



A Rule Based Human Skin Detection Method in CMYK Color Space

Abdelkrim Sahnouné^(✉) , Djamila Dahmani , and Saliha Aouat 

Artificial Intelligence Laboratory (LRIA), Computer Science Department, University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria
{asahnouné, ddahmani, saouat}@usthb.dz

Abstract. Skin detection is a very important task in computer vision, since we can find it in many applications such as face detection and recognition, face tracking, gesture analysis, content-based image retrieval systems and human machine interaction systems. In this paper we present a novel rule-based skin detection method in the Cyan Magenta Yellow Key (CMYK) color space. This space is a subtractive color space used in color printing, and poorly explored in image processing and still less in skin detection tasks. Our method uses thresholds which are based on the relation between CMYK color components in order to recognize skin pixels, two thresholding models were proposed and we have considered the most performing one. The proposed method has been tested on two public skin image databases and has achieved very satisfactory qualitative and quantitative results against other widely used rule-based methods.

Keywords: Skin detection · Rule-based method · CMYK color space

1 Introduction

Skin detection can be defined as pixels classification into skin pixels and non-skin pixels. This task is very important since we can find it in many computer vision applications such as face detection and recognition, video surveillance, gesture analyse and human machine interaction systems to name but a few.

Due to its simplicity and effectiveness, its non-sensitivity to: rotation, scaling and partial occlusions. Color is the most used attribute in skin detection methods. However choosing this attribute presents some drawbacks like:

- Sensibility to luminance variation and camera characteristics.
- Color variation depending on ethnicity and individual characteristics.
- Complex backgrounds which means the existence within the input (image, video) of some objects with skin-like color such as wood and sand.

These drawbacks lead to erroneous detection of skin segment. In particular, complex backgrounds increase the false positive rate *i.e.* non-skin regions

detected as skin regions. Hence, other attributes can be associated with color such as texture, form and motion.

Skin detection methods can be classified into machine-learning methods and rule-based methods. We can also find hybrid methods which combine the two classes mentioned before. Principal skin detection works are presented in [7, 13] and [16]. The major difference between machine-learning-based and rule-based methods lies in the fact that unlike the first class, the second class does not require a training data set. Machine-learning methods provide better results than rule based methods, but as mentioned above they require a training data set and that can be considered as a drawback, learning and classification time can also be considered as a disadvantage especially in real time applications.

Since we are aiming to have a fast, simple and effective solution that does not depend on any dataset and can be used in real-time applications we opted for a rule-based method. Therefore, we propose in this paper an explicit method in a color space rarely explored in computer vision namely the Cyan Magenta Yellow Key CMYK color space in order to accomplish the skin detection. This method is based on thresholds resulting from fitting the CMYK space. For this purpose we have developed two models and considered the most performing one.

The experimentations we have done showed that our method allows us to have good results both in terms of detection rate and execution time. And so it is suitable for skin detection related applications particularly interaction applications.

The remaining part of the paper will be organized as follows, Sect. 2 describes briefly the related works, in Sect. 3 we present our method, experimental results are reported in Sect. 4 and a conclusion is given in the Sect. 5.

2 Related Works

As mentioned in the Introduction color is the most used attribute in skin detection, hence it is very important to start with selecting an adequate color space. Different color spaces were used to this aim and we note that RGB, HSV and YCbCr are the most found in the literature. Excluding RGB, the other spaces are issued from linear or nonlinear transformations of the first one. And contrary to RGB, in the latter spaces chrominance and luminance are separated which improves the distinction between skin and non-skin colors. There are two major classes of skin detection methods which are the rule-based and machine-learning based classes.

In the Rule-based methods (also known as explicit methods or thresholding methods) we try to generate rules related the color distribution in the selected color space, since skin colors occupy a little place in the color space and tend to cluster in one region. Rule-based methods use color only as attribute and go from a simple one component thresholding to a more or less complex rules. A simple thresholding in different spaces is used in [8] and [19] for the RGB space, [1] and [12] for the HSV space and [2] and [4] for the YCbCr space. In [3] a constructive induction algorithm is used to construct a three component decision rules from

normalized RGB components through simple arithmetic operations and in [6] a heuristic rules in the RGB color space are used to detect skin regions. In [18] they estimated correlation rules between YCb and YCr sub spaces to perform the skin detection relying on the calcul of some distances and thresholds. Combining more than one space is also common in skin detection methods. [10] used both YCbCg and YCgCr spaces which are variation of YCbCr while [14] used an RGB + Cr (from YCbCr) thresholding. [19] combined RGB,HSV and YCbCr to achieve the skin detection. We note also that many works proposed a dynamic method in RGB where the thresholds change according to the image [15].

