Augmentation of Skin Segmentation

Rehanullah Khan
PRIP, TU-Wien, Austria
Email: rehan@prip.tuwien.ac.at

Allan Hanbury
IRF, Vienna, Austria
Email: a.hanbury@ir-facility.org

Julian Stöttinger
PRIP, TU-Wien, Austria
Email: julian@prip.tuwien.ac.at

***Abstract*—Skin detection is used in applications ranging from face detection, tracking body parts and hand gesture analysis, to retrieval and blocking objectionable content. We present a systematic approach for skin segmentation with graph cuts by using local skin information, universal skin information and skin augmentation using the off-line learned model. The skin segmentation process starts by exploiting the local skin information of detected faces. The detected faces are used as foreground seeds for calculating the foreground weights of the graph. If local skin information is not available, we opt for a highly adaptive universal seed. To increase the robustness we learn a decision tree based classifier. The learned model is used to augment the universal seed for skin segmentation when no local information is available from the image. Experiments on a database of 8991 images with annotated pixel-level ground truth show that the systematic skin augmentation approach outperforms universal seed only approach, universal seed plus face detection approach and the learned decision tree based classifier (J48), showing its suitability for robust skin segmentation. We also report the behavior of a skin detection system in the presence of skin only images and images which do not contain skin at all.**

Keywords: Skin detection, segmentation, graph cuts, classification.

I. Introduction

Skin detection has a wide range of applications both in human computer interaction and content based analysis. Applications such as: detecting and tracking of human body parts [1], face detection [2], naked people detection, people retrieval in multimedia databases [3] and blocking objectionable content [4]; all benefit from skin detection.

The most attractive properties of color based skin detection are: potentially high processing speed, invariance against rotation, partial occlusion and pose change. However, standard skin color detection techniques are affected by changing lighting conditions, complex backgrounds and surfaces having skin-like colors.

The primary objective of skin detection or classification is building a decision rule that will differentiate between skin and non-skin pixels. The most widely used approach to identify skin colored pixels involves creating a static skin filter, a volume into which most skin pixels would fall in a given color space [5]. There is a set of techniques which estimate the distribution of skin color by a training phase. These methods are often referred to as non-parametric skin models [6]. Finally, other methods include parametric skin distribution models, such as the Gaussian skin color model [7].

Our contribution is as follows: We opt for a systematic approach for skin segmentation by exploiting the local skin information in terms of face seeds, the global skin information in terms of universal seed and the graph weights augmentation in terms of learned decision tree classifier (J48). We model the skin segmentation problem as a min-cut problem on a graph defined by the image color characteristics. The vertices of the graph represent the image colored pixels and edges represent weights or costs for labeling the vertices as skin or non-skin. The segmentation process is based on a skin seed or skin template patch. If a face is present in the image, we use it as a seed. If no face is detected, we load a universal seed. With the universal seed we successfully remove the need for local foreground seeds from an image. With the weight augmentation technique we improve the over-all skin segmentation performance.

The skin segmentation process is summarized in a block diagram, see Figure 1. If a face is detected we compute foreground histogram based on the face template. For the face seed/template generation we use the well known face detection approach from Viola Jones [8]. If a face is not detected we load a universal seed and compute the foreground histogram based on this universal seed. The background histogram is calculated from the whole image itself. The foreground and background histograms are used to create foreground and background weights. The foreground/background weights and the neighborhood weights are used to create a graph. If the universal seed is used, the foreground weights are augmented with the weights computed given by the probabilities of the decision tree based classifier (J48). A graph is constructed incorporating proper adjustment of the neighborhood weights and background/foreground weights. We use a very efficient algorithm [9] for finding min-cut/max-flow in a graph, finally segmenting the desired skin.

Experiments are performed and the augmentation technique is compared to the universal seed only approach, the universal seed with face detection approach and the learned model (J48) only approach. The results show that the augmented approach is well suited to robust skin segmentation.

In Section II we summarize some of the related work regarding skin detection. Section III discusses the graph building process and the weights assignment. Section IV describes the creation of local and universal seeds and augmentation of the universal seed. Experimental details and the dataset used are given in Section V. Section VI concludes.

