Human Body Segmentation with Multi-limb Error-Correcting Output Codes Detection and Graph Cuts Optimization

Daniel Sánchez[1], Juan Carlos Ortega[1], Miguel Ángel Bautista[1,2], and Sergio Escalera[1,2]

[1] Dept. Matemàtica Aplicada i Anàlisi, UB, Gran Via 585, 08007, Barcelona, Spain
[2] Centre de Visió per Computador, Campus UAB, Edifici O, 08193, Barcelona, Spain
gammarl@gmail.es, juancarloslinux@gmail.com, mbautista@ub.edu, sergio@maia.ub.es

Abstract. Human body segmentation is a hard task because of the high variability in appearance produced by changes in the point of view, lighting conditions, and number of articulations of the human body. In this paper, we propose a two-stage approach for the segmentation of the human body. In a first step, a set of human limbs are described, normalized to be rotation invariant, and trained using cascade of 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 GMM color modelling and binary Graph Cuts optimization procedure. The proposed methodology is tested in a novel limb-labelled data set. Results show performance improvements of the novel approach in comparison to classical cascade of classifiers and human detector-based Graph Cuts segmentation approaches.

Keywords: Human Body Segmentation, Error-Correcting Output Codes, Cascade of Classifiers, Graph Cuts.

1 Introduction

Human body segmentation 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 the human/s body appearing in an image, discriminating the human body from the rest of the image. Usually, human body segmentation is treated in a two-stage fashion. First, a human body part detection is performed, and then, these detections are used as a prior knowledge to be optimized by segmentation strategies.

In the first stage, Bourdev et. al. [1] used body part detections in an AND-OR graph to obtain the pose estimation. Ramanan et. al [2] detect body parts and use a tree pictorial structure to infer the final pose of the human. The work of

J.M. Sanches, L. Micó, and J.S. Cardoso (Eds.): IbPRIA 2013, LNCS 7887, pp. 50–58, 2013.

Dalal et. al in [3] detects the human body using a cascade of classifiers architecture with SVM and HOG features. This is one of the main approaches used to initialize posterior pose estimation and segmentation approaches. In the second stage, once the human body pose is obtained, many methods can be applied in order to obtain a human/background segmentation. Vinet et. al proposed to use Conditional Random Fields based on body part detectors to obtain a complete person/background segmentation. In addition, there is a current tendency to use Graph Cuts optimization to obtain either a multi-limb or a complete human body segmentation [4].

In this paper, we present a novel two-stage approach for the segmentation of the human body on RGB data. We propose to use cascade of classifiers as body part detectors in a tree structure combining their outputs in an Error-Correcting Output Codes framework. Once the body-pose estimation is obtained, it is used as an initialization of a GMM color modelling and posterior binary Graph Cut segmentation optimization. Furthermore, we provide a novel dataset consisting of 90 images in which 14 limbs were manually tagged. As a result of our two-stage segmentation methodology, we show performance improvements in comparison to classical cascade of classifiers and human detector-based Graph Cuts segmentation approaches.

The rest of the paper is organized as follows: Section 2 introduces the proposed method. Section 3 presents the novel dataset and the experimental results. Finally, Section 4 concludes the paper.

2 Methodology

In the following subsections we describe the novel two-stage human segmentation approach. For this task, we split the procedure in a) Body part learning using cascade of classifiers, b) Tree structure body part learning, c) ECOC multi-limb detection, and d) GrabCut optimization for foreground extraction.

2.1 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 of body part detectors in the literature follow a cascade of classifiers architecture. Cascade of classifiers is 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}. In particular, 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 [5] as the base classifier in the cascade architecture.

In addition, in order to make the body part detection rotation invariant, all the samples in the target class (number of samples of body parts) are normalized to the dominant region orientation. Once all positive and negative samples are normalized with respect to the dominant region orientation, Haar-like features are used to describe the body-parts.

Because of its properties, cascade of classifiers is usually trained to split one visual object from the rest of possible instances 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 (body parts in our case). However, if we define our problem as a multi-limb detection procedure, some body parts are similar in appearance, and thus, grouping them makes sense. 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 [6].

