Technical Report

Pattern Recognition and Image Processing Group
Institute of Visual Computing and Human-Centered Technology
TU Wien
Favoritenstrasse 9-11/193-03
A-1040 Vienna AUSTRIA
Phone: +43 (1) 58801 - 18661
Fax: +43 (1) 58801 - 18697
E-mail: 1029201@student.tuwien.ac.at
URL: http://www.prip.tuwien.ac.at/

PRIP-TR-142 April 2, 2018

Skin Detection in frontal-view faces[1]

Robin Melán

Abstract

Skin detection plays an important role in a wide range of image processing applications such as image classification, face detection, face tracking, content-based image retrieval, gesture analysis and various human computer interaction domains. In recent years, the number of skin segmentation approaches has grown. However, skin detection remains an open problem due to its challenges illumination, complex background, camera characteristics and ethnicity. This report presents a new model-based approach using classification learners with supervised learning on frontal-view face images. The proposed solution is based on independent pixel classifiers, namely weighted kNN and decision trees. Both classifiers are trained from automatically labeled data and extend it by using Viola-Jones eyes and nose detectors and Active Contour Model (ACM) to extract sample pixels of both skin and non-skin classes. Our evaluation gives a comparative study with baseline state-of-the-art explicit thresholding methods. This methodological approach is seen as a preprocessing step of the following master thesis *Automatic human-head and shoulder segmentation of frontal-view face images*.

[1]Supported by Dipl.Ing. Dr.techn. Walter G. Kropatsch and Univ.Ass. Dr.techn. MSc Nicole M. Artner

1 Introduction

Skin detection is the process of finding skin colored pixels and regions in an image or a video [12]. It is the process of separating skin and non-skin pixels. In many computer vision applications this step is used as a preprocessing step to find regions that potentially have human faces and/or limbs in images. There is a lot of research on the topic of skin detection since many applications rely on it. Examples for these applications are face detection and tracking [9] or hand detection and tracking [11], retrieval of humans in databases and internet, automatic annotation, archival and retrieval [1], and content filtering, parental control software and criminal investigation. Concerning the last one, digital forensic experts are increasingly confronted with large amounts of data and judging whether it contains digital contraband or not [32].

There is still no universal method developed for the detection and segmentation of human skin since the skin appearance in images is affected by several factors such as illumination, background, camera characteristics, and ethnicity (see Section 1.2). These methods first detect skin pixels or regions based on their color [12], then transform them into an appropriate color space and finally use a simple skin classifier to label the pixel (see Section 2.1). Another approach is training a model on a particular dataset, resulting in a model-based method (see Section 2.2), or the third possibility is incorporating spatial information, using a region-based methodology (see Section 2.3). Most of the research in this area have focused on detecting skin pixels and regions based on their color [12]. Few approaches attempt to use texture information as another possibility to classify skin pixels [26]. For more information on the current state-of-the-art algorithms see Section 2.

1.1 Motivation

Skin detection is an important feature for several computer vision applications and often used as a preprocessing step, deciding whether human beings are found in an image or not. Such an image classification is for instance the preliminary step to automatically assess data as pornographic material for content filters, parental control software and criminal investigations [32]. Here the success rate is measured on the correctly selected images rather than the correct classification of every single pixel in the image.

The automatic retrieval of images from a database, a problem known as

content-based image retrieval (CBIR) is also an application, where skin-color detection is often used as a preliminary step [21]. An early application proposed by Gunsel et al. [16] uses skin detection to locate the anchorperson in TV shows for automatic video annotation, archival, and retrieval. Liensberger et al. [27] similarly provide a skin-color detection approach for online video annotation.

Another motivation for skin detection is hand detection, hand tracking, or further more detecting hand pose or gesture analysis. This is widely used in Virtual Reality (VR) or Augmented Reality (AR) for navigation and object manipulation [20].

Skin color information can be added to other features such as shape and geometry and can be used to build accurate face detection system [17] or track heads and faces in an image sequence [42].

In this report, we present a skin detection approach based on supervised classification learners, decision trees and weighted kNN. We concentrate on images with frontal-view faces only and look at skin detection as a preprocessing step for the following master thesis on *Automatic human-head and shoulder segmentation of frontal-view face images*. Therefore our main interest lies on the classification of skin pixels around the silhouette of the face and neck of humans (see Section 4.5). In the following sections of this report we will call our approach Skin Detection based on Supervised Classification Learners (SDSCL).

1.2 Challenges

For computer vision systems skin detection is prone to many challenges and still an open problem, while for the human visual system skin detection is easy. Spillmann [38] describes the human perception with an example of seeing a blue ball, were we all can agree in that the ball is perceived blue as whole, and not as a ball having blue patches and some other color patches produced by differences in illumination. Furthermore, the human visual system can dynamically adapt to varying illumination conditions, so it can preserve the actual color of the object [21]. In literature this is called color constancy [12] or chromatic adaptation [8].

Most of the literature on human skin detection has focused on imaging of the visible spectrum and using color information on those kind of images can be a challenging task as the skin color in images is sensitive to various factors [21]:

- *Illumination:* Changing the light source distribution, light source position or the illumination level (e.g. indoor, outdoor, diffuse or specular light, shadows, non-white lights) produces a change in the color of the skin in the image as well (= color constancy problem). Most skin detection approaches in literature are concentrating on this illumination variation problem.

- *Camera characteristics:* The behavior of different cameras can differ even under the same illumination. Hence the skin-distribution for the same person differs depending on the camera sensors. Moreover, not all cameras can capture the same level of dynamic range (the range of light intensities from the darkest shadows to the brightest highlights).

- *Ethnicity:* Different ethnic groups and people across different regions vary in skin color. For instance, under the same illumination condition the skin color of African, Asian, Caucasian and Hispanic groups differ from one another and range from dark, brown, yellow to white.

- *Individual characteristics:* Age, sex and different body parts of a person have an effect on the appearance of skin color.

- *Other factors:* Complex backgrounds containing surfaces and objects with skin-like colors produce false positive detection of skin. The persons hairstyle, make-up and glasses, can produce unwanted shadows, reflection or unusual skin colors. Moreover, color bleeding (the colored reflection of indirect light from a nearby object) and motion captured with slow shutter-speed can also influence the quality of the image due to blurring of colors and respectively the skin color appearance.

Regarding the imaging of the non-visual spectrum such as infrared (IR) [21] and spectral imaging [4] some of the problems can be overcome, such as partially illumination conditions, ethnicity, shadows and make-up. Illumination conditions are depending on the location where the picture was taken, since IR-cameras have difficulties outdoors in bright daylight. Moreover, the expensive equipment for these methods is another disadvantage and limitation for specific application areas.

This report concentrates on skin detection techniques applicable on images of the visible spectrum or single frames of videos.

Spillmann's [38] example of the blue ball shows that the human perception is **high-level**. Human beings can detect skin in real scenes, in pictures

or videos without problems [33]. However, when it comes to classifying single pixels as skin or non-skin the task becomes difficult. The reason is that human skin detection is not a simple **low-level** process, but a process in which high-level mechanisms are involved, incorporating visually detecting hair, clothes, etc and also some spatial diffusion mechanisms such as color and texture [38].

1.3 Overview of the Report

The remainder of this report is organized as follows: Section 2 gives a brief description of numerous state-of-the-art techniques in literature for skin detection using color. In Section 3 our supervised skin segmentation approach (SDSCL) is presented. Section 4 presents evaluations of the proposed algorithm on multiple datasets concerning particularly frontal-view face images and it discusses them in an extensive comparative study with baseline state-of-the-art approaches. Section 5 provides the summary and conclusions of this report.

2 State-of-the-Art

Saxen and Al-Hamadi [36] categorize color-based skin segmentation approaches found in the literature into the following three groups of methods:

1. Threshold-based methods: simple decision rules and easy to implement.
2. Model-based methods: need training and testing procedures and thus require datasets.
3. Region-based methods: incorporate neighboring pixels and are often computational expensive and rarely used.

These three groups of skin segmentation methods are described in the following subsection in detail.

2.1 Threshold-based Methods

One of the simplest and most commonly used human skin detection method is to define a fixed decision boundary for different color space components [39]. For each color space component single or multiple ranges of threshold values are defined. The pixel values of the input image that fall within those predefined ranges are labeled as skin pixels, all the others are defined as non-skin.

It is important to select a color space, where skin color is a compact cluster in order to be able to tightly model the skin class [12]. In the literature a variety of color spaces have been used in skin detection with the aim of finding a color space, where skin color is invariant to illumination or ethnicity conditions. In a threshold-based approach the choice of the color space affects the shape of the skin cluster, which further affects the detection process. Figure 1 shows density plots for skin-colored pixels from different people from different ethnicity groups: Asian, African and Caucasian in different color spaces.

In the following subsections some of the most common color spaces are discussed. For a better survey of different color spaces (e.g., RGB, YCbCr, HSV, CIE Lab, CIE Luv and normalized RGB) for skin-color representation and skin-pixel segmentation methods the reader is referred to Kakumanu et al. [21].

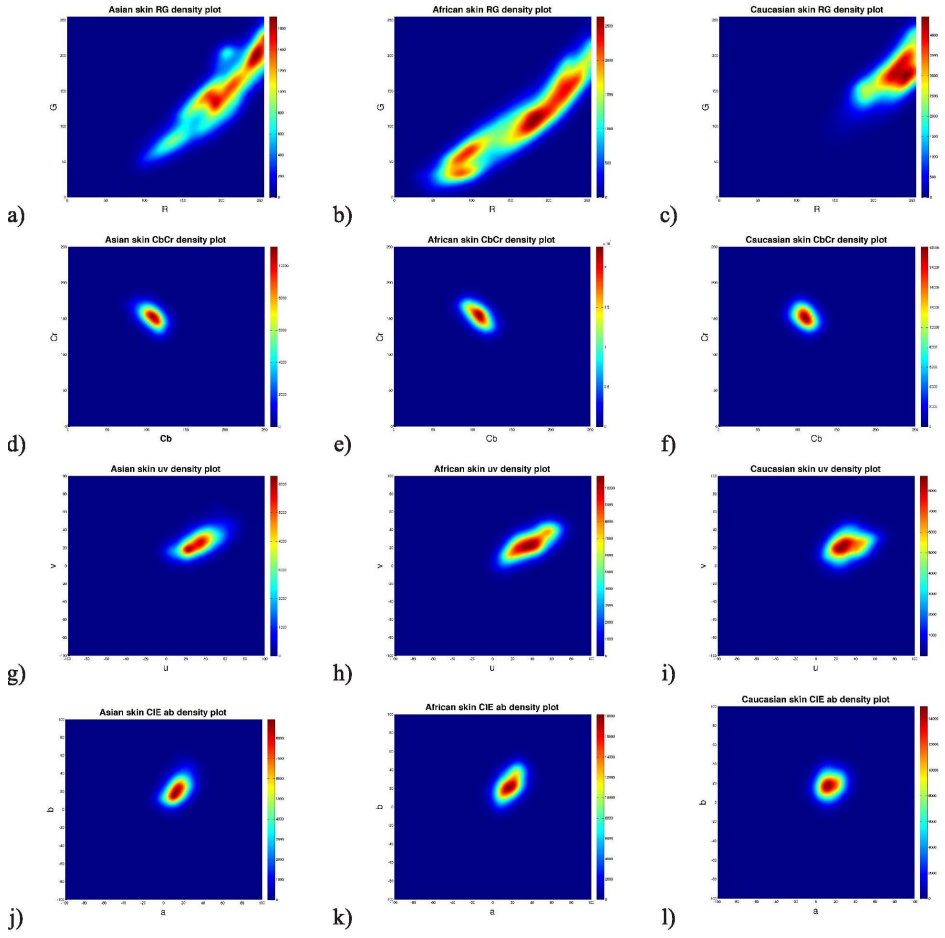

Figure 1: Density plots of Asian, African and Caucasian skin in different color spaces from Elgammal et al. [12]

2.1.1 RGB Color Space and Skin Detection

The **RGB** color space is the most commonly used color space in digital images [12]. It encodes colors as an additive combination of three primary colors: red (R), green (G) and blue (B). One of the drawbacks of working in the **RGB** color space is that luminance and chrominance cannot be separated. The R,G,B components are highly correlated, so changing the luminance of a given skin patch affect all three (R, G, and B) components. This can be observed in the first row of Figure 1, where skin patches from images of Asian, African and Cau-

casian people are taken at random illumination and plotted in RG space [12]. Furthermore, the skin color clusters for patches from different ethnicity groups are located at different locations in the **RGB** color space. Liensberger et al. [27] are applying for their online video annotation a combination of YCbCr, normalized RGB and RGB for skin detection. The final decision is made by taking votes out of the three color spaces.

