

Color-Based Skin Detection and its Application in Video Annotation

Christian Liensberger, Julian Stöttinger, and Martin Kampel

Institute of Computer Aided Automation,
Pattern Recognition and Image Processing Group,
University of Technology Vienna, Austria
{liensberger, julian, kempel}@prip.tuwien.ac.at

Abstract *Skin detection in visual data cannot be solved by analyzing the low level image features only: In an extensive online experiment we are able to show that even humans are not able to detect skin color reliably without knowing the context of their perception. To compensate for this major drawback of many approaches, we combine a state of the art recognition algorithm with color model based skin detection. Detected faces in videos are the basis for adaptive skin color models, which are propagated throughout the video, providing a more precise and accurate model in its recognition performance than pure color based approaches. The approach is able to run in real-time and does not need prior data-specific training. We received challenging online videos from an online service provider and use additional videos from public web platforms covering a grand variety of different skin colors, illumination circumstances, image quality and difficulty levels. In an extensive evaluation we estimated the best performing parameters and decided on the best model propagation techniques. We show that adaptive model propagation outperforms static low level detection.*

1 Introduction

User generated content became very popular in the last decade. With the international success of several Web 2.0 websites (platforms that concentrate on the interaction aspect of the internet) the amount of publicly available content from private sources is growing rapidly. One of the visible trends is that the number of videos that are being uploaded every day is way beyond the means of the operating companies that are in charge to classify the content properly.

The mostly used approach to content blocking on the internet is based on contextual keyword pattern matching technology that categorizes URLs by means of checking contexts of web pages or video names and then traps the websites [18]. This does not hold true for websites which allow uploading videos like e.g. YouTube and MSN Video because the uploaded videos do not explicitly need to report (valid) keywords for the content. Due to no reliable automated process is available the platforms rely on their user community. This method is rather slow and does not guarantee that inappropriate videos are blocked from the beginning.

Therefore, an automated method to categorize videos based on skin color will help the service providers to gain more information about the videos' content as soon as they have been uploaded to the platform.

Skin color detection is also used as a preliminary step in a broad range of computer vision tasks, including gesture analysis, tracking, or content-based image retrieval systems [9]. We evaluate a fast and straightforward adaptive skin detection method for videos. We adapt our decision rules upon a first step of face detection using the well known approach from Viola et al. [32]. We propose a method which takes high advantage of the temporal relationship between frames in an image sequence and deals well with time dependent illumination changes.

There are other approaches that rely upon a successful face detection for skin classification e.g. [20, 34]. For a fast classification technique for low quality videos using multiple models at one time [17] we extend their propagation model and add trainable parameters to the framework. Additionally, we use different color spaces which are combined using voting. It can be carried out as real-time classification of the video and may therefore be highly useful for an automated pre-selection and classification for large video databases.

Further, we deal with multiple models at one time. Per detected face, one new decision model is applied and broadens our classification knowledge about the scene and possible skin which might appear in the video. Additionally, we vote between multiple color spaces.

The main drawback of using static color decisions is the high number of false positive detections [27]. The multiple model approach can reduce this number dramatically which is shown in an extensive evaluation.

We suggest a method for extracting meaningful key frames from the videos for the possible task of filtering adult content in the video. Based on the results of the skin coverage graph we are able to extract the meaningful frames for further manual classification. Our algorithm even generates an output that groups frames with the similar amount of skin together. This allows easier post processing of the video and data.

In the following Section 2 we describe the state of the art in low level skin detection and its adaption towards time-varying color circumstances and video segmentation. Section 3 describes the multiple model approach for fast skin

detection. The experiments and results are outlined in Section 4. A conclusion is given in Section 5.

2 Related Work

The main goal of skin color detection or classification for skin contents filtering is to build a decision rule that will discriminate between skin and non-skin pixels. There are different ways to classify skin by color in frames of videos. They can be grouped into three types of skin modeling: parametric, nonparametric 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 [14]. The second approach, nonparametric methods, estimate the skin color from the histogram that is generated by the used training data [15].

An easy and most often 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. [23] and explicitly defines the boundaries of the skin clusters in a given color space. The underlying hypothesis here is that skin pixels have similar color coordinates in the chosen color space, which means that skin pixel are found within a given set of boundaries in a color space. The main drawback of this method is not the resulting true positives, but a comparably high number of false detections [16]. We are able to compensate for this issue in our approach by using a multiple adaptive model approach.

