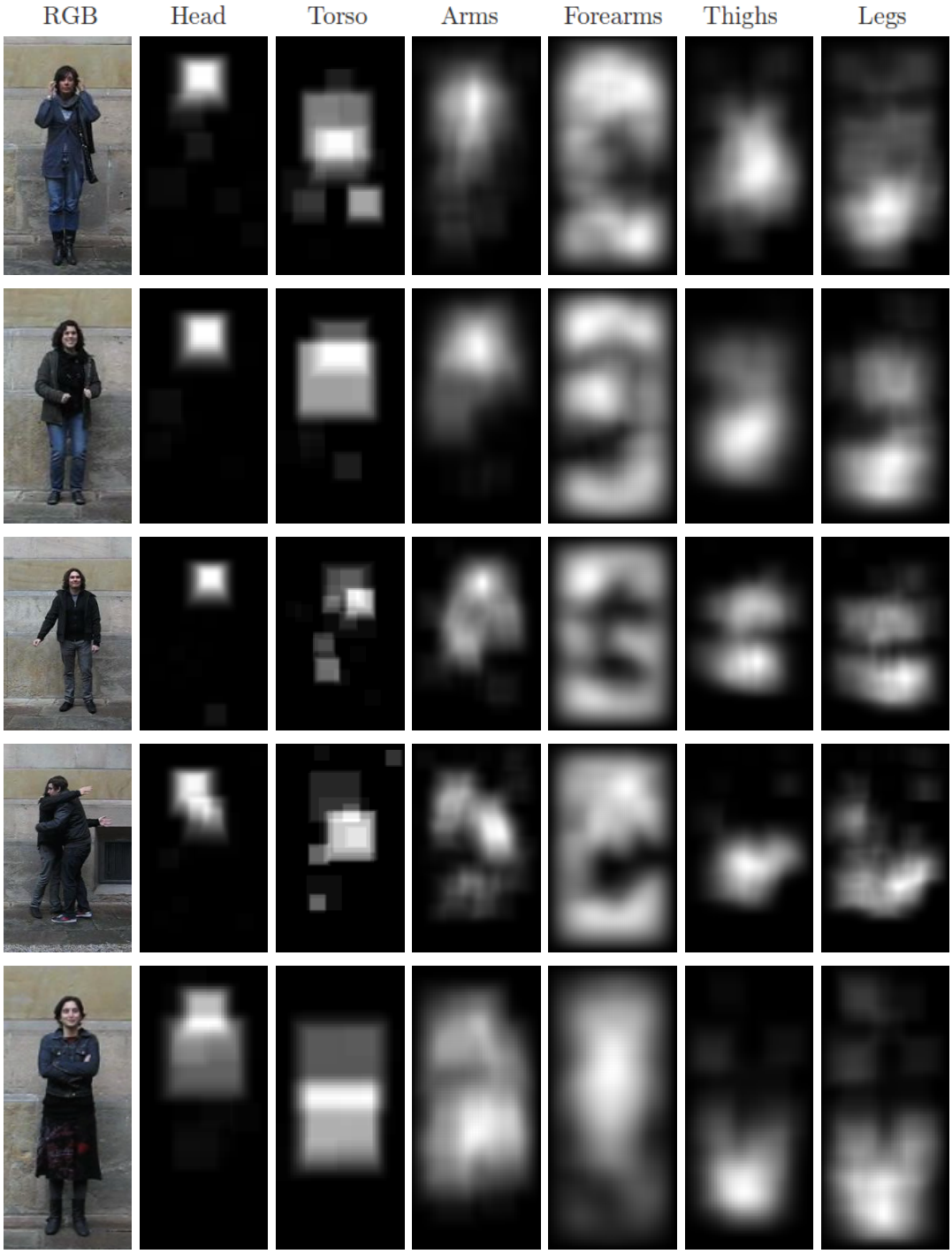


Figure 16: Body-like probability maps obtained by applying HOG descriptors and SVM classifiers. From left to right, the columns represent show the RGB image, head, torso, arms, forearms, thighs and legs.

