This is the peer reviewd version of the followng article:

A complete system for garment segmentation and color classification / Manfredi, Marco; Grana, Costantino; Calderara, Simone; Cucchiara, Rita. - In: MACHINE VISION AND APPLICATIONS. - ISSN 0932-8092. - STAMPA. - 25:4(2014), pp. 955-969. [10.1007/s00138-013-0580-3]

Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

19/04/2024 08:21

(Article begins on next page)

Machine Vision and Applications manuscript No.
(will be inserted by the editor)

A complete system for garment segmentation and color classification

Marco Manfredi · Costantino Grana · Simone Calderara · Rita Cucchiara

Received: date / Accepted: date

Abstract In this paper we propose a general approach for automatic segmentation, color based retrieval and classification of garments in fashion stores databases, exploiting shape and color information. The garment segmentation is automatically initialized by learning geometric constraints and shape cues, then it is performed by modeling both skin and accessories colors with Gaussian Mixture Models. For color similarity retrieval and classification, in order to adapt the color description to the users' perception and the company marketing directives, a color histogram with an optimized binning strategy, learned on the given color classes, is introduced and combined with HOG features for garment classification. Experiments validating the proposed strategy, and a free-to-use dataset publicly available for scientific purposes, are finally detailed.

Keywords image retrieval · segmentation · color clustering · graph-cut

1 Introduction

Internet shopping has grown incredibly in the last years, and fashion created an interesting application field for image understanding and retrieval, since hundreds of thousands images of clothes constitute a challenging dataset to be used for automatic or semi-automatic segmentation strategies, color analysis, texture analysis, similarity retrieval, automatic piece of clothing classification and so on. Image processing and understanding, in particular, could be beneficial in this context. In fact, it could improve the quality control of the manual annotations of operators, as well as accelerate the process itself by exploiting objective measurements on every piece of clothing, gathered with visual cues. A correct automatic analysis and retrieval has the potential for dramatically improving the user experience as well as the industrial process, but at the same time a strong effectiveness is mandatory: suggesting a wrong garment to an unwilling buyer, or inconsistently categorizing different pieces of clothing, has a direct impact on the perception of the quality of the system itself. This is indeed particularly important in fashion e-commerce, where the quality of the product cannot be appreciated on the real object, but is strongly linked to the quality of its presentation.

In this paper, we address the problem of automatic segmentation, color retrieval and classification of fashion garments. Depending on the availability of manually photo retouched images (as often happens) background removal is performed with simple thresholding or with a more sophisticated approach detailed in Sec. 3.1. A Random Forest classification on projection features is used to classify the product category (Sec. 3.2), while Gaussian Mixture Models (GMM) are exploited to select the interesting piece of clothing of the picture, with an automated initialization procedure(Sec. 3.3). Upon the results of Garment Segmentation, color and type classification procedures are outlined in Sec. 3.4 and 3.6 respectively. A novel color histogram specifically optimized on the color distribution of the dataset classes is finally employed for similarity retrieval.

To summarize, we combine the state of the art representation in image segmentation techniques with a

Marco Manfredi · Costantino Grana · Simone Calderara · Rita Cucchiara
Università degli Studi di Modena e Reggio Emilia
via Vignolese 905/b - Modena, Italy
Tel.: +39-059-2056270
Fax: +39-059-2056129
E-mail: marco.manfredi@unimore.it, costantino.grana@unimore.it, simone.calderara@unimore.it, rita.cucchiara@unimore.it

problem specific initialization and a powerful color description to create a complete fashion images analysis system. We demonstrate its effectiveness for automatic color based retrieval and garment classification. The main contributions of our work are:

- our method employs a GMM color modeling to describe *non interesting* parts, such as skin and additional garments (not the item which is advertised by the image), and creates a segmentation by removing them;
- we propose a novel color descriptor which provides a discriminative summary of the color distribution of the region of interest; moreover we provide a solution based on an extension of integral images to allow its fast computation;
- we provide a large dataset, used in our experiments, in order to allow the scientific community to test their solutions in comparison with our choices.

2 Related work

Image segmentation is the process of partitioning the original image into different sub regions of homogeneity, which conveys saliency properties to be used in the following stages of image understanding. The literature on the topic is huge: please refer to works like [1, 2] for some recent surveys. The use of segmentation for clothing can therefore be considered just another application scenario, in which the same techniques (based on supervised learning, clustering, and so on) are used not to describe the image given the parts, but to remove irrelevant regions, in order to focus the image retrieval by similarity on the interesting part (i.e. the current piece of clothing). Hu *et al.* [3] for example performed the segmentation via graph cuts with foreground and background seeds estimated by a constrained Delaunay triangulation. Bertelli *et al.* [4] integrated object-level top down information with low-level image cues into a kernelized structural SVM learning framework.

Unlike usual image retrieval strategies, built upon the bag-of-words model [5] in various flavors and with dictionaries created from several kinds of local descriptors [6], there are contexts in which a global representation focused on specific visual cues is much more suitable. Let's consider for example the case of interest in this paper, in which pieces of apparel and fashion garments are effectively represented by means of their dominant and more salient colors, even by the expert personnel itself. Here local information is not very significant, leading to an inaccurate visual summary, mostly driven to boundaries (the shape) and distracting interest points (folds, illumination changes due to the body of the model and the position of lights). Instead, after an adequate shape segmentation and target detection (the main apparel worn by a model) we need global features that provide a compact summary of the visual content, typically by aggregating some chromatic information extracted at every pixel location of the target. Color Histograms [7] are one of the most common descriptors which describe the visual content ignoring its spatial arrangement. Many variations of them have been proposed in the last two decades, along with different distance measures (histogram intersection, Bhattacharyya, χ^2).

A consistent improvement in accuracy is reached with the adoption of adaptive binning on every image [8]. Since the resulting histograms can vary in length, the comparison requires more complex solutions. The Earth Mover's Distance (EMD) [9] is a cross-bin distance that addresses this alignment problem. EMD is defined as the minimal cost that must be paid to transform one histogram into the other, where there is a "ground distance" between the basic features that are aggregated into the histogram. The EMD as defined by Rubner is a metric only for normalized histograms, limiting the possibility of using it with fast algorithms for nearest neighbor searches, fast clustering and large margin classiers. However, recently Pele *et al.* [10] suggested a variation called $\widehat{EMD}$, which is a metric also for unnormalized histograms, allows the use of thresholded ground distances, and is suitable for very fast implementations [11]. Recently the bag-of-colors approach has been proposed by Wengert *et al.* [12] for image search: the color signature of the image is produced by a k-means quantization over a training set. This procedure is shown to improve retrieval accuracy in terms of mAP.

