A Global-Local approach to extracting deformable fashion items from Web images

Lixuan Yang[1,2], Helena Rodriguez[2], Michel Crucianu[1], and Marin Ferecatu[1]

[1] Conservatoire National des Arts et Metiers,
292 Rue Saint-Martin, 75003 Paris, France
firstname.lastname@cnam.fr
[2] Shopedia SAS,
55 rue La Boétie, 75008 Paris, France
firstname.lastname@shopedia.fr

Abstract. In this work we propose a new framework for extracting deformable clothing items from images by using a three stage global-local fitting procedure. First, a set of initial segmentation templates are generated from a handcrafted database. Then, each template initiates an object extraction process by a global alignment of the model, followed by a local search minimizing a measure of the misfit with respect to the potential boundaries in the neighborhood. Finally, the results provided by each template are aggregated, with a global fitting criterion, to obtain the final segmentation. The method is validated on the Fashionista database and on a new database of manually segmented images. Our method compares favorably with the Paper Doll clothing parsing and with the recent GrabCut on One Cut foreground extraction method. We quantitatively analyze each component, and show examples of both successful segmentation and difficult cases.

Keywords: Clothing extraction, Segmentation, Active Contour, GrabCut

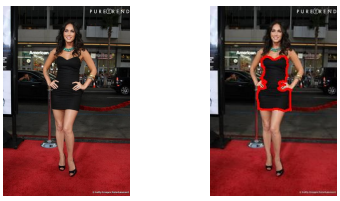
1 Introduction and related work

With the recent proliferation of fashion web-stores, an important goal for online advertising systems is to propose items that truly correspond to the expectations of the users in terms of design, manufacturing and suitability. We put forward here a method to extract, without user supervision, clothes and other fashion items from web images. Indeed, localizing, extracting and tracking fashion items during web browsing is an important step in addressing the needs of professionals of online advertising and fashion media: present the users with relevant items from a clothing database, based on the content of the web application they are consulting and its context of use. Users usually look for characteristics expressed by very subjective concepts, to describe a style, a brand or a specific design. For this reason, recent research focused in the development of detection, recognition and search of fashion items based on visual characteristics [11].

A popular approach is to model the target items based on attribute selection and high-level classification, for example [5] trains attribute classifiers on fine-grained clothing styles formulating the retrieval as a classification problem, [2] extracts low-level

features in a pose-adaptive manner and learns attribute classifiers by using conditional random fields (CRF), while [3] introduced a novel double-path deep domain adaptation network for attribute prediction by modeling the data jointly from unconstrained photos and the images issued from large-scale online shopping stores. A complementary approach is to use part-based models to compensate for the lack of pose estimation. The idea is to automatically align patches of human body parts by using different methods, for example sparse coding as in[16] or graph parsing technique as in [12].

Segmentation and aggregation to select cloth categories was employed either by using bottom-up cloth parsing from labels attached to pixels [19] or by over-segmentation and classification [8]. Deep learning was also used with success for clothing retrieval (deep similarity learning [13], Siamese networks [18]) or to predict fashionability [15].



(a) Original image (b) Desired output

Fig. 1. Our goal is to produce a precise segmentation (extraction) of the fashion items as in (b).

Unlike the above-mentioned methods, our proposal aims to *precisely* segment the object of interest from the background (foreground separation, see Fig. 1(b)), without user interaction and without using an extensive training database. Extracting such complex objects by simply optimizing a local pixel objective function is likely to fail without an awareness of the object's global properties. To take this into account, we propose a Global-Local approach based on the idea that a local search is likely to converge to a better fit if the initial state is coherent with the expected global appearance of the object.

Our method is validated on the Fashionista database [19] [3]and on a new database of manually segmented images that we specifically built to test fashion objects extraction and that we make available to the community. Our method compares favorably with the well-known Paper Doll [19] clothing parsing and with the recent GrabCut on One Cut [17] generic foreground extraction method. We provide examples of successful segmentation, analyze difficult cases and also quantitatively evaluate each component.