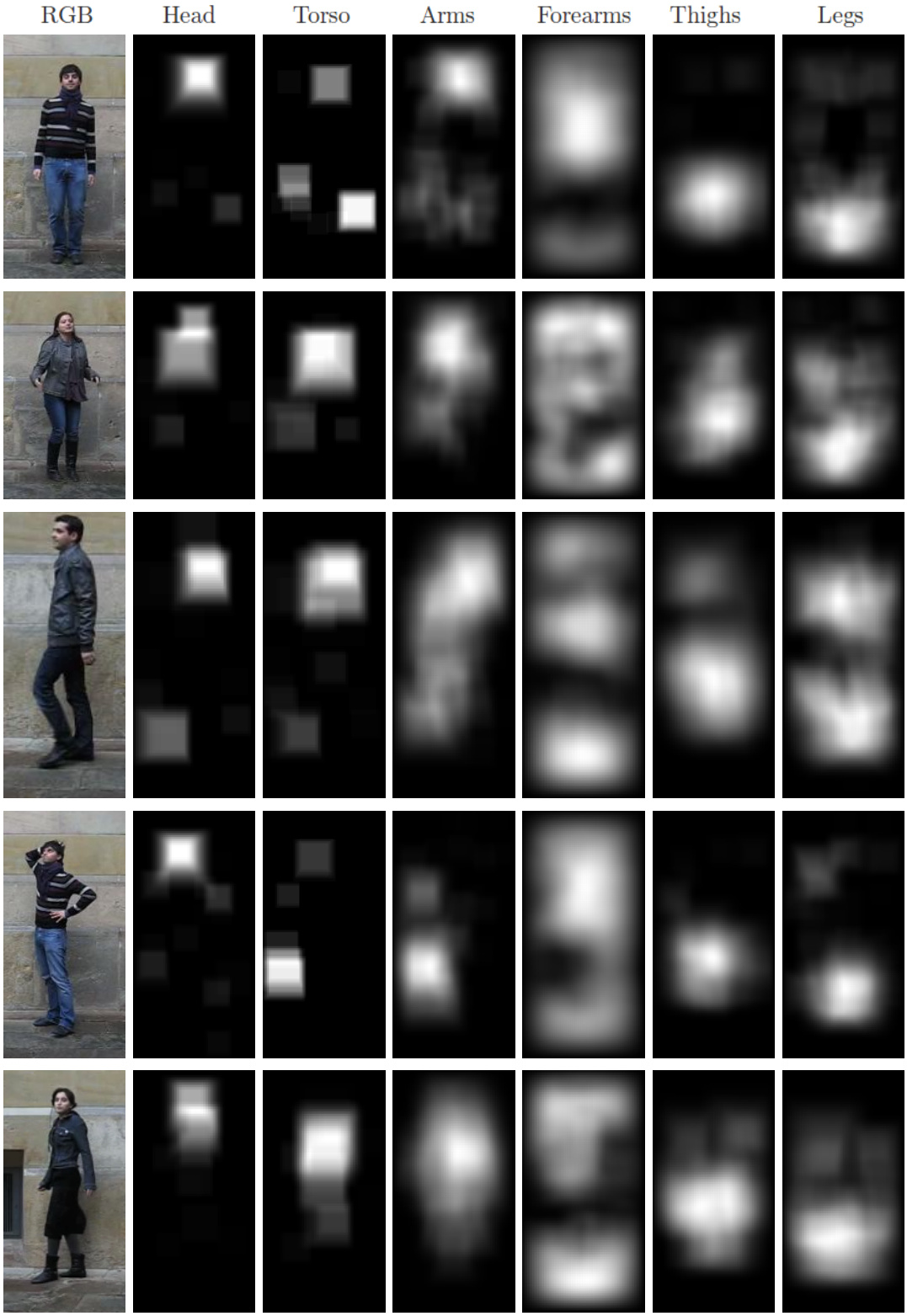


Figure 17: Body-like probability maps obtained by applying HOG descriptors and SVM classifiers. From left to right, the columns represent show the RGB image, head, torso, arms, forearms, thighs and legs.