Machine-learning methods differ from the first ones by the use of training data sets of both skin and non-skin elements in order to generate models capable of completing the skin detection task. This class can be divided into two categories: Non-parametric methods, these methods estimate the distributions of skin colors by probability calculations made on color histograms constructed from the training data. the most popular and best-performing method based on histograms is proposed in [5]. In which they create probability lookup tables from RGB histogram built from skin and non-skin data sets and use Bayes rules to calculate the probabilities of being a skin or non-skin entity for a given pixel. The classification is achieved by thresholding the probabilities. In this same category we also have the artificial neural networks in which we generally associate texture information to color information [17]. Parametric methods, they use probabilistic Gaussian models to represent the distribution of skin colors in color spaces and thus predict the class of each pixel. We can have models using one Gaussian which are called Single Gaussian Models SGM and other models called Gaussian mixture models GMM that use more than one Gaussian. These methods are widely used since they provide good results with significantly less training data [9,15].

Finally we have hybrid methods that combining both rule-based and machine-learning methods such as the work of [11].

3 The Proposed Method

Since our method uses color as a unique attribute, our first step was to choose the right color space. So we opted for CMYK (Cyan, Magenta, Yellow, Key -black-) a color mode used to represent colors in the RGB space. The idea behind the use of this mode of color is of artistic and industrial inspiration. In effect, contrary to the default space used in digital images RGB which considers that the main colors are red, green and blue. In art and printing the primary colors are Cyan, Magenta and Yellow; as well as black, whose addition allows for darker colors depending on its value. This can be seen as a separation between the luminance: K and the chrominance: CMY like the YCbCr and HSV spaces, which justifies our choice to adopt this color space.

The first use of CMYK in industry dates back to 1906 when the Eagle Printing Ink Company first used four colors (CMYK) for printing, discovering that these four colors can be combined to produce an almost unlimited number of tones [22]. Since then, this system has been adopted in all printers.

A CMYK image is visually identical to an RGB image. The difference lies, as mentioned above, in the use of different primary colors. So, to represent a given color we will have a different distribution for RGB and CMYK. Figure 1 shows this different distribution for the same colour.


	RGB code	CMYK code
	65 - 83 - 107	100 - 57 - 0 - 148

Fig. 1. CMYK and RGB color code for the same color.

Figure 2 represents the CMYK color wheel which gives a global view of the different colors with their CMYK distribution.

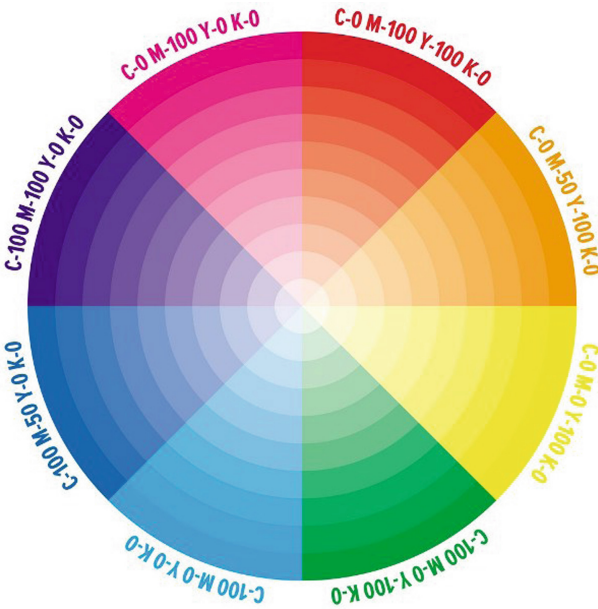


Fig. 2. CMYK color wheel.

By observing the Color Wheel, we can notice that the non-skin colors have a value of C different from zero. So just by applying this simple thresholding we will eliminate a large part of the non-skin objects this step also justifies the choice of this mode. To confirm this we also tried to do the same with the default space RGB by considering the condition $B = 0$ as Cyan is a variant of blue and the results were largely unsatisfactory.