Color is a low level feature [37]. It is broadly used for real-time object characterization, detection and localization [18]. Following Kakumanu et al. [16] the major difficulties in skin color detection are caused by various effects:

Illumination circumstances: Any change in the lighting of a scene changes color and intensity of a surface's color and therefore changes the skin color present in the image. This is known as the color constancy problem and is the most challenging one in detecting skin detection. There have been different approaches [33] to overcome this problem. We use our own approach to make sure that we deal with different illumination properly.

Camera characteristics: The color distribution of a picture is highly dependent on the sensitivity and the internal parameters of the capture device.

Ethnicity: The great variety of skin color from person to person and between ethnic groups challenges the classification approaches. There are different techniques available that try to minimize this problem. We apply some of them into our algorithm to work around this problem.

Individual characteristics: Age, sex and body parts affect the skin color appearance. Detecting context (such as face or other body parts) helps to overcome some of these problems.

Other factors: Makeup, hairstyle, glasses, sweat, background colors, and motion influence the skin color. Context is also very important here. Detecting faces and classifying the content of the face as skin helps to overcome these problems.

Regarding user-generated video content, we face addi-

tional problems:

Video resolution of 640x360 pixel and below: The video material is often shot by amateurs with mobile devices or cheap cameras that operate at the resolution of 1 mega pixel or below. Online portals restrict the resolution of their videos to minimize their server load and bandwidth. YouTube, for example, recommends are resolution of 640x360 for 16:9 or 480x360 for 4:3¹. This can be seen as the limit that is provided by the platform but does not mean that the quality that is uploaded by the users is at that level.

Amount of noise: Capture devices with low aperture like mobile phones and web cams produce a higher amounts of noise than professional devices that are properly equipped. This is especially present in scenes with low illumination and contrast. Further, these cameras most often are not equipped with additional lights and therefore tend to produce dark videos to begin with. Additionally, many videos are compressed several times in the work flow of user-generated video publishing. YouTube recommends to upload already pre-compressed videos to their platform¹ but does another compression step automatically after the video has been uploaded to the portal.

Amount of data: Dealing with the typical amount of data that has to be processed by a video platform (YouTube allows uploads of files up to 1 GB¹) the runtime of the algorithm should be real-time or faster to be of use for this task.

No presumptions: In many tasks of computer vision, certain presumptions of the appearance of skin or scene circumstances can be made. Online users upload anything they want, nothing about the content can be assumed.

An approach for detecting skin in this kind of content has therefore to be fast, reliable and needs to be stable against noise and artifacts caused by compression. Additionally, it must be very flexible against varying lighting conditions.

2.1 Skin color

To model and classify skin color properly the choice of the appropriate color space is crucial. Yang et al. [36] observed that during skin detection the intensity is more likely to change than chrominance. Therefore, most of the approaches disregard the intensity information in their detection process. Additionally they showed that clusters in normalized RGB are an appropriate model for skin color and therefore many successful approaches rely on this color space e.g. [25, 2, 3]. Still, the normalized RGB color space suffers from instability with dark colors.

Color spaces like the HS^* family model the RGB cube onto a transformed color space by following perceptual features. The *Hue* component gives the perceptive idea of a color as humans are able to define the hue value as the tone of the color. The *Saturation* gives a perceptual measure of the colorfulness. The HS^* color spaces are known to be invariant to illumination change. This property is helpful in the process of skin detection and that is why they are often used to detect skin in images. Examples are found in [2, 8, 9].

Orthogonal color spaces like YC_bC_r , YC_gC_r , YIQ ,

¹http://www.youtube.com/handbook_popup_produce_upload?pcont=bestformats

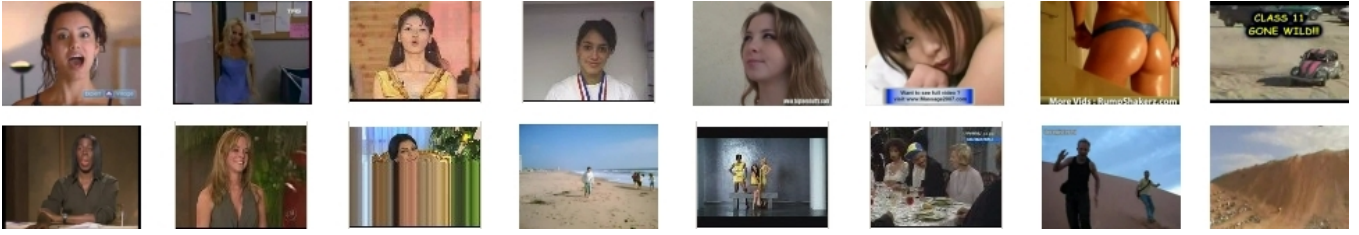


Figure 1: Example frames from the video dataset used.