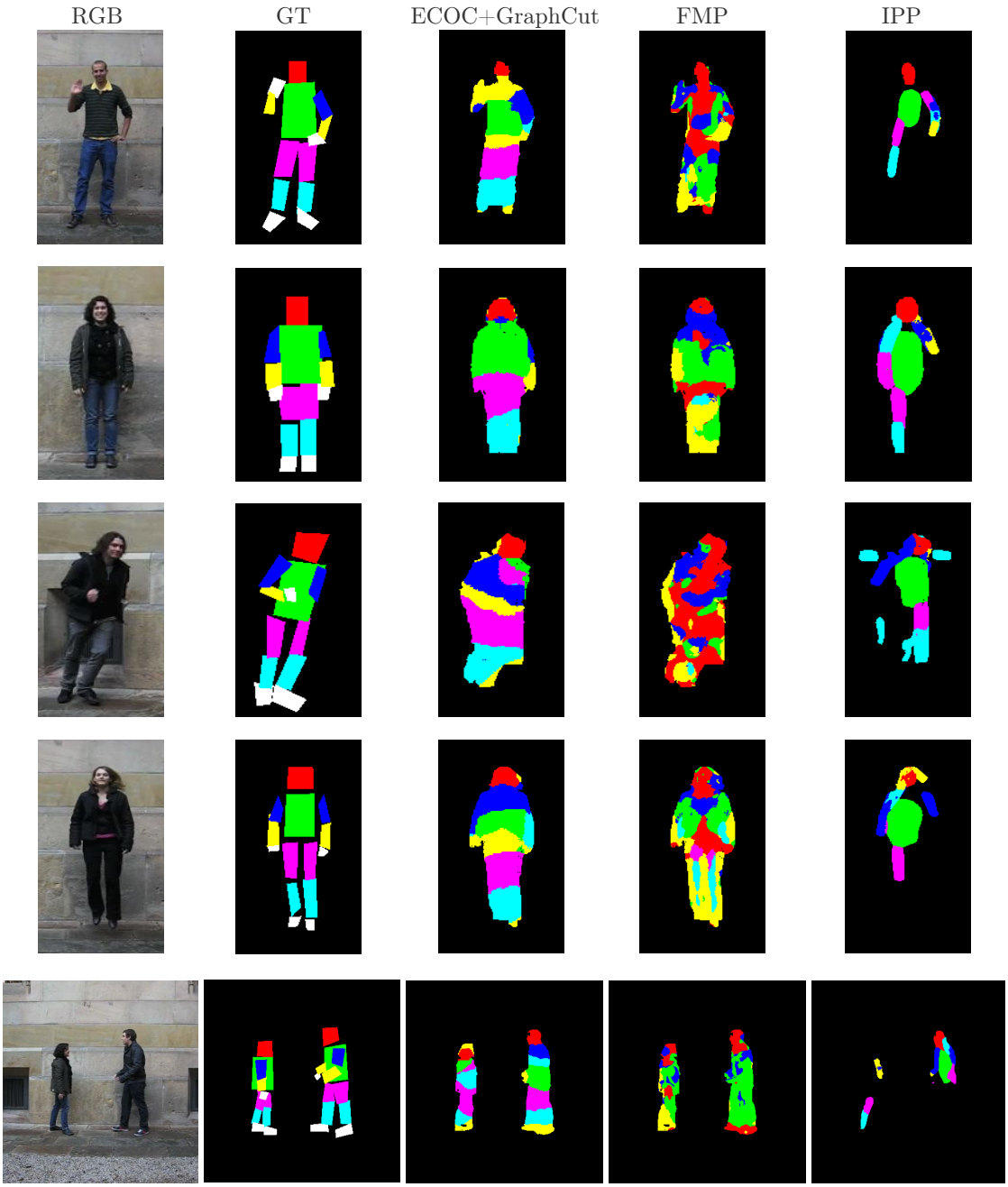


Figure 18: Multi-limb segmentation results for the three methods, for each sample, we also show the RGB image and the ground-truth (GT).

Figure 19: Multi-limb segmentation results for the three methods, for each sample, we also show the RGB image and the ground-truth (GT).

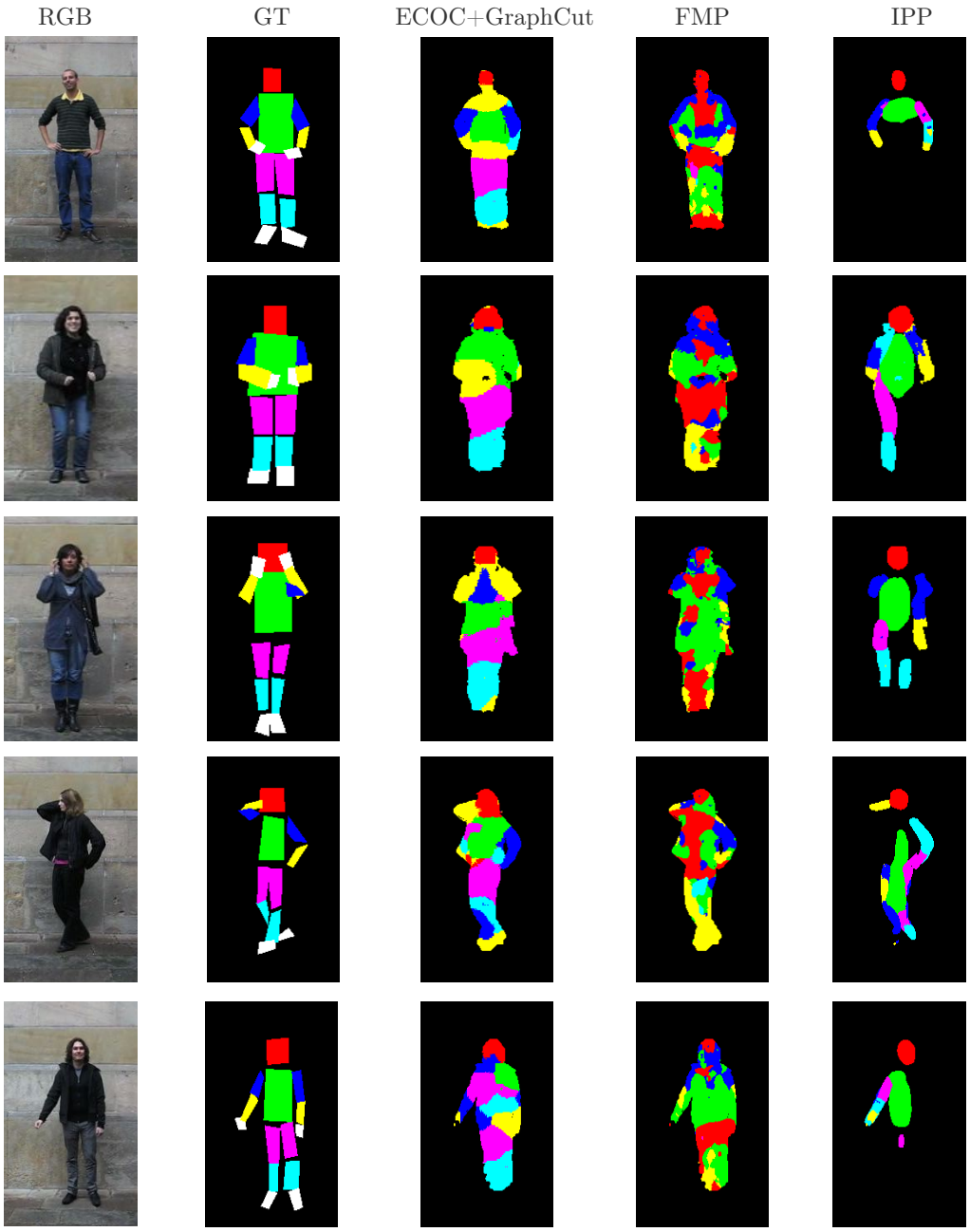


Figure 20: Multi-limb segmentation results for the three methods, for each sample, we also show the RGB image and the ground-truth (GT).