Bossard *et al.* [13] tackle the problem of garment classification using a Random Forest Classifier. Their approach is focused only on upper body clothing and, exploiting training data crawled from the web, learns to classify the garment category and some visual attributes like color and material. Cheng *et al.* [14] proposed a clothes search system based on garments visual attributes and higher level properties like "suitable occasions" or "feeling description". Recently, Yamaguchi *et al.* [15] proposed an interesting solution to garment parsing, that is able to classify 53 different garment categories from fashion photographs. The method is able to separately segment each garment exploiting superpixels and flexible parts model for human pose estimation. They focused on a detailed parsing of the whole figure, but they do not analyze the color retrieval problem in depth, just providing few visual results. Liu *et al.* [16] followed a different strategy, avoiding parsing and seg-

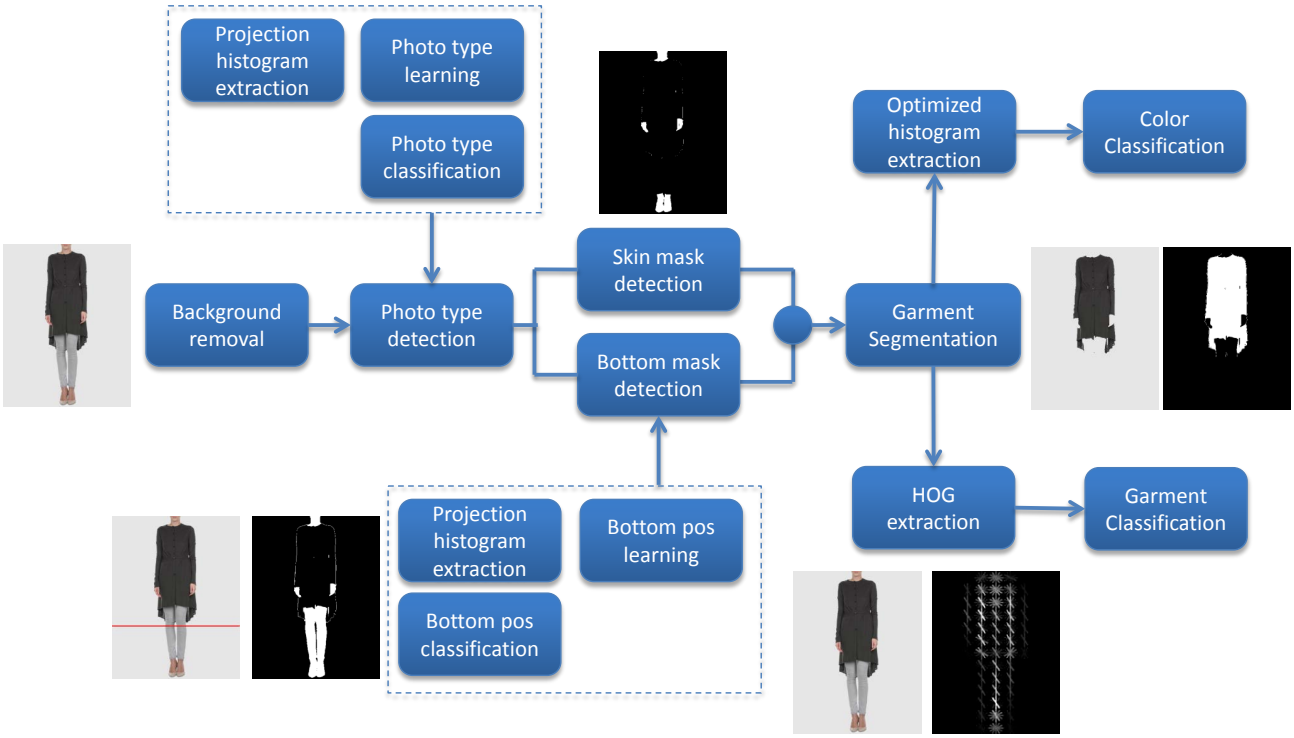


Fig. 1 Overall schema of the system.

mentation, focusing instead on human parts. By avoiding segmentation and parsing, they are able to match images also in a cross-scenario, i.e. from street images to controlled studio shootings. They also do not provide quantitative results on color retrieval, even if they stress that color is one of the most salient attributes when people search for favorite clothing. An important note is that their definition of colors is limited to 10 nuances and 1 multicolor category, so making the problem much easier when comparing to the fashion industry definitions, which are definitely more demanding. None of the previous works on fashion images proposes a solution for a complete description of clothes, including garment segmentation, color retrieval and category classification.

3 System description

We propose a complete system for garment segmentation and color classification from images taken from on-line fashion stores. The system is composed of several modules depicted in Fig. 1 and every single module will be detailed in the following subsections. Roughly, given an image, background removal is performed in order to obtain a binary mask. The *photo type detection* module first classifies the masks according to the shooting type (e.g. model is present in the image or mannequin is used . . .). Consequently, according to the shooting type, both skin and additional garments and accessories are removed to obtain a clear picture of the object of interest. Finally, a garment color descriptor or HOG based descriptors of garment shapes and textures are computed on the selected object and used for color retrieval or classification.

3.1 Background removal

Background removal is the procedure of separating the main object of an image from the background, creating a binary mask M as a result. Background removal can be used to choose the appropriate background color for a certain object, or to perform further analysis on the object of interest. Background removal can be easily done on photo retouched images, where shadows and minor objects are removed, providing a uniform background of a known color. A simple processing of the images with a threshold based on the reference background color may be employed to recover the binary mask M of the object.

We have investigated the capability of the proposed system to deal with original (not photo-retouched) images, that therefore present a nonuniform background and possibly shadows. The purpose is to replace the manual background subtraction with an automatic one.

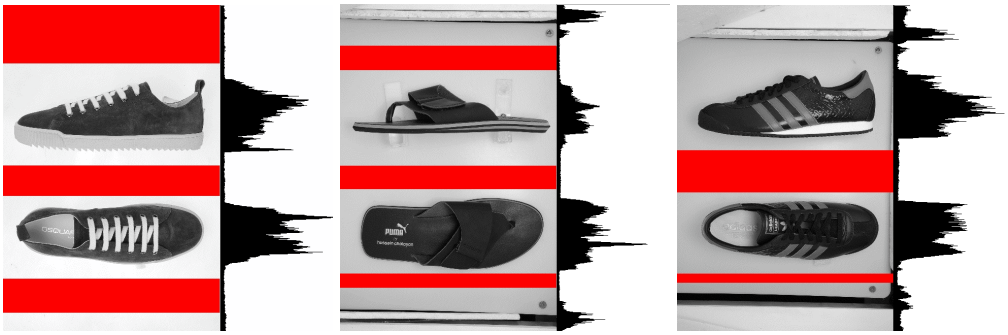


Fig. 2 Vertical gradient projection histograms are used to highlight slices that probably belong to the background (highlighted in red).