2.1.2 Orthogonal Color Space and Skin Detection

A different class of color spaces are the orthogonal color spaces, which include **YUV**, **YIQ** and **YCbCr**. Transforming from **RGB** into any of these spaces is a linear transformation [12]. All these color spaces separate the illumination component (Y) from the two orthogonal chrominance components (**UV**, **IQ**, **CbCr**). Therefore, unlike the **RGB** color space the location of the skin color in the chrominance components is not affected by changing the intensity of the illumination [12]. As can be observed in the second and third row of Figure 1 the skin color of different ethnicity groups almost co-locates in the chrominance channels. The simplicity of the transformation and invariant properties made these color spaces widely used in skin detection applications e.g. [37], [13].

2.1.3 Perceptual Color Space and Skin Detection

Perceptual color spaces are described by **HSI**, **HSV/HSB**, and **HSL**. They separate three components: hue (H), saturation (S) and brightness, also called intensity, value or lightness (I,V, or L). These color spaces are deformations of the **RGB** color cube and are computed by a non-linear transformation. The boundary of the skin color class is specified in terms of hue and saturation.The brightness component **I**, **V** and **L** is often dropped to reduce illumination dependency of skin color. Shaik et al. [37] as well as Gasparini and Schettini [13] and Platzer et al. [32] used these color spaces in their skin detection approaches.

2.2 Model-based Methods

A commonly used model-based method in literature are the histogram-based Bayes classifier, also called skin probability map (SPM). It models the distribution of skin tones, is simple and computationally fast [21]. Yoo and Oh [43] use in their approach a histogram model with naive Bayes classification for face detection. The histogram is quantized into a number of histogram bins, where

each bin stores the count associated with the occurrence of the bin color in the training data set. These bins are converted into probability distributions, which corresponds to the likelihood a given color belongs to skin or the likelihood it belongs to non-skin.

Other widely used model-based methods in literature are Gaussian classifiers or Gaussian Mixture Models (GMMs) to approximate the skin-color distribution [21]. Greenspan et al. [15] show a mixture of Gaussians as a robust representation that can accommodate large color variations, as well as highlights and shadows. They trained GMM with two components, where one component captures the distribution of the skin color while the other captures the distribution of the highlighted regions of the skin.

Lee and Yoo [25] compare the performance of a single Gaussian model (SGM) with a GMM of six components. Under controlled illumination condition, skin colors of different individuals in a orthogonal color space cluster in a small region. Hence, in these conditions the skin color distribution can be modeled through an elliptical Gaussian joint probability distribution function (pdf). Once other image conditions have to be considered a SGM is not sufficient and GMMs with multiple components have to be considered. The key idea behind using multiple components is that different parts of the skin regions are illuminated in a different manner and they can be modeled by different components [21].

Lü and Huang [28] propose a skin detection method based on the cascaded adaptive boosting (AdaBoost) classifier, which consists of minimum-risk based Bayesian classifier and models in different color spaces such as HSV (hue, saturation, value), YCgCb (brightness, green, blue) and YCgCr (brightness, green, red). Ma et al. [29] proposed the Semantically Constraint Skin Detection (SCSD) method based on Random Forests. The semantic constraint is based on the dependence between skin pixels and human body parts, to limit the influence of background skin-like pixels. Khan et al. [24] compare their random forest based skin detection approach with other classification learners like Bayesian network, Multilayer Perceptron, SVM, AdaBoost, Naive Bayes and RBF network. For their evaluation a dataset consisting of 25 videos from the internet with 8991 images was used with annotated pixel-level ground truth. Their results show that the random forest with 10 trees performs best in terms of accuracy, precision and recall and F-score.

2.3 Region-based Methods

A common region-based method used for skin segmentation is Region Growing [36]. The problem with Region Growing is the need of seed points. Abdullah-Al-Wadud et al. [2] use a color distance map and based on this map they generate some skin as well as non-skin seed pixels. Then they grow them to capture the appropriate regions. With this approach they do not generate much noisy segments and do not need any prior training session. Saxen and Al-Hamadi [36] propose a region growing approach computing the seed points by a Bayes approach.

Khan et al. [23] propose a skin segmentation approach using graph cuts. They model the skin segmentation as a min-cut problem on a graph defined by the image color characteristics and a universal seed to overcome the potential lack of successful seed detections. The advantage of their approach is that it is only based on skin sampled training data making it robust to unseen backgrounds. It exploits the spatial relationship among the neighboring skin pixels providing more accurate and stable skin blobs.

In this paper we present a model-based supervised classification learner based on independently decision trees and weighted kNN. We include high-level information of the query image into the training set and show how this improves the correct skin detection. The databases used as training and testing sets were transformed from RGB color space into the orthogonal color space YCbCr from which the two chrominance channels Cb and Cr represent the feature space.

3 Our Method

The author of this report proposes a novel skin detection algorithm based on classification learners. In pattern recognition, classification is considered an instance of supervised learning, e.g. learning where a training set of correctly identified observations is available. In literature there are a number of algorithms including [30]: Linear Classifiers, Support Vector Machines (SVM), Kernel estimation like k-Nearest Neighbor (kNN), Boosting (meta-algorithms), decision trees and neural networks (NN).

The novelty of the proposed approach lies in our improvements on the training set of the kNN and decision trees classifier (see Section 3.3).

3.1 Recall: Decision Trees

Decision trees [7] are characteristic in having fast prediction speed[1], small memory usage[2] and being easy to interpret. A disadvantage can be that they have low predictive accuracy and tend to overfit, if the depth of the splits is not pruned to a maximum number of splits [18].

In Figure 2 a simple tree with a maximum number of 4 splits can be observed. To predict a response, the decisions in the tree beginning from the root node down to a leaf node are followed. Each step in a prediction involves checking the value of one predictor. Observing the decision tree in Figure 2 the first predicator is the root node, and its decision is $Cr < 140.5$ to follow branch on the left or $Cr \geq 140.5$ to follow the branch on the right. When a branch reaches a leaf node, the data is classified either as type 0 (non-skin) or 1 (skin).

Increasing the number of splits on the decision tree usually increases the accuracy on the training data [18]. However, predicting with an independent test set might not show similar accuracy compared to the validation accuracy of the training set. It can be said that a decision tree is as all classification learners highly dependent on the training set and provides comparable accurate results on new test samples if they are similar to the training set. This is one of the reasons why the author chose the decision tree as classification learning model for skin segmentation. Furthermore the author chose a maximum number of splits of 100, gaining a huge number of leaves to make many fine distinctions between the two classes. Adding information of the input image into the train-

[1] Speed: Fast 0.01 sec.; Medium 1 sec.; Slow 100 sec.

[2] Memory: Small 1MB; Medium 4MB; Large 100MB

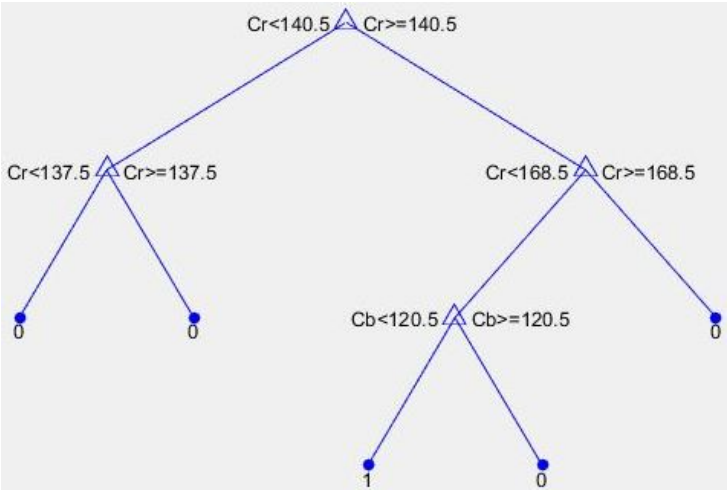

Figure 2: A simple decision tree classifier with a maximum number of 4 splits trained with the *UCI* database (see more information on the database in Section 4.1).

ing set as described in Section 3.3 improves the accuracy of the correctly classified pixels in the remainders of the input image.

3.2 Recall: Weighted k-Nearest Neighbor (kNN)

Nearest Neighbor classifiers [3] are characteristic in having slow to medium prediction speed[1], medium memory usage[2] and being harder to interpret compared to decision trees. They typically have good predictive accuracy in low dimensions. As dimensionality increases, the distance to the nearest data point approaches the distance to the farthest data point, which might lower the prediction accuracy [5]. In the k-Nearest Neighbor (kNN) algorithm categorizing a query point is based on its closest k neighbors in the training examples. Regarding the weighted kNN the distances to the neighboring points are weighted. Choosing a high number of neighbors can be time consuming to fit. After experimental results a number of 10 neighbors and a squared inverse distance weight was defined fitting our purpose.

3.3 Skin Detection based on Supervised Classification Learners (SDSCL)

To improve the performance of classification learners in particular decision tree and weighted kNN regarding skin detection, we propose to extend the training set by adding high-level information of the query image. A series of preprocessing steps were performed on the input image to extract this pixel information to be incorporated into the training set.

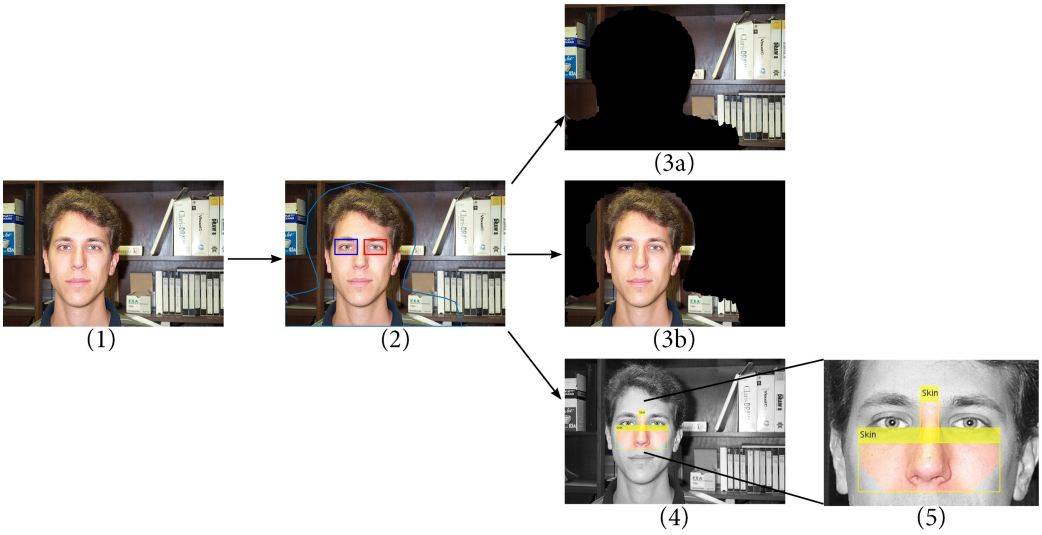

Figure 3: Overview of the preprocessing steps: (1) Input image. (2) Detect eyes and placing control points for the initial ACM mask (see blue line). (3*) ACM Results (3a) *Foreground* & (3b) *Background*. (4) Extracting skin pixels (in color) (5) Zoom into the selection of the extracted skin information.