2.2 Tree Structure Body Part Learning

The first issue to combine a set of cascade of classifiers is how to define the groups of limbs to be learnt by each individual cascades. For this task, we propose to train a tree structure of 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 tree structure body part defined taking into account these issues for a set of 7 body limbs is show

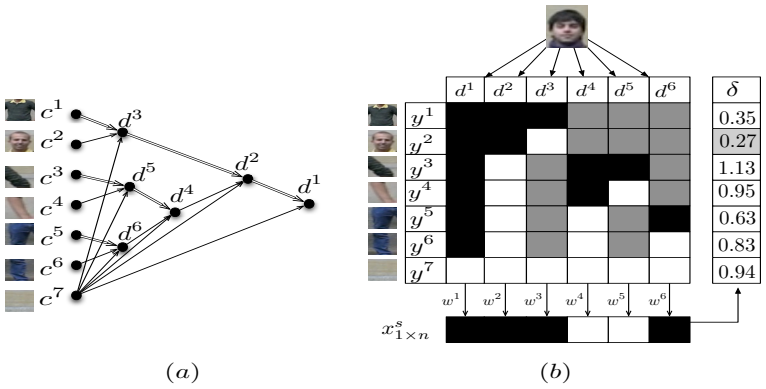

Fig. 1. (a) Tree structure classifier of body parts, where nodes represent the defined dichotomies. Notice that the single or double lines indicate the meta-class defined. (b) ECOC decoding step, in which a head sample is classified. The coding matrix codifies the tree structure of (a), where black and 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.

in Fig. 1(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 7 is codified as the background.

2.3 Error-Correcting Output Codes 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 [7]. 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 [6].

ECOC coding 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 [8]. 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. 1(b) shows the coding matrix codification of the tree structure in Fig. 1(a).

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 [6] the authors proposed a problem-dependent decoding function (distance function that takes into account classifier performances) obtaining very satisfying results. Following the core idea in [6], 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, matrix of weights M_w corresponds to the product of cascade accuracies at each stage. Thus, each column i of M_w is assigned a weight w^i as,

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

for a cascade of classifiers of k stages, where d_j^i 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. An example of the decoding step is shown in Fig. 1(b) for an input face. Notice that, even though classifier d^3 misses its prediction, the sample is not miss-classified due to the use of the weighted decoding taking into account classifier performances and the Error-Correcting principles. Finally, a body-like probability map $P^{bl} \in [0, 1]^{l \times w}$ is build. This map contains, at each position P_{ij}^{bl}, 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. 3(e) and (g), respectively.

2.4 GrabCut Optimization for Foreground Extraction

GrabCut [4] 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,

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

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 C around each pixel,

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

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 $P^{bl} \in [0,1]^{l \times w}$ defined in previous section. 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 Experimental Results

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

3.1 HuPBA-90 Data Set

Given the lack of a public available dataset to test for multi-limb human pose detection and segmentation evaluation, we present a fully limb-labelled dataset, called HuPBA-90 dataset. The HuPBA-90 dataset is formed by 90 RGB images in which different actors appear portraying a certain pose. Across all the dataset, the point of view, lightning conditions and background conditions remain invariant. The 90 images are obtained from 9 actors (equal proportion of males and females) and for each image 14 limbs were manually tagged: **Head, Torso, R-L Upper-arm, R-L Lower-arm, R-L Hand , R-L Upper-leg, R-L Lower-leg, R-L Foot**. Some samples of the HuPBA-90 dataset are shown in Fig. 2 [1]. Notice the pose variability and the extremely fitted limb labelling.

3.2 Methods and Settings

Methods: We compared 3 different methods for person/background segmentation. The first method is the well-known Person Detector of [3] followed by GrabCut segmentation. The second one is the cascade of classifiers proposed by Viola and Jones [9], training one cascade of classifiers per limb (aggregating R-L limbs in the same category) and GrabCut segmentation. Finally, the third method is the proposed ECOC tree structure body part classifier and automatic GrabCut segmentation.