In Sec. 2 we describe our proposal, followed by a detailed presentation of each component. After the experimental validation in Sec. 3, we conclude the paper with Sec. 4 by a discussion of the main points and extension perspectives.

2 Our proposal

Detecting clothes in images is a difficult problem because the objects are deformable, have large intra-class diversity and may appear against complex backgrounds. To extract objects under these difficult conditions and without user intervention, methods solely relying on optimizing a local criterion (or pixel classification based on local features) are unlikely to perform well. Some knowledge about the global shape of the class of

[3] http://vision.is.tohoku.ac.jp/~kyamagu/research/paperdoll/

objects to be extracted is necessary to help a local analysis converge to a correct object boundary. In this paper we use this intuition to develop a framework that takes into account the local/global duality to select the most likely object segmentation.

We investigate here fashion items that are worn by a person. This covers practically most of the situations encountered by users of fashion and/or news web sites, while making possible the use of a person detector to restrict the search regions in the image and to serve as reference for alignment operations.

First, we prepare a set of images containing the object of interest and we manually segment them. These initial object masks (called templates in the following) provide the prior knowledge used by the algorithm. Of course, a given manual segmentation will not match exactly the object in an unknown image. We use each segmentation (after a suitable alignment) as a template to initiate an active contour (AC) procedure that will converge closer to the true boundaries of the real object in the current image. We then extract the object with a suitable GrabCut procedure to provide the final segmentation. Thus, at the end we have as many candidate segmentations as hand-made templates. In the final step we choose the best of them according to a criterion that optimizes the coherence of the proposed segmentation with the edges extracted from the image. In the following subsections we detail each of these stages (see also Fig. 2 for an illustration).

(a) (b) (c) (d) (e) (f) (g)

Fig. 2. Different stages of our approach: (a) original image, (b) a template segmentation, (c) output of the person detector, (d) result after the alignment step, (e) result after the active contour step, (f) the GrabCut band, (g) result after the GrabCut step.

To summarize, the main contributions of this paper are: we introduce a new framework for the extraction of fashion items in web images that combines local and global object characteristics, framework supported by a new active contour that optimizes the gap with respect to the global segmentation model, and by a new measure of fit of the proposed segmentation to the real distribution of the contours. Also, we prepare a new benchmark database and make it available to the community.

2.1 Person detector

For clothing extraction, it is reasonable to first apply a person detector. As in many other studies (*e.g.* [12], [8], [20]), we use the person detector with articulated pose estimation algorithm from [21] that was extensively tested and proved to be very robust in several other fashion-related works (see Sec. 1). It is based on a deformable model that sees the object as a combination of parts [21]. The detection score is defined as the fit to parts minus the parts deformation cost. The mixture model not only encodes object structure but also captures spatial relations between part locations and co-occurrence relations between parts. The output of the detector is a set of parts (rectangular boxes) centered

on the body joints and oriented correctly. The boxes are used as reference points for alignment by translation and re-scaling in several stages of our proposal (see below).

To train the person detector, we manually annotate a set of 800 images. Each person is annotated with 14 joint points by marking the articulations and main body parts. When the legs are covered by long dresses, the lower parts are placed on the edges of the dress rather than on the legs. This not only improves detection accuracy, but also hints to the location of the contours. Fig. 2(c) shows the output of the person detector on an unannotated image. Boxes usually slightly cover the limbs and body joints.

2.2 Template selection

As we have seen, each initial template can provide a candidate segmentation for a new, unknown image. However, this is redundant and may slow down unnecessarily the procedure. Since we focus on the fashion items that are worn by a person, the number of different poses in which an object may be found is relatively small, and many initial templates are thus quite similar. Intuitively, templates that are alike in shape should also produce similar segmentation masks. To reduce their number, the initial templates are clustered into similar-shape clusters by using the K-Medoid procedure [9]. We employ 8 clusters for each object class, which is a reasonable choice in our case because the number of person poses is not very large. Each resulting cluster is a configuration of deformable objects that share a similarity in pose, viewpoint and clothing shape. The dissimilarity of two object masks is defined by the complement of the Jaccard index:

$$\mathrm{d}(S_1, S_2) = 1 - \frac{\mathrm{Surface}(S_1 \cap S_2)}{\mathrm{Surface}(S_1 \cup S_2)}$$

where S_1 and S_2 are the binary masks of two objects.