Following the enumeration of the subtasks in Figure 3, at first, the eyes in the input image Figure 3(1) are detected by Viola-Jones [41] to place the control points for an Active Contour Model (ACM) [22] in Figure 3(2). The shape mask identified with the blue contour line in Figure 3(2) is the initial mask for ACM. The purpose of this rough separation into foreground and background is to segregate most of the background out of the image resulting in an incomplete background mask (3a) and a foreground mask (3b) with spurious segments including the subject. The last preprocessing step is the extraction of human skin information of the query image, shown in Figure 3(4). With the

same Viola-Jones algorithm [41], but different Haar-like features the nose of the person is found in the image underneath the eyes location. Skin pixels are extracted with this information from the region between the eyes location and the nose bounding box, represented as the two skin boxes in Figure 3(5). The first bounding box concentrates on skin extraction between the eyes of the subject, the second one under the eyes including the nose. Since the face is round the author extracts a half disk-shape of skin information to reduce the possibility of false positive background pixels around the cheeks of the persons face (observe Figure 3(5) only colored skin pixels are considered).

In the further evaluation in Section 4 different independent variations of supervised classification learners including pixel information of the input image are discussed. These variations are defined in Table 1. *tree* and *kNN* are generated using the *UCI* dataset (see description in 4.1) as training set. Every following variation of the classification learners include information on the input image in their training set to make them more individual in their prediction/testing phase to the respective input image. *tree-bg* and *kNN-bg* include the background mask segregated after ACM (see Figure 3(3b)) and *tree-skin* and *kNN-skin* respectively the skin information from the query image computed after ACM (see Figure 3 (4)). The last three variations include both background and skin information in the training set, once with the *UCI* database (*tree-SDSCL* and *kNN-SDSCL*), and once without the *UCI* database (*tree-exclusive* and *kNN-exclusive*). The last variation includes the information multiple n times (*tree-SDSCL-multiple* and *kNN-SDSCL-multiple*).

Evaluations show that adding information of the input image into the training set noticeably improves the accuracy of the correctly classified pixels in the remainders of the input image.

Table 1: Different variations of semi-supervised classification learners including pixel information of the input image. *UCI* is the database used as training set in most of the variation. *bg* means that the training set includes background pixels from the input image computed after ACM. *skin* means that the training set includes skin pixels from the input image. *n* is a variable, which defines the weighting of the additional training information.

Name (Abbr.)	Method	Training Set
tree	decision tree	UCI
kNN	weighted kNN	UCI
tree-bg	decision tree	UCI + bg
kNN-bg	weighted kNN	UCI + bg
tree-skin	decision tree	UCI + skin
kNN-skin	weighted kNN	UCI + skin
tree-SDSCL	decision tree	UCI + bg + skin
kNN-SDSCL	weighted kNN	UCI + bg + skin
tree-exclusive	decision tree	bg + skin
kNN-exclusive	weighted kNN	bg + skin
tree-SDSCL-multiple	decision tree	UCI + n * (bg + skin)
kNN-SDSCL-multiple	weighted kNN	UCI + n * (bg + skin)

4 Evaluation

The proposed approaches based on the classification learners decision tree and weighted kNN and their improvements were implemented in MATLAB[3]. In the following qualitative and quantitative evaluations we compare them with baseline skin detection based on *explicit thresholding in the YCbCr Color Space* [12] (thresholdYCbCr), *HSV Color Space* [13] (thresholdHSV) and *RGB Color Space* [13] (thresholdRGB).

The performed evaluations use the databases described in Section 4.1. They provide ground truth on images with varying skin color and background conditions as well as illumination and camera characteristics, i.e. the ground truth dataset is representative for real applications.

In a further analysis, the ground truth is altered, concentrating the evaluation on our region of interest: the silhouette of the face and neck of the subject (see Section 4.5). For this evaluation quantitative and qualitative evaluations

[3]MATLAB: `https://de.mathworks.com`

are performed.

Representative sample images from different databases were selected for qualitative evaluations to demonstrate the performance and limitations of the proposed approaches. Regarding quantitative evaluations the segmentation results of the approaches were compared against the ground truth. In the context of skin classification,

- *true positives* are skin pixels that the classifier correctly labels as skin.
- *true negatives* are non-skin pixels that the classifier correctly labels as non-skin.
- *false positives* are non-skin pixels that the classifier erroneously labels as skin.
- *false negatives* are skin pixels that the classifier erroneously labels as non-skin.

The goal of a good classifier is to have low false positive and false negative rates. As in any classification problem, there is a trade-off between false positives and false negatives [12]. Having a soft class boundary the false negative rate is low and the false positive rate is high, which results in a high recall value. Having a tighter class boundary the false negatives are high and the false positives low. This normally results in a higher precision value.

In the following evaluations, well known evaluation measurements were computed including accuracy, precision, recall / true positive rate (TPR), false positive rate (FPR), F1 score (harmonic average of the precision and recall), and the sums of true positives (TP), false positives (FP), false negatives (FN), true negatives (TN).

4.1 Databases

Experiments are conducted using the following public datasets which all except for the last one provide a ground truth. The databases were transformed from RGB color space into the orthogonal color space YCbCr, from which the two chrominance channels Cb and Cr represent the two-dimensional feature space.

It is important to mention that for this report the primary focus is on images, where the face can be easily found with state-of-the-art face detection algorithms, so the subject in the image is not occluded and face and shoulders

are facing the camera. Therefore to filter out images, not satisfying the defined criteria, the commonly used face detection algorithm from Viola and Jones [41] was performed reducing the number of images per database. This removed images, where no face or eyes were found, i.e. hand and leg pictures, head pose sideways images and therefore not detected, no humans in the image, face occluded, sun glasses occluding eyes etc. If no filtering is specifically stated in the following descriptions of the databases then no images were removed.

- *UCI [6],[10]:* is collected by randomly sampling B,G,R values from face images of various age groups (young, middle, old), ethnicity groups (white, black, and Asian), and genders obtained from FERET database and PAL database. The dataset provides ground truth and contains 245.057 pixel entries (50.859 skin and 194.198 non-skin).

- *dbSkinChile [33],[34],[35]:* is collected from random *in the wild* images, containing a variation of age, ethnicity groups (majority white), genders, illumination and camera characteristics. The database provides ground truth and contains multiple or single subjects, resulting, after filtering out not satisfying images, into a total of 36 images.

- *Pratheepan [39]:* is collected randomly from Google and images are captured with a range of different cameras, using different color enhancement, under different illuminations, variation of age (young, middle), ethnicity groups (white, Asian), and genders. The database provides ground truth and contains 32 face images.

- *Faces*[4]*:* this frontal face dataset is collected at California Institute of Technology, capturing 27 people under different light conditions, facial expression, ethnicity groups (mostly white and Asian), gender and complex backgrounds. It provides images under different conditions with a complex background, where the orientation of the head and shoulders is facing the camera according to the defined criteria we are focusing on in this paper. The database does not provide any ground truth. Therefore, for a small set of images ground truth was generated manually and those samples were used in qualitative evaluations.

[4]Collected by Markus Weber at California Institute of Technology `http://www.vision.caltech.edu/html-files/archive.html`

4.2 Evaluation of Classification Learners

To evaluate which classification learning approach is better suited for skin detection, different classifiers were tested using the *UCI* database (4.1) considering 90% as training set and 10% as testing set (see results in Table 2).

In Table 2 the validation accuracy represents the success rate on the respective training set computed by 5-fold cross validation. This validation method protects against overfitting by partitioning the data set into folds and estimating accuracy on each fold. The procedure of training the classifiers was carried out with an Intel 3.4GHz quad processor i7-3770 and 16 Gbytes of RAM, running a 64-bit Windows 10 operating system.

Observing the results in Table 2 the classification learner decision tree and weighted kNN perform best regarding validation accuracy. Regarding the training time decision tree outperforms the other classifiers, being twice as fast as the logistic regression, about ten times faster than weighted kNN and boosted trees and hundred times faster than SVM. As for the prediction phase all classification learners perform similarly except for logistic regression. For further evaluation the author selected the two most promising classification learners decision tree and weighted kNN and performed the improvements on them regarding the training set as described in Section 3.3.

Table 2: Evaluating classification learners with *UCI* database (90% training set and 10% testing set)

Classification Learners	Validation Accuracy	Training time (sec)	Accuracy	Precision	Recall	F1
Decision Tree	**0.9988**	**3.60**	**0.999**	**0.995**	**0.999**	**0.997**
Weighted kNN	**0.9988**	28.15	0.998	**0.995**	0.998	0.996
Logistic Regression	0.8993	9.66	0.898	0.764	0.735	0.749
SVM	0.9985	255.56	0.998	0.994	0.998	0.996
Boosted Trees	0.9973	38.91	0.998	0.993	**0.999**	0.996

Regarding the parameters distance weight and the number of neighbors k multiple values have been evaluated for weighted kNN with *UCI* database as training set. For the distance weight the following values were considered:

- *Equal:* no weight
- *Inverse:* $\frac{1}{\text{Euclidean distance}}$

- *Square Inverse:* $\frac{1}{(\text{Euclidean distance})^2}$

For the number of nearest neighbors k three different values were evaluated: a fine number of 10, middle 30 and a coarse number of 100. Having many neighbors can be time consuming to fit. The best results were achieved with $k = 10$ and a distance weight of square inverse.

4.3 Quantitative Evaluation of Classification Learners tree and kNN

The classification learners *tree* and *kNN* (without any improvements on the training set) are evaluated and compared with state-of-the-art approach *explicit thresholding skin detection in YCbCr Color Space* [12] (thresholdYCbCr), *explicit thresholding skin detection in HSV Color Space* [13] (thresholdHSV) and *explicit thresholding skin detection in RGB Color Space* [13] (thresholdRGB) in Table 3.

The *tree* and *kNN* are trained with 90% (220.550 pixels) of the *UCI* dataset. The remaining 10% (24.506 pixels) were used as testing samples for the five classifiers. The split was chosen randomly for ten iterations and the mean of all iterations was calculated. The quantitative results show that for this particular dataset the results of *kNN* and *tree* are very similar outperforming the three explicit thresholding skin detection algorithms regarding accuracy, precision and F1 score. Recall is higher for *thresholdYCbCr* and *thresholdRGB*, because both find more skin pixels (FN = 0) but as a drawback also categorize a large number of background pixels as skin (see false positives (FP)).

Table 3: Quantitative evaluation on the *tree*, *kNN* and the state-of-the-art explicit thresholding approaches thresholdingYCbCr [12], thresholdHSV [13] and thresholdRGB [13] with *UCI* dataset.

Approach	Accuracy	Precision	Recall	F1	TP	FP	FN	TN
tree	**0.9989**	0.9947	0.9991	0.9969	5128.4	26.7	4.1	19346.8
kNN	**0.9989**	**0.995**	0.9993	**0.9971**	**5128.8**	**26.2**	3.7	**19347.3**
thresholdYCbCr	0.9874	0.9432	**1**	0.9709	5075.6	305.5	**0**	19124.9
thresholdHSV	0.9441	0.9351	0.7846	0.8534	3983	276.6	1092.6	19153.8
thresholdRGB	0.9599	0.8376	**1**	0.9116	5075.6	985.1	**0**	18445.3

For the training set of both classification learners the orthogonal color space YCbCr was chosen. Observing the histograms of the *UCI* database once in RGB

color space (Figure 4) and in YCbCr color space only considering the chrominance components Cb and Cr (Figure 5) can be observed that in the RGB the non-skin pixels overlap completely with the skin pixels making the correct classification harder. For the YCbCr color space only a smaller overlap can be observed for this particular database, which makes it more suiting for a correct classification.