(a) (b) (c)

Fig. 3 The background extraction procedure. In (a) background slices are highlighted, (b) shows the distance map from the background histogram, (c) shows the segmentation results obtained with GrabCut.

The automatic background extraction process starts computing a gradient map with the Sobel operator, to highlight the uniform and low-textured areas, which are typically used as supporting plane in photographic studios. Then the gradients are projected either vertically or horizontally depending on the shape of the object class, obtaining a projection histogram which reflects the probability of having an object in a specific image slice. Therefore a direct thresholding on the histogram values is employed to highlight image portions containing only background values. These are then validated with an empirical requirement of being compact and at least a few lines wide (10 in our experiments was found satisfactory). Fig. 2 shows some examples on shoe images.

From the validated slices, an initial background model is built as a uniformly sampled 3D RGB color histogram H_b with 16 values per channel. Each pixel's vote is distributed among eight color bins with trilinear interpolation. A background probability map B_p is generated, where the probability of each pixel is represented by the corresponding background histogram value. These values are linearly scaled in the range $[0, 1]$. A pixel x with $B_p(x) = 0$ has a color that was never found on the selected background slices; on the other hand, when $B_p(x) = 1$, the pixel x belongs to the set of colors which is most likely to be background.

The final step of the segmentation procedure is composed of the GrabCut algorithm [17], that can be initialized with a mask GC_m indicating how each pixel is likely to be foreground or background. Each pixel of GC_m can take one of four labels: certainly background (BGD), probably background (PR_BGD), probably foreground (PR_FGD) and certainly foreground (FGD). All the pixels belonging to the previously extracted background slices take the BGD label.

To assign a proper label to the other pixels, B_p is progressively thresholded, so as to assign the FGD label to pixels with the lowest background probability. The three necessary thresholds have been found empirically as $T_{BGD} = 0.95$, $T_{PR_BGD} = 0.5$ and $T_{PR_FGD} = 0.0$, and are applied in the following way: if $B_p(x) > T_{BGD}$, then x is background, else if $B_p(x) > T_{PR_BGD}$, then it is probably background, else if $P(x) > T_{PR_FGD}$, then it is probably foreground, otherwise it is foreground. Note that in our experiments we ended up con-

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

Fig. 4 Samples of the three photo types taken into consideration in our system: Model (a,b), Mannequin (c,d) and Still Life (e,f).

sidering as certainly foreground only those pixels which never appeared in the background slices. Figure 3 shows the steps of the background extraction method.

3.2 Photo Type Identification

Different fashion products categories require specific presentations in terms of photographic shooting to better highlight the product characteristics. Common fashion standards define three principal guidelines differentiating the presence of a human model wearing the garment (*Model*), a mannequin dressed with the product (*Mannequin*) or shoes and accessories which are imaged without any distracting element (*Still Life*). We will refer to this problem as *photo type identification* (see Fig. 4 for some visual samples).

Following the idea that the product is always at the center of the image, the information of the photo type is related to the shape of the object, and since we can avoid aligning the images, projection histograms of the binary masks are a good candidate for this task.

We project the binary mask on the horizontal and vertical axes obtaining a first rough shape descriptor. Considering that, not all the bins are equally informative, it is important to identify which elements are characteristic of the different photo types. For this reason we chose to use a discriminative approach which involves a feature selection process.

The two principal solution employed in literature are Boosting (initially proposed by Schapire in [18] and then developed in a huge amount of varieties) and Decision Trees Classifiers. In particular Random Forest classifiers [19] have been chosen because they can handle easily multiclass problems providing an inherent feature selection mechanism. The random forest is trained using the concatenated projection histograms (both vertical and horizontal) bin values where the final number of classes corresponds to the three different photo types.

3.3 Segmentation of garments of interest

The Still Life category does not require any further analysis, since the object mask perfectly identifies the presented product.

Conversely both Model and Mannequin classes contain additional elements in order to nicely present the product to the public. For this motivation it is mandatory to find a solution capable of extracting the portion of the image containing only the product of interest. We refer to this problem as *Garment Segmentation*, and we remark that the absence of any information coming from text annotation and product description make the task challenging.

Most of the solutions to garment segmentation tackle the problem under the hypothesis that clothes are worn by people, so finding where people is (for example using face detection) provides a good starting point for following steps. Skin represents one of the most valuable indicator of people presence and skin detection and removal is often adopted. When dealing with real fashion photo shootings and product advertising rules, most of these hypotheses are broken, because photographers aim to make the product appealing for the consumer, neglecting objective color reproduction and photo realism. This limits the trivial application of skin color identification on fashion photographs as can be immediately seen from images in Fig. 5.

To deal with these problems adaptive skin detection approaches [20] have been used as an instrument to refine global skin detection results using the current image features. Among different skin color descriptors, Gaussian Mixture Models in color domain proved to be one of the state of the art approaches [21]. Since Gaussian Mixture Models training using the EM algorithm [22] is computationally expensive, we follow the iterative energy minimization approach used in the GrabCut algorithm [17]. The GrabCut segmentation algorithm is an iterative procedure which aims to mini-

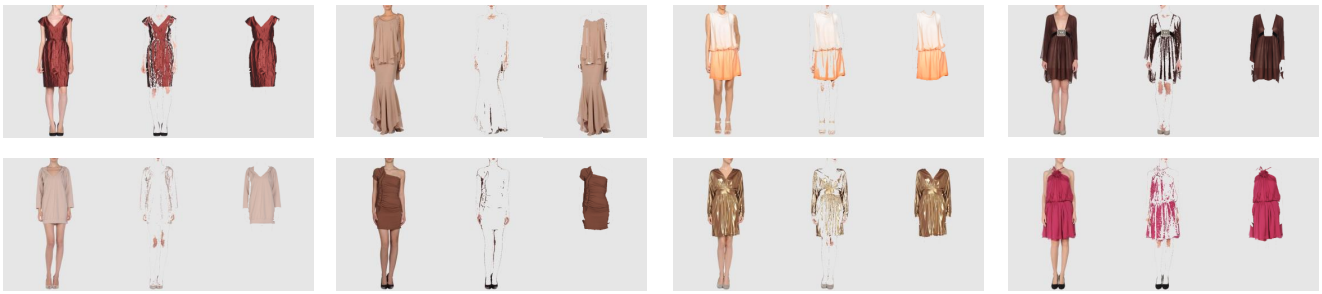


Fig. 5 Examples of mask extraction. For every triplet of images, the leftmost is the original image, the center one is the image after removing skin with the method proposed in [20] while the rightmost one is obtained with our proposal.

