# Using Skin Segmentation to Improve Similar Product Recommendations in Online Clothing Stores

Noran Hasan, Ahmed Hamouda, Tamer Deif , Motaz El-Sabban and Ramy Shahin
*Microsoft Research Advanced Technology Labls in Cairo, 306 Cornish El Maadi, Basatin District, Cairo, Egypt*

Keywords: Skin Segmentation, Image Matching and Retrieval.

Abstract: Image matching and retrieval in the domain of clothing, as used in online shopping for recommending similar products, is often distracted by the existence of a mannequin/model wearing the product. The existence of a model adds clutter to both the shape and color features of the product. In this paper, we propose a novel image pre-processing pipeline that minimizes skin and background segments generated from generic GraphCut segmentation. Experiments judged by human subjects show very promising gains of around 23% in retrieval precision of the top 25 similar products compared to the baseline system.

## 1 INTRODUCTION

Online shopping allows consumers to browse and purchase products online. Online stores often rely on images, some textual descriptions, and sometimes videos, to represent and showcase products. To help consumers find products that better match their needs, or merely to expose to them other options, online stores try to recommend to consumers products that are similar to the ones they are browsing.

Recommending similar products requires a metric that would capture product similarity. For some products, such as clothing, how the product looks is an integral factor in the consumer's decision to purchase, therefore the visual similarity between products is an important dimension of product similarity. It is reasonable to assume that consumers would be interested in clothing items that are similar in form and/or color to the items they are browsing. That is in contrast to other products, such as books, where the visual similarity of products is insignificant.

There are a number of challenges in image retrieval such as complex backgrounds, viewpoint variation, etc. A challenge in the domain of clothing that we focus on in this paper is that the items are sometimes worn by a model in the product's image. Therefore, the visual representation of the product, including shape and color, would be cluttered by the model's shape and skin color. Consequently, the recommended products will be affected by whether or not these products are worn by a model in their images.

There is little work done on retrieval of clothing images. Recently, Grana et al, 2012, have presented a color-based technique for fashion-retrieval. They have a pre-processing step to remove skin or mannequin parts but they do not provide details of their skin removal technique or an analysis of the effect of skin removal on retrieval. Our contributions include:

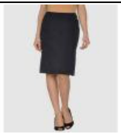
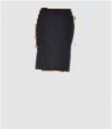
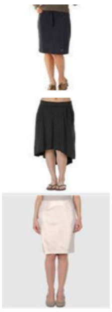
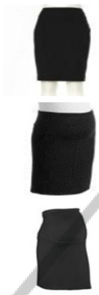
- Proposing a novel skin region removal pipeline tailored to enhancing visual product search quality
- conducting experiments and reporting very promising evaluation results on a large image dataset of 1 million product images from a commercial search engine

To motivate the work in this paper, we illustrate in Table 1 how the top results of image matching are different for the exact same product with and without a human model. With a model, most of the results returned also include a model. When the model was manually removed from the image, the results returned were mostly images that do not contain a model. This supports the assumption that the existence/absence of a model influnces the image matching system's decision that should ultimately measure similarity based on the products' visual features independent of a possibly existing model.

In this paper, we present a technique that uses skin detection to identify body parts and

Hasan N., Hamouda A., Deif T., El-Sabban M. and Shahin R..
Using Skin Segmentation to Improve Similar Product Recommendations in Online Clothing Stores.
DOI: 10.5220/0004299906930700
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 693-700
ISBN: 978-989-8565-47-1

Table 1: Bias of results to existence/absence of a model.

| | With a model | Without a model |
|---|---|---|
| Query Image |  |  |
| Top Results |  |  |

automatically remove those before the visual descriptors are extracted.

In Section 2, we refer to some related work in the fields of image retreival, image indexing, fashion retrieval, and skin detection. In Section 3, we describe the system into which we apply skin removal and in Section 4, we describe our skin removal technique. In Section 5, we present our judging methodology and results. Finally in Section 6, we conclude the paper and discuss future work.