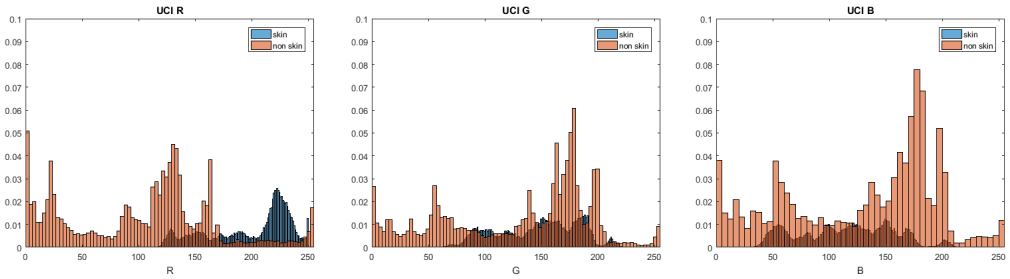

Figure 4: 1-D histograms of skin vs. non-skin pixels of the *UCI* database in RGB color space.

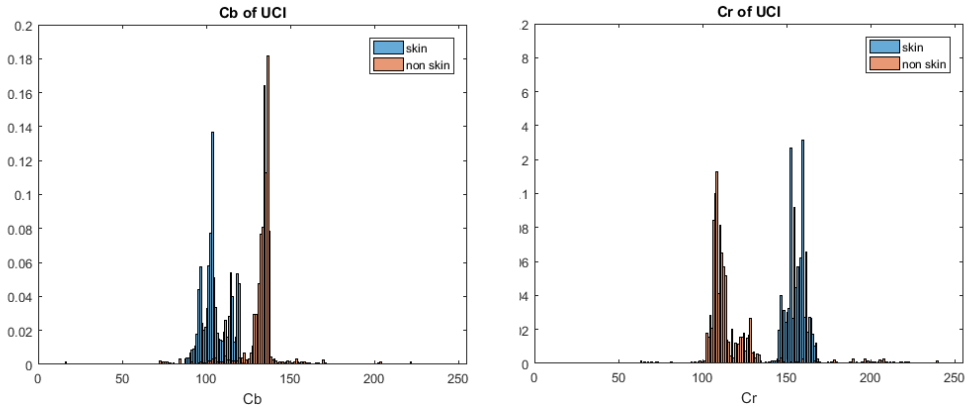

Figure 5: 1-D histograms of skin vs. non-skin pixels of the *UCI* database considering the chrominance components of the YCbCr color space.

Figure 6 shows the incorrect classified pixels after training a decision tree with *UCI* dataset once in RGB color space and once in YCbCr color space only considering the chrominance components. In orange are the false positive and in blue the false negatives. The decision tree trained in CbCr color space classifies 0.14% less incorrectly than the decision tree trained in RGB color space regarding the *UCI* database.

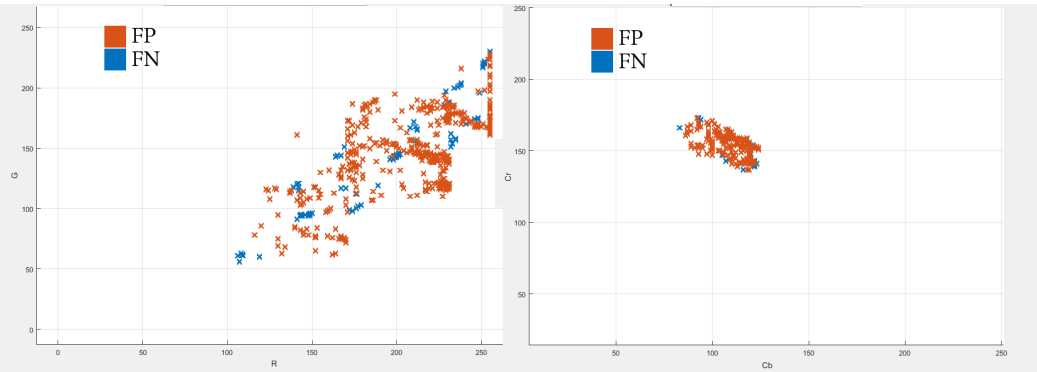

Figure 6: Incorrectly classified pixels from decision tree. Left: Result of trained decision tree in RGB color space. Right: Result of trained decision tree in YCbCr color space. In orange are the false positive and in blue the false negatives.

4.4 Qualitative Evaluation of tree and kNN

In Figure 7 some qualitative results are illustrated: the first column shows the original image and in the corner the binary result of the skin ground truth. Columns two to six show the results of the five classifiers.

Different samples from distinct databases were selected: captured with a range of different cameras, under different illuminations (in Figure 7 e.g. (2) having a slightly bluish tone, (4) being overexposed and (5) being underexposed), variation of age (young, middle), ethnicity groups and backgrounds (from uniform in (6) and (7) to complex e.g. in (2)-(5)).

Similar to the conclusions of the quantitative results, for these particular samples it can be observed that *thresholdYCbCr* and *thresholdRGB* are classifying too much as skin leading to large false positive regions. This hinders the further process of segmenting out the background from the person, as can be seen specially in sample (3) and (4). *Tree* detects fewer skin pixels inside the face, but shows better results around the silhouette of the person.

As mentioned in Section 2.1, RGB color space is not suited for color based skin detection and color analysis because of mixing of color (chrominance) and intensity (luminance) information and its non-uniform characteristics. Orthogonal and perceptual color spaces discriminate color and intensity information even under uneven illumination conditions. Comparative studies by Shaik et al. [37] as well as our experimental results in Table 3 and Figure 7 show that *thresholdYCbCr* outperforms other explicit thresholding-based approaches for

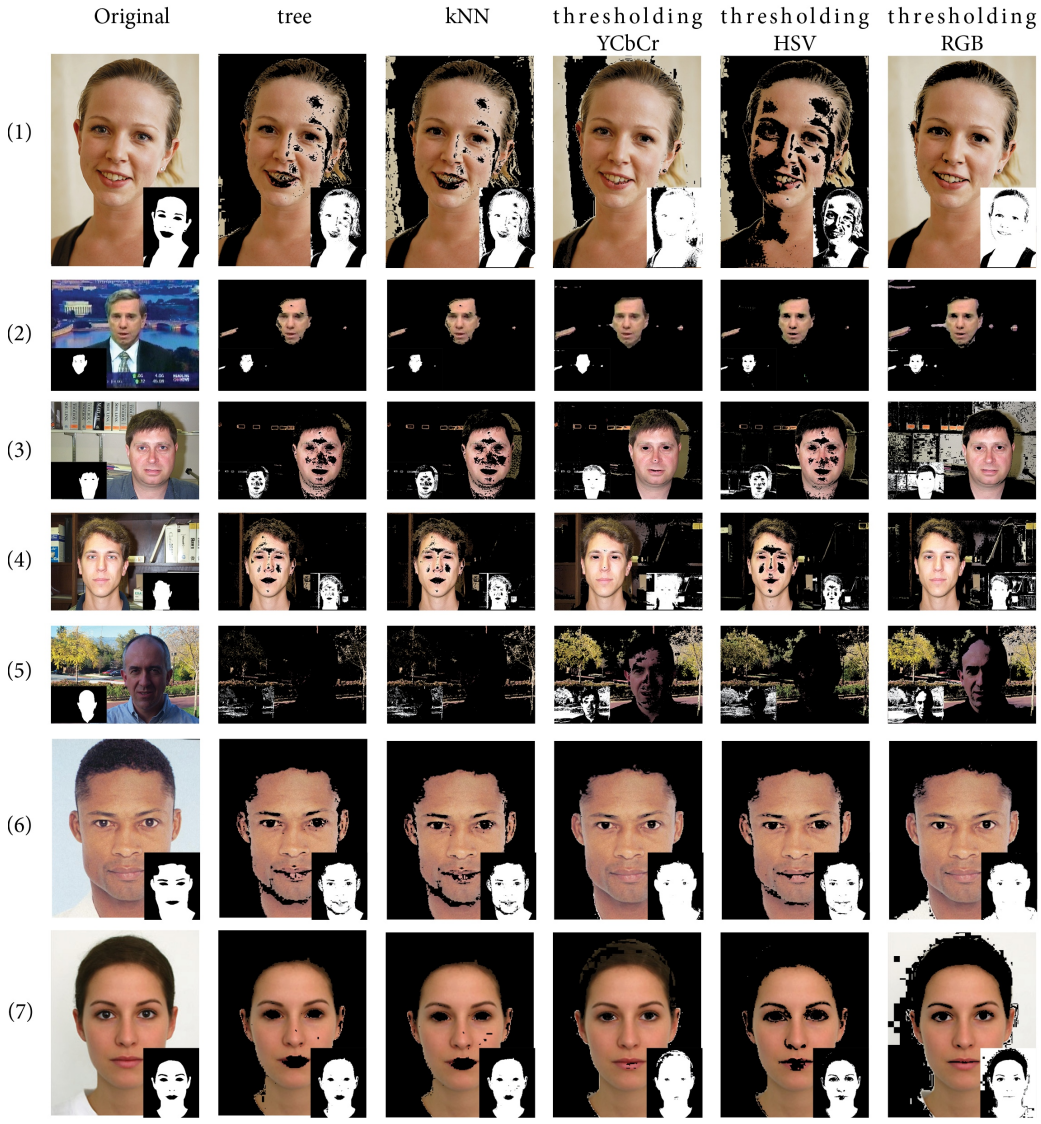

Figure 7: Seven examples: (1), (6), (7) are from *Pratheepan*; (2) is from *dbSkin-Chile*; (3), (4), (5) are from *Faces* (California Institute of Technology)

the segmentation and detection of skin color in images.

4.5 Silhouette Ground Truth

Since in this report the author analyzes the skin segmentation as a preprocessing step for the following master thesis (*Automatic human-head and shoulder segmentation of frontal-view face images*), the region of interest of skin detection lies on the silhouette of the persons face and neck. If the pixels around the silhouette are correctly classified then the rest inside the silhouette can be labeled as face and everything outside the silhouette as background. For further evaluations the ground truth is altered with morphological image processing algorithms of erosion and dilation [40]. This way the evaluation concentrates on the silhouette of the face and neck of the subject, for visual examples observe Figure 8.

Both morphological operators are used with a disk-shaped structuring element [14] of radius 10. The fixed radius leads to the evaluation of ten non-skin and ten skin pixels from the silhouette origin. With erosion the ground truth boundary shrinks and with dilation pixels are added to the boundary. This new ground truth is computed for all described databases in Section 4.1 except *UCI*, since this dataset is provided in a csv-table format with randomly selected pixels from a number of images, where the original images were not provided with the dataset.

4.5.1 Compare evaluations of Complete Ground Truth and Silhouette Ground Truth

The region of interest on our skin detection approach is around the silhouette of the persons face and neck. Similar to the evaluations done before (see Section 4.3) the *UCI* dataset was chosen again as training set for the proposed classification learners *tree*, *kNN*. The segmentation results of both approaches are compared with the state-of-the-art explicit thresholding *thresholdingYCbCr* approach by Elgammal et al. [12]. As test dataset randomly 25.000 (20.000 non-skin and 5.000 skin) pixels of the *dbSkinChile* were selected, first from all pixels and second only around the silhouette of the subjects in the images. This size of test dataset was chosen to be able to compare the quantitative results with the evaluation done in Section 4.3. The test samples of this particular dataset are taken, since the ground truth can be modified as explained in Section 4.5.