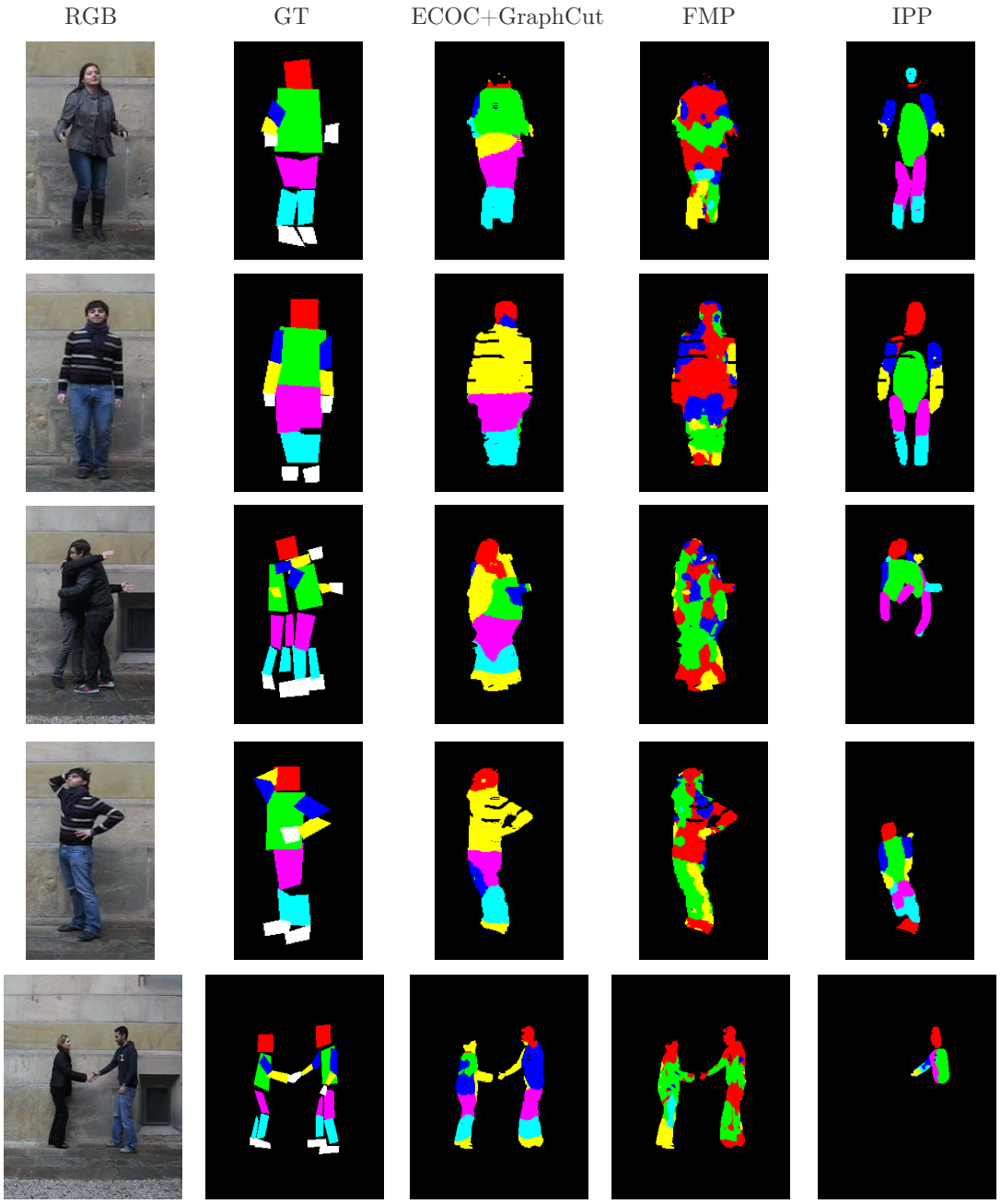


Figure 21: Multi-limb segmentation results for the three methods, for each sample, we also show the RGB image and the ground-truth (GT).