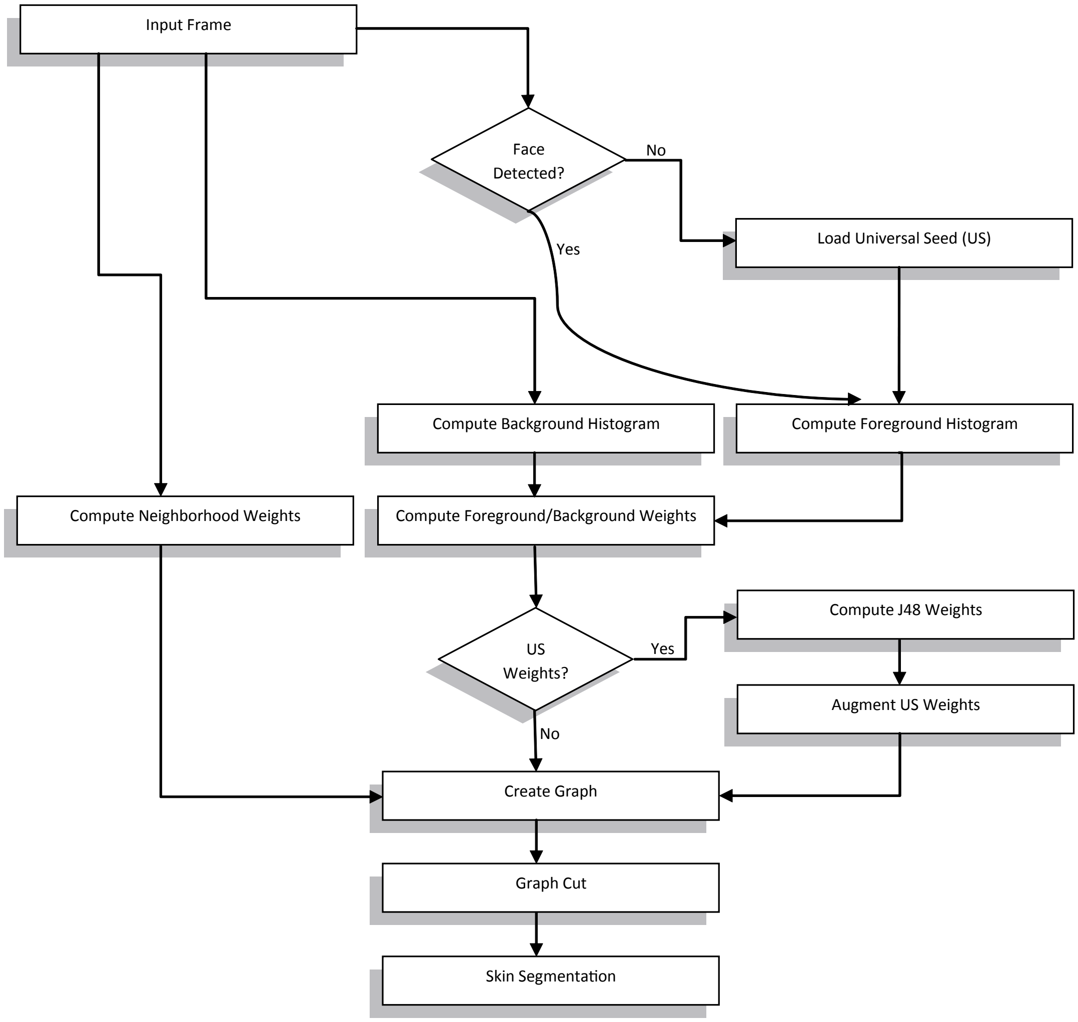


Fig. 1. Steps for skin segmentation.

II. Related Work

In computer vision, skin detection is used as a first step in face detection, e.g. [10], and for localization in the first stages of gesture tracking systems, e.g. [1]. It has also been used in the detection of naked people [11], [12] and for blocking objectionable content [4]. The latter application has been developed for videos, too.

The approaches to classify skin in images can be grouped into three types of skin modeling: parametric, non-parametric and explicit skin cluster definition methods. The parametric models use a Gaussian color distribution since they assume that skin can be modeled by a Gaussian probability density function [7]. Non-parametric methods estimate the skin-color from the histogram that is generated by the training data used [6].

An efficient and widely used method is the definition of classifiers that build upon the approach of skin clustering. This thresholding of different color space coordinates is used in many approaches, e.g. [13] and explicitly defines the boundaries of the skin clusters in a given color space, generally termed as static skin filters. The underlying hypothesis is that skin pixels have similar color coordinates in the chosen color space, which means that skin pixels are found within a given set of boundaries in a color space. The static skin filters in YCbCr and RGB color spaces are reported in [14] and [15] respectively.

Although static filters approach is extremely rapid, its main drawback is a comparably high number of false detections [16]. This problem is addressed in [17] by adapting the skin-color model according to reliably detected faces. When more than one detected face exists in a frame and face information is properly extracted, multiple adapted models are used. The multiple model approach makes it possible to filter out skin for multiple people with different skin tones and reduce its false positives. The dynamic multiple model approach outperforms static approaches.

Color is a low level feature that is computationally inexpensive. For many applications in computer vision, it is suitable for real-time object characterization, detection and localization [12]. The main goal of skin color detection or classification is to build a decision rule that will discriminate between skin and non-skin pixels. Following [16] the major difficulties in skin color detection are caused by various effects such as illumination circumstances, camera characteristics, ethnicity, individual characteristics and other factors like makeup, hairstyle, glasses, sweat, and background colors. An approach for reliably detecting skin has therefore to be stable against noise, artifacts and very flexible against varying lighting conditions.