In Table 4 the comparison of both evaluation methods on the two different ground truths are illustrated, first considering the complete image (All Pixels)

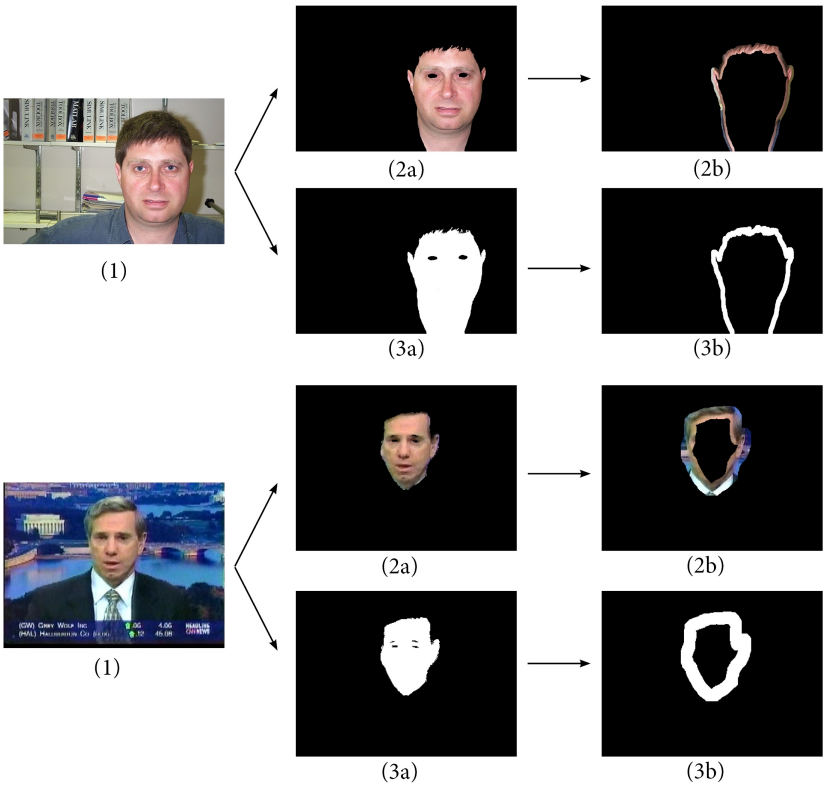

Figure 8: Focus in the evaluation of skin detection on the silhouette of the person. First image is from *Faces* (California Institute of Technology), second image from *dbSkinChile*.

and second only pixels around the silhouette (Silhouette). It can be observed that *thresholdYCbCr* performs better than both classification learners concerning skin detection in the entire image regarding this test set, but when it comes to the pixels around the silhouette of the regarding dataset the performance of the classification learners is better (observe box plot in Figure 9).

A F1-score of 56.16% for *tree* for the skin detection around the silhouette is fairly poor. The difficulty is that around the silhouette skin pixels are darker due to shadows or other illumination conditions of the scene and the transition from skin to hair or background is captured depending on the quality of the camera and its sensor. Moreover the manually generated ground truth by humans is prone to have errors specially around the silhouette. In a study by

Table 4: Quantitative evaluation on the proposed *tree*, *kNN* and the state-of-the-art explicit thresholding approaches *thresholdingYCbCr* [12] first a test dataset of pixels randomly selected in dataset *dbSkinChile (All Pixels)* and second a test dataset of pixels only around the silhouette of *dbSkinChile (Silhouette)*. For ten iterations the mean is computed for accuracy, precision, recall and F1.

Test set	Approach	Accuracy	Precision	Recall	F1
All Pixels	tree	**0.8649**	**0.6718**	0.6333	0.652
	kNN	0.861	0.6522	0.6527	0.6524
	thresholdYCbCr	0.8255	0.539	**0.8808**	**0.6688**
Silhouette	tree	**0.7906**	**0.4826**	0.6713	**0.5616**
	kNN	0.7782	0.463	0.6819	0.5517
	thresholdYCbCr	0.643	0.3486	**0.9048**	0.5032

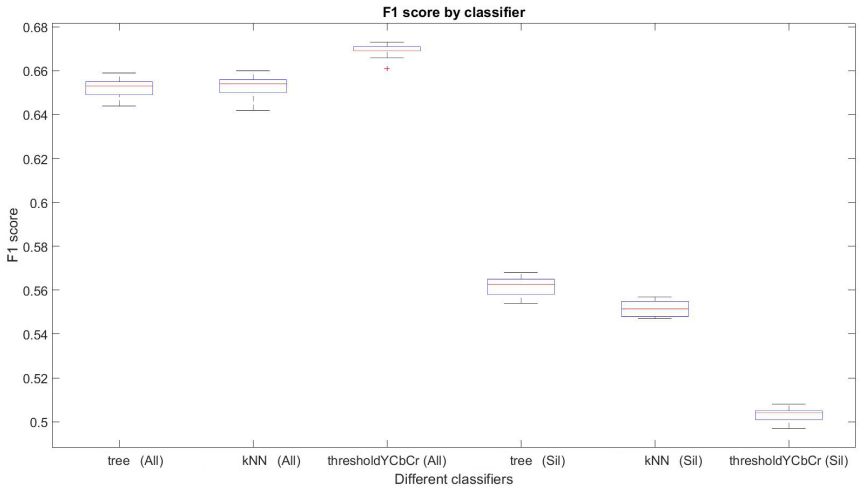

Figure 9: Comparing the different classifiers with first a test dataset of pixels randomly selected in dataset *dbSkinChile (All)* and second a test dataset of pixels only around the silhouette of *dbSkinChile (Silhouette)*. For ten iterations the mean and standard deviation of F1 score is illustrated as a box plot.

Liensberger et al. [27] people were asked to rate fragments of images as whether they contain skin or not and their results point out that humans are not able to detect skin without context. Classifying skin pixels manually around the sil-

houette is difficult for humans as well.

In Figure 10, the normalized histograms of *UCI* database (first row) and the reduced *dbSkinChile* database (second row) regarding only pixels around the silhouette are plotted. Comparing the histograms with each other, for Figure 10(1) the two peeks skin and non-skin pixels are disjoint whereas for the silhouette dataset in Figure 10(2) the two peeks of skin and non-skin pixels are closing onto each other making a segregation more difficult. This is the reason why the performance in Section 4.3 for all classifier is better regarding the *UCI* database than with the reduced *dbSkinChile* dataset.

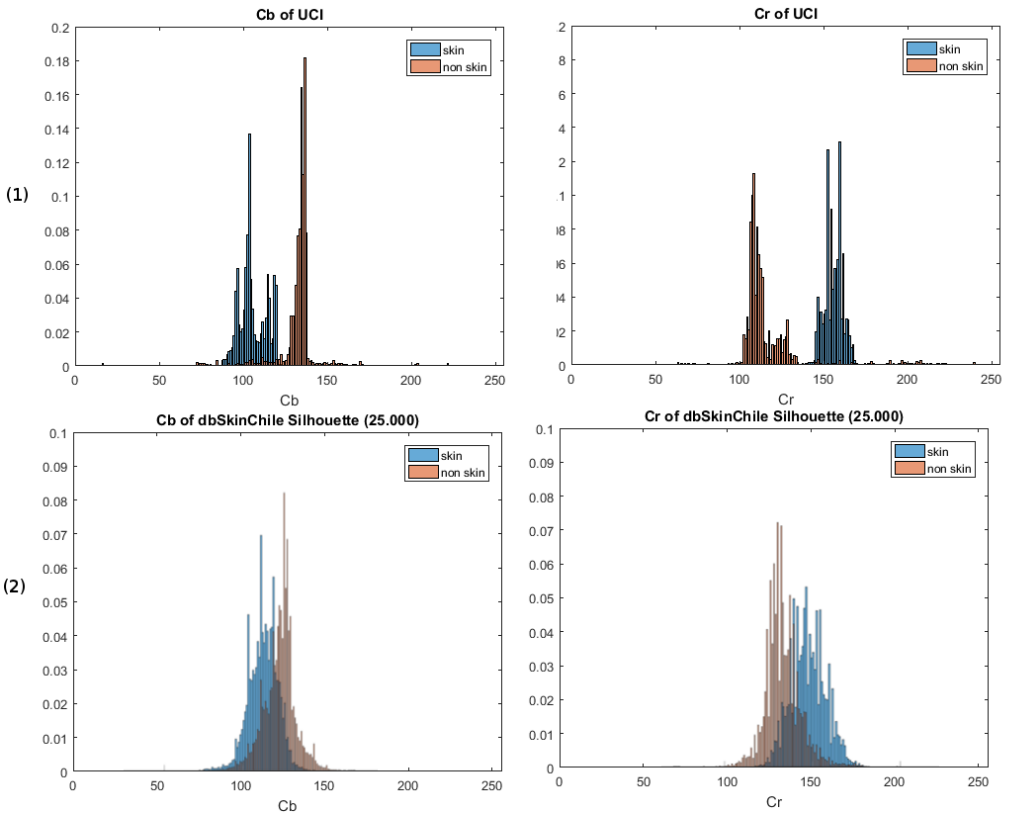

Figure 10: Normalized histograms: Comparing in YCbCr color space skin-pixels with non-skin pixels from *UCI* database in (1) and pixels around the silhouette from *dbSkinChile* in (2).

4.6 Evaluating different variations of training set extensions

For the two classification learners *tree* and *kNN* background pixels or skin pixels or both pixel informations from the query image are included in the training set, leading to five different variations, as described above in Section 3.3. These variations are compared with the two simple approaches *tree* and *kNN*, which have no high-level information incorporated in their training set, and the *thresholdYCbCr* in a qualitative evaluation.

Two representative samples were selected and the Figures (11 and 12) are arranged as following: for every sample the first row illustrates the original image, ground truth of skin, the generated silhouette representing the region of interest (pink being skin pixels and yellow non-skin pixels). The last image on the first row shows the result of *thresholdYCbCr*. The following two rows are the results of the *tree* and *kNN* classification learners and respective variations with the extended training set (Section 3.3). In the results skin pixels are visualized as white and non-skin pixels as black. Regarding the area around the silhouette, green pixels are correctly classified pixels, so all true positives and true negatives, and misclassified pixels are red, equivalent to all false positives and false negatives.

Both examples show that the supervised variations of the classification learners (explained in Section 3.3) improve the results, specially for *tree-SDSCL* and *kNN-SDSCL*, which include background as well as foreground pixels of the input image in the training set. Moreover, it can be concluded that the decision tree as a classifier with the support of information on the input in the training set does provide better results than weighted kNN on most samples, e.g. in Figure 12 detecting the hat of the baby as non-skin. The results in Figure 11 are very similar throughout all the supervised classification learner variations, detecting correctly most of the hair as non-skin, and segregating most of the background correctly (except for *tree-skin* and *kNN-skin*). The output of the supervised learners, which only use information of the input image in the training *tree-exclusive* and *kNN-exclusive* minimize the false positive rate, but therefore have a higher false negative rate. Adding the input image information multiple times in the training set and weighting it higher has minimal impact on the results when comparing *tree-SDSCL*, *kNN-SDSCL* with *tree-SDSCL-multiple*, *kNN-SDSCL-multiple*.

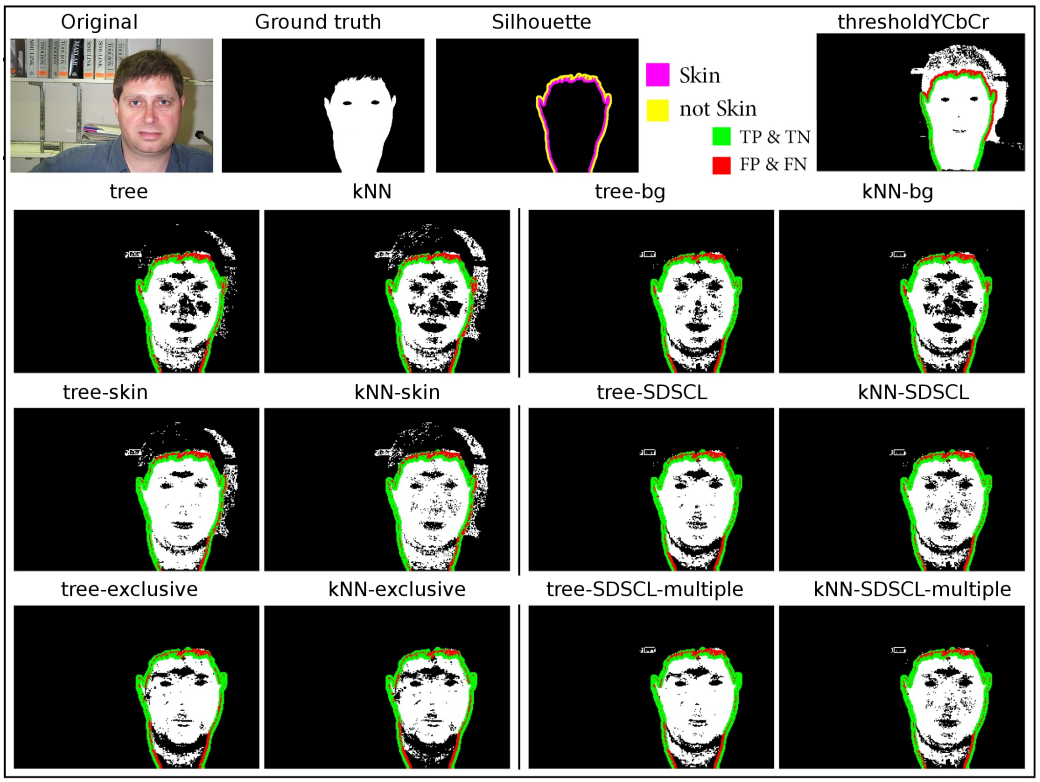

Figure 11: Sample from *Faces* (California Institute of Technology) and the results of *thresholdYCbCr*, the classification learners *tree* and *kNN*, and the five variations on the training set. The detected skin is marked as white, the correct classified pixels around the silhouette are marked green and the misclassified are visualized as red.