YUV , YES try to form as independent components as possible. YC_bC_r is one of the most successful approaches for skin-detection and used by e.g. [35, 26].

According to [12, 13, 1] a single color space may limit the performance of the skin color filter and that better performance can be achieved using two or more color spaces. Using the most distinct invariant color coordinates of different color spaces increases the performance under certain conditions. The combination of different color spaces eliminates also a lot of the false positives since the combination stabilizes the area that is used for skin detection.

2.2 Skin detection

In many approaches pixel level skin detection is used as one of the first step for a successful face detection. Among others [9, 4, 35, 26, 36] use the detection of skin color for the estimated position of the faces and further face detection, face recognition and gesture tracking. This is a valuable assumption since the human face is most of the time not completely covered and at least some skin is visible.

Viola et al. [32] introduced a very successful and stable face detection algorithm based on their “integral image”, Haar-like features and a cascade structure that applies more specialized filters as the cascade is walked. The algorithm is applicable in real-time (see Section 3). The performance and simplicity of the face detector inspired several authors for using this approach as an initial step for further skin color estimation [20, 34]. In contrary to our approach they use more sophisticated classifiers, which rely on different assumptions and just one model at a time.

Skin detection under varying illumination in image sequences is addressed in [27, 36, 28, 33]. Some of these approaches try to map the illuminance of the image into a common range. They want to compensate the variance in the image’s illuminance to make sure that skin in all the different images expose the same illumination and tone. These methods work under certain conditions but also fail completely under others, like e.g. the color of the skin is reproduced with a bluish tone.

Neural networks [19], Bayesian Networks e.g. [25], Gaussian classifiers e.g. [15] or self organizing maps (SOM) [2] are high level classifiers that try to overcome some of the issues of low level classifier and try increase the classification accuracy. These methods have often the issue of being too slow for real-time classification and therefore are not suitable for high speed classification as required in the described scenarios.

3 Method

We address the problem of changing light conditions, different skin colors and varying image quality in videos in adapting the skin color model according to reliably detected faces. Prior to any detected face, the combination of the static YC_bC_r , normalized RGB and RGB skin model (see Section 3.3) is applied for skin detection. These three color spaces are used in a combination and if two of them vote for a pixel to be skin it is classified as such. This has been evaluated and found to be more robust than only using one of the three color spaces. Since there are videos where no face is visible this “voting algorithm” is used for the whole video.

Due to its real-time performance, Khan et al. [17] use the Viola et al. [32] face detector for parameters cue in their model propagation. Wimmer et al. [34] point out that the performance of this detection algorithm allows a precise and reliable estimation of the skin color. Any detected face introduces a new skin color model, which allows to detect skin of different color and under different light conditions.

3.1 Skin sample localization through face detection

Viola-Jones [32] describe a face detection algorithm that uses three new contributions to detect faces without the usage of the color in an image.

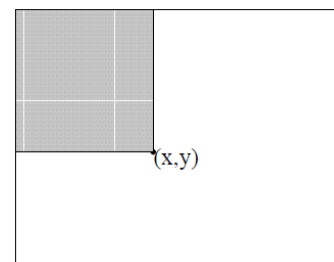


Figure 2: The integral image that is computed for the pixel at the coordinate x and y . [32]

The first contribution is a new image representation that they call “integral image” (see Figure 2). The idea of the integral image was motivated by the work of Papageorgiou et al. [22] because they suggest in their work not to directly use the intensities of the image when doing object detection. Instead, Viola-Jones use a set of features, which are similar to Haar basis functions but also extend them to complexer features that can not be modeled with the Haar basis functions only. The integral image was introduced to model these features very rapidly at many scales. It is very similar to the

summed area tables, which are used by some algorithms in computer graphics to do proper texture mapping [6]. Computing the integral image is done with only a few operations per pixel:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

where $ii(x, y)$ is the integral image and $i(x', y')$ is the original image (see Figure 2). By using the following pair of recursions:

$$\begin{aligned} s(x, y) &= s(x, y - 1) + i(x, y) & (2) \\ ii(x, y) &= ii(x - 1, y) + s(x, y) & (3) \end{aligned}$$

(where $s(x, y)$ is the cumulative row sum, $s(x, -1) = 0$ and $ii(-1, y) = 0$) the integral image can be computed in one pass over the original image.

Once the integral image is computed the Haar-like features can be evaluated in constant time.