Color spaces like the HS* family transform the RGB cube into a cylindrical coordinates representation. They have been widely used in skin detection scenarios, such as [18], [19], [20]. To simulate the primates visual attention, perceptually uniform color spaces like the CIELAB, CIELUV are used for skin detection e.g. in [2]. Orthogonal color spaces like YCbCr, YCgCr, YIQ, YUV, YES try to form as independent components as possible. YCbCr is one of the most successful color spaces for skin detection and used in e.g. [21], [10].

Skin detection under varying illumination in image sequences is addressed in [22], [23], [24]. These approaches try to map the illuminance of the image into a common range. They compensate for the variance of changing lighting to equalize the appearance of skin color throughout different scenes. These methods are dependent heavily on the lighting correction techniques and their ability to estimate the illuminant.

Neural networks [25], Bayesian Networks e.g. [26], Gaussian classifiers e.g. [6], and self organizing maps [18] have been used to try to increase the classification accuracy.

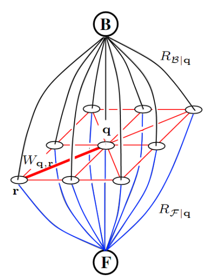
In the literature of segmentation, Graph-cuts provide a globally optimal solution for N-dimensional segmentation when the cost function has specific properties as defined in [27]. A semi-automatic method for general image segmentation is created in [27]. A user puts marks on the image, acting as a cue for being counted as segments and updating the marks without graph reconstruction. The method in [28] consists of two steps: an object marking task as in [27] and the pre-segmentation, followed by a simple boundary editing process. [29] segments the image into many non-overlapping regions introducing normalized graph cuts. The method has often been used in combination with computing pixel neighborhood relations using brightness, color and texture cues [30], [31], [32].

In graph cuts, background and foreground seeds are usually specified from within an image. An assumption is made in [33] that each textured or colored region can be represented by a small template, called the seed and positioning of the seed across the input image gives many possible sub-segmentations of the image. A probability map assigns each pixel to just one most probable region and produces the final pyramid representing various detailed segmentations. Each sub-segmentation is obtained as the min-cut/max-flow in the graph built from the image and the seed. In our work if available, we use detected faces as seeds for skin segmentation.

Graph cuts is used for skin segmentation in [34] using both foreground and background seeds. In the form of faces, we only use the foreground local seeds. In absence of local seed information, we introduce the concept of a global universal seed thereby removing the need for strict requirement of local seeds from with in an image. Additionally we also augment the weights calculated based on universal seed and thus improving the over-all segmentation performance.

III. Graph Representing the Skin Image

We use a skin segmentation technique based on an interactive graph cut method as used in [33] and [27]. Before segmentation we construct a graph. The graph is shown in Figure 2 for a 9 pixel image and 8-point neighborhood N with representative pixels q and r. For an image, the number of graph nodes equals the pixel count plus two extra nodes labeled as F, B representing foreground and background respectively. There



edge	cost	region
$\{\mathbf{q},\mathbf{r}\}$	$W_{\mathbf{q},\mathbf{r}}$	$\{\mathbf{q},\mathbf{r}\} \in \mathcal{N}$
$\{\mathbf{q},F\}$	$\lambda R_{\mathcal{F}\|\mathbf{q}}$	$\forall \mathbf{q}$
$\{\mathbf{q},B\}$	$\lambda R_{\mathcal{B}\|\mathbf{q}}$	$\forall \mathbf{q}$

Fig. 2. Left: Graph representation for 9 pixel image. Right: Table defining the costs of graph edges. λ defined in the text.

are two types of weight to be set, the foreground/background (skin/non-skin) weights and the neighborhood weights. If a face is detected, the foreground weights are calculated from the face seed. If no successful face is detected we load a universal seed the foreground weights are computed based on the universal seed. The background weights are calculated from the whole image itself. For min-cut/max-flow a very efficient algorithm [9] is used. An example of skin segmentation based on graph cut and its principle is shown in Figure 3.

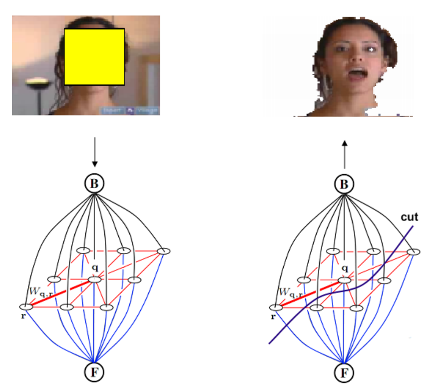


Fig. 3. An example of skin segmentation if local skin information (in terms of faces) is available. In case of unavailability of local information, we load a universal seed and base of our segmentation on the universal seed. Note that hair is detected as skin because the seed covers the hair portion.