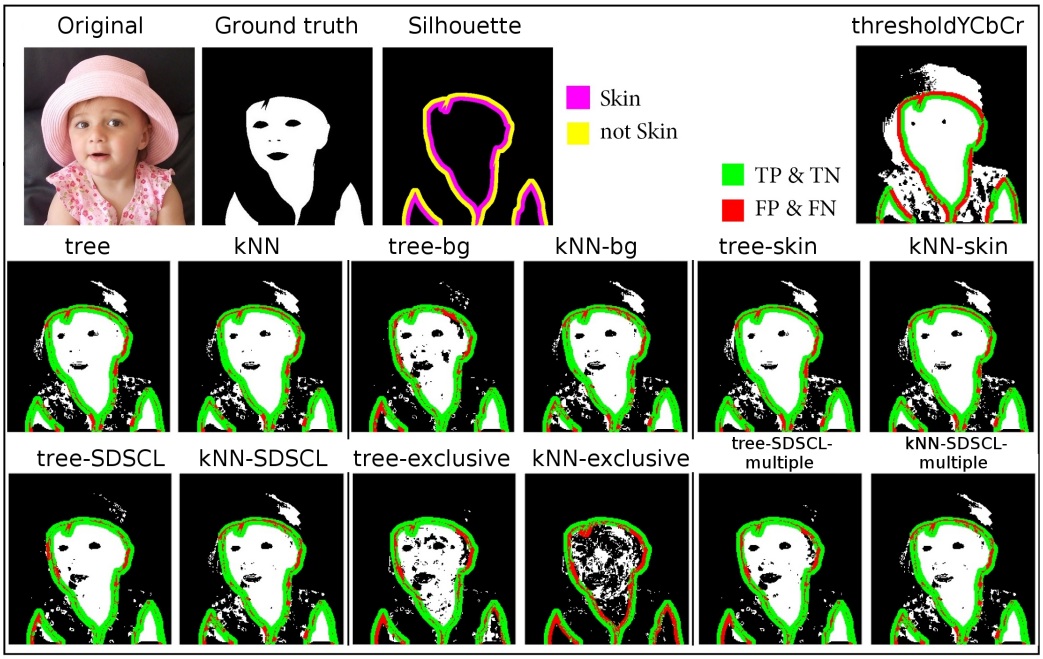

Figure 12: Sample from *Pratheepan* and the results of *thresholdYCbCr*, the classification learners *tree* and *kNN*, and the five variations on the training set. The detected skin is marked as white, the correct classified pixels around the silhouette are marked green and the misclassified are visualized as red.

4.7 Evaluation of SDSCL

In this subsection we are discussing quantitative and qualitative results concerning the proposed SDSCL approach and compare it with state-of-the-art algorithms. For the evaluation we are using *UCI* database as training set for the classification learners *tree* and *kNN*. As described in Section 3.3 *tree-SDSCL* and *kNN-SDSCL* include in the training phase information on the input image and the *UCI* database. Both quantitative results in Tables 5 and 6 are realized with *Pratheepan* as testing set. In the first Table 5 the complete provided ground truth has been considered. In the second Table 6 the results are regarding only the correct classification around the silhouette of the subjects skin region. Qualitative results of three different skin detection databases *Pratheepan*, *dbSkinChile* and *Faces* (see their description in Section 4.1) are provided in Figures 13, 14 and 15.

The results concerning the complete ground truth of skin are shown in Table 5. The best performance regarding accuracy, precision and F1 measure is our *tree-SDSCL*. All classification learners outperfom the state-of-the-art explicit thresholding methods regarding the *Pratheepan* database as testing set. The explicit thresholding methods *thresholdYCbCr* and *thresholdRGB* are prone to generally classify more pixels as skin, leading to a high value of true positives but also false positives. This can also be observed in the precision value, which considers the false positive rate in its calculation.

Table 5: Evaluation on the testing database *Pratheepan* concentrating on the complete ground truth.

Approach	Accuracy	Precision	Recall / TPR	FPR	F1
tree-SDSCL	**0.934**	**0.852**	0.848	**0.052**	**0.841**
kNN-SDSCL	0.926	0.818	**0.869**	0.067	0.831
tree	0.908	0.796	0.842	0.080	0.797
kNN	0.910	0.794	0.860	0.083	0.807
thresholdYCbCr	0.690	0.348	0.774	0.356	0.450
thresholdHSV	0.738	0.319	0.419	0.215	0.323
thresholdRGB	0.695	0.330	0.657	0.320	0.409

The results of Table 6 are evaluating the classification only around the silhouettes ground truth. Our proposed classification learners do not outperform the explicit thresholding methods even though the same testing database of

Pratheepan has been used for both evaluations (Tables 5 and 6). The difference is the alteration of the ground truth as described in Section 4.5 and observing the results of the F1-measure the best performing algorithm is *thresholdRGB*. Also regarding the recall value *thresholdYCbCr* and *thresholdRGB* perform better for a small margin. This is due to the high number of true positives, which are considered in the calculation of the recall value. Regarding accuracy and precision the supervised classification learner based on decision tree *tree-SDSCL* outperforms the other algorithms.

Table 6: Evaluation on the testing database *Pratheepan* concentrating on the silhouette as ground truth.

Approach	Accuracy	Precision	Recall / TPR	FPR	F1
tree-SDSCL	**0.797**	**0.799**	0.778	0.205	0.772
kNN-SDSCL	0.788	0.771	0.801	0.245	0.770
tree	0.764	0.757	0.778	0.264	0.743
kNN	0.765	0.753	0.793	0.274	0.751
thresholdYCbCr	0.698	0.64	**0.914**	0.515	0.745
thresholdHSV	0.720	0.754	0.600	**0.181**	0.644
thresholdRGB	0.775	0.732	0.882	0.333	**0.789**

To give here a further comparison, in the latest survey of skin-color modeling and detection methods by Kakumanu et al. [21], the authors compare skin detection strategies and their performance in terms of the *true positive rate (TPR)* and *false positive rate (FPR)*. Obviously it is difficult to compare these different published methodologies, since their is no uniform benchmark dataset on skin detection like there is on general image segmentation and boundary detection (Berkley Segmentation Dataset and Benchmark [31]). Therefore we have to keep in mind that the results listed in this report are all concerning their own dataset with respective ground truth.

The best performing algorithms regarding the quantitative results listed in the report, show a confidence value of around 88.5%-99.4% TPR and 10%-15.5% FPR. In our report regarding the *Pratheepan* dataset on the complete ground truth (see Table 5), we can observe that for *tree-SDSCL* a 84.8% TPR, which is for a small margin below the state-of-the-art results reported in the survey, and 5.2% FPR is achieved, which shows better performance. For a more detailed discussion on the skin detection methods and their comparison we allow to refer to Kakumanu et al. [21]'s survey.

In Figures 13, 14 and 15 from three different testing databases *Pratheepan*, *dbSkinChile* and *Faces* respectively three sample input images were selected, giving a total of nine qualitative results. They are arranged as follows: for every sample (1)-(3) in a figure the first row illustrates the original image, ground truth of skin and the generated silhouette representing the region of interest (pink being skin pixels and yellow non-skin pixels). The second row shows the results of the classification learners (*tree-SDSCL*, *kNN-SDSCL*, *tree* and *kNN*) and the thresholding methods (*thresholdYCbCr*, *thresholdHSV* and *thresholdRGB*). In the results skin pixels are visualized as white and non-skin pixels as black. Regarding the area around the silhouette, green pixels are correctly classified pixels (true positives and true negatives), and misclassified pixels are red (false positives and false negatives).

For the qualitative examples illustrated in this report we selected images with a variety of different skin tones, background and illumination to give a good representation on the tested samples. In the second sample in Figure 13 the face is illuminated from the side causing a shadow in the background and different skin tone patches in the face of the subject. For the explicit thresholding algorithms as well as *tree*, *kNN* these areas are difficult to distinguish and classify correctly. Also the simple background shows difficulties for the thresholding algorithms (observe *thresholdYCbCr* in Figure 13(2) and *thresholdRGB* in Figure 13(3)) as for our improvements background information is included into the training set and therefore the remaining background after ACM is classified correctly. Comparing the results from *Pratheepan* dataset in Figure 13 of the classification learners *tree*, *kNN* with our improved *tree-SDSCL* and *kNN-SDSCL*, it can be concluded that our improvements have an noticeable impact on the true and false positive skin detection and are concerning these samples also performing better than state-of-the-art.

Looking at the qualitative results from testing dataset *dbSkinChile* in Figure 14 similar observation can be made. In sample (1) and (3) *tree-SDSCL* and *kNN-SDSCL* show an improvement regarding false positives compared to the simple classification learning results of *tree* and *kNN* and their results are also noticeably better than state-of-the-art. The second sample (in Figure 14(2)) is actually a negative example, where *tree-SDSCL* does not improve results around the silhouette and performs worse than *thresholdYCbCr*.