## 2 RELATED WORK

Although there is a lot of work being done on image retrieval in general, there is little work done on the specific domain of clothing retrieval. Recently, Grana et al, 2012, presented their work on fashion retrieval based solely on color using a color bag of words signature. They describe garments by a single dominant color and therefore focus only on images with a unique color classification. Arguing that uniform color space division and color space clustering don't reflect fashion color jargon, they use color classes that label garments in their training set to split the color space in a way that minimizes error between these color classes. They use automatic pre-processing to remove skin and mannequin parts and then use GrabCut (Rother et al., 2004) to remove clothing items that are not the main garment depicted in an image. However, they don't provide a description of their skin removal approach in this pre-processing step and its impact on retrieval.

Skin detection has been approached by researchers with different methodologies including explicit color space thresholding and histogram models with naïve Bayes classifiers which we discuss later (Kakumanu et al., 2007). However, we noticed that the precision of most of the proposed techniques is not high. That is mainly because those techniques depend on analysing the images in the visible color spectrum without any attention to the context. This is not optimal because many factors (like illumination, camera characteristics, shadows, makeup, etc…) affect the skin color significantly. A workaround is to move the problem to the non-visible color spectrum (Infra-red range), in which the skin color seems to be more consistent across different conditions. However, the equipment needed is more expensive and usually not available in consumer devices.

## 3 EXISTING SYSTEM

We integrate our skin removal component in an existing clothing retrieval system (running on a commercial search engine) which we briefly describe in this section. Figure 3 shows a high-level overview of the system. In the coming subsections, we briefly describe the features extracted. In the following section, we describe our skin removal component and how it fits in this system.

The features generated for each image are contours to capture shape, and a single RGB value that captures the most dominant color. The image indexing and retrieval system is based upon the Edgel index by Cao et. Al, 2011.

### 3.1 Visual Representation

When a query image is submitted, a list of candidate similar edges is retrieved from an inverted index (Sivic and Zisserman, 2003). This list is ranked based on a composite score of the edge similarity, salient color similarity, and textual description similarity. Our interest is in improving the edge similarity score by removing unwanted edges and therefore improving this metric's semantic quality. By removing such edges, we also potentially impact the salient color extracted as motivated in figure 2.

#### 3.1.1 Image Pre-processing

To reduce computation and storage costs while preserving information, the image is first downsized to a maximum dimension of 200 pixels (Cao et. Al, 2011). The downsized image is then segmented using GraphCut (Felzenszwalb and Huttenlocher, 2004). The output is a segmented image where each segment is colored by the mean color of its pixels (see figure 1). It is then passed on to the salient color extractor. It is also converted to grayscale and passed on to the edge detection component.

### 3.1.2 Edge Detection

The grayscale segmented image is fed to a Canny detector to find contours. Contours are broken up at inflection points and very short edges are discarded. These edges are then used to create an inverted-index for the images. This inverted-index is used to retrieve the most similar images in terms of shape. The salient color is then used to rank the returned results.

### 3.1.3 Salient Color Extraction

When the image is segmented, the output image of segmentation is colored by the mean color of the pixels inside this segment. After edge detection, and based on the assumption that the background is a simple and homogeneous, the foreground bounding box is determined by the minimum box that encompasses all detected edges. Once this bounding box is found, an RGB histogram is created for this area of the segmented RGB image. The top color in the histogram is considered the salient color. To further avoid the background color, that is usually white in this domain, if the top color is white, the next color is considered the salient color.

## 3.2 Index Generation

An inverted-index based on edges is used for fast retrieval of similar images. It is out of the scope of this paper to delve into the details of this component, but what we like to stress is the importance of the edges and their direct impact on the quality of similar images retrieved. Therefore, our work explained in the next section attempts to remove unwanted edges before they are fed to the index generation component.

# 4 SKIN REMOVAL METHODOLOGY

The purpose of our work is to eliminate from products' images edges that define the model's body rather than the product itself. Since edge detection is performed on the segmented image as explained in Section 3, we need to blend the skin segments with background segments so that the edge detector doesn't detect these unwanted edges. Figure 3 shows where our contribution is integrated to the system. Figure 5 shows an overview of our skin removal technique. Figure 4 shows an examples on how the edges changed with skin removal.

In the following sub-sections, we explain why we chose this point of integration, how we do the skin to background blending, issues that we faced, and how we handled them.