The second contribution of the Viola-Jones' work is to construct a classifier that selects only a small subset of important features by using Ada-Boost [7]. To make the learning algorithm fast a large number of available features need to be ignored, since within any tested sub-window (represents a part of the whole image) the total number of Haar-like features is larger than the actual number of pixels in that part of the image. It is important that the learning algorithm focuses only on a small subset of critical features. The Ada-Boost learner is modified so that each returned weak classifier can depend on only one single feature. This modification is based on the work of Tieu and Viola [29]. As a result each stage of the boosting process that selects a new weak classifier can be seen as a feature selection process. The benefit of Ada-Boost is that it provides an effective learning algorithm and is also very strong on generalization performance [21]. That is why it is has been selected as learning algorithm for the face detector.

The third major contribution is a method for chaining more complex classifiers in a cascade structure, which dramatically increases the performance of the detector by only focusing on promising regions of the image. The main idea is to assume that a simple classifier is often able to detect whether an image contains an object of interest faster [30, 24, 11]. These promising regions are then the target of a more sophisticated processing. The image is split in a recursive way, where a more sophisticated classifier is then applied to each one of the regions. The regions that may contain interesting information are used for further processing. The key measure for this approach is the calculation of the "false negative" rate, while walking the cascade structure. In that case all, or most of the important object instances, are selected by the filter process.

Large frames tend to slow down the classifier. If a frame exceeds are certain size it is resized to speed up the face detection. We are also only using each second frame of the video (if the video frame rate is above a certain threshold) to speed up the face detection and later skin detection.

From time to time certain parts of the video are detected as face despite being one. This might happen since certain parts of the image, due to compression and artifacts, look

like faces to the algorithm. This problem is overcome by the face tracking algorithm. If a face's detection time span is under a certain time threshold it is discarded as noise and not used for the skin detection at all. The evaluation (see Section 4) helped us to find a threshold that is around 30-40 frames for the videos in our dataset. With that threshold all of the noise got eliminated. The algorithm also, from time to time, detects a bigger area as face as the one that is found in the ground truth. This behavior might be caused by the compression and only rests for a few frames. To avoid that a mechanism has been put in place that detects this behavior and resizes the rectangle properly.

3.2 Color Space for Skin Color Tracking

Choosing a color space that is relatively invariant to minor illuminance changes is crucial to any skin color tracking system. The transformation simplicity and explicit separation of luminance and chrominance components makes YC_bC_r attractive for skin color modeling [31]. YC_bC_r is an encoded nonlinear RGB signal, commonly used by European television studios and for image compression work, such as JPEG (Joint Photographic Experts Group) and MPEG (Moving Picture Experts Group). Color is represented by luma which represents the luminance and computed from the nonlinear RGB . It is constructed as a weighted sum of the RGB values, and two color difference values C_b and C_r that are formed by subtracting luma from RGB red and blue components. For 24 bit color depth, the following values apply:

$$\begin{aligned} Y &= (0.299 * (R - G)) + G + (0.114 * (B - G)) \\ C_b &= (0.564 * (B - Y)) + 128 \\ C_r &= (0.713 * (R - Y)) + 128 \end{aligned}$$

The favorable property of this color space for skin color detection is the stable separation of luminance, chrominance, and its fast conversion from RGB . These points makes it suitable for our real-time skin detection scenarios.

3.3 Introducing a new face as model

There are two ways of introducing a face as a model. This depends on the preferences of the algorithm:

- *One Pass Skin Detection:* We rely on a very simple tracker and confidence check for reasons of the runtime of the algorithm: For every given detected area A_n in the frame number n is regarded as a new and trustful face if

$$(A_n \cap A_{n-1}) \wedge (A_n \cap A_{n+3}) \geq 0.5 \quad (4)$$

The parameters $n+3$ and 0.5 are chosen after the parameter training. This condition suppresses every background detection in the test set given in Section 4 as they tend to "flicker" through the scene. No true face is disregarded.

Once the face is lost the model is still applied, unless a new face is found. The idea behind this approach is that in subsequent frames the face detector may fail to detect a face, because of occlusion, face rotation etc. This simple approach works perfectly for practical purposes.

- *Two Pass Skin Detection*: The second approach runs through the video twice. During the first run the faces in the video are detected. Faces that are found in a frame are tracked through the video to make sure that "flickering" is avoided. In a second run the detected faces are used as models for the skin detection algorithm. Since all the faces have been detected during the first run sophisticated modeling approaches might be taken into account or the faces might be merged depending on the approach that is used and the speed that should be achieved.

The main assumption of this approach is that a detected face contains a certain amount of skin color and is the base for a new model. The Viola-Jones face detection system returns detected faces as a square that contains the face. The square is arranged in such a way that it covers hair and parts of the background to the left and right region of face.

The key problem here is to remove the hair and background information from the returned face, so that a simple means for the updated model can be computed in order to account for the real-time performance. We chose the simple rule of truncating the square to make sure that only the skin area of the face is returned. It effectively returns the face area, removing hair and side background: the area returned only contains the face region and predominantly skin pixels.