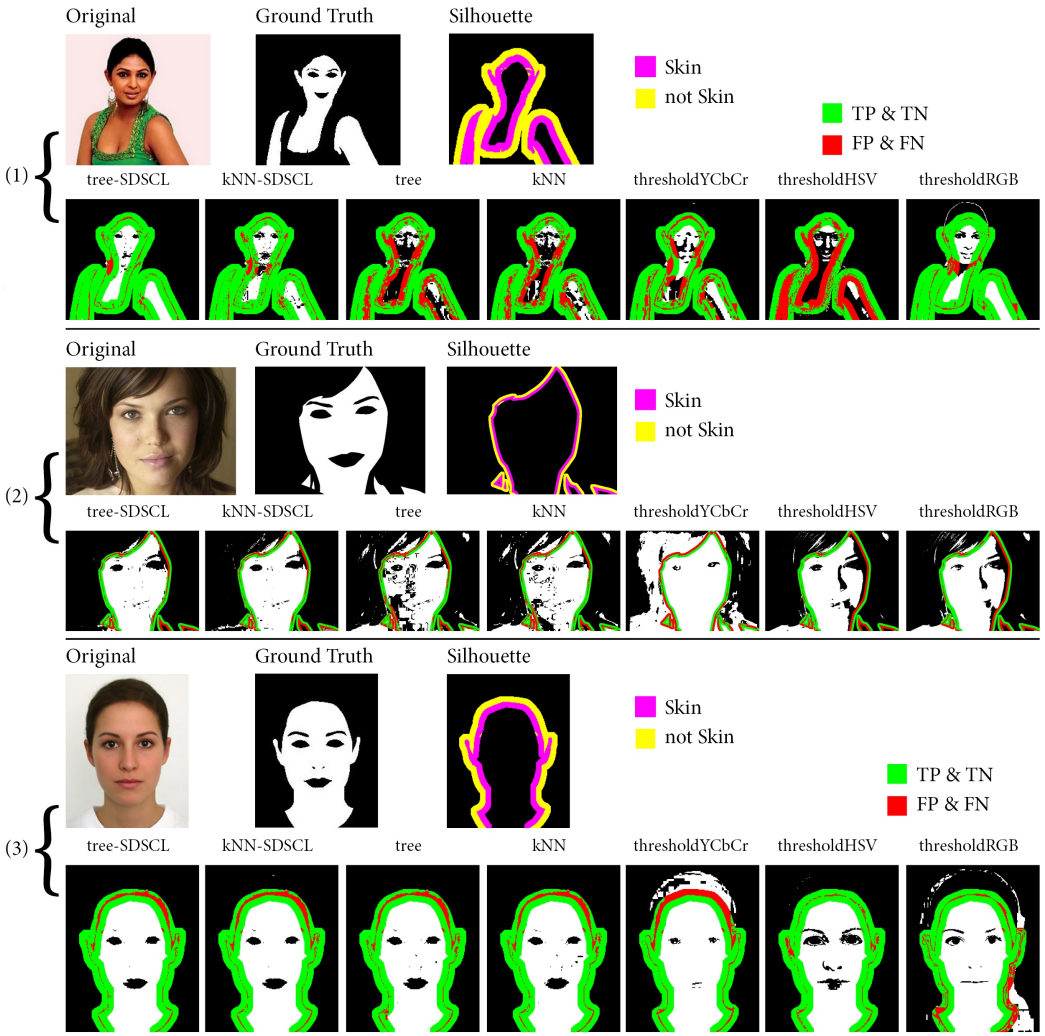

Figure 13: Qualitative results from *Pratheepan*: White pixels are skin, black non-skin and around the silhouette green represent all true positives (TP) and true negatives (TN) and red all false positives (FP) and false negatives (FN).

The qualitative results in Figure 15 concerning testing database *Faces*, show the difficulty with complex background. Specially for the first two samples (1) and (2) the explicit thresholding methods, *tree* and *kNN* fail completely in distinguishing skin from background, whereas *tree-SDSCL* and *kNN-SDSCL* show significant improvements, leading to acceptable results.

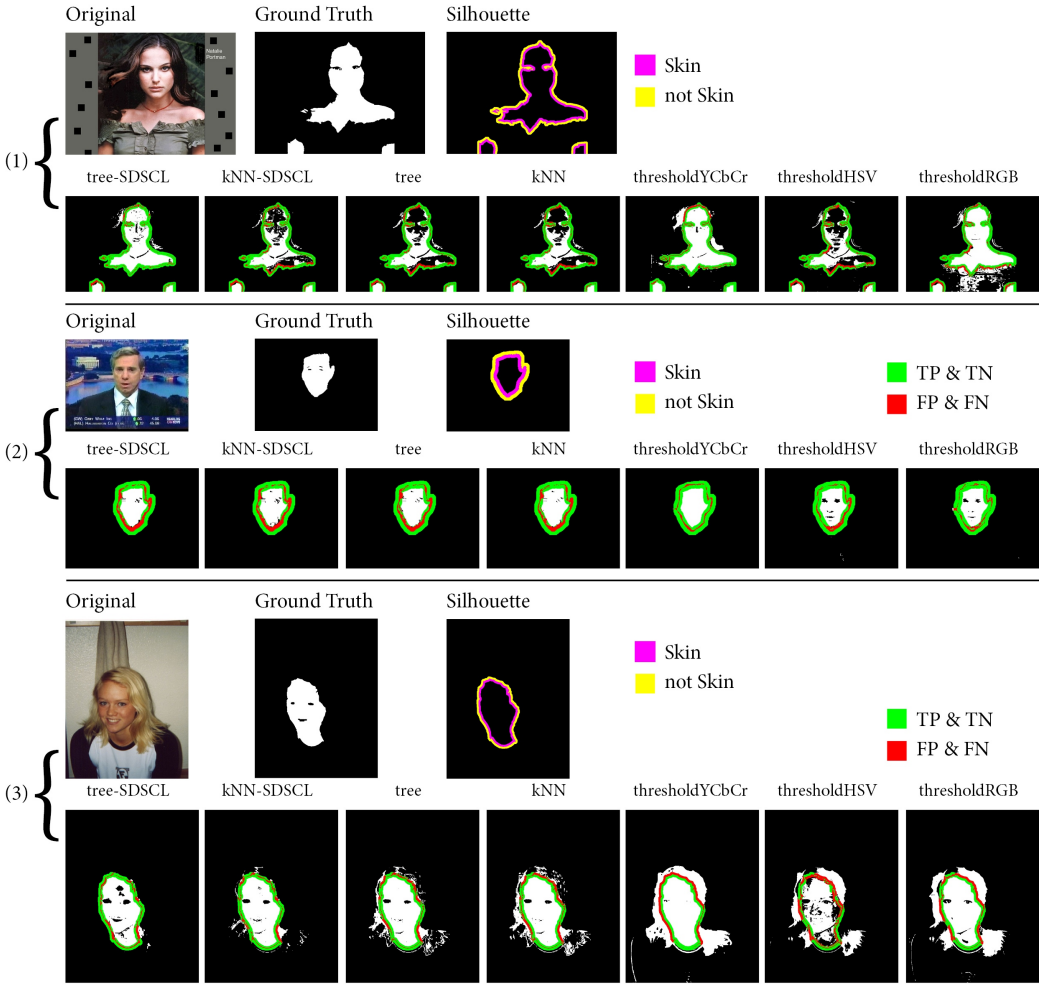

Figure 14: Qualitative results from *dbSkinChile*: White pixels are skin, black non-skin and around the silhouette green represent all true positives (TP) and true negatives (TN) and red all false positives (FP) and false negatives (FN)

It can be concluded that our novel supervised skin classifier improve results significantly when we are dealing with complex backgrounds, different ethnicities and different illumination conditions. This simple state-of-the-art approaches and the simple classification learners *tree* and *kNN* have problems distinguishing between skin-similar pixels in the background and actual skin pixels of the person since no contextual information is available. Nearly all nine

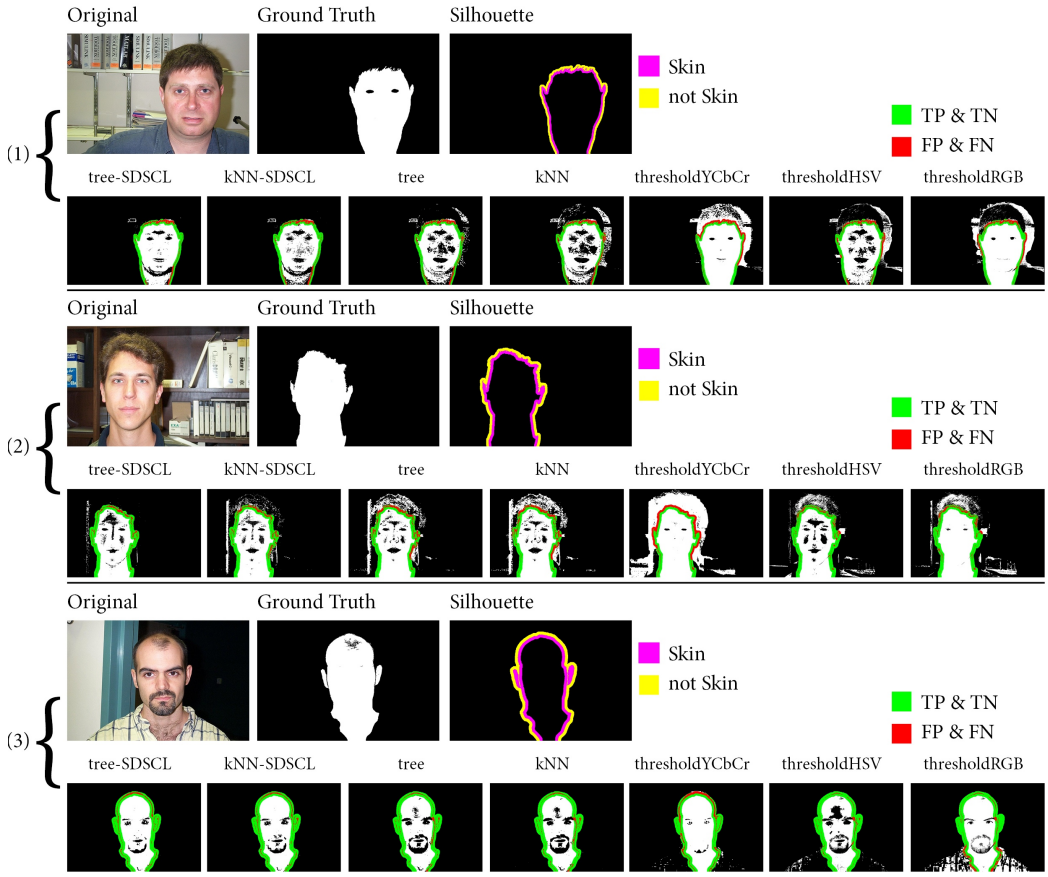

Figure 15: Qualitative results from *Faces*: White pixels are skin, black non-skin and around the silhouette green represent all true positives (TP) and true negatives (TN) and red all false positives (FP) and false negatives (FN)

examples demonstrate the typical behavior of *tresholdYCbCr* and *thresholdRGB* classifying more pixels as skin, leading to a high true positive and false positive rate.

Allowing too much or too little light during exposure makes images darker or brighter, respectively changing the natural tone of skin. A color space such as YCbCr allows to compensate this problem by splitting color into the luminance and chrominance components. In Figure 16 an example of over- and under-exposed image can be seen, where the *thresholdYCbCr* results are spurious not finding most of the skin pixels. Using the idea of classification learners in par-

ticular looking at *tree* the results are even worse, but after adding high-level information (skin pixels and background pixels of the input image) in *tree-SDSCL* the results improve but having still erroneous regions.

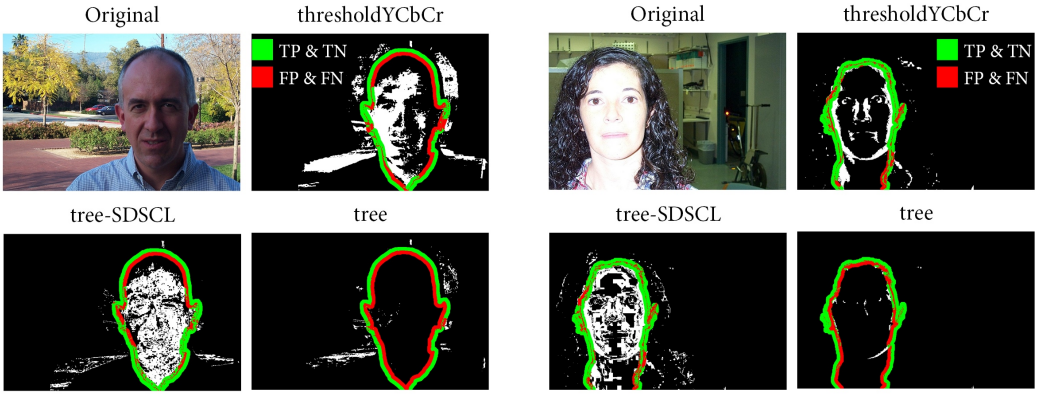

Figure 16: Examples of a under- and overexposed image where the results of *thresholdYCbCr* and *tree* fail completely. *Tree-SDSCL* improves the skin detection but not sufficiently enough.

5 Summary and Conclusions

Over the years different human skin segmentation solutions have been introduced, but most of them are prone to errors and are not able to cope with the variety of challenges arising with human skin on camera.

Firstly, **low accuracy**: when there is a wide variety of skin colors across different ethnicity, complex backgrounds and a variation of illumination conditions, false positive skin detection is a common problem.

Secondly, **luminance-invariant space**: through the use of luminance invariant color spaces, like the use of the chrominance components Cb and Cr in the color space YCbCr, results reveal that in certain instances they achieve more robustness, but in others the absence of the luminance component decreases performance [19].