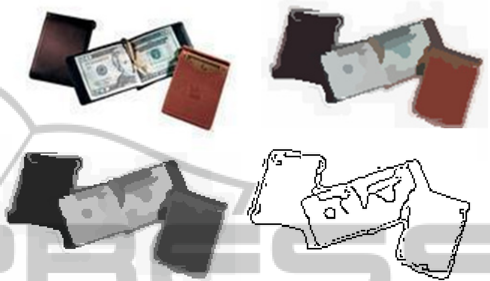


Figure 1: Stages leading to edge extraction. Top-left: The input image. Top-right: The segmented image with each segment colored by the average color of its pixels. Bottom-left: Grayscale of segmented image. Bottom-right: Edges detected by the Canny detector.

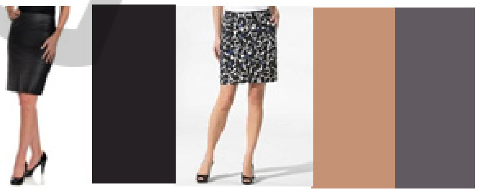


Figure 2: Two examples of salient colors extracted. The example on the left shows an accurate salient color, and the example on the right shows how the model's skin dominated the salient color, and a better salient color was produced after skin removal.

## 4.1 Skin Detection

The majority of state-of-the-art techniques rely on color for skin detection. Leading approaches either use explicit color space thresholding or model skin color within the color space. Therefore, we investigated two skin detection techniques that where suggested in the excellent survey by Kakumanu et al, 2007; explicit thresholding in the YCbCr color space, and histogram modelling with a naïve Bayes classifier. In the first technique, the image is converted into the YCbCr color space, and then a threshold is used to classify skin vs. non-skin pixels. The ranges we used were applied on the Cb and Cr components (Cb=[77,127], and Cr=[133,173]).

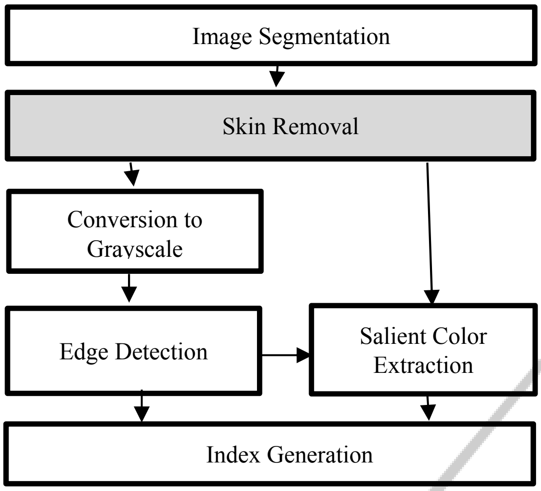


Figure 3: Overview of the system after skin removal integration.

As described in (Kakumanu et al, 2007) and (Jones and Rehg, 2002), we developed a histogram model with naïve Bayes classifiers. The idea is to approximate the probability distributions of color for skin pixels and for non-skin pixels. Training images are first converted to the Lab color space and then a 3D bin model is built by clustering the pixels according to their L, A, and B values into 64 clusters. The centroid of each cluster is chosen as its representative. The next step is creating a histogram model by counting the pixels that fall in each bin. This was done by measuring the Euclidean distance between each training pixel and the 64 clusters and then assigning it to the closest cluster. Once the model is ready, it is used to classify pixels in the test images. For each pixel, the probabilities of it being a skin vs. a non-skin pixel are compared and it is classified accordingly.

To evaluate the different skin detection techniques, we manually labelled 265 images, trained the histogram model on 185 images and used 80 for evaluation. The dataset includes challenging samples where the product color is similar to skin colors (e.g. beige, brown). Figure 7 shows samples of the data and how they were labelled.

To blend skin with the background, skin pixels need to be identified. When skin classification is done on the pixel-level on the original image before segmentation, misses can lead to non-homogenous areas (see middle of figure 6) that would result in more unwanted edges being detected. However, classifying all pixels in a segment collectively ensures that no new unwanted edges are introduced by skin removal (see right of figure 6).