3.4 Adaptive skin color modeling

At the starting frame of a video we use the static range for skin detection. After a face has been detected its color is examined: The range for the C_b and C_r components (of the YC_bC_r color space) are used to generate a newly adapted range model. The Y component is ignored since it encodes only the luminance.

We use every detected face in each frame to adapt our model continuously as the lighting in the scene changes or a new face, which might mean a new kind skin-color, is introduced to the sequence.

We do not use the original C_b and C_r ranges that has been found in the face but rather generate the average values for each of them. The possible ranges are estimated by using a "clamping" value for each of the two channels. This is done since the face usually still contains certain parts that are not skin, such as possible open eyes, mouth, eye brows etc. The clamping values are percentage values of the static ranges for the C_b and C_r channels that have been published in several times [5]. We model the adapted skin color as a range by starting from the detected average and expanding it by the percentages defined in the clamping. In other words we take the static and general assumption and narrow it down by the average skin color and a reasonable clamping variance.

If we are not able to detect a face in the frame/video the static values are used for detection of skin.

3.5 Multiple Models

A separate model is used for each separate face. For example if there are three people in the frame, and three faces detected, a separate model is maintained for each face and the skin that is detected is based on one of these models. This approach solves the problem of multiple people with different skin tones.

4 Experiments

Our experiments include evaluation of different parameters for "clamping", evaluation of a set of color spaces and an online poll to understand how people classify skin. All the videos used in the experiments have been annotated by hand to have a valid ground truth that we can evaluate the videos against.

The results have been evaluated per pixel using four categories: *True Positive* are pixels classified as skin in the ground truth and by the algorithm. *True Negatives* are not classified as skin in the ground truth and by the algorithm. *False Positive* are not skin in the ground truth but detected by the algorithm. We encounter *False Negatives* when pixel are classified as skin in the ground truth but not by the algorithm.



Figure 3: Classification screen shot: White represents true negatives, green is true positives, blue is false positive and red is false negative

Figure 3 shows one frame with these four categories marked in different colors. With these numbers, calculated for each frame and each video, the two ratios "Precision" and "Recall" have been calculated:

$$Precision = \frac{tpcount}{tpcount + fpcount} \quad (5)$$

$$Recall = \frac{tpcount}{tpcount + fncount} \quad (6)$$

4.1 Skin classification by humans

To better understand the importance of context for humans when classifying something as skin we did an online poll where we asked people to rate pieces of images as whether they contain skin. To make sure a lot of people were reached a link to the poll was published in several forums where we knew that there is a board distribution of people coming from different regions of the world. In total we got 403 people participating who rated 18338 fragments.

The poll consisted of a set of random frames from the various videos in our dataset. These frames were then cut into very small pieces/windows (see Figure 4). Each person who participated got presented with a set of pieces (one by one and selected in a random manner) and asked whether or not that piece contained skin and how much skin was visible.

The results clearly pointed out that humans are somehow clueless whether or not something is skin without having a valid context. Without context people tend to fall back to only use color to understand whether something is skin and therefore in some scenarios, e.g. sand, fail completely (see Figure 5).

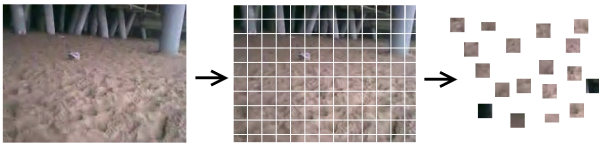


Figure 4: Images where cut into small pieces (windows) to remove the context.



Figure 5: Humans do not classify skin properly without context. The darker the red the more people thought it is skin.

4.2 Performance evaluation

The dataset that was used is a set of 25 videos. Half of it have been downloaded from YouTube². The other half has been given by an external company that is in need for a skin detection application for their online platform.

Most of the sequences also contain scenes with multiple people and/or multiple visible body parts and scenes shot both indoors and outdoors, with steady or moving camera. The lighting varies from natural light to directional stage lighting. Sequences contain shadows and minor occlusions. Collected sequences vary in length from 100 frames to 500 frames. The videos are generally challenging as they contain near skin color content as pink backgrounds, beaches, sand, cork boards or similar which are detected as false positives easily (see Figure 1).