Each cluster represents a segmentation configuration and its prototype is used in the next stages of the procedure. However, we do not simply choose the medoid as the prototype of the cluster, but rather the element in the cluster that is visually closest to the corresponding box parts produced by the person detector on the unknown image. To do so, we apply the object detector on both the unknown image and the template image and we compare the boxes that contain the object in the template with the corresponding ones in the unknown image by using the Euclidean distance. To represent the content of the boxes we first considered HOG features [4] (to favor similar shape content) but finally settled for Caffe features [7] that provide better results. This suggests that mid-level features give better clues to identifying the correct pose of an object compared to local pure shape features. Shape is relevant for comparing the boundaries of two objects but less so when comparing what is inside those boundaries.

Specifically, we use the AlexNet model in [10] within the Caffe framework [7]. The network was pre-trained on 1.2 million high-resolution images from ImageNet, classified into 1000 classes. To fine-tune the network to our image domain, we replace the last layer by a layer of ten outputs (the number of classes considered here) and then retrain the network on our training database with back-propagation to fine-tune the weights of all the layers. After the fine-tuning, the feature we employ is the vector of responses for layer fc7 (second to last layer) obtained by forward propagation.

To illustrate this step we show in Fig. 3 the medoids (centers) of the 8 clusters obtained for three classes of our benchmark database. We notice the diversity in poses,

scale and topology. For example, some coats are segmented into several disjoint parts, some have openings and some jeans are covered by a vest.

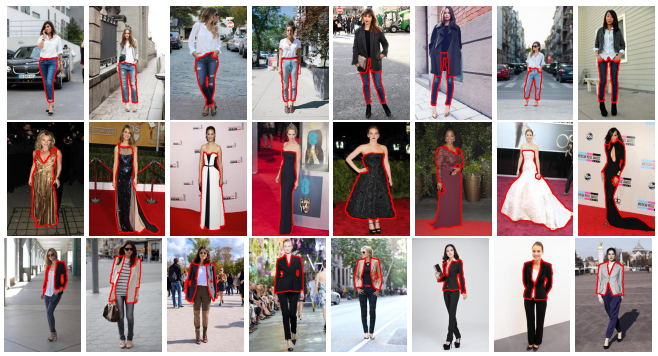


Fig. 3. Medoids of the 8 clusters of template segmentations for three classes: jeans (top), long dress (middle) and coat (bottom).

2.3 Template Alignment

The output of the previous stage is a set of segmentation templates (8 in our case) for each object class. They will be used one by one to initiate an active contour process. But they first need to be aligned into the unknown image at the right site and with the correct angle and scale. We propose an SVM alignment technique based on the observation that the person detector places the boxes centered on the body joints. Thus, the line joining adjacent boxes represents the body limbs. Since the clothing's spatial distribution highly depends on the pose of human body, and thus on limb placement, we use the vector of distances from a pixel to the limbs as a feature vector to learn a pixel-level SVM classifier that predicts if a pixel belongs to the object. Learning is performed on the template image and prediction on the unknown image. Pixels predicted as positives form the mask whose envelope serves as initialization for the active contour step. The SVM uses a Gaussian kernel with a scale parameter $\sigma = 1$ found through experiments.

2.4 Active Contour

Once the template is embedded in the image, we use it to initialize an active contour (AC) that should converge to the boundaries of the object. The result is highly dependent on the initial contour, but usually one of the 8 segmentation templates leads to a final contour that is quite close to the true boundary. The AC is initialized with the aligned segmentation contour produced by the previous step and has as input the gray-level image. We use the AC introduced in [1] because it can segment objects whose boundaries are not necessarily well-supported by gradient information. The AC minimizes an energy defined by contour length, area inside the contour and a fitting term:

$$F(c_1, c_2, C) = \mu \cdot \text{Length}(C) + \nu \cdot \text{Area}(\text{in}(C)) + \lambda_1 \int_{in(C)} |u(x,y) - c_1|^2 + \quad (1)$$
$$+ \lambda_2 \int_{out(C)} |u(x,y) - c_2|^2$$

where C is the current contour, c_1 and c_2 are the average pixel gray-level values $u(x, y)$ inside and respectively outside the contour C. The curvature term is controlled by μ and the fitting terms by λ_1 and λ_2. The averages c_1 and c_2 are usually computed on the entire image. Because of the large variability of the background in real images, these values can be meaningless locally. Consequently, in our case we replace them by averages computed in a local window of size 40×40 pixels around each contour pixel.

To reinforce the influence of the global shape of the template on the position of the AC, we include a new term in the energy function (Eq. 1) that moderates the tendency to converge too far away from the template:

$$F_t(C) = \eta \int_{on(C)} D_m(x, y) \tag{2}$$

where $D_m(x, y)$ is the distance between pixel (x, y) and the template. By including this term, the contour will converge to those image regions that separate best the inside from the outside and, at the same time, are not too far away from the template contour.

2.5 Segmentation

The contours obtained in the previous step suffer from two implicit problems: (1) only the grey-level information is used by the AC process, and (2) possible alignment errors may affect the result. To compensate for these problems, an "exclusion band" of constant thickness is defined around the contour produced by the previous step, then the inside region is labeled as "certain foreground" and the outside area as "certain background". A GrabCut algorithm [14] is then initialized by these labels to obtain the final result. GrabCut takes into account the global information of color in the image and will correct the alignment errors within the limits of the defined band.

2.6 Object Selection

After obtaining the segmentation proposals initiated from each template, we need to select a single segmentation as the final result. For this, we propose a score based on a global measure of fit to the image:

$$F(C) = \frac{\int_{on(C)} D_e(x, y) ds}{\int_{on(C)} ds} \tag{3}$$

where $D_e(x, y)$ is the distance from the current pixel to the closest edge detected by [6] and C is the boundary of the segmentation proposal. This score measures the average distance from the segmentation boundary to the closest edges in the image. A small value indicates a good fit to the image. See Table 1 for an illustration of this step.

3 Experimental results

To assess the performance of the proposed method, we perform two sets of experiments. In the first set, our method is compared to a recent improvement of GrabCut [14] that is the standard approach in generic object extraction, on a novel fashion item benchmark we built. The second set of experiments compares our proposal to the recent PaperDoll [19] fashion item annotation method on the Fashionista database [20].

Edges	2.5133	2.3492	**1.5187**	2.8724	2.8082	2.8112	1.9229	2.7667

Table 1. Segmentation selection from the results based on the 8 templates of the class, using the corresponding fit values. The test image is given top left, with the extracted edges shown bottom left. The best score is the smallest (outlined in boldface).

Fig. 4. Qualitative evaluation: are original images and associated segmentation results.

3.1 RichPicture Database

Since, to our knowledge, at this time there is no public benckmark specifically designed for clothing extraction from fashion images, we introduce a novel dataset called RichPicture, consisting of 1000 images from Google.com and Bing.com. It has 100 images for each of the following fashion items: Boots, Coat, Jeans, Shirt, T-Shirt, Short Dress, Mid Dress, Long Dress, Vest and Sweater. Each target object in each class is manually segmented. To train the person detector (see Sec. 2.1), images are also annotated by 14 key points. This database will be made available with the paper and open to external contributions. We shall further extend it with new classes and more images per class.

3.2 Comparison with GrabCut in One Cut

In this set of experiments, we compare our proposal to GrabCut in one cut [17], a recent improvement on the well-known GrabCut [14] foreground extraction algorithm,

which is frequently used as a baseline method in the literature. GrabCut in One Cut was shown in [17] to have higher effectiveness, is less resource demanding and has an open implementation. These reasons makes it a good candidate as a benchmark baseline. For the purpose of this evaluation, we split each class of our database in 80 images for training (template selection) and 20 images for test.