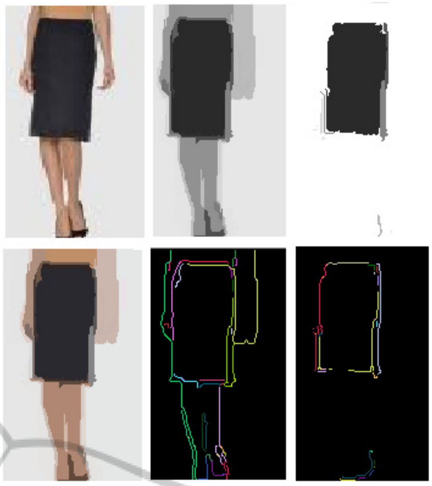


Figure 4: Effect of skin removal on detected edges (colors denote different contour segments). Top left: Input image. Bottom left: Segmented colored image. Middle top: Grayscale of segmented image. Middle bottom: Edges detected without skin removal. Right top: Segmented image after removing skin segments. Right bottom: Edges detected after removing skin segments.

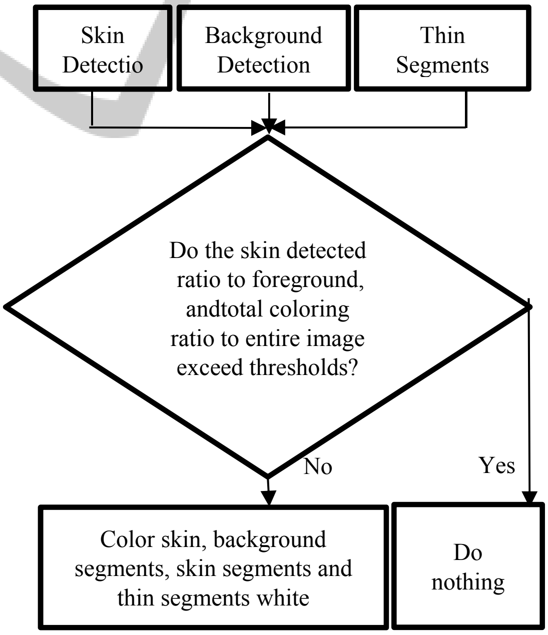

Skin Detectio
Background Detection
Thin Segments
Do the skin detected ratio to foreground, andtotal coloring ratio to entire image exceed thresholds?
No
Yes
Color skin, background segments, skin segments and thin segments white
Do nothing

Figure 5: Overview of skin removal technique.

Moreover, segment-level classification is much faster since all pixels in a segment only need a single classification, rather than once for each pixel (or for each possible RGB value in the image). Therefore, we found that it makes more sense to perform skin detection and removal on the segmented image rather than the raw image.

Table 2: Comparison of explicit  thresholding and training our own histogram model for skin detection.

| | Explicit Threshold | Trained Histogram Model, threshold at: | | |
|---|---|---|---|---|
| | | 0.5 | 1.5 | 2.5 |
| Precision | 24% | 29% | 39% | 60% |
| Recall | 99% | 65% | 36% | 6% |
| Accuracy | 79% | 87% | 92% | 93% |
| False Positive Rate | 76% | 71% | 61% | 40% |

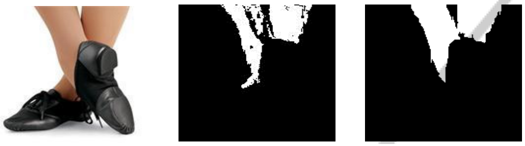


Figure 6: Left: Input image. Middle: White areas show per-pixel positive skin classifications. Right: White areas show per-segment positive skin classifications.

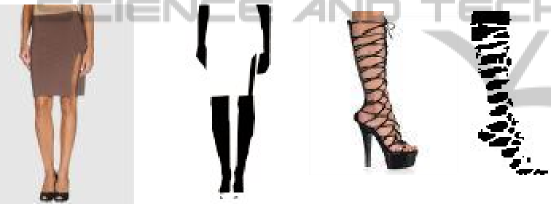


Figure 7: Examples of images labelled for skin.

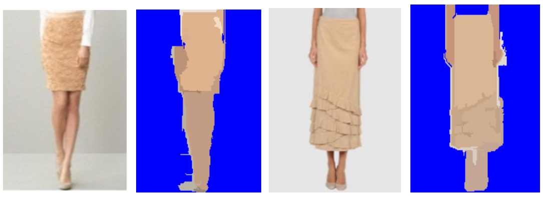


Figure 8: Blue areas mark segments classified as background.