For all of the videos has been generated ground truth. We wanted to use Viper GT³ but since it is only able to mark simple polygons we switched to Macromedia Flash⁴ because that allowed us to mark the skin in the video more precisely. Flash allowed us to output a binary ground truth video with a per pixel precision that was a lot easier to process than using the XML that Viper GT would have produced for us. We used this approach for making the evaluation a simple pixel by pixel comparison between the dynamically updated model and ground truth data. In a second step we also evaluated the static model with the ground truth. Figure 6 shows a graph that compares the skin percentage that has been found by the algorithm in one video with the ground truth that has been generated for that video. During the generation of the ground truth it was made sure that eye brows, open mouth and eyes were excluded if these were distinguishable in the video. Sometimes, due to the low resolution of the video it was not possible to exactly mark some of these non-skin elements (see Figure 7).

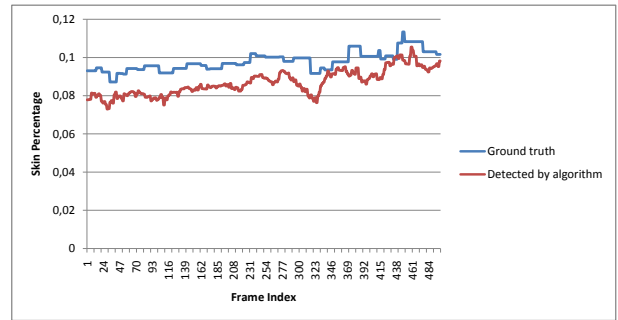


Figure 6: Graph showing the skin distribution percentage compared with the ground truth data in one of the videos of our dataset. This video has been run with face detection. For the C_b clamping has been used 30% and for the C_r clamping 17.5%.

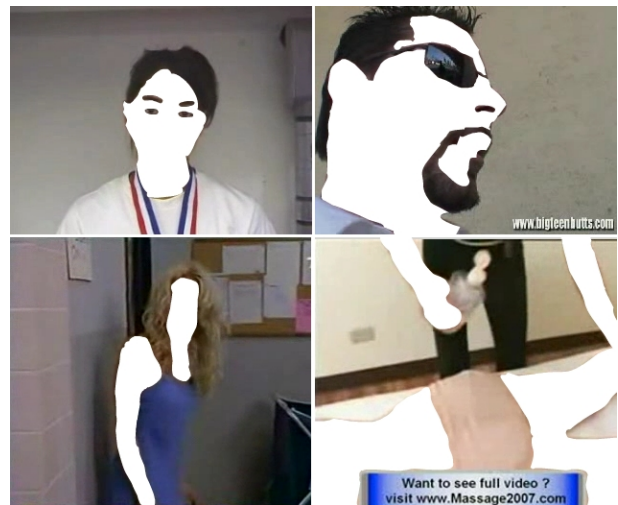


Figure 7: Image frames showing the generated ground truth

4.3 Clamping parameter evaluation

We tested the algorithm against our dataset with various percentage values for C_b and C_r and got the best results by using 30% for C_b and 17.5% for C_r . All parameters have been evaluated with the two skin detection approaches (one pass skin detection, two pass skin detection). The full set of results for the one pass skin detection approach is found in Figure 8 and the results for the two pass skin detection are found in Figure 9. The results show that the one pass skin detection algorithm performs very similar to the one that needs two passes. The results are very similar because the largest amount of variation in lighting is only encoded in the Y component of the selected YC_bC_r color space and therefore does not influence the C_b or C_r components.

4.4 Evaluation of color spaces

To get a broad overview of the static approach by using various fixed ranges for color spaces we processed all the videos in the dataset also by using the HSI , $NRGB$, RGB and YC_bC_r color spaces. The static ranges for these color spaces are available in various papers, such as [5, 10]. The results are found in Figure 10 and it shows that the usage of a combination of color spaces results in a more robust detection than only using a single color space.

²www.youtube.com

³<http://viper-toolkit.sourceforge.net/>

⁴www.macromedia.com

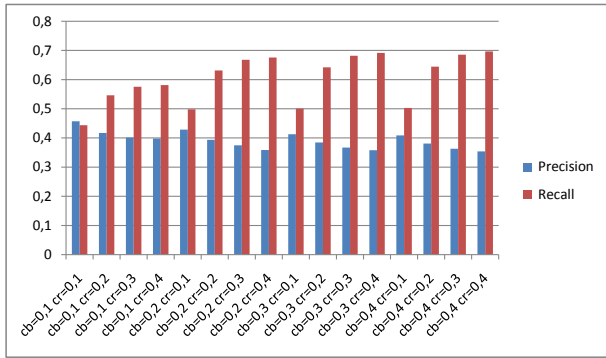


Figure 8: The average dataset results by using various clamping values for C_b and C_r and with the one pass skin detection algorithm.