A. Neighborhood Weights

For a black and white image the neighborhood weights (weight matrix) $W_{q,r}$, as reported by [27] are

$$W_{q,r} \propto e^{-\frac{||I_q - I_r||^2}{2\sigma^2}} . \frac{1}{||q-r||} \tag{1}$$

where I_q and I_r are the intensities at point q and point r, $||q-r||$ is the distance between these points and σ is a parameter. For color images we modify the above function to take color into account as in [35] obtaining,

$$W_{q,r} = e^{-\frac{||c_q - c_r||^2}{\sigma_1}} . \frac{1}{||q-r||} \tag{2}$$

where c_q and c_r are the YCbCr vectors of points at the position q and r. $||q-r||$ is the distance between these points and σ_1 is a parameter. For skin detection purpose a value of σ_1 = 0.02 is used, which is the optimized value for segmentation in [35] and is obtained experimentally for giving the best performance on a large database of images. We are using a neighborhood window of size 21×21. For skin segmentation we are using a sampling rate of 0.3. This means that we are only selecting at random 30% sample of all the pixels in the window. There are two reasons: Firstly by using only a fraction of pixels we are reducing the computational demands and secondly only a fraction of pixels allows the use of larger windows and at the same time preserve the spatial relationship between the neighboring pixels.

B. Foreground/Background Weights

For foreground/background weights, the regional penalty of a point as being "skin" (foreground) $\mathcal{F}$ or "non-skin" (background) $\mathcal{B}$ [35] is

$$R_{\mathcal{F}|q} = -\ln p(\mathcal{B}|c_q) \quad and \quad R_{\mathcal{B}|q} = -\ln p(\mathcal{F}|c_q) \tag{3}$$

where $c_q = (c_Y, c_{Cb}, c_{Cr})^T$ stands for a vector in $\mathbb{R}^3$ of YCbCr values at the pixel q. The posterior probabilities in Equation 3 are computed as follows,

$$p_{(\mathcal{B}|c_q)} = \frac{p(c_q|\mathcal{B})p(\mathcal{B})}{p(\mathcal{B})p(c_q|\mathcal{B}) + p(\mathcal{F})p(c_q|\mathcal{F})} \tag{4}$$

For the skin segmentation problem we first demonstrate it on $p(\mathcal{B}|c_q)$, for $p(\mathcal{F}|c_q)$ the steps are analogous. Initially we fix $p(\mathcal{F}) = p(\mathcal{B}) = 1/2$ and thus,

$$p_{(\mathcal{B}|c_q)} = \frac{p(c_q|\mathcal{B})}{p(c_q|\mathcal{B}) + p(c_q|\mathcal{F})} \tag{5}$$

where the "skin" and "non-skin" prior probabilities are

$$p(c_q|\mathcal{F}) = f^Y_{c_Y} . f^{Cb}_{c_{Cb}} . f^{Cr}_{c_{Cr}} \quad and \quad p(c_q|\mathcal{B}) = b^Y_{c_Y} . b^{Cb}_{c_{Cb}} . b^{Cr}_{c_{Cr}} \tag{6}$$

and $f_i^{\{Y,Cb,Cr\}}$, resp. $b_i^{\{Y,Cb,Cr\}}$, represents the foreground, resp. the background histogram of each color channel separately at the ith bin. After smoothing $\omega_f = \lambda R_{\mathcal{F}|q}$, where ω_f is the foreground weight, λ is set to 1000 and controls the importance of penalties for foreground and background against the neighborhood weights. Similarly the background weight ω_b is given by $\omega_b = \lambda R_{\mathcal{B}|q}$.

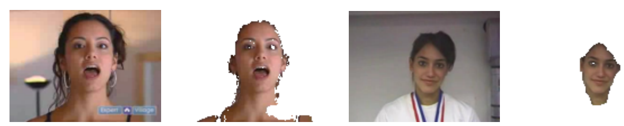
If at least one face is detected, the foreground histogram is computed from the face area only. In case if no face is detected, the universal seed is used to estimate the foreground histogram. Since $\sum_{i=1}^{N} \bar{b}_i$=1, the probability $p(c_q|\mathcal{B})$ gives smaller values than $p(c_q|\mathcal{F})$ for the "skin" colors present in the universal seed therefore we compute the background histogram from all image pixels. Finally in case of universal seed the foreground weights are augmented.

IV. Seed Generation and Weight Augmentation

If we detect a face, we base our foreground weights on this local seed. If on the other hand no face is present, we opt for a universal seed. For robustness we augment the universal seed based weights.