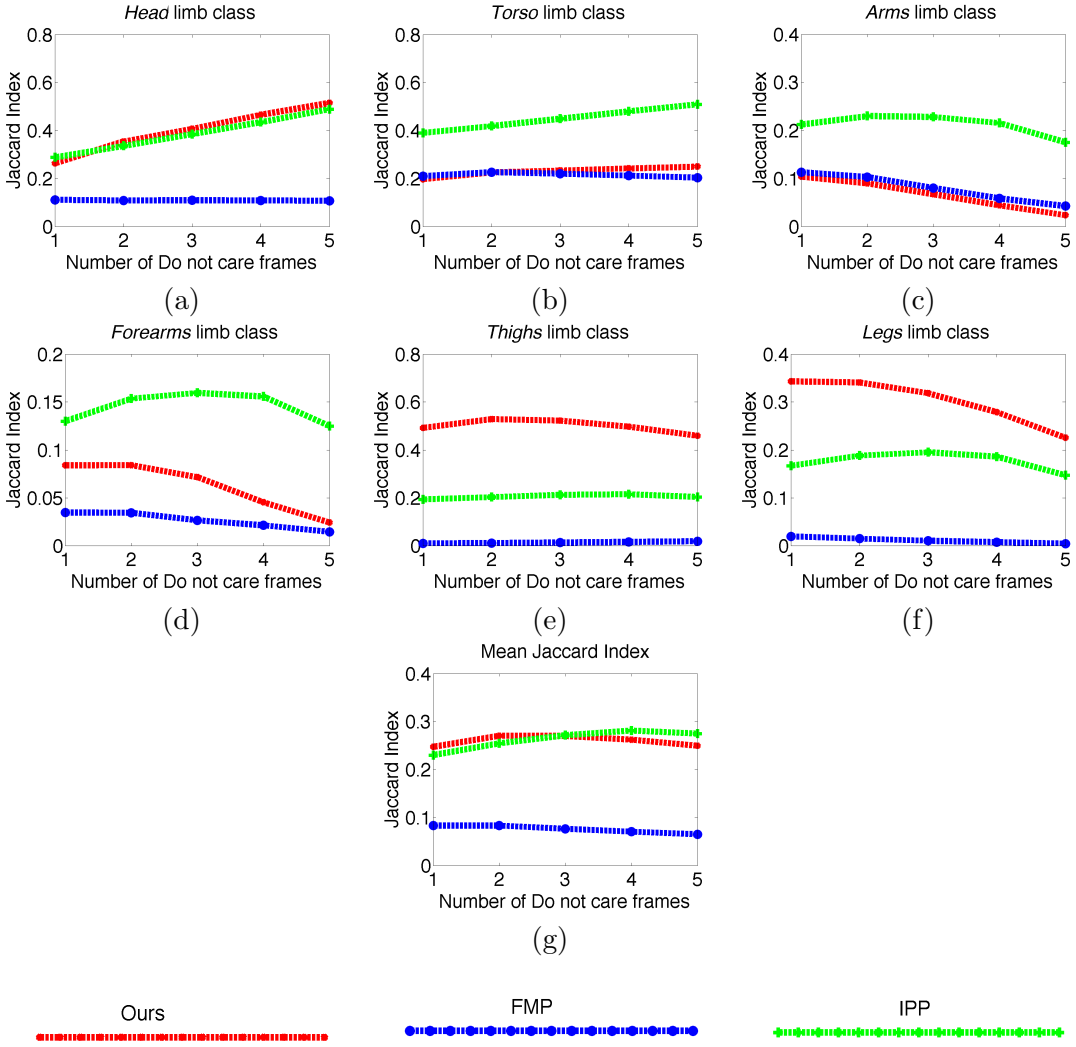


Figure 22: Jaccard Indexes for the different limb categories from (a) to (f). (g) Mean Jaccard Index among all limb categories.

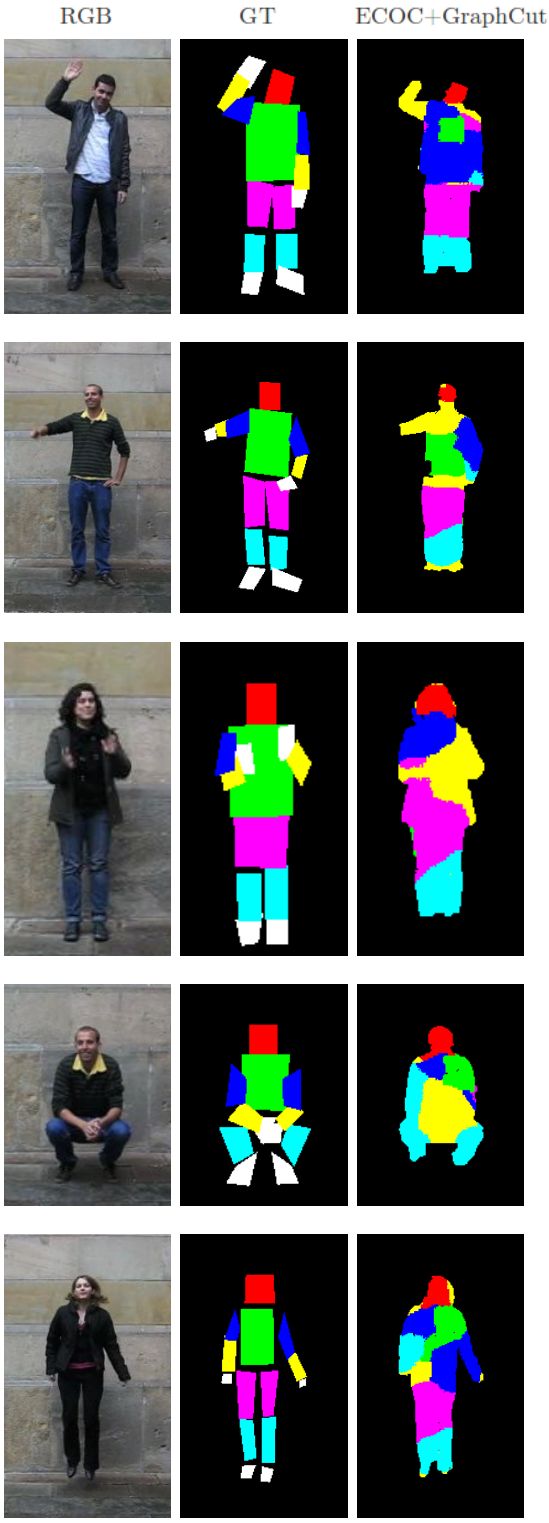


Figure 23: Gesture categories with our multi-limb segmentation approach, for each sample, we also show the RGB image and the ground-truth (GT). For each row, top down, we show the categories: wave, point, clap, crouch and jump.