mize the following Gibbs energy:

$$E(\boldsymbol{\alpha}, \boldsymbol{k}, \theta, \boldsymbol{z}) = U(\boldsymbol{\alpha}, \boldsymbol{k}, \theta, \boldsymbol{z}) + V(\boldsymbol{\alpha}, \boldsymbol{z}) \tag{1}$$

where all boldface quantities are N-dimensional vectors, with N the number of pixels in the image (one value per pixel). $\boldsymbol{z}$ is the image vector, $\boldsymbol{\alpha}$ is the segmentation mask ($\alpha_n \in \{0, 1\}$). $\boldsymbol{k}$ is the vector, with $k_n \in \{1, \dots K\}$, assigning each pixel to a unique GMM component, from either the background or the foreground model. θ is the set of parameters of the GMMs, that is K color means (3-dimensional vectors), full covariances (3×3 matrices) and mixture weighting coefficients. The data term $U(\cdot)$ is defined as

$$\begin{aligned} &U(\boldsymbol{\alpha}, \boldsymbol{k}, \theta, \boldsymbol{z}) \\ &= \sum_n -log(p(z_n|\alpha_n, k_n, \theta)) - log(\pi(\alpha_n, k_n)), \end{aligned} \tag{2}$$

where $p(\cdot)$ is a Gaussian probability distribution, and $\pi(\cdot)$ are mixture weighting coefficients. The smoothness term $V(\cdot)$ in Eq. 1 is

$$V(\boldsymbol{\alpha}, \boldsymbol{z}) = \gamma \sum_{(m,n) \in C} [\alpha_n \neq \alpha_m] e^{-\beta \|z_m - z_n\|^2}, \tag{3}$$

where $[\cdot]$ denotes the indicator function (taking values $0, 1$), C is the set of pairs of neighboring pixels (8-way connectivity is used in our case). This energy term encourages coherence in regions of similar colors. As originally proposed in [23], the constant β is chosen to be

$$\beta = \left(2 \left\langle \|z_m - z_n\|^2 \right\rangle\right)^{-1}, \tag{4}$$

where $\langle \cdot \rangle$ denotes expectation over an image sample. This choice of β ensures that the exponential term in 3 switches appropriately between high and low contrast.

The minimization of Eq. 1 is efficiently performed by repeatedly estimating the GMM parameters θ and using minimum cut to estimate the segmentation mask and the hard assignment vector.

The majority of the approaches for image segmentation simplify the problem, by assuming some level of human interaction (seeds selection, bounding box drawing). This allows to obtain a starting point defining what is foreground and what is background, allowing to initialize a further refinement process. Even if this turns out to be an effective solution, especially in the application scenarios for which these algorithms usually are designed (i.e. assistance for the image post-processing in studio), when a fully automated system is desired and no process changes in the company's production flow are possible, an automatic initialization becomes necessary.

To solve this problem, we can consider that in most fashion datasets for internet e-commerce applications, models faces are located in the top-center of the image, so it is safe to expect that by selecting part of the object mask from the top, skin tones and hair will be identified. Following these photographic guidelines it is feasible to assume that the *skin mask* can be initialized by selecting few lines from the top of the image, proportionally w.r.t. image size, use this element as the initializers to solve the optimization problem of Eq. 1 and then retain the generated mask.

Conversely, it is not equally safe to hypothesize that, by selecting part of the object mask from the bottom, some items and parts of the image which are not relevant to define the main clothing under interest (legs potentially left out by the skin segmentation, shoes, trousers and so on) will be identified. In fact, depending on the product type its presentation is specifically conceived to properly advertise it. For these reasons sometimes full shot of the body are taken (e.g. long dresses), while in shirts and jackets a zoomed shooting is preferred to highlight garments details. This elements concur to raise the degree of variability of the bottom part of the image, and fixed geometric constraints (i.e. choosing a fixed number of lines from the bottom) typically lead to under/over detection.

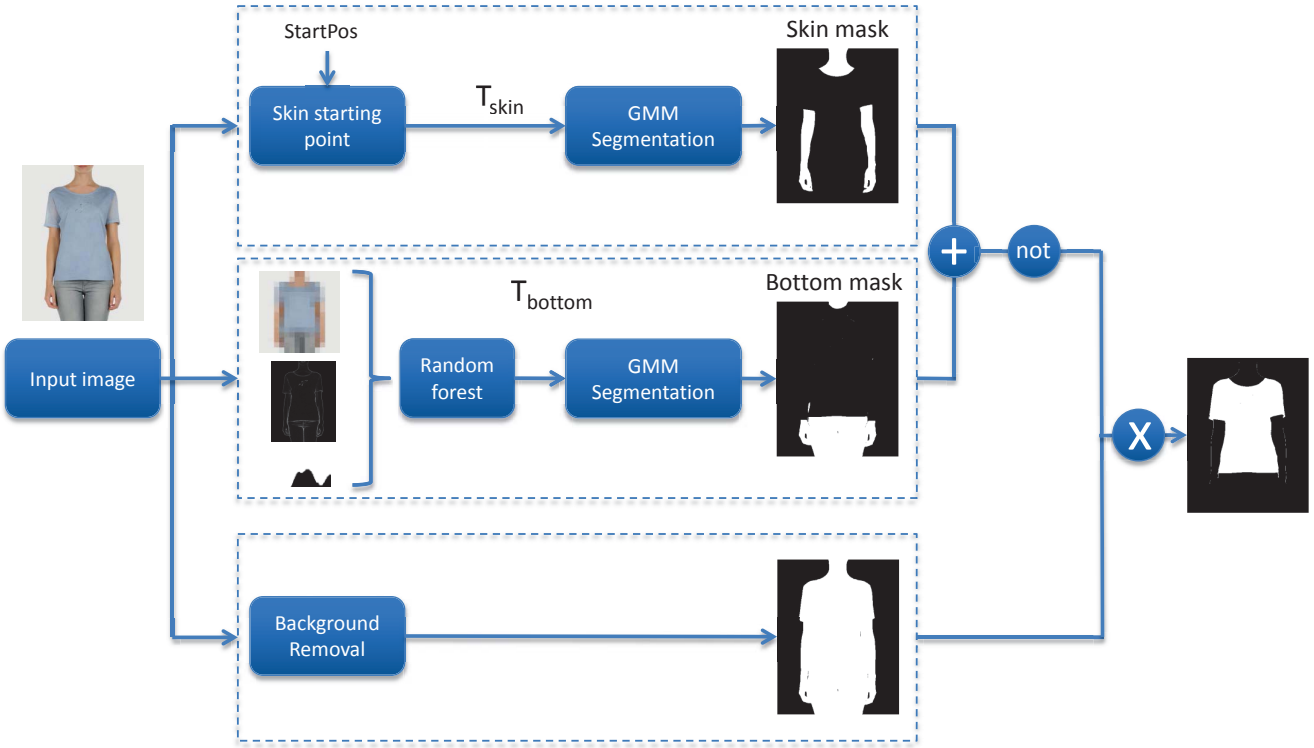


Fig. 6 Block Diagram of the Garment Segmentation algorithm.