A. Local Seed/Template Generation

We can incorporate the information provided by the automatic seed/template from with in the skin image. Manually a user can click on the image and select some small patch for skin from the image. We want to automate this process. For skin segmentation this is possible as we are using images having humans and thus probably faces visible in these images. We use the Viola Jones [8] face detector frame work for



(a) (b) (c) (d)

Fig. 4. Skin detection based on local (face) skin information.

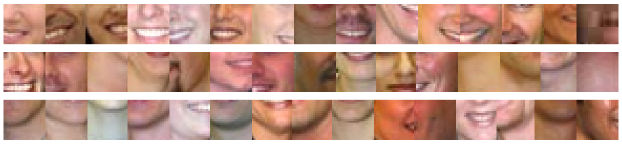
automatic seed generation from images. The face detector finds faces and returns a rectangle around the face. Skin detection based on face detection is shown in Figure 4.

The Viola-Jones object detection framework can be trained for a variety of objects, primarily motivated for the problem of face detection. The features used by the detection framework involve the sums of image pixels within rectangular areas [36]. With the use of the *integral image* as image representation, rectangular features can be evaluated in constant time, which gives them a considerable speed advantage.

The object detection framework employs a variant of the learning algorithm AdaBoost [37] to both select the best features and to train classifiers that use them. AdaBoost is sensitive to noisy data and outliers as it is using fast and simple classifiers and subsequent classifiers are adapted based on the success of previous classifiers. This results in processing gain, and can be embedded in real-time applications.

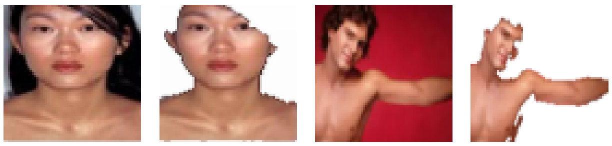
B. From Local Seed to Universal Seed

We investigate the following: Can we, without having a large amount of manually labeled ground truth, produce a seed/template that is as general as possible and can be used as a filter? We denote such a seed as the *universal seed*. We collect different skin tones from faces covering different ethnicities, see Figure 5. These images are not taken from the



(a)

Fig. 5. Skin samples used for universal seed.



(a) (b) (c) (d)

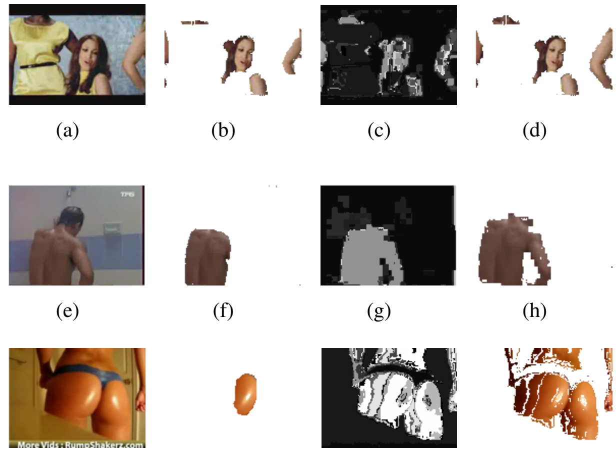
Fig. 6. Skin detection based on universal seed.

testing database. For the universal seed calculation, different faces/skin patches are aligned and a static filter is used to block the unwanted information which is mainly hair and in some cases the background information. A mask image is created which is pure skin patches extracted through the steps described. We denote it as the universal seed. The foreground histogram in Equation 6 for a new image is calculated based on this seed. Skin detection based on the universal seed is shown in Figure 6. The universal seed is highly adaptable as for a new patch to be added to universal seed we just have to merge it with skin patches. We augment foreground weights based on universal seed with the learned model.

C. Weight Augmentation Using Decision Trees (J48)

The popularity of tree classifiers is their intuitive appeal and easy training procedures. We use it because for our noisy data, the tree classifiers generalizes well and gives good over-all performance compared to the other algorithms (SVM, AdaBoost, Bayesian Network, Naive Bayesian, Multilayer Perceptron) that we tested for the same problem. J48 builds decision trees (binary trees) from a set of training data using the concept of information entropy [38]. Decision tree algorithms begin with a set of cases, or examples, and create a tree data structure that can be used to classify new cases. Each case is described by a set of attributes (or features) which can have numeric or symbolic values. Associated with each training case is a label representing the name of a class. Each internal node of a decision tree contains a test, the result of which is used to decide what branch to follow from that node. In case of a binary decision tree, if the test is true, then the case will proceed down the left branch, and if not then it will follow the right branch. The leaf nodes contain class labels instead of tests. In classification mode, when a test case (which has no label) reaches a leaf node, it is classified using the label stored there. The foreground probabilities are estimated based on the decision functions throughout the visited internal nodes. We train the classifier based on annotated pixel values in YCbCr color space.