## 4.2 Skin Segment Coloring

To prevent skin edges from being detected, we need to blend skin segments with the background. Intuitively, we considered coloring skin segments with the same color as the closest background segment. The measure of proximity can be done in a number of ways. We tried using the distance between segments' centroid as the measure of proximity but that sometimes resulted in that the nearest segment selected is not adjacent to the skin segment. Therefore, the segment edges are still visible. Even when that problem doesn't occur, the background that appears homogenous may include slightly different colors, and therefore when each segment is colored by the color of its nearest background segment, some edges are desirably lost but other edges persist (see figure 9). Therefore, even if the proximity measure ensures that the segments are adjacent, the results would still be unsatisfactory.

### 4.2.1 Background Detection

Due to the nature of online clothing images, the vast majority of images have simple backgrounds. Taking advantage of that and to avoid more complex background detection algorithms, such as GrabCut (Rother et al, 2004), we devised a simple algorithm to detect the background. Since, the image is already segmented, we classify a segment as background if it intersects with the image border. This approach has demonstrated satisfactory results (see figure 8).

Consequently, we decided to color all the skin and background segments by the same color. However, this approach surfaced a problem with thin segments that we discuss next.

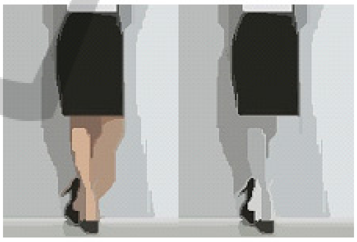


Figure 9: The effect of coloring a skin segment by the color of the nearest background segment in terms of distance between segment centroids.

### 4.2.2 Thin segments

We noticed that in some images, there are thin spurious segments that exist between skin and background segments that have been missed by both skin and background detectors. Although they are initially not very visible, they become very visible when the skin and background segments are colored white (see figure 10). To classify a segment as thin or not, we first need to identify its bounding box as the minimum and maximum values it reaches in the x and y dimensions. We then calculate the area of the segment's bounding box as well as the actual area of the segment which is the count of pixels that belong to this segment. We then define the area ratio which aims to capture curved or diagonal thin segments as:

$$\text{Area ratio} = \frac{\text{Actual area}}{\text{Bounding box area}} \quad (1)$$

We also calculate the segments' elongation which aims to capture thin, fairly straight thin segments that are vertical or horizontal. We define it as the ratio between the horizontal and vertical dimensions:

$$\text{Elongation} = \frac{\text{maximum}(\text{width}, \text{height})}{\text{minimum}\ (\text{width}, \text{height})} \quad (2)$$

A segment is classified as thin if its elongation is greater than 10 or its area ratio is less than 20%.

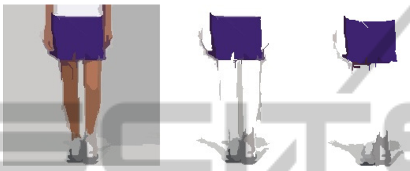


Figure 10: Left: Segmented colored image image. Middle: After skin removal and without handling thin segments. Right: After removing thin segments.

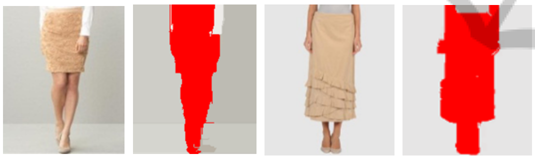


Figure 11: Two examples of products that are very similar in color to skin. The red areas mark those erroneously classified as skin.

### 4.3 False-positive Handling

Some products have a color that is very close to skin color and are therefore erroneously classified as skin (see Figure 11). To overcome this issue, we make sure that skin removal is only performed if the percentage of the foreground detected as skin is less than a certain threshold. With some experimentation we found that 70% is a reasonable threshold.

To combat cases of where background and thin segments may have been falsely detected parts of the garment, we also check that the total coloring does not exceed 90% of the entire image. If this threshold is exceeded, skin removal is skipped for that image.

## 5 RESULTS

### 5.1 Data Selection

To evaluate the effectiveness of our skin removal approach in improving the visual relevance of the retrieved product matches, three product categories are selected: jackets, dresses and skirts. For each category, a representative sample of twenty images is selected to be used as queries, for which the visually-similar matches from the same category are retrieved. The query is issued against a database of around 24,000 dresses, 49,500 jackets and 32,500 skirts. For each of the selected queries, the top 25 matches are retrieved. A total number of 1500 (3 categories * 20 queries * 25 matches) pairs are judged as described later in the following section. To select the best thresholds, skin coloring techniques, etc., multiple experiments were run and their results were evaluated. It is worth noting that the number of pairs to be judged exceeded the total number above because of evaluating multiple experiments. Each experiment results in the generation of a new index which potentially results in new matches per query.