We face this problem as a supervised learning task where the irrelevant lower part (*bottom mask*) is directly inferred combining image features and operators experience. Using a regression random forest classifier we obtain the number of probably irrelevant lines from the bottom of the image. The features used are again the projection histograms of the segmentation mask, along with two low resolution maps of the colors and of the gradients. These two maps help distinguish cases in which the silhouette by itself is not sufficient, for example cases with boots, leggings, transparencies and so on. The use of redundant features is mitigated by the feature selection stage of the random forest classifier that chooses the best ones for the assigned task. In particular, color and gradient maps have non-zero feature weights in most of the cases. As for the skin mask case, we solve the optimization of Eq. 1 using the learnt bottom lines for initialization to obtain a binary mask of the bottom non-informative part.

The garment segmentation is finally obtained by the product of the *binary mask* and the complement of the binary OR combination of the two aforementioned masks.

For the sake of clarity, the procedure to accomplish the computation of the garment segmentation mask is sketched in Fig. 6.

3.4 Color signature definition and extraction

Working on large datasets for search and classification purposes fixed length signatures are often adopted, being more easily used in different hashing and indexing tasks. Color histograms are a common solution, and using a non uniform division of the color space could better describe the distribution of color aspects in the dataset. Since we aim at exploiting color for visual object classification, we would like to employ a dynamic binning which emphasizes the classes peculiarities. This is different from extracting a set of colors based on the data only (for example by clustering in the color space), but is a task of feature selection by incorporating data classification information in the definition of the color signature. A color in a color space $\mathcal{C}$ is denoted by c. Given an image I, the color distribution for the image is

$$p(c|I) = \frac{\#\{I(x,y) = c\}}{\#I}. \tag{5}$$

Given a class of images C_j, with $j = 1, \ldots, J$, described by a training set of images I, we can define $p_j(c)$ as the L1-normalized sum of the color distributions of all images in that class. We approach the problem of finding a class optimized binning with a greedy procedure inspired to the median cut algorithm [24].

Algorithm 1 Class Based Color Space Partitioning

```
Compute cumulative histograms of training images p_j for all classes
b ← (0,0,0),(255,255,255)  ▷ Start with the whole color space
FindBestSplit(b)
insert(list, b)  ▷ list contains the color space partition
while size(list)< N do
    b ← max_delta(list)
    b_0, b_1 ← GetSplits(b)
    FindBestSplit(b_0)
    insert(list, b_0)
    FindBestSplit(b_1)
    insert(list, b_1)
end while
```

A box is the set of colors contained within a parallelepiped defined by two extreme colors low (l) and high (h), with $l, h \in \mathcal{C}$:

$$b = \{c \in \mathcal{C} : l_k \leq c_k \leq h_k, k = 0, 1, 2\} \tag{6}$$

To simplify the equations, from now on we will assume a three-channel color space. We will equivalently write $b = (l, h)$. We call $m_j(b)$ the mass of box b in class j, such as

$$m_j(b) = \sum_{c \in b} p_j(c). \tag{7}$$

The total mass of b is

$$M(b) = \sum_{j=1}^{J} m_j(b). \tag{8}$$

We also denote $C(b)$ as the class associated with box b, that is the class with maximum mass for the box:

$$C(b) = \arg\max_j m_j(b). \tag{9}$$

The error induced by considering colors in b to be all of class C_j is defined as:

$$E(b) = \sum_{j \neq C(b)} m_j(b) = M(b) - m_{C(b)}(b). \tag{10}$$

We define a split of a box as $s = (v, k)$, meaning that we divide the box along channel k at position v. Splitting a box has the purpose of better describing the colors of that box, thus it is reasonable to assume that this will lower the error. We call $\delta(b, s)$ the difference between the current error caused by the box b and the one obtained after the splitting s. $\delta(b) = \max_s \delta(b, s)$ is the error induced by the *best split*. We will then choose to split the box which maximizes its δ.

The algorithm employs a list of boxes, initially containing a single box enclosing the whole 3D color space, described as b_0. For example in an 8-bit RGB color space $b_0 = ((0, 0, 0), (255, 255, 255))$. At each iteration step we extract from the list the box which has the maximum delta value, then it is split so as to minimize the sum of the errors after the split. The resulting boxes are put back in the list. The algorithm proceeds until the required number of boxes/histogram bins is obtained. Pseudo code is given in Algorithm 1.

To compute the mass of a box, and therefore the search for the best split, we integrated $p_j(c)$ for all c which belong to b in three different directions, that is the three color channels. This procedure is computational demanding, but it can be accomplished offline, potentially adapting to the available computational capacity with a suitable quantization of the color distribution.

3.5 3D Integral Color Histograms

The algorithm described in previous section requires to compute the mass of a box, and the search for the best split requires to compute it at all possible positions of a split. The straightforward solution to this problem is to integrate $p_j(c)$ for all c which belong to b in three different directions, that is the three color channels. This is computational demanding and must be performed on all class distributions, making it unfeasible as soon as the distributions are not heavily quantized.

In this work we propose to simplify the selection of the best search by defining a 3D extension of the "integral images" approach introduced in [25]. The integral image contains at every point (x', y') the sum of all pixels with $x < x'$ and $y < y'$. This allows to compute the sum of all values of a rectangle by combining just four values of the integral image. This was successfully employed in the past also for extracting histograms on an arbitrary rectangular region of an image [26].

We propose to apply the same process to 3D color histograms, to compute *in constant time* the mass of a box. To this aim the first step is to extract the 3D integral color histogram from a conventional 3D color histogram. Having $p(c)$, the normalized histogram at color c, the 3D integral histogram $ih(c)$ is defined as

$$ih(c) = \sum_{\substack{x : x_k < c_k \\ k=0,1,2}} p(x) \tag{11}$$

As for integral images, it is possible to compute the 3D integral histogram by a single sweep over the original histograms, taking advantage of the recursive nature of