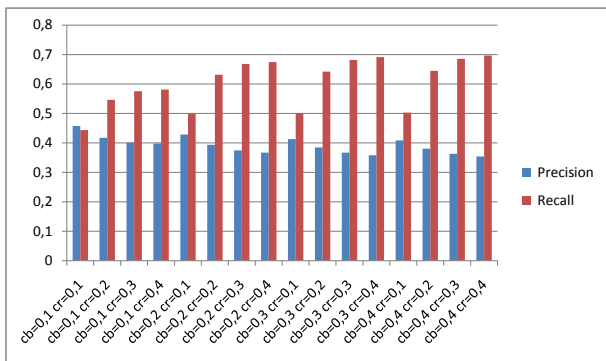


Figure 9: The average dataset results by using different clamping values for C_b and C_r and with the two pass skin detection algorithm.

5 Conclusion

The results of the online poll show that even humans are not able to reliably classify skin color when there is no context involved. People tend to randomly guess upon the color whether or not something is skin. We argue that this also holds for low level detection approaches, which are not able to handle this drawback and therefore some context, like the one that is introduced with face detection or other object detection methods, needs to be in place to overcome some of these issues.

By using a combination of color spaces and an adaptive multiple model approach to dynamically adapt skin color decision rules we are able to significantly reduce the number of false positive detection and the classification results become more reliable. The runtime of the algorithm is still real-time and can be carried out in parallel because the frames are independent of each other.

Acknowledgement

This work was partly supported by the Austrian Research Promotion Agency (FFG), project OMOR 815994, and the CogVis⁵ Ltd. However, this paper reflects only the authors' views; the FFG or CogVis Ltd. are not liable for any use that may be made of the information contained herein.

⁵<http://www.cogvis.at/>

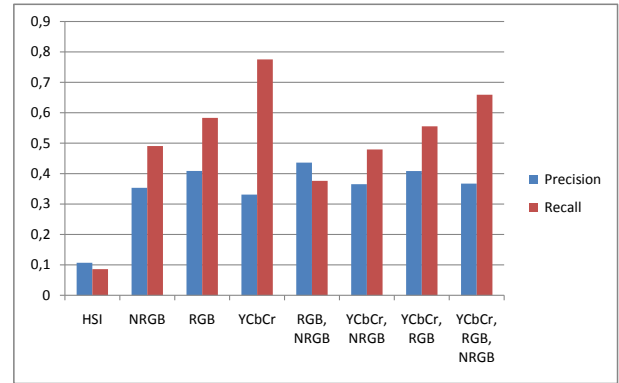


Figure 10: The average dataset results by using different color spaces and combination of color spaces.

References

- [1] J. Brand and J.S. Mason. A comparative assessment of three approaches to pixel-level human skin-detection. *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 1:1056–1059, 2000.
- [2] D. Brown, I. Craw, and J. Lewthwaite. A som based approach to skin detection with application in real time systems. In *BMVC'01: Proceedings of the British Machine Vision Conference*, pages 491–500, 2001.
- [3] T.S. Caetano and D.A.C. Barone. A probabilistic model for the human skin color. *Image Analysis and Processing, 2001. Proceedings. 11th International Conference on*, pages 279–283, 26–28 Sep 2001.
- [4] J. Cai and A. Goshtasby. Detecting human faces in color images. *Image and Vision Computing*, 18:63–75, December 1999.
- [5] D. Chai and K.N. Ngan. Locating facial region of a head-and-shoulders color image. *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 124–129, 14–16 Apr 1998.
- [6] Franklin C. Crow. Summed-area tables for texture mapping. In *SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, pages 207–212, New York, NY, USA, 1984. ACM Press.
- [7] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995.
- [8] Zhouyu Fu, Jinfeng Yang, Weiming Hu, and Tieniu Tan. Mixture clustering using multidimensional histograms for skin detection. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4*, pages 549–552, Washington, DC, USA, 2004. IEEE Computer Society.
- [9] C. Garcia and G. Tziritas. Face detection using quantized skin color regions merging and wavelet packet analysis. *Multimedia, IEEE Transactions on*, 1(3):264–277, Sep 1999.
- [10] Francesca Gasparini and Raimondo Schettini. Skin segmentation using multiple thresholding. volume 6061. SPIE, 2006.