A weight image is created by replacing the pixel values with



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

Fig. 7. Augmentation of Skin segmentation. First column: original images. Second column: Universal seed based skin segmentation. Third column: Weighted image. Fourth column: Skin Augmentation. (Best viewed in color)

the probability weights of skin pixels. These probabilities p_j are obtained from the J48 classifier. Then using the classifier probability a new weight $w_j = -\ln(p_j)\lambda$ is calculated, the final new weight to be assigned to the edge E is

$$w_{new} = (w_f + w_j) \tag{7}$$

where w_f is the foreground weight generated using universal seed. We denote this fusion of classification results as pixel weight augmentation. We add non-spatial based classification results of the J48 to the basis of spatial classification of the graph cuts. Augmentation of skin detection using universal seed is shown in Figure 7. The augmentation of skin weights is only applicable to the foreground weights and not to the neighborhood weights. The neighborhood weights are calculated as explained in the Section III-A and are not augmented.

V. Experiments

First we introduce the dataset used and then present the evaluation of universal seed skin segmentation, universal seed with face detection, learned model (J48) and the skin augmentation approach. We are also interested in evaluation of skin segmentation algorithm on images with skin and on images which do not contain any skin at all.

A. Data Sets

We used images extracted from 25 videos provided by an Internet service provider that requires a skin detection application for their on-line platform. The sequences contain scenes with multiple people and/or multiple visible body parts and scene shots both indoors and outdoors, with steady or moving camera, see Figure 8. The lighting varies from natural light to directional stage lighting. Ground truth has been generated for all of the 25 videos on a per pixel basis. The data set is available one-line[1]. We arrange the dataset into

[1]http://www.prip.tuwien.ac.at/people/julian/skin-detection

three sets. The first set contains 5817 images, every image of which contains some skin (referred to as skin-only images). The second set contains 3174 images, all the images of which are with out skin (referred to as non-skin images). The third set consists of 8991 images having both skin-only and non-skin images (referred to as hybrid images).

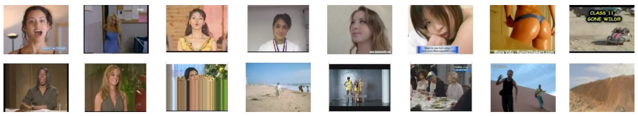


Fig. 8. Example frames from the annotated video data-set.

Our first experiment consists of evaluating the four approaches which are the universal seed only approach, the combination of universal seed and face detection, the decision tree classifier (J48) approach and the augmentation approach where we combine universal seed, face detection and augmentation using J48, on 8991 hybrid images. We use F-score and accuracy as a measure of evaluation. The F-score is calculated by evenly waiting precision and recall. As shown in Figure 9, the augmented technique has higher F-score than the other three approaches. Comparable high accuracy is shown in Figure 10 for augmentation technique, outperforming the three other approaches. Examples of skin detection based on the augmentation technique can be seen in Figure 7. The areas of skin in Figure 7, which were missed by universal seed are detected properly by the augmentation technique.

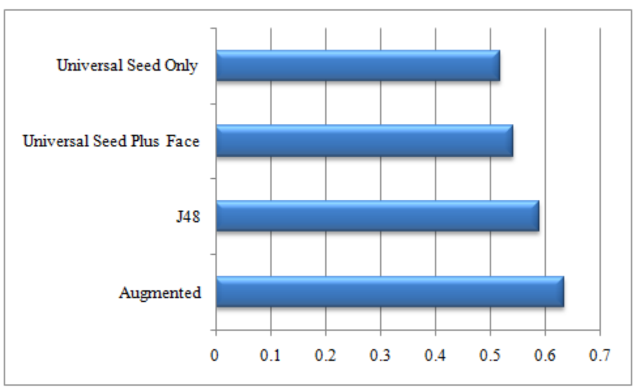


Fig. 9. Average F-score of 8991 images for universal seed, universal seed and face combination, J48 and the augmentation technique.

In the second experiment we are interested in investigating the behavior of skin detection algorithms when applied to skin-only and non-skin images. Such an evaluation is novel and valuable as other state-of-the-art approaches suffer from a high number of false positives. We also want to determine valid evaluation measures for skin-only and non-skin images. The evaluation measures given are Precision, Recall, F-score, Accuracy, and Specificity (true negative rate). As a skin detection algorithm, we are using the same augmentation technique described in the first set of experiments. Table I shows the evaluation measures for skin-only (5817 images) and non-skin (3174 images). For non-skin images the valid

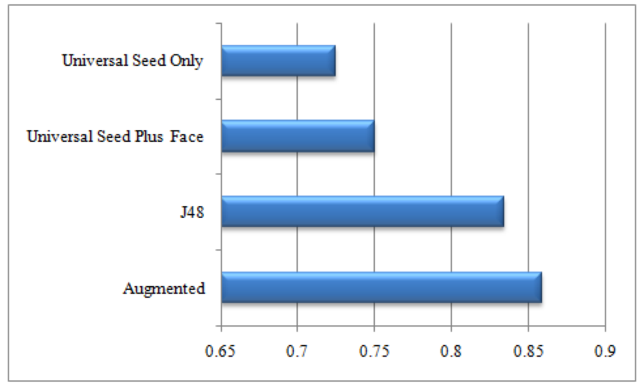


Fig. 10. Accuracy of 8991 images for universal seed, universal seed and face combination, J48 and the augmentation technique.