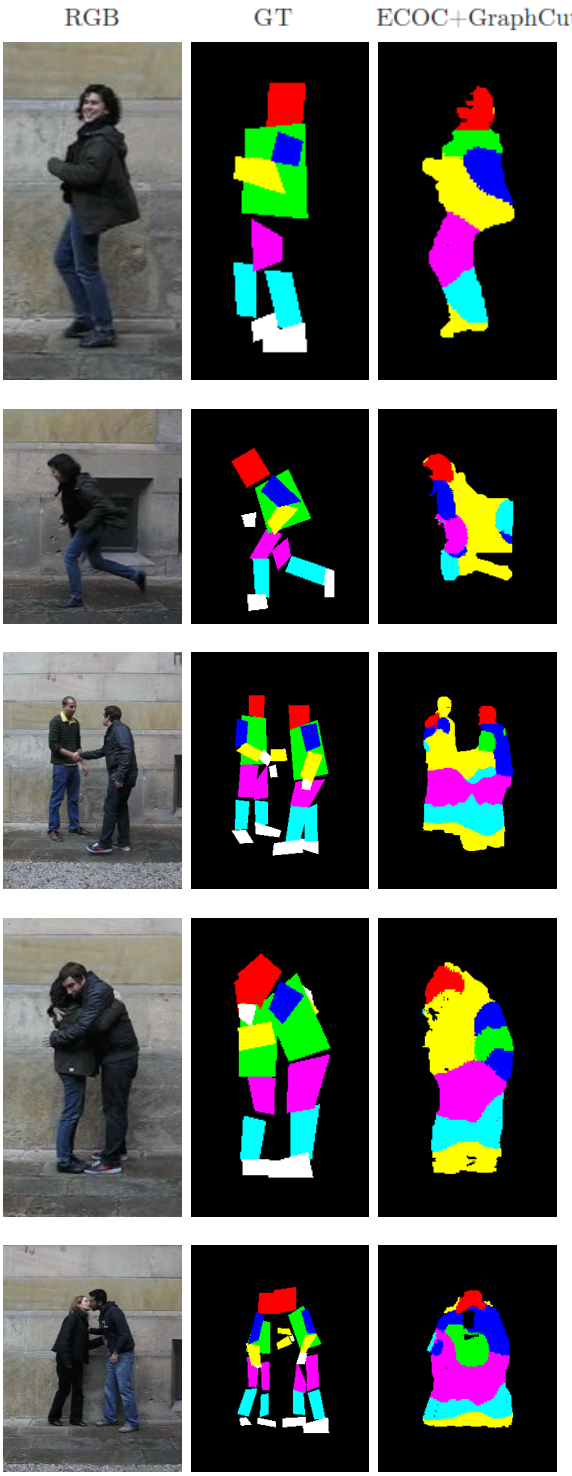


Figure 24: Gesture categories with our multi-limb segmentation approach, for each sample, we also show the RGB image and the ground-truth (GT). For each row, top down, we show the categories: walk, run, shake hands, hug and kiss.

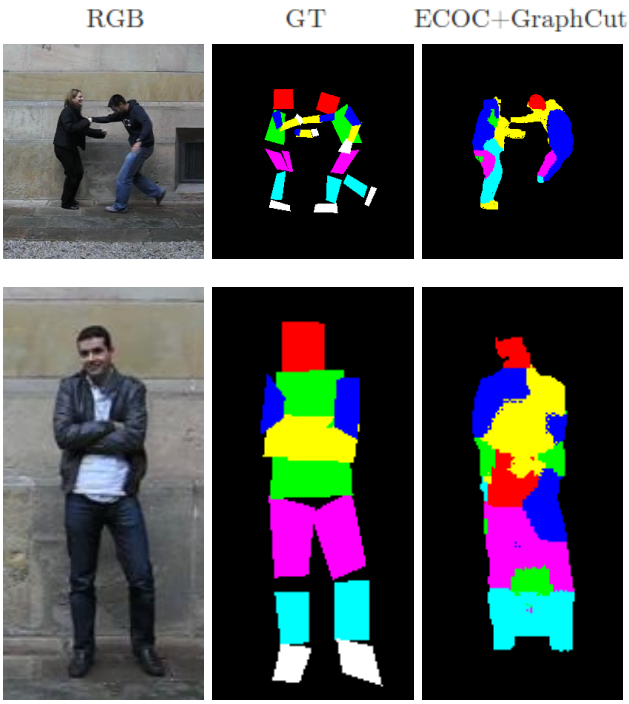


Figure 25: Gesture categories with our multi-limb segmentation approach, for each sample, we also show the RGB image and the ground-truth (GT). For each row, top down, we show the categories: fight and idle.