The segmentation produced by the algorithms is tested against the ground truth obtained by manual segmentation. As performance measure we employ the Jaccard index, traditionally used for segmentation evaluation, and averaged over all the testing images of a class. To outline the object for One Cut we use the external envelope of the relevant parts (the ones that contain parts of the object) identified by the person detector. Table 2 shows a class by class synthesis of the results (best results are in boldface).

Table 2. Comparison with the One Cut algorithm. The comparison measure is the Jaccard index.

Class	Boots	Coat	Mid dress	Jeans	Shirt	T-shirt	Short dress	Long dress	Vest	Pull
Our method	**0,54**	**0,74**	**0,84**	**0,78**	**0,77**	**0,67**	**0,80**	**0,74**	**0,65**	**0,74**
One Cut	0,26	0,31	0,54	0,71	**0,77**	0,45	0,47	0,57	0,35	0,36

It can be seen that the proposed method performs significantly better on all the classes except "Shirt" where the scores are equal. While both segmentation methods are automatic (do not require interaction), these results speak in favor of including specific knowledge into the algorithm (by the use of segmentation templates in our case).

3.3 Comparison with Paper Doll

To our knowledge, there is no published method concerning fashion retrieval that aims to precisely extract entire fashion items from arbitrary images. The closest we could find is the Paper Doll framework, cited above, that in fact attributes label scores to a set of blobs in the image. By taking the union of all the blobs that correspond to a same clothing class, one can extract objects of that class. The authors of Paper Doll also introduced the Fashionista database, used to test annotation algorithms, which we use for this evaluation. Table 3 presents the synthesis of the results of Paper Doll *vs.* One Cut *vs.* our method. The object classes we selected for tests are those that correspond to fashion items that are worn by persons (compatible with our method).

For our method, training and template selection are performed on the same part of the database that Paper Doll employed for training. As seen from Table 3, on most object classes we compare favorably to Paper Doll. For objects like "Boots", our method needs a more dedicated alignment process, since the object is very small compared to the frame given by the person detector that serves as alignment reference. For objects of the "Jeans" class, the problem also comes from the alignment stage, because the boxes proposed by the person detector are not very well positioned when the legs are crossed. It is necessary to increase the number of training examples with this specific pose.

3.4 Qualitative evaluation

We illustrate here the results of the proposed method with some examples taken from the our test database. First, Table 1 shows the final segmentation selection stage, based

Table 3. Comparison (using Jaccard index) with Paper Doll and One Cut on Fashionista.

Class	Vest	Jeans	Shirt	Boots	Coat	Dress	Skirt	Sweater
Our method	**0,32**	0,72	**0,35**	0,35	**0,56**	**0,52**	**0,62**	**0,52**
Paper Doll	0,19	**0,74**	0,24	**0,44**	0,28	**0,52**	0,52	0,07
One Cut	0,23	0,62	0,29	0,01	0,23	0,33	0,32	0,25

Fig. 5. Comparison with OneCut: original image (left), our method (middle) and OneCut (right)

on the values of fit associated to the results obtained from each of the 8 templates of the class. Visually, the object segmentation in the test image is close to the template.

A first example of successful segmentation was shown in Fig. 2(g). In Fig. 4 we present other difficult but successful segmentations: (a, c) for small object extraction, (e, g, i) for clothes against confusing or cluttered background, and (k, m, o) for deformed clothes. Fig. 4 also shows examples where the segmentation is not perfect: in (r) the extracted object includes some hair and in (t) also part of the leggings. These inclusions probably occur here because the energy term we introduced in the active contour encourages the contour to stay close to the global shape of the segmentation template.

A visual comparison with One Cut is shown in Fig. 5. As hinted by the quantitative results, One Cut includes larger parts of external objects, mainly due to the lack of prior shape information. This occurs on most of the images in the database, explaining the significantly lower performance of One Cut in Table 2 and in Table 3.