Algorithm 2 Pseudo code of fast best splitting algorithm

```
procedure FindBestSplit(b, ih)
    error ← +∞
    for k ∈ [0, 2] do                                  ▷ For all channels
        for j ← 1, J do                  ▷ Precompute columns limits
            m_min_j = col(b, k, l_k − 1)
            m_max_j = col(b, k, h_k)
        end for
        for x ∈ [l_k, h_k − 1] do                      ▷ For all possible splits
            M^0 ← 0                                    ▷ Total masses for splits
            M^1 ← 0
            for j ∈ [1, J] do
                c ← col(b, k, x)
                m_j^0 ← c − m_min_j
                M^0 ← M_j^0 + m_j^0
                m_j^1 ← m_max_j − c
                M^1 ← M_j^1 + m_j^1
            end for
            E^0 ← M^0 − max_j m_j^0                    ▷ Errors of splits
            E^1 ← M^1 − max_j m_j^1
            if error > E^0 + E^1 then
                error ← E^0 + E^1                      ▷ Update best split
                SetSplit(b, k, x)
                δ ← E(b) − error
            end if
        end for
    end for
end procedure
```

the definition. In particular:

$$\begin{aligned} ih(c) = p(c) &+ ih(c_0-1, c_1, c_2) + ih(c_0, c_1-1, c_2) \\ &- ih(c_0-1, c_1-1, c_2) + ih(c_0, c_1, c_2-1) \\ &- ih(c_0-1, c_1, c_2-1) - ih(c_0, c_1-1, c_2-1) \\ &+ ih(c_0-1, c_1-1, c_2-1) \end{aligned} \tag{12}$$

This assumes that the value of the 3D integral histogram is 0 whenever any of the color coordinates is negative. Given a 3D integral histogram, the computation of the mass within a box $b = (l, h)$ is quite similar to the use of integral images for rectangle area calculations:

$$\begin{aligned} m(b) = ih(h_0, h_1, h_2) &- ih(l_0-1, h_1, h_2) \\ &- ih(h_0, l_1-1, h_2) + ih(l_0-1, l_1-1, h_2) \\ &- ih(h_0, h_1, l_2-1) + ih(l_0-1, h_1, l_2-1) \\ &+ ih(h_0, l_1-1, l_2-1) - ih(l_0-1, l_1-1, l_2-1) \end{aligned} \tag{13}$$

This formulation holds, again assuming that $ih(c) = 0$ if $c_k = -1$ for any k.

A further observation may help while exhaustively searching for the best split. We define the quantity *column* of b along dimension d at height x as the mass of the box with limits $(\tilde{l}, \tilde{h})$, where

$$\tilde{l}_k = \begin{cases} 0 & \text{if } k = d \\ l_k & \text{otherwise} \end{cases} \quad \tilde{h}_k = \begin{cases} x & \text{if } k = d \\ h_k & \text{otherwise} \end{cases} \tag{14}$$

This means that, using Eq. 13, for example when $d = 0$ we have

$$\begin{aligned} col(b, 0, x) &= ih(x, h_1, h_2) - ih(-1, h_1, h_2) \\ &\quad - ih(x, l_1-1, h_2) + ih(-1, l_1-1, h_2) \\ &\quad - ih(x, h_1, l_2-1) + ih(-1, h_1, l_2-1) \\ &\quad + ih(x, l_1-1, l_2-1) - ih(-1, l_1-1, l_2-1) \\ &= ih(x, h_1, h_2) - ih(x, l_1-1, h_2) \\ &\quad - ih(x, h_1, l_2-1) + ih(x, l_1-1, l_2-1) \end{aligned} \tag{15}$$

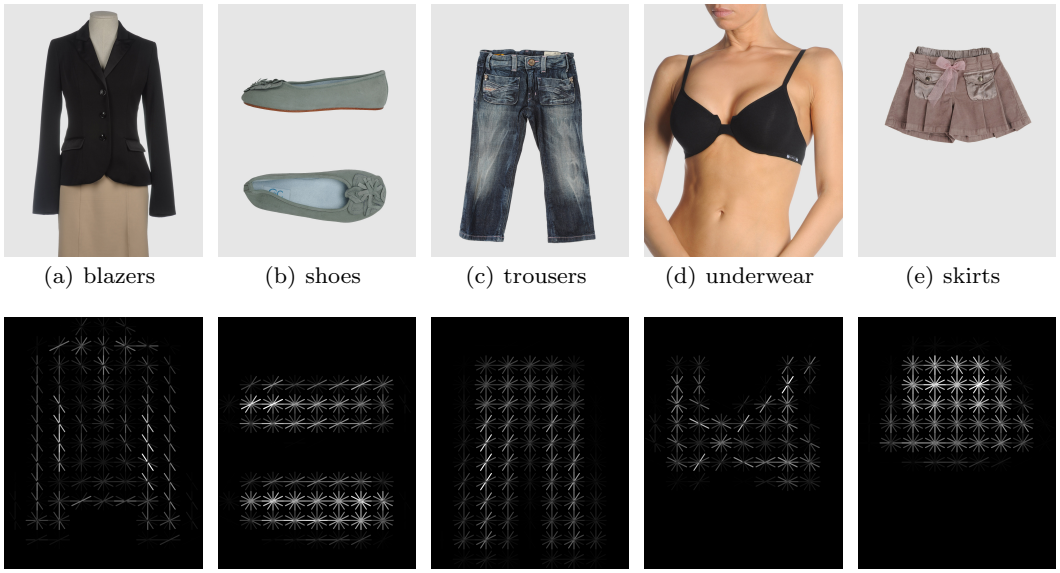
Plugging back Eq. 15 in Eq. 13 we can write:

$$m(b) = col(b, k, h_k) - col(b, k, l_k - 1), \tag{16}$$

thus enabling us to compute the best split by keeping fixed the second term (the "column base"), while progressively increasing the first one. The pseudo code is provided in Algorithm 2.

3.6 Garment Classification

An interesting feature of the proposed system is the automatic classification of garment categories, starting from the masks extracted in Sec. 3.3. Given a binary mask, we can extract projection histograms as previously mentioned, that roughly describe the shape of the object and give us some important clues about which category the garment belongs to. To complete the description of the shape of the object we exploit rectangular HOG (R-HOG) features [27], computing gradients on the mask, then dividing the image with a 9×13 grid of cells (117 cells), and grouping them in partially overlapped blocks of 3×3 cells each. Trilinear interpolation between histogram bins and cells is appropriately applied. The HOG feature is computed using 9 bins to quantize the orientation. We chose a set of 9 representative garment categories that can cover the entire dataset, while exhibiting a substantial shape difference: shirts, dresses, skirts, trousers, short pants, blazers, underwear, shoes and accessories. A multiclass linear Support Vector Machine is then trained, concatenating the projection histograms with the HOG features. For a comparison with nonlinear SVMs see Sec. 4.



(a) blazers (b) shoes (c) trousers (d) underwear (e) skirts

Fig. 7 Garment Classification: image samples with their relative HOG descriptors.