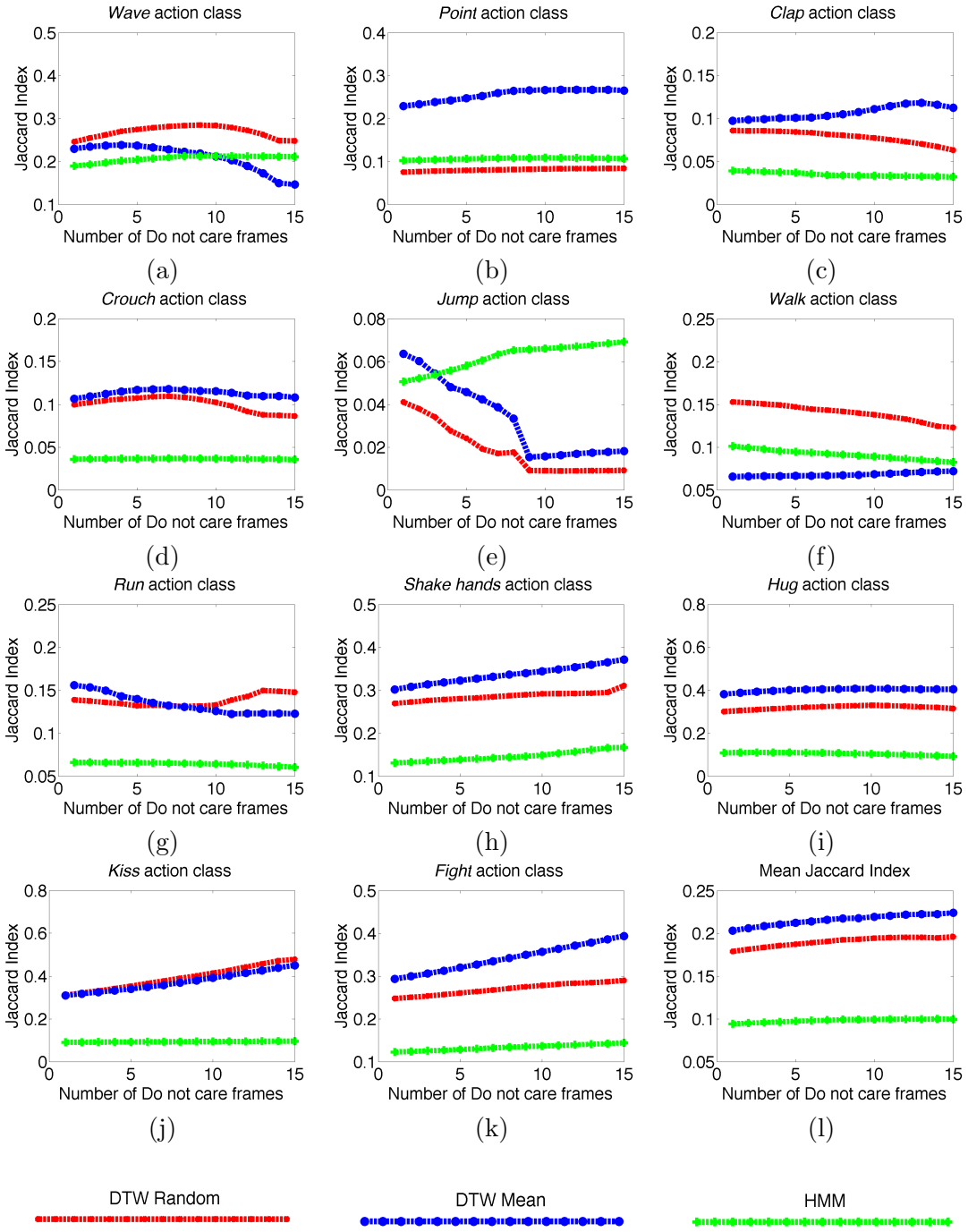


Figure 26: Jaccard Indexes for the different action categories from (a) to (k). (l) Mean Jaccard Index among all action categories.

5 Conclusions

In this work, we introduced the *HuPBA* $8K+$ dataset, which represents the largest available multi-limb dataset on RGB data up to date, with more than 120000 manually labeled limb regions. In addition, we proposed a novel two-stage method for human multi-limb segmentation in RGB images. In the first stage, we perform a person/background segmentation by training a set of body parts using cascades of classifiers embedded in an ECOC framework. In the second stage, to obtain a multi-limb segmentation we applied multi-label Graph Cuts to a set of limb-like probability maps obtained from a problem-dependent ECOC scheme.

We compared our proposal with state-of-the-art pose-recovery approaches on the novel dataset, obtaining very satisfying results in terms of both person/background and multi-limb segmentation steps. For completeness, the novel dataset was also labeled with different human actions drawn from an 11 gesture/action dictionary, including isolate and collaborative behaviors. In this sense, we also provided with action recognition baseline results on the novel dataset considering DTW and HMM strategies.

References

[1] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1014–1021. IEEE, 2009.

[2] Miguel Ángel Bautista, Sergio Escalera, Xavier Baró, and Oriol Pujol. On the design of an ecoc-compliant genetic algorithm. *Pattern Recognition*, 47(2):865–884, 2014.