4 Conclusion

We proposed a novel framework for extracting deformable clothing objects from web images. Our proposal combines a three stage global-local approach that injects specific knowledge about the object by using segmentation templates to guide an active contour process. Comparisons with GrabCut in One Cut and with Paper Doll show that the proposed approach is promising and performs favorably compared to generic or more dedicated object extractors. The method can easily be extended to new object classes at relatively low cost, *i.e.* by manually segmenting objects from these classes. We intend to continue adding new object classes to the RichPicture database. Also, a better alignment solution should benefit the proposed method, as well as the annotation of more images for training the person detector with other class-specific poses.

References

1. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Image Proc. 10(2), 266–277 (2001) 5
2. Chen, H., Gallagher, A., Girod, B.: Describing clothing by semantic attributes. In: ECCV. pp. 609–623 (2012) 1
3. Chen, Q., Huang, J., Feris, R., Brown, L.M., Dong, J., Yan, S.: Deep domain adaptation for describing people based on fine-grained clothing attributes. In: CVPR (June 2015) 2
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. pp. 886–893 (2005) 4
5. Di, W., Wah, C., Bhardwaj, A., Piramuthu, R., Sundaresan, N.: Style finder: Fine-grained clothing style detection and retrieval. In: IEEE Intl. Workshop Mobile Vision, CVPR. pp. 8–13 (June 2013) 1
6. Dollár, P., Zitnick, C.L.: Fast edge detection using structured forests. ArXiv (2014) 6
7. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014) 4
8. Kalantidis, Y., Kennedy, L., Li, L.J.: Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos. In: ACM Intl. Conf. Multimedia Retrieval. pp. 105–112 (2013) 2, 3
9. Kaufman, L., Rousseeuw, P.: Clustering by means of medoids. In: Dodge, Y. (ed.) Statistical Data Analysis Based on the L1-Norm and Related Methods. pp. 405–416. North-Holland (1987) 4
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Bartlett, P., Pereira, F., Burges, C., Bottou, L., Weinberger, K. (eds.) NIPS 25, pp. 1106–1114 (2012) 4
11. Liu, S., Feng, J., Song, Z., Zhang, T., Lu, H., Xu, C., Yan, S.: Hi, magic closet, tell me what to wear! In: ACM Multimedia. pp. 619–628. ACM (2012) 1
12. Liu, S., Song, Z., Liu, G., Xu, C., Lu, H., Yan, S.: Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In: CVPR. pp. 3330–3337 (2012) 2, 3
13. M. Hadi, K., Xufeng, H., Svetlana, L., Alexander, C.B., Tamara, L.B.: Where to buy it:matching street clothing photos in online shops. In: ICCV (2015) 2
14. Rother, C., Kolmogorov, V., Blake, A.: GrabCut – interactive foreground extraction using iterated graph cuts. ACM Trans. on Graphics pp. 309–314 (2004) 6, 7
15. Simo-Serra, E., Fidler, S., Moreno-Noguer, F., Urtasun, R.: Neuroaesthetics in Fashion: Modeling the Perception of Fashionability. In: CVPR (2015) 2
16. Song, Z., Wang, M., sheng Hua, X., Yan, S.: Predicting occupation via human clothing and contexts. In: ICCV. pp. 1084–1091. IEEE Comp. Soc., Washington, DC, USA (2011) 2
17. Tang, M., Gorelick, L., Veksler, O., Boykov, Y.: Grabcut in one cut. In: ICCV. pp. 1769–1776. IEEE Computer Society, Washington, DC, USA (2013) 2, 7, 8
18. Veit*, A., Kovacs*, B., Bell, S., McAuley, J., Bala, K., Belongie, S.: Learning visual clothing style with heterogeneous dyadic co-occurrences. In: ICCV. Santiago, Chile (2015) 2
19. Yamaguchi, K., Hadi, K., Luis, E., Tamara, L.B.: Retrieving similar styles to parse clothing. IEEE TPAMI 37, 1028–1040 (2015) 2, 6
20. Yamaguchi, K., Kiapour, M.H., Berg, T.L.: Paper doll parsing: Retrieving similar styles to parse clothing items. In: ICCV. pp. 3519–3526. Washington, DC, USA (2013) 3, 6
21. Yang, Y., Ramanan, D.: Articulated human detection with flexible mixtures of parts. IEEE TPAMI 35, 2878–2890 (2013) 3