- [11] Donald Geman, Yali Amit, and Ken Wilder. Joint induction of shape features and tree classifiers. *IEEE Trans. PAMI*, 19, 1997.
- [12] Moheb R. Girgis, Tarek M. Mahmoud, and Tarek Abd-El-Hafeez. An approach to image extraction and accurate skin detection from web pages. In *Proceedings of World Academy of Science, Engineering and Technology*, pages 367–375, 2007.
- [13] G. Gomez, M. Sanchez, and Luis Enrique Sucar. On selecting an appropriate colour space for skin detection. In *MICAI '02: Proceedings of the Second Mexican International Conference on Artificial Intelligence*, pages 69–78, London, UK, 2002. Springer-Verlag.
- [14] Ming hsuan Yang and Narendra Ahuja. Gaussian mixture model for human skin color and its application in image and video databases. In *Its Application in Image and Video Databases. Proceedings of SPIE 99 (San Jose CA)*, pages 458–466, 1999.
- [15] Michael J. Jones and James M. Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, 2002.
- [16] P. Kakumanu, S. Makrogiannis, and N. Bourbakis. A survey of skin-color modeling and detection methods. *Pattern Recogn.*, 40(3):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 *Proceedings of the 1st ACM International Workshop in Analysis and Retrieval of Events/Actions and Workflows in Video Streams (AREA 2008)*, Vancouver, Canada, October 2008.
- [18] Jiann-Shu Lee, Yung-Ming Kuo, Pau-Choo Chung, and E-Liang Chen. Naked image detection based on adaptive and extensible skin color model. *The journal of pattern recognition society, Pattern Recognition*, 40(8):2261–2270, 2007.
- [19] Jae-Young Lee and Yoo Suk-in. An elliptical boundary model for skin color detection. In *International Conference on Imaging Science, Systems and Technology*, pages 579–584, 2002.
- [20] Shi lin Wang, Hong Hui1, Sheng hong Li, Hao Zhang, Yong yu Shi, and Wen tao Qu. Exploring content-based and image-based features for nude image detection. In *Fuzzy Systems and Knowledge Discovery*, pages 324–328, 2005.
- [21] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 130–136, 1997.
- [22] C. P. Papageorgiou, M. Oren, and T. Poggio. 1998, a general framework for object detection. *Computer Vision, 1998. Sixth International Conference on*, pages 555–562, 1998.
- [23] Member-Son Lam Phung, Sr. Member-Abdesselam Bouzerdoum, and Sr. Member-Douglas Chai. Skin segmentation using color pixel classification: Analysis and comparison. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(1):148–154, 2005.
- [24] J. R. Quinlan. Simplifying decision trees, 1986.
- [25] Nicu Sebe, Ira Cohen, Thomas S. Huang, and Theo Gevers. Skin detection: A bayesian network approach. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 2*, pages 903–906, Washington, DC, USA, 2004. IEEE Computer Society.
- [26] Andrew Senior, Rein-Lien Hsu, Mohamed Abdel Mottaleb, and Anil K. Jain. Face detection in color images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):696–706, 2002.
- [27] L. Sigal, S. Sclaroff, and V. Athitsos. Skin color-based video segmentation under time-varying illumination. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(7):862–877, July 2004.
- [28] M. Storrang, H.J. Andersen, and E. Granum. Estimation of the illuminant colour from human skin colour. *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 64–69, 2000.
- [29] Kinh Tieu and Paul Viola. Boosting image retrieval. In *International Journal of Computer Vision*, pages 228–235, 2000.
- [30] John K. Tsotsos, Sean M. Culhane, W. Y. K. Winky, Yuzhong Lai, Neal Davis, and Fernando Nuflo. Modeling visual attention via selective tuning. *Artif. Intell.*, 78(1-2):507–545, 1995.
- [31] Vladimir Vezhnevets, Vassili Sazonov, and Alla Andreev. A survey on pixel-based skin color detection techniques. In *Graphicon-2003, 13th International Conference on the Computer Graphics and Vision*, pages 85–92, 2003.
- [32] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 1:I–511–I–518 vol.1, 2001.
- [33] Ching-Chih Weng, H. Chen, and Chiou-Shann Fuh. A novel automatic white balance method for digital still cameras. *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 3801–3804 Vol. 4, 23–26 May 2005.
- [34] Matthias Wimmer, Bernd Radig, and Michael Beetz. A person and context specific approach for skin color classification. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 39–42, Washington, DC, USA, 2006. IEEE Computer Society.
- [35] K.W. Wong, K.M. Lam, and W.C. Siu. A robust scheme for live detection of human faces in color images. *SPIC*, 18(2):103–114, 2003.
- [36] Jie Yang, Weier Lu, and Alex Waibel. Skin-color modeling and adaptation. In *ACCV '98: Proceedings of the Third Asian Conference on Computer Vision-Volume II*, pages 687–694, London, UK, 1997. Springer-Verlag.
- [37] C. Zhang and T. Chen. *From Low Level Features to High Level Semantics*, chapter 27. CRC Press, 2003.