4 Dataset Description and Experimental Results

In order to verify the correctness of our proposal we built, in collaboration with a worldwide leader in fashion e-commerce, a dataset for fashion garment retrieval and color classification. We did not find in literature dataset of fashion products with complete ground truth data on both garment segmentation and classification. During the construction of the dataset, a high number of fashion operators have been employed to define in an objective way the color characteristics and the correctness of the automatic procedures. The dataset is composed of 60204 images of different pieces of clothes and accessories from most famous fashion designers. These are divided into the three photo type categories, mentioned in Sec. 3.2, where 23% are mannequin shootings, 52% models and the remaining 25% still life. For all these pictures the fashion retailer provides us the color category and the garment class they belong to, but this categorization is subject to possible errors because human operators are prone to mistakes during ordinary products management flow. We then choose to verify accurately 10000 images where the annotations have been checked and confirmed by different operators using a voting scheme. This verified subset has been selected by uniformly sampling the images on both the categories and photo types. The color categories comprise a range of 60 nuances typically adopted in fashion industries unevenly distributed in the color space. Additionally, in order to evaluate the effectiveness of garment segmentation, our output masks, on the verified subset, have been manually evaluated by the same operators which produced a list of correct/incorrect garment segmentation masks. The dataset and its annotation are publicly available at the website: `imagelab.ing.unimore.it/imagelab/fashion_dataset/dataset.zip`. In the dataset, we also provide our binary garment segmentation masks and color classification results in order to encourage comparisons and improvements to our solution.

To test the effectiveness of the background removal algorithm, pairs of original-retouched images are taken, comparing the extracted background masks. The binary masks extracted from the retouched images, due to their high accuracy (see 3.1), serve as ground truth data. In the photo-retouching phase, for aesthetic reasons all the images are slightly transformed to obtain the correct location, orientation and scale of the object of interest; then they are manually retouched to delete the background. For the evaluation of the automatic background removal algorithm a perfect alignment of ground truth data and original images is needed. To get the geometric transformation that transforms the retouched image into the original one we matched SIFT [28] descriptors extracted from the two images. We used RANSAC [29] to filter out noisy detections and to get the most likely transformation. It is known that the manually applied geometric transformations are composed of a translation step, a scaling step (equal in height and width) and, if necessary, a rotation step. We have to find only 5 parameters to describe the transformation, hence we consider 3 points at a time in the RANSAC voting procedure. The most challenging situ-

Fig. 8 Results of the system on four different samples. First Column is the Original Image. Second Column is the Garment Segmentation output. Finally, the Third Column shows the five most similar garments using the proposed color descriptor.

ations are those in which the object is slightly textured and only few descriptors are found.

In order to quantify the effectiveness of the garment segmentation algorithm, we do not have a sufficient amount of manually drawn segmentations to be used as ground truth. For this reason, we randomly picked 5000 images and segmented them using our proposal and a regular GrabCut, initialized as follows: the initialization mask is set to PR_BGD; a rectangle centered on the image center and with sides reduced to x percent of the image dimensions is set to PR_FGD. This general purpose initialization of the GrabCut algorithm, which assumes roughly centered objects, is a commonly adopted technique used for comparison in object segmentation. This same approach (with $x = 50\%$) was used in [30] and in [4]. We tried different values for x and found a reasonable compromise value at $x = 30\%$. To quantify the improvement of our garment segmentation strategy, we asked an operator to answer the question: "Does the segmentation outline only the garment of interest, without large missing parts?". Our approach provided very good results and was confirmed in 86% of the cases, while the general purpose initialization simply could not cope with any variation in the target position and other complementary garments (it was accepted only in 41% of the cases). The main reason is that our pipeline progressively finds not interesting regions (background, skin, bottom): it is easier to model their color distribution one at a time, than trying to create a model of all of them together against the model of the interesting part (which is often composed of many different tones). This is only possible thanks to the structured environment in which a specific initialization strategy can be designed.

For the photo type classification task, Sec. 3.2, we obtain an accuracy of 99.6% on the complete dataset of 60204 images, this task does not carry strong intrinsic difficulties because visual separation of data is quite clear. The garment segmentation phase presents a higher degree of complexity because images are subjected to the noise introduced by the product advertising rules. we still are capable of obtaining promising results reaching a 98.4% accuracy, tested on the verified subset of 10000 images. Some garment segmentation results can be shown in the second column of Fig. 8.
Additionally, in many practical situations, top fashion

Descriptor	bins	mAP
RGB Histogram	512	0.457
RGB Histogram	4096	0.496
L*a*b* Histogram	784	0.391
Bag of colors signature	512	0.492
Class-based Color Bag of Words	512	0.558
Class-based Color Bag of Words	1024	0.566

Table 1 Comparison of image retrieval results in terms of Mean Average Precision.

retailers, focused on worldwide internet distribution, manage a huge volume of images per day. In our specific scenario, the partner company handles typically more than 5000 products and photographs daily that are bulked in a set which is mandatory to process all at once. We report an endemic temporal constraint, due to factory workflow, of 4 hours processing, while our proposal is able to conclude the metadata extraction in less than half a second per image, i.e. less than one hour for 5000 images.

Finally, color based retrieval and classification is evaluated on selected garments masks and the following color descriptors have been tested:

- RGB Color Histogram: the three channels are uniformly quantized to 8 or 16 values, giving a 512 or 4096 bins descriptor.
- L*a*b* Color Histogram: the image is converted to the CIE L*a*b* color space, with the hypothesis that the source images are in the sRGB color space. The three channels are then quantized to 4,14,14 bins as suggested in [31].
- Bag-of-colors signature: this approach consists in clustering the colors extracted from the training set with k-means, followed by Inverse Document Frequency weighting, power law, and L1 normalization [12].
- The proposed Class-based Color Bag of Words.

The first test was performed to assess the ability of the different descriptors to retrieve similar images in terms of colors. Histograms were compared with the histogram intersection metric. Results are shown in Table 1. It is clear that the CIE L*a*b* color space fails to correctly represent the color information. This may be due to the assumption of sRGB color space which is not confirmed. While this color space could help in a general image retrieval context, when the collection is uniform, the raw RGB values are probably more reliable. Adapting the histogram binning to the dataset characteristics is indeed useful, even if these results do not show the consistent improvement reported in [12]. The proposed bin selection shows the best performance which is not matched by the RGB color histogram even rising the number of bins to 4096. Some example results are shown in the third column of Fig. 8.

Descriptor	RBF kernel	HI kernel
RGB Histogram	71.11	69.96
Bag of colors signature	69.48	72.27
Class-based Color BoW	73.80	74.38

Table 2 Accuracy results using SVMs with RBF and Histogram Intersection kernels

Descriptor	Linear SVM	HI kernel
HOG	82.5	80.30
HOG + Proj. Histogram	90.22	86.30

Table 3 Comparison of garment classification accuracy with different features and SVMs classifiers.