TABLE I
STATISTICS FOR SKIN-ONLY AND NON-SKIN IMAGES

Data	Precision	Recall	F-Score	Accuracy	Specificity
Skin-only	0.763	0.626	0.688	0.836	0.921
Non-skin	—	—	—	0.887	0.891

statistical parameters to be considered are accuracy and specificity. For skin-only images the most interesting parameters are the precision and recall, because the objective is to increase both precision and recall. In practice however the best can be achieved by a compromise between precision and recall. Figure 11 shows the precision and recall space for 5817 skin-only images. The dominant cluster can be found at the top right, reporting good over-all performance for these images.

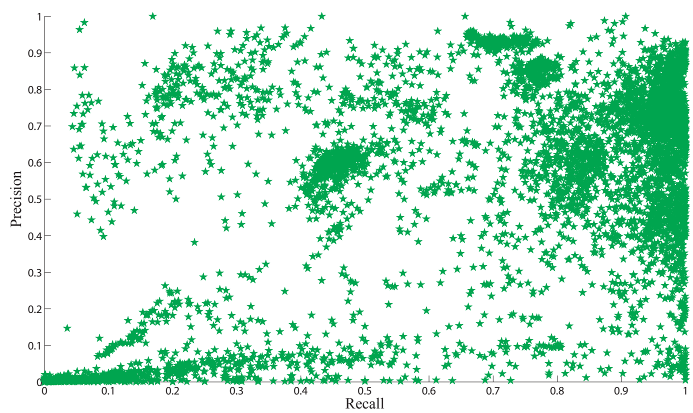


Fig. 11. Precision and recall space for 5817 skin-only images.

VI. Conclusion

We presented a systematic approach for skin segmentation using graph cuts by exploiting local skin information, universal skin information and skin augmentation using the learned model. The detected faces are used as foreground seeds for the graph. If local skin information is not available, we opt for a universal seed approach. To increase robustness we use the learned decision tree based classifier for augmenting the universal seed based foreground weights. Experiments on a database of 8991 images with annotated pixel-level ground truth show that the systematic approach is well suited to the skin segmentation. A robust skin detection should not only detect skin when it is present, but should detect nothing in images without skin. We pay attention to this in our evaluation by calculating evaluation measures on two subsets dividing the test dataset into skin-only and non-skin images. The suitability of specificity and accuracy as a measure for datasets containing non-skin images is demonstrated.

References

[1] Antonis A. Argyros and Manolis I.A. Lourakis, "Real-time tracking of multiple skin-colored objects with a possibly moving camera," in *ECCV*, 2004, pp. 368–379.

[2] J. Cai and A. Goshtasby, "Detecting human faces in color images," *Image and Vision Computing*, vol. 18, pp. 63–75, 1999.

[3] Liang-Liang Cao, Xue-Long Li, Neng-Hai Yu, and Zheng-Kai Liu, "Naked people retrieval based on adaboost learning," in *MLC*, 2002, pp. 1133–1138.

[4] Julian Stöttinger, Allan Hanbury, Christian Liensberger, and Rehanullah Khan, "Skin paths for contextual flagging adult videos," in *ISVC*, 2009, pp. 303–314.

[5] Vladimir Vezhnevets, Vassili Sazonov, and Alla Andreev, "A survey on pixel-based skin color detection techniques," in *ICCGV*, 2003, pp. 85–92.

[6] Michael J. Jones and James M. Rehg, "Statistical color models with application to skin detection.," *IJCV*, vol. 46, no. 1, pp. 81–96, 2002.

[7] Minghsuan Yang and Narendra Ahuja, "Gaussian mixture model for human skin color and its application in image and video databases," in *SPIE*, 1999, pp. 458–466.

[8] Paul Viola and Michael J. Jones, "Robust real-time face detection," *IJCV*, vol. 57, no. 2, pp. 137–154, 2004.

[9] Yuri Boykov and Vladimir Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *PAMI*, vol. 26, no. 9, pp. 1124–1137, 2004.

[10] R.L. Hsu, M. Abdel-Mottaleb, and A.K. Jain, "Face detection in color images," *PAMI*, vol. 24, pp. 696–706, 2002.

[11] Margaret M. Fleck, David A. Forsyth, and Chris Bregler, "Finding naked people," in *ECCV*, 1996, pp. 593–602.