Our method starts by converting the input image from the RGB space to the CMYK space. The conversion is done according to the following formulas [23]:

$$R' = \frac{R}{255} \quad (1)$$

$$G' = \frac{G}{255} \quad (2)$$

$$B' = \frac{B}{255} \quad (3)$$

$$K = 1 - \max(R', G', B') \quad (4)$$

$$C = \frac{1 - R' - K}{1 - K} \quad (5)$$

$$M = \frac{1 - G' - K}{1 - K} \quad (6)$$

$$Y = \frac{1 - B' - K}{1 - K} \quad (7)$$

Thus, we will have values between 0 and 1 corresponding to percentages. To obtain value in [0.255] interval, we will just multiply them by 255. After the conversion we apply a threshold for each pixel in order to perform the skin classification. This threshold includes, as mentioned before, the condition $C = 0$ to eliminate non skin-like colors. We must also take into account objects with colors similar to skin colors, *i.e.* pixels with $C = 0$ but which are not skin pixels. To solve this problem we have proposed two thresholding models for the three remaining components M Y and K.

In the first one noted $M1$, we were inspired by the basic threshold methods. Mainly the most used one, the RGB model [19]. So model $M1$ was conceived after intense experimentation that followed an analysis of the CMYK color wheel Fig. 2 as well as the analysis of many skin images. The aim of the latter was to find a relationship between the MYK components that allows distinguishing skin pixels from non-skin pixels. This work allowed us to come out with a condition discriminating skin and non-skin regions and which we confirmed by experimentation. At the end of the experimentation the optimal quantitative values were obtained by applying the following threshold:

$$C = 0 \text{ and } M > 19 \text{ and } M - Y < 33 \quad (8)$$

For this first model the experimentation showed that the value of K does not influence the results, which led us to ignore this component.

In the second model noted *M2* we wanted to rely on statistical measures in order to have more reliability regarding our method. Hence, we collected 60 skin images with different light conditions and with complex backgrounds and whose skin masks are available. Which gives us about $3,35 \times 10^6$ skin tone pixels, in order to represent the distribution of skin colours in the CMYK space. However, since the CMYK space is 4-Dimensional, we considered the 2D subspaces of CMYK to be able to perform the analysis. Figure 3 shows the 2D projections of CMYK subspaces, skin colors distribution in each subspace is represented with a blue cluster.