[3] Miguel Ángel Bautista, Sergio Escalera, Xavier Baró, Petia Radeva, Jordi Vitriá, and Oriol Pujol. Minimal design of error-correcting output codes. *Pattern Recogn. Lett.*, 33(6):693–702, April 2012.

[4] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: Applications to image and text data. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 245–250, New York, NY, USA, 2001. ACM.

[5] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, pages 1365–1372. IEEE, 2009.

[6] Y. Boykov and V. Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *CVPR*, pages 26–33, 2003.

[7] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient nd image segmentation. *International Journal of Computer Vision*, 70(2):109–131, 2006.

[8] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, 2001.

[9] Yu-Ting Chen and Chu-Song Chen. Fast human detection using a novel boosted cascading structure with meta stages. *Image Processing, IEEE Transactions on*, 17(8):1452–1464, 2008.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886 –893 vol. 1, 2005.

[11] Miguel Angel Bautista Sergio Escalera Daniel Sanchez, Juan Carlos Ortega. Human body segmentation with multi-limb error-correcting output codes detection and graph cuts optimization. In *Proceedings of InPRIA*, pages 50–58, 2013.

[12] Fernando De la Torre, Jessica K. Hodgins, Javier Montano, and Sergio Valcarcel. Detailed human data acquisition of kitchen activities: the cmu-multimodal activity database (cmu-mmac). Technical report, RI-TR-08-22h, CMU, 2008.

[13] Markus Enzweiler and Dariu M Gavrila. Monocular pedestrian detection: Survey and experiments. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2179–2195, 2009.

[14] Sergio Escalera, Jordi Gonzalez, Xavier Baro, Miguel Reyes, Isabelle Guyon, Vassilis Athitsos, Hugo Jair Escalante, Leonid Sigal, Antonis Argyros, Cristian Sminchisescu, Richard Bowden, and Stan Sclaroff. Chalearn multi-modal gesture recognition 2013: grand challenge and workshop summary. *ChaLearn Multi-modal Gesture Recognition Grand Challenge and Workshop, 15th ACM International Conference on Multimodal Interaction*, pages 365–368, 2013.

[15] Sergio Escalera, Jordi Gonzàlez, Xavier Baró, Miguel Reyes, Oscar Lopes, Isabelle Guyon, Vassilis Athitsos, and Hugo J Escalante. Multi-modal gesture recognition challenge 2013: Dataset and results. *ChaLearn Multi-modal Gesture Recognition Grand Challenge and Workshop, 15th ACM International Conference on Multimodal Interaction*, pages 445–452, 2013.

[16] Sergio Escalera, Oriol Pujol, and Petia Radeva. On the decoding process in ternary error-correcting output codes. *PAMI*, 32:120–134, 2010.

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[18] Pedro F Felzenszwalb and Daniel P Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[19] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[20] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *EuroCOLT*, pages 23–37, 1995.

[21] A. Hernández-Vela, N. Zlateva, A. Marinov, M. Reyes, P. Radeva, D. Dimov, and S. Escalera. Graph cuts optimization for multi-limb human segmentation in depth maps. In *CVPR*, pages 726–732, 2012.

[22] Antonio Hernández-Vela, Miguel Ángel Bautista, Xavier Perez-Sala, Víctor Ponce-López, Sergio Escalera, Xavier Baró, Oriol Pujol, and Cecilio Angulo. Probability-based dynamic time warping and bag-of-visual-and-depth-words for human gesture recognition in rgb-d. *Pattern Recognition Letters*, 2013.

[23] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12.

[24] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[25] Krystian Mikolajczyk, Cordelia Schmid, and Andrew Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *Computer Vision-ECCV 2004*, pages 69–82. Springer, 2004.

[26] D. Ramanan, D.A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *PAMI*, 29(1):65 –81, jan. 2007.

[27] Deva Ramanan. Learning to parse images of articulated bodies. In *Advances in neural information processing systems*, pages 1129–1136, 2006.

[28] Miguel Reyes, Gabriel Dominguez, and Sergio Escalera. Featureweighting in dynamic timewarping for gesture recognition in depth data. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1182–1188. IEEE, 2011.

[29] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, August 2004.

[30] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. (1):43–49, 1978.

[31] Benjamin Sapp, Chris Jordan, and Ben Taskar. Adaptive pose priors for pictorial structures. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 422–429. IEEE, 2010.

[32] Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 32–36. IEEE, 2004.

[33] Thad Starner and Alex Pentland. Real-time american sign language recognition from video using hidden markov models. In *Motion-Based Recognition*, pages 227–243. Springer, 1997.

[34] Duan Tran and David Forsyth. Improved human parsing with a full relational model. In *Computer Vision–ECCV 2010*, pages 227–240. Springer, 2010.

[35] V. Vineet, J. Warrell, L. Ladicky, and P. Torr. Human instance segmentation from video using detector-based conditional random fields. In *BMVC*, 2011.

[36] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, volume 1, 2001.

[37] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1794–1801. IEEE, 2009.

[38] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1385–1392. IEEE, 2011.

[39] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures-of-parts. 2012.

[40] Feng Zhou, Fernando De la Torre, and J Hodgins. Hierarchical aligned cluster analysis for temporal clustering of human motion. 2013.

[41] Qiang Zhu, Mei-Chen Yeh, Kwang-Ting Cheng, and Shai Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1491–1498. IEEE, 2006.