Category	Precision	Recall
accessories	0.9245	0.9800
blazers	0.9130	0.8400
dresses	0.8750	0.9800
shirts	0.8621	1.0000
shoes	0.9778	0.8800
short pants	0.9000	0.5200
skirts	0.6389	0.9200
trousers	1.0000	1.0000
underwear	1.0000	1.0000
Average	0.8990	0.9022

Table 4 Per-class breakdown of the garment classification precision and recall, using HOG+Proj.Histogram and linear SVM.

The second test tackles the color classification task. We trained a multiclass SVM using a 1-vs-1 learning strategy [32] using the three methods with 512 bins. We compared the RBF kernel and the Histogram Intersection kernel optimizing C and σ (only for RBF) with grid search. Table 2 reports the accuracy results. The bag-of-colors signature has the lowest accuracy with RBF kernel, which is much improved by the HI kernel. The opposite happens with RGB histograms. Our solution is still better, even if the improvement obtained using the HI kernel is not so significant.

The computational time of the proposed solution is much lower than what required by k-means, and most of the complexity is the precomputation of the 3D integral histograms, which requires (without quantization) to sweep $256^3 \cdot N_c$ values, with N_c the number of classes. The main limitation is the memory requirements of this approach which is $256^3 \cdot N_c \cdot 4$ bytes when using 32-bit floats, that is 64 MB for each class. In our experiments we were using 60 classes for a total memory allocation of 3.75 GB.

The garment classification algorithm was tested on a selected dataset of 900 images belonging to 9 categories (see Sec. 3.6), equally split in training and test-

Fig. 9 Results of garment classification on three categories: skirts, dresses and short pants. In the first column a training image for each class is presented. Second, third and fourth column are correctly classified garments. In the last two columns misclassified examples are reported.

ing set. A linear multiclass SVM trained with HOG features and projection histograms obtains the best results (see Tab. 3). Due to the high resolution of the involved images (1571 × 2000), we resized them to 314× 400, without observing any accuracy loss, considerably reducing the computational time. Linear SVMs perform very well, probably due to the high dimensionality of the feature vectors (around 2000 dimensions), but are still prone to errors. A per-class breakdown of the garment classification precision and recall is provided in Tab. 4. Sometimes, garments of similar shape, like short pants and skirts, are misclassified. Some visual results are provided in Fig. 9.

5 Conclusions

In this paper we proposed a complete system for garment segmentation and color classification which has the great advantages of being efficient in terms of computational time, adaptable to different fashion rules and accurate enough to compete with human operators' performance on the same data. The massive use of learning techniques allows to reconfigure the system rules by simply exhibiting new examples to the different classifiers. We are currently testing the system in the industrial workflow of a world leader e-commerce retailer.

References

1. Zhang H, Fritts JE, Goldman SA (2008) Image segmentation evaluation: A survey of unsupervised methods. Computer Vision and Image Understanding 110:260–280.
2. Freixenet J, Muñoz X, Raba D, et al (2002) Yet Another Survey on Image Segmentation: Region and Boundary Information Integration. In: Proceedings of the European Conference on Computer Vision(ECCV), volume 2352, pp 408–422.
3. Hu Z, Yan H, Lin X (2008) Clothing segmentation using foreground and background estimation based on the constrained Delaunay triangulation. Pattern Recognition 41:1581–1592.
4. Bertelli L, Yu T, Vu D, et al (2011) Kernelized structural SVM learning for supervised object segmentation. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp 2153–2160.

5. Dance CR, Csurka G, Fan L, et al (2004) Visual categorization with bags of keypoints. In: ECCV Workshop on Statistical Learning in Computer Vision, pp 1–22.
6. van de Sande KEA, Gevers T, Snoek CGM (2010) Evaluating Color Descriptors for Object and Scene Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 32:1582–1596.
7. Swain MJ, Ballard DH (1991) Color indexing. International Journal of Computer Vision 7:11–32.
8. Leow WK, Li R (2001) Adaptive binning and dissimilarity measure for image retrieval and classification. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp 234–239.
9. Rubner Y, Tomasi C, Guibas LJ (1998) A Metric for Distributions with Applications to Image Databases. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp 59–66.
10. Pele O, Werman M (2008) A Linear Time Histogram Metric for Improved SIFT Matching. In: Proceedings of the European Conference on Computer Vision(ECCV), pp 495–508.
11. Pele O, Werman M (2009) Fast and robust Earth Mover's Distances. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp 460–467.
12. Wengert C, Douze M, Jégou H (2011) Bag-of-colors for improved image search. In: Proceedings of the ACM International Conference on Multimedia (ACM MM), pp 1437–1440.
13. Bossard L, Dantone M, Leistner C, et al (2012) Apparel Classification with Style. In: Proceedings of the Asian Conference on Computer Vision (ACCV), pp 1–14.
14. Cheng CI, Liu DSM (2008) An intelligent clothes search system based on fashion styles. In: Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC), pp 1592–1597.
15. Yamaguchi K, Kiapour M, Ortiz L, et al (2012) Parsing clothing in fashion photographs. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp 3570–3577.
16. Liu S, Song Z, Liu G, et al (2012) Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp 3330–3337.
17. Rother C, Kolmogorov V, Blake A (2004) "GrabCut": interactive foreground extraction using iterated graph cuts. In: Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques, pp 309–314.
18. Schapire RE (1990) The Strength of Weak Learnability. Machine Learning 5:197–227.
19. Breiman L (2001) Random Forests. Machine Learning 45:5–32.
20. Sun HM (2010) Skin detection for single images using dynamic skin color modeling. Pattern Recognition 43:1413–1420.
21. Kakumanu P, Makrogiannis S, Bourbakis NG (2007) A survey of skin-color modeling and detection methods. Pattern Recognition 40:1106–1122.
22. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society Series B (Methodological) 39:1–38.
23. Boykov Y, Jolly MP (2001) Interactive graph cuts for optimal boundary amp; region segmentation of objects in N-D images. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on, volume 1, pp 105–112 vol.1.
24. Heckbert P (1982) Color image quantization for frame buffer display. In: Proceedings of the ACM SIGGRAPH International Conference on Computer Graphics and Interactive Techniques, pp 297–307.
25. Viola P, Jones MJ (2004) Robust Real-Time Face Detection. International Journal of Computer Vision 57:137–154.
26. Porikli F (2005) Integral histogram: a fast way to extract histograms in Cartesian spaces. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp 829–836.
27. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp 886–893.
28. Lowe DG (2004) Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60:91–110.
29. Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 24:381–395.
30. Kuettel D, Ferrari V (2012) Figure-ground segmentation by transferring window masks. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp 558–565.
31. Xiao J, Hays J, Ehinger K, et al (2010) SUN database: Large-scale scene recognition from abbey to zoo. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp 3485–3492.
32. Hsu CW, Lin CJ (2002) A comparison of methods for multiclass support vector machines. Neural Networks, IEEE Transactions on 13:415–425.