### 5.2 Judging Process

In an effort to better simulate the consumer's online shopping experience, five human judges were asked to judge the similarity between each product and its corresponding top matches. All judges were assigned exactly the same set of pairwise product-matches from the three product categories mentioned above. Each judge is presented with a pairwise comparison between a product and a visually-similar candidate match. The pairs are presented in a random order from all categories. The judge is asked to rate each pair, based on the visual similarity, on a scale of 1 to 4 where: (1 = Very Different, 2 = Somewhat Different, 3 = Somewhat similar, 4 = Very Similar). Figure 12 illustrates the description of each rating along with example pairs.

### 5.3 Evaluation Technique

The goal of the evaluation process is to measure the amount of improvement achieved due to adopting the skin removal approach in the index generation process. As mentioned before, we have tried several variations of our skin removal technique and for each, a new index is generated. Once a new index is ready, each of the selected twenty queries per category is issued to get the top 25 matches. The new pairs then go into the judging cycle. Each pair gets five rates from the judges. We define the Average Judging Rate (AJR) as the average of all rates provided by the judges. It is worth noting that each pair gets exactly a single rate per judge, regardless of index that resulted in that rate.

The evaluation is done by means of precision that is defined as:

$$\text{Precision} = \frac{\text{TP}}{TP + FP} \quad (3)$$

where TP (True Positive) represents the number of similar products retrieved by the system and got an AJR between 2 and 4. FP (False Positive) represents the number of similar products retrieved by the system and got an AJR less than 2. The preferred metric is precision because it captures the user experience in terms of whether the retrieved results are relevant or not. Recall, however, is not used because it is unfeasible to comprehensively evaluate all matches of each query against the entire database.

Using the evaluation scheme above, an initial version of the system (without adopting any skin removal method) is evaluated and considered the baseline. Consequently, each skin removal experiment conducted is evaluated and compared to the baseline to measure the amount of improvement.

## 5.4 Results

Tables 3 and 4 show the top 5 and 25 matches, respectively, for different skin coloring techniques. We compare 3 different techniques:

- *Nearest Background Segment Color (NBSC):* Coloring skin segments by the color of its nearest background segment where the proximity measure is based on the distance between segment centroids
- *Dominant Background Color (DBC):* Coloring skin and background segments by the color of largest background segment.
- *Single Color (SC):* Color skin and background segments by a single new color (white).

The results show that the best technique is Single Color coloring where the average precision increased from the baseline precision 0.56 to 0.64 on the top 5 matches, and 0.43 to 0.53 on the top 25 matches. As discussed in section 4.3, coloring a segment by its nearest background segment does not remove all skin edges even if skin segments are correctly classified. That is because backgrounds that appear homogenous probably include multiple similar shades, and therefore some edges will persist (refer to figure 9). Dominant Background Color performs comparable to Single Color in the top 5 matches but is worse in the top 25 matches. It is possible that the dominant color negatively influences the salient color, while white doesn't (refer to section 3.2).

To visually demonstrate the impact of skin removal on retrieval, figure 13 shows an example of the top 4 matches for a query dress image before and after using our best skin removal technique using Single Color coloring. Table 5 shows the AJRs for both the baseline and post-skin-removal matches. The precision for the top 4 matches is shown calculated as explained in section 5.3 where an AJR of 2 or more is considered a TP, and an AJR below 2 is considered a FP. In this particular example, the precision went up from 0.25 to 0.75.