[12] Jiann-Shu Lee, Yung-Ming Kuo, Pau-Choo Chung, and E-Liang Chen, "Naked image detection based on adaptive and extensible skin color model," *PR*, vol. 40, no. 8, pp. 2261–2270, 2007.

[13] Som Lam Phung, Abdesselam Bouzerdoum, and Douglas Chai, "Skin segmentation using color pixel classification: Analysis and comparison," *PAMI*, vol. 27, no. 1, pp. 148–154, 2005.

[14] D. Chai and K.N. Ngan, "Locating facial region of a head-and-shoulders color image," in *Int. Conf. Automatic Face and Gesture Recognition*, 1998, pp. 124–129.

[15] Peter Peer, Jure Kovac, and Franc Solina, "Human skin colour clustering for face detection," in *EUROCON*, 2003, pp. 144–148, vol.2.

[16] P. Kakumanu, S. Makrogiannis, and N. Bourbakis, "A survey of skin-color modeling and detection methods," *PR*, vol. 40, no. 3, pp. 1106–1122, 2007.

[17] Rehanullah Khan, Julian Stöttinger, and Martin Kampel, "An adaptive multiple model approach for fast content-based skin detection in on-line videos," in *ACM MM, AREA workshop*, 2008.

[18] D. Brown, I. Craw, and J. Lewthwaite, "A SOM based approach to skin detection with application in real time systems," in *BMVC'01*, 2001, pp. 491–500.

[19] Zhouyu Fu, Jinfeng Yang, Weiming Hu, and Tieniu Tan, "Mixture clustering using multidimensional histograms for skin detection," in *ICPR*, Washington, DC, USA, 2004, pp. 549–552.

[20] C. Garcia and G. Tziritas, "Face detection using quantized skin color regions merging and wavelet packet analysis," *MM*, vol. 1, no. 3, pp. 264–277, Sep 1999.

[21] K.W. Wong, K.M. Lam, and W.C. Siu, "A robust scheme for live detection of human faces in color images," *SPIC*, vol. 18, no. 2, pp. 103–114, 2003.

[22] L. Sigal, S. Sclaroff, and V. Athitsos, "Skin color-based video segmentation under time-varying illumination," *PAMI*, vol. 26, no. 7, pp. 862–877, July 2004.

[23] Jie Yang, Weier Lu, and Alex Waibel, "Skin-color modeling and adaptation," in *ACCV*, London, UK, 1997, pp. 687–694, Springer-Verlag.

[24] M. Storring, H.J. Andersen, and E. Granum, "Estimation of the illuminant colour from human skin colour," *AFGR*, pp. 64–69, 2000.

[25] Jae-Young Lee and Yoo Suk-in, "An elliptical boundary model for skin color detection," in *ISST*, 2002, pp. 579–584.

[26] Nicu Sebe, Ira Cohen, Thomas S. Huang, and Theo Gevers, "Skin detection: A Bayesian network approach," in *ICPR*, 2004, pp. 903–906.

[27] Y. Y. Boykov and M. P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images," in *ICCV 2001*, 2001, vol. 1, pp. 105–112 vol.1.

[28] Yin Li, Jian Sun, Chi K. Tang, and Heung Y. Shum, "Lazy snapping," in *SIGGRAPH*, New York, NY, USA, 2004, pp. 303–308.

[29] Jianbo Shi and Jitendra Malik, "Normalized cuts and image segmentation," *PAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[30] Jitendra Malik, Serge Belongie, Thomas K. Leung, and Jianbo Shi, "Contour and texture analysis for image segmentation," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 7–27, 2001.

[31] Charless Fowlkes, David Martin, and Jitendra Malik, "Learning affinity functions for image segmentation: Combining patch-based and gradient-based approaches," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, pp. 54, 2003.

[32] David R. Martin, Charless C. Fowlkes, and Jitendra Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *PAMI*, vol. 26, no. 5, pp. 530–549, 2004.

[33] Branislav Micusík and Allan Hanbury, "Automatic image segmentation by positioning a seed," in *ECCV (2)*, 2006, pp. 468–480.

[34] Zhilan Hu, Guijin Wang, Xinggang Lin, and Hong Yan, "Skin segmentation based on graph cuts," *Science and Technology*, vol. 14, no. 4, pp. 478 – 486, 2009.

[35] Branislav Micusík and Allan Hanbury, "Steerable semi-automatic segmentation of textured images," in *SCIA*, 2005, pp. 35–44.

[36] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Computer Vision and Pattern Recognition*, vol. 1, pp. I–511–I–518, 2001.

[37] Yoav Freund and Robert E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *EuroCOLT*, 1995, pp. 23–37.

[38] Ross J. Quinlan, *C4.5: programs for machine learning*, Morgan Kaufmann Publishers Inc., 1993.