[1] The dataset is available at
`http://www.maia.ub.es/~sergio/sergio%20escalera%20homepage_008.htm`

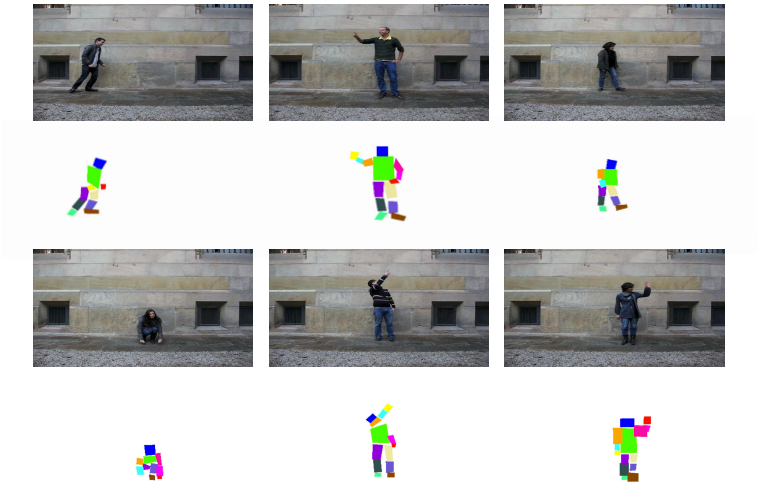

Fig. 2. HuPBA-90 dataset samples with 14 labelled limbs per person

Settings: For the Person Detector and cascade of classifiers we used the OpenCV implementation [2]. For the cascade of classifiers, 7 classifiers were trained (one per each body part), forcing a 0.99 false positive rate and maximum of 0.4 false alarm rate during 8 stages. Finally, the initialization values provided to the GrabCut algorithm were tuned via a 5-fold cross-validation for all methods. Final segmentation is computed using overlapping with the Jaccard Index ($J = \frac{A \bigcap B}{A \bigcup B}$) with the dataset limb masks grouping all them as person body ground truth, training all the methods in a ten-fold evaluation procedure and computing the confidence interval with a two-tailed t-test.

3.3 Validation

Qualitative results: Fig. 3 shows an example of the person/background segmentation obtained by the compared methodologies. In particular, we can see in Fig. 3(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. 3(e) and 3(f), the improvement in the body-like probability map obtained by the ECOC+GbCut approach over the cascade class.+GbCut method are clearly significant.

Quantitative results: In order to evaluate the performance of the compared methodologies, Fig. 4 shows a bar plot of the mean overlapping obtained on the 90 images of the dataset. 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