Thirdly, **require large training set**: model-based methods require training sets, and there is always a trade-off between the size of the training set and classifier performance [39].

Fourthly and last, **high computational cost**: specially region-based methods incorporate neighboring pixels and are often computationally expensive [36]. There is no general method suited for any problem regarding skin detection.

Skin segmentation is seen in this report as a preprocessing step of the following master thesis *Automatic human-head and shoulder segmentation of frontal-view face images*, where the importance lies on the *silhouette of the face* rather than the correct classification of skin pixels *in* the face of the subject. This report presents a novel approach based on supervised classification learners, called Skin Detector based on Supervised Classification Learners (SDSCL). We propose to extend the training set of the kNN and decision tree classifiers by adding automatically labeled subset of pixels extracted from the query image. The skin pixels are cropped from the area below and between the eyes region that is found by Viola-Jones detector. The non-skin pixels are extracted from the area outside the face contour that is found by ACM landmark detector.

Evaluations on multiple datasets with frontal-view face images were discussed, and results were compared with explicit thresholding methods. Furthermore we discussed the results with skin detection strategies summarized in the survey report by Kakumanu et al. [21] measuring the performances in terms of TPR and FPR. The evaluation shows improvements over several baselines and is above the average of the best performing state-of-the-art algorithms re-

garding FPR. Including information of the input image into the training set and applying SDSCL on the remainder of the image, allows to reduce the number of false positive detections significantly and the classification results around the silhouette become more reliable.

References

[1] M. Abdel-Mottaleb and A. Elgammal. Face detection in complex environments from color images. In *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*, volume 3, pages 622–626 vol.3, 1999. doi: 10.1109/ICIP.1999.817190.

[2] M. Abdullah-Al-Wadud, Mohammad Shoyaib, and Oksam Chae. A skin detection approach based on color distance map. *EURASIP J. Adv. Signal Process*, 2008:199:1–199:10, January 2008. ISSN 1110-8657. doi: 10.1155/2008/814283. URL http://dx.doi.org/10.1155/2008/814283.

[3] N. S. Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185, 1992. doi: 10.1080/00031305.1992.10475879. URL http://www.tandfonline.com/doi/abs/10.1080/00031305.1992.10475879.

[4] E. Angelopoulo, R. Molana, and K. Daniilidis. Multispectral skin color modeling. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 2, pages II–635–II–642 vol.2, 2001. doi: 10.1109/CVPR.2001.991023.

[5] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. *When Is "Nearest Neighbor" Meaningful?*, pages 217–235. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999. ISBN 978-3-540-49257-3. doi: 10.1007/3-540-49257-7_15. URL https://doi.org/10.1007/3-540-49257-7_15.

[6] R. B. Bhatt, G. Sharma, A. Dhall, and S. Chaudhury. Efficient Skin Region Segmentation Using Low Complexity Fuzzy Decision Tree Model. In *2009 Annual IEEE India Conference*, pages 1–4, December 2009. doi: 10.1109/INDCON.2009.5409447.

[7] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[8] Michael H. Brill. The relation between the color of the illuminant and the color of the illuminated object. *Color Research and Application*, 20(1):70–76, 1995. ISSN 1520-6378. doi: 10.1002/col.5080200112. URL http://dx.doi.org/10.1002/col.5080200112.

[9] J. Chatrath, P. Gupta, P. Ahuja, A. Goel, and S. M. Arora. Real time human face detection and tracking. In *2014 International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 705–710, February 2014. doi: 10.1109/SPIN.2014.6777046.

[10] Abhinav Dhall, Gaurav Sharma, Rajen Bhatt, and Ghulam Mohiuddin Khan. *Adaptive Digital Makeup*, pages 728–736. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-642-10520-3. doi: 10.1007/978-3-642-10520-3_69. URL https://doi.org/10.1007/978-3-642-10520-3_69.

[11] A. Diplaros, T. Gevers, and N. Vlassis. Skin detection using the EM algorithm with spatial constraints. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 4, pages 3071–3075 vol.4, October 2004. doi: 10.1109/ICSMC.2004.1400810.

[12] Ahmed Elgammal, Crystal Muang, and Dunxu Hu. Skin detection-a short tutorial. *Encyclopedia of Biometrics*, pages 1–10, 2009.

[13] Francesca Gasparini and Raimondo Schettini. Skin segmentation using multiple thresholding. In *Internet Imaging VII, Proceedings of SPIE*, volume 6061, pages 128–135, 2006.

[14] Rafael C. Gonzalez, Richard E. Woods, and Steven L. Eddins. *Digital Image Processing using MATLAB*. Pearson, 2004.

[15] Hayit Greenspan, Jacob Goldberger, and Itay Eshet. Mixture model for face-color modeling and segmentation. *Pattern Recognition Letters*, 22 (14):1525–1536, 2001.

[16] B. Gunsel, A. M. Ferman, and A. M. Tekalp. Video indexing through integration of syntactic and semantic features. In *, Proceedings 3rd IEEE Workshop on Applications of Computer Vision, 1996. WACV '96*, pages 90–95, December 1996. doi: 10.1109/ACV.1996.572007.

[17] Erik Hjelmås and Boon Kee Low. Face detection: A survey. *Computer vision and image understanding*, 83(3):236–274, 2001.

[18] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: with Applications in R*. Springer, New

York, 1st ed. 2013, corr. 7th printing 2017 edition, September 2017. ISBN 978-1-4614-7137-0.

[19] S. Jayaram, S. Schmugge, M. C. Shin, and L. V. Tsap. Effect of colorspace transformation, the illuminance component, and color modeling on skin detection. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–813–II–818 Vol.2, June 2004. doi: 10.1109/CVPR.2004.1315248.

[20] Dennis Jensch, Daniel Mohr, and Gabriel Zachmann. A Comparative Evaluation of Three Skin Color Detection Approaches. *Journal of Virtual Reality and Broadcasting*, 12(2015)(1), January 2015. ISSN 1860-2037. doi: 10.20385/1860-2037/12.2015.1. URL `http://www.jvrb.org/past-issues/12.2015/4088`.

[21] P. Kakumanu, S. Makrogiannis, and N. Bourbakis. A Survey of Skin-color Modeling and Detection Methods. *Pattern Recogn.*, 40(3):1106–1122, March 2007. ISSN 0031-3203. doi: 10.1016/j.patcog.2006.06.010. URL `http://dx.doi.org/10.1016/j.patcog.2006.06.010`.

[22] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, January 1988. ISSN 0920-5691, 1573-1405. doi: 10.1007/BF00133570. URL `https://link.springer.com/article/10.1007/BF00133570`.

[23] Rehanullah Khan, Allan Hanbury, and Julian Stöttinger. Universal seed skin segmentation. *Advances in Visual Computing*, pages 75–84, 2010.

[24] Rehanullah Khan, Allan Hanbury, and Julian Stöttinger. Skin detection: A random forest approach. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 4613–4616. IEEE, 2010.

[25] Jae Y. Lee and Suk I. Yoo. An elliptical boundary model for skin color detection. In *In Proc. Int. Conf. on Imaging Science, System and Technology*, 2002.

[26] Y. Lei, W. Xiaoyu, L. Hui, Z. Dewei, and Z. Jun. An algorithm of skin detection based on texture. In *2011 4th International Congress on Image and Signal Processing*, volume 4, pages 1822–1825, October 2011. doi: 10.1109/CISP.2011.6100627.

[27] C. Liensberger, J. Stöttinger, and M. Kampel. Color-based and context-aware skin detection for online video annotation. In *2009 IEEE International Workshop on Multimedia Signal Processing*, pages 1–6, October 2009. doi: 10.1109/MMSP.2009.5293337.

[28] Wan Lü and Jie Huang. Skin detection method based on cascaded adaboost classifier. *Journal of Shanghai Jiaotong University (Science)*, 17(2): 197–202, Apr 2012. ISSN 1995-8188. doi: 10.1007/s12204-012-1252-6. URL https://doi.org/10.1007/s12204-012-1252-6.

[29] Binbin Ma, Changqing Zhang, Jingjing Chen, Ri Qu, Jiangjian Xiao, and Xiaochun Cao. Human skin detection via semantic constraint. In *Proceedings of International Conference on Internet Multimedia Computing and Service*, ICIMCS '14, pages 181:181–181:184, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2810-4. doi: 10.1145/2632856.2632885. URL http://doi.acm.org/10.1145/2632856.2632885.

[30] I.G. Maglogiannis. *Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in EHealth, HCI, Information Retrieval and Pervasive Technologies*. Frontiers in artificial intelligence and applications. IOS Press, 2007. ISBN 9781586037802. URL https://books.google.at/books?id=vLiTXDHr_sYC.

[31] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[32] Christian Platzer, Martin Stuetz, and Martina Lindorfer. Skin Sheriff: A Machine Learning Solution for Detecting Explicit Images. In *Proceedings of the 2Nd International Workshop on Security and Forensics in Communication Systems*, SFCS '14, pages 45–56, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2802-9. doi: 10.1145/2598918.2598920. URL http://doi.acm.org/10.1145/2598918.2598920.

[33] Javier Ruiz-del-Solar and Rodrigo Verschae. Skin detection using neighborhood information. In *Proceedings of the 6th International Conference on Automatic Face and Gesture Recognition (FG2004)*, pages 463–468, Seoul, Korea, May 17-19 2004.

[34] Javier Ruiz-del-Solar and Rodrigo Verschae. Robust skin segmentation using neighborhood information. In *the Eleventh International Conference on Image Processing (ICIP 2004), pp. 207-210, October 24-27, 2004, Singapore, IEEE Press.*, 2004. URL `files/cr2455.pdf`.

[35] Javier Ruiz-del-Solar and Rodrigo Verschae. SKINDIFF. Robust and Fast Skin Segmentation. Technical report, UCH-DIE-VISION-2006-01, Universidad de Chile, 2006. URL `files/TR_UCH-DIE-VISION-2006-01.pdf`.

[36] F. Saxen and A. Al-Hamadi. Color-based skin segmentation: An evaluation of the state of the art. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 4467–4471, October 2014. doi: 10.1109/ICIP.2014.7025906.

[37] Khamar Basha Shaik, P. Ganesan, V. Kalist, B. S. Sathish, and J. Merlin Mary Jenitha. Comparative Study of Skin Color Detection and Segmentation in HSV and YCbCr Color Space. *Procedia Computer Science*, 57:41–48, January 2015. ISSN 1877-0509. doi: 10.1016/j.procs.2015.07.362. URL `http://www.sciencedirect.com/science/article/pii/S1877050915018918`.

[38] Lothar Spillmann. *Visual Perception: The Neurophysiological Foundations*. Academic Press, 1990. ISBN 978-0-12-657676-4.

[39] W. R. Tan, C. S. Chan, P. Yogarajah, and J. Condell. A Fusion Approach for Efficient Human Skin Detection. *IEEE Transactions on Industrial Informatics*, 8(1):138–147, February 2012. ISSN 1551-3203. doi: 10.1109/TII.2011.2172451.

[40] Rein Van Den Boomgaard and Richard Van Balen. Methods for fast morphological image transforms using bitmapped binary images. *CVGIP: Graphical Models and Image Processing*, 54(3):252–258, 1992.

[41] Paul Viola and Michael Jones. Robust Real-time Object Detection. In *International Journal of Computer Vision*, 2001.

[42] Ming-Hsuan Yang, D. J. Kriegman, and N. Ahuja. Detecting faces in images: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, January 2002. ISSN 0162-8828. doi: 10.1109/34.982883.

[43] Tae-Woong Yoo and Il-Seok Oh. A fast algorithm for tracking human faces based on chromatic histograms. *Pattern Recogn. Lett.*, 20(10):967–978, October 1999. ISSN 0167-8655. doi: 10.1016/S0167-8655(99)00053-7. URL http://dx.doi.org/10.1016/S0167-8655(99)00053-7.