Table 3: Comparison of the precision of the top 5 matches for different skin segment coloring techniques.

| | Dresses | Jackets | Skirts | Average |
|---|---|---|---|---|
| Baseline | 0.54 | 0.62 | 0.51 | 0.56 |
| NBSC | 0.54 | 0.70 | 0.58 | 0.61 |
| DBC | 0.62 | 0.68 | 0.60 | 0.63 |
| SC | 0.64 | 0.68 | 0.60 | 0.64 |

Table 4: Comparison of the precision of the top 25 matches for different skin segment coloring techniques.

| | Dresses | Jackets | Skirts | Average |
|---|---|---|---|---|
| Baseline | 0.37 | 0.52 | 0.39 | 0.43 |
| NBSC | 0.44 | 0.57 | 0.48 | 0.49 |
| DBC | 0.38 | 0.38 | 0.59 | 0.45 |
| SC | 0.50 | 0.58 | 0.51 | 0.53 |

Table 5: AJR for results in figure 13 and the precision for the top 4 results.

| Rank | Baseline AJR | After skin removal AJR |
|---|---|---|
| 1 | 1.6 | 2 |
| 2 | 1.8 | 2 |
| 3 | 2 | 1.8 |
| 4 | 1.2 | 2.8 |
| Precision | 0.25 | 0.75 |

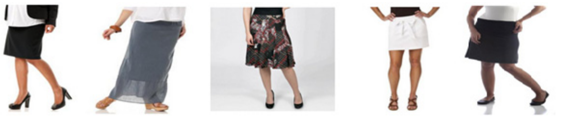


Figure 12: Left-to-right: An image of a query product, a very different product (very different form and color), a somewhat different product (different color and different form, but both color and form are somewhat similar), a somewhat similar product (either the form or color is similar, and the other is different)., a very similar (similar color and similar form).

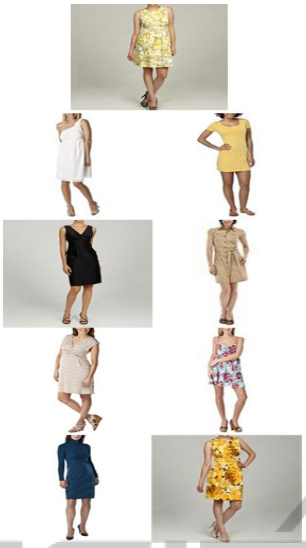


Figure 13: Example top 4 matches before and after skin removal. Top row: query image. Left column: baseline results. Right column: results after skin removal.

## 6 CONCLUSION & FUTURE WORK

We used skin removal to improve similar products retrieval in the domain of clothing. We tried our technique on 3 types of clothing; dresses, jackets and skirts. The results show improvement in precision as measured for up to the top 25 matches. Skin removal helped remove unwanted edges and improve salient color extraction, which in turn increased relevance.

Potential future work includes developing the technique to handle more complex cases such as complex backgrounds, multiple viewpoints of a product in the same image, and multiple products depicted in the same image, e.g. a top and skirt where the skirt is the product of interest. It is possible to leverage knowledge of the product category to better choose representative edges and better identify the region that has the color of interest. Also, the product metadata, such as color description, can be used to better localize the product region in the image. In addition, a more complex color descriptor can be devised to better describe products that consist of multiple colors.

## REFERENCES

Canny, J., 1983. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*

Cao, Y., Wang, C., Zhang, L., Zhang, L., 2011. Edgel Index for Large-Scale Sketch-based Image Search. *IEEE Conference on Computer Vision and Pattern Recognition.*

Datta, R., Joshi, D., Li, J., Wang, J. Z. 2008. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys.*

Felzenszwalb, P. F., Huttenlocher, D. P., 2004. Efficient Graph-Based Image Segmentation . *International Journal of Computer Vision.*

Grana, C., Borghesani, D., Cucchiara, R., 2012, Class-based Color Bag of Words for Fashion Retrieval. *IEEE International Conference on Multimedia and Expo.*

Jones, M. J., Rehg, J. M., 2002. Statistical Color Models with Application to Skin Detection. *International Journal of Computer Vision.*

Kakumanu, P., Makrogiannis, S., Bourbakis, N., 2007. A survey of skin-color modeling and detection methods. *Pattern Recognition.*

Martin, D. R., Fowlkes, C. C., Malik, J., 2004. Learning to detect natural image boundaries using local brightness, color, and texture cues. *Pattern Analysis and Machine Intelligence.*

Rother, C., Kolmogorov, V., Blake, A., 2004. GrabCut: interactive foreground extraction using iterated graph cuts. *ACM SIGGRAPH.*

Sivic, J., Zisserman, A., 2003. Video Google: a text retrieval approach to object matching in videos. *Proceesings of the 9th IEEE International Conference on Computer Vison.*