[2] OpenCV library: `http://opencv.willowgarage.com`

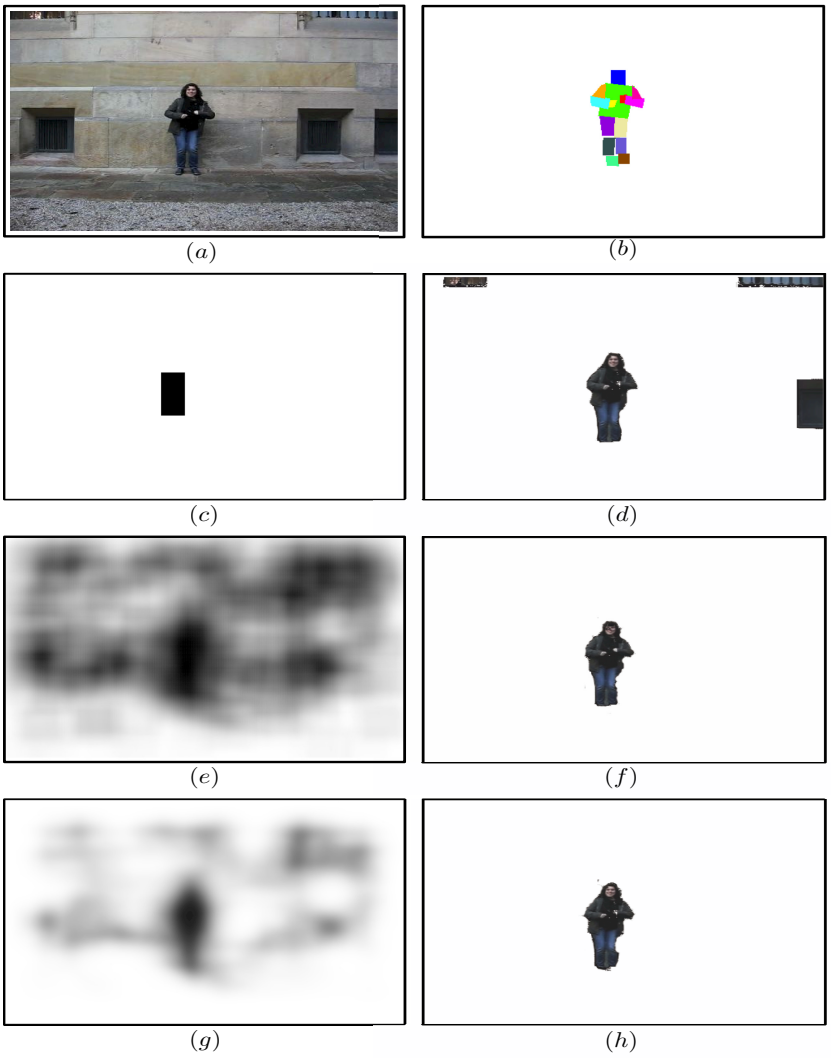

Fig. 3. (a) Original RGB image. (b) Multi-limb ground truth. (c) Probability map obtained by the Person Detector method. (d) Person/background segmentation of the Person Detector+GbCut approach. (e) Probability map yielded by the cascade class. method. (f) Person/background segmentation of the cascade class method. (g) Probability map obtained from the ECOC method. (h) RGB segmentation obtained by the ECOC+GbCut approach.

the ECOC framework. The latter, is the tree structure definition of the coding matrix, which allows base classifiers to obtain accurate results.

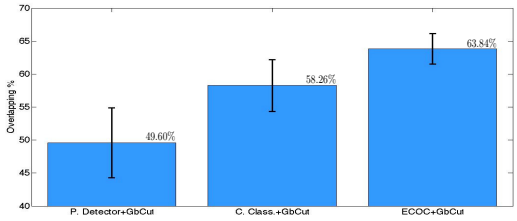

Fig. 4. Mean overlapping and standard deviation measures

4 Conclusion

We proposed a novel two-stage method for human segmentation in RGB images. In the first stage, a set of body parts are trained in a novel body part tree structure architecture using a cascade of classifiers, which is embedded into an ECOC framework. The method is applied in a new test image producing a body-like probability map. In a second stage, human body probability maps are used to automatically initialize a GrabCut segmentation procedure. The system was tested on a novel limb-labelled dataset, showing performance improvements in comparison to classical cascade of classifiers and human detector-based Graph Cuts segmentation procedures.

References

1. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3d human pose annotations. In: IEEE ICCV, pp. 1365–1372 (2009)
2. Ramanan, D., Forsyth, D., Zisserman, A.: Tracking people by learning their appearance. PAMI 29(1), 65–81 (2007)
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, vol. 1, pp. 886–893 (2005)
4. Hernández-Vela, A., Zlateva, N., Marinov, A., Reyes, M., Radeva, P., Dimov, D., Escalera, S.: Graph cuts optimization for multi-limb human segmentation in depth maps. In: CVPR, pp. 726–732 (2012)
5. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: Vitányi, P. (ed.) EuroCOLT 1995. LNCS, vol. 904, pp. 23–37. Springer, Heidelberg (1995)
6. Escalera, S., Pujol, O., Radeva, P.: On the decoding process in ternary error-correcting output codes. PAMI 32, 120–134 (2010)
7. Escalera, S., Tax, D., Pujol, O., Radeva, P., Duin, R.: Subclass problem-dependent design of error-correcting output codes. PAMI 30(6), 1–14 (2008)
8. Bautista, M.A., Escalera, S., Baró, X., Radeva, P., Vitriá, J., Pujol, O.: Minimal design of error-correcting output codes. Pattern Recogn. Lett. 33(6), 693–702 (2012)
9. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR, vol. 1 (2001)