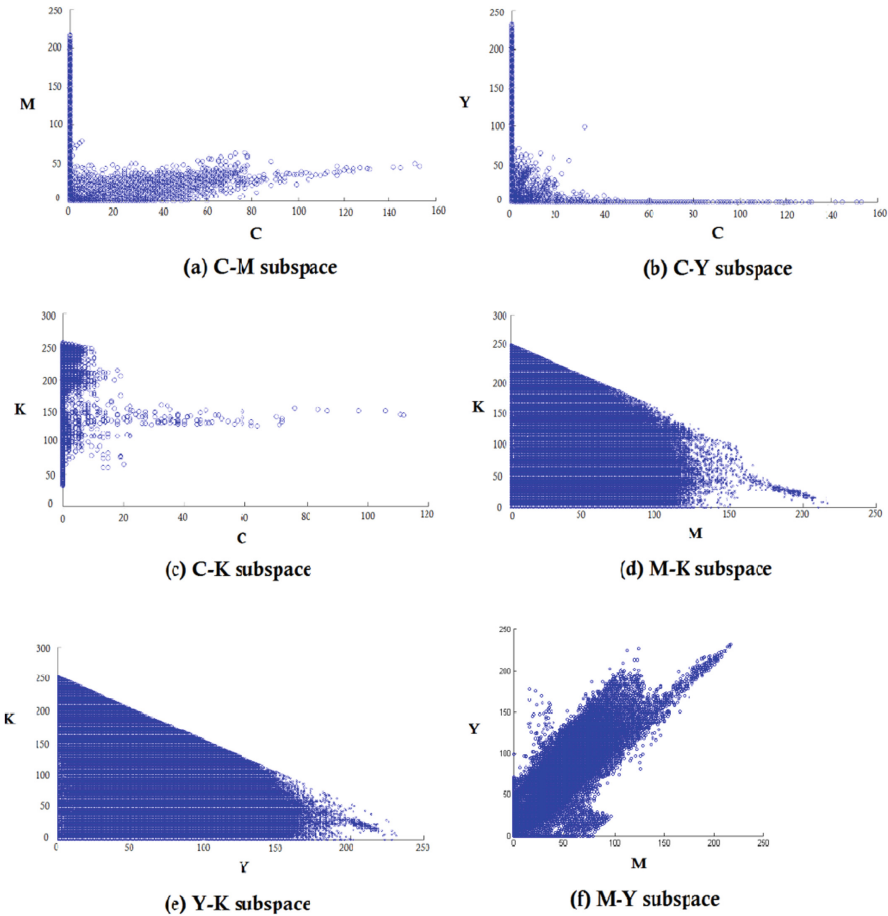


Fig. 3. Skin colors distribution in CMYK subspaces: (a) C-M, (b) C-Y, (c) C-K, (d) M-K, (e) Y-K, (f) M-Y. Blocs in blue represent skin colors clusters.

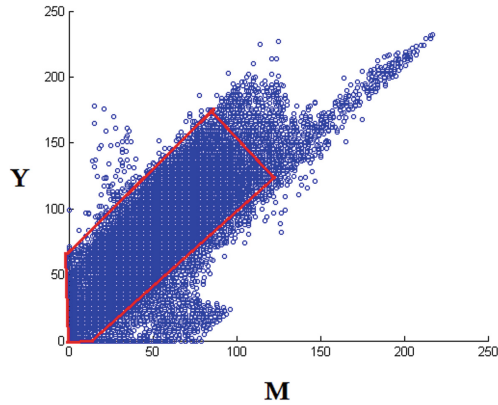


Fig. 4. M-Y subspace: skin color cluster represented in blue, the dense region highlighted with red rectangle.

By observing the subspaces in which the C component is involved we can see that the majority of skin pixels have a $C = 0$. Indeed, on the $3,35 \times 10^6$ skin pixels the number of pixels having $C \neq 0$ does not exceed 3×10^4 and this confirms what we mentioned above. From Fig. 3 we can see on the skin colors cluster in the MY subspace, especially the dense region highlighted in Fig. 4 that the difference between the components M and Y is practically constant and can be thus bounded by a threshold that we have determined graphically as being $0 < Y - M < 80$. We can also notice from this dense region that the value of Y is always greater or equal to the value of M . From Fig. 4 as well, we can notice that the value of M for skin pixels are limited and belongs to the interval $[0, 120]$, this was also been taken into account for the thresholding. Finally, the relationship between the skin colors and the K was not drawn from the graphs. It was deduced by experimentation. We note that we have started experimenting with thresholding directly issued from the graphs *i.e.*:

$$C = 0 \text{ and } M > 0 \text{ and } M < 120 \text{ and } Y - M < 80 \text{ and } Y \geq M \quad (9)$$

These values were modified after experimentation according to the results obtained. Thus, after experimentation a pixel is classified as a skin pixel if it meets the following condition:

$$C = 0 \text{ and } M > 30 \text{ and } M < 97 \text{ and } K < 160 \text{ and } Y - M < 65 \text{ and } Y \geq M \quad (10)$$

Algorithm 1 summarizes the steps of the proposed method.

Algorithm 1: Our skin detection algorithms

```

Result: segmented image (skin and non-skin regions)
read the input image i;
convert i from RGB to CMYK;
for each pixel p in i do
    if the CMYK components satisfy the threshold* then
        | p = skin pixel;
    else
        | p = non-skin pixel;
    end
end
* (8) for model M1 and (10) for model M2

```

4 Experimental Results

To prove its effectiveness, our method was tested on different datasets and compared to other rule-based skin detection methods. First, we carried out a comparison between our two models *M1* and *M2*. These two models were also compared to the method proposed in [19] which is the most common threshold skin detection method. The latter was used as a reference to validate our results. Skin detection results are presented as binary images where white segments represent skin regions and black segments represent non-skin regions. Figure 5 shows some qualitative results for the three methods and Fig. 6 illustrates the region covered by the threshold of each model (*M1* and *M2*) in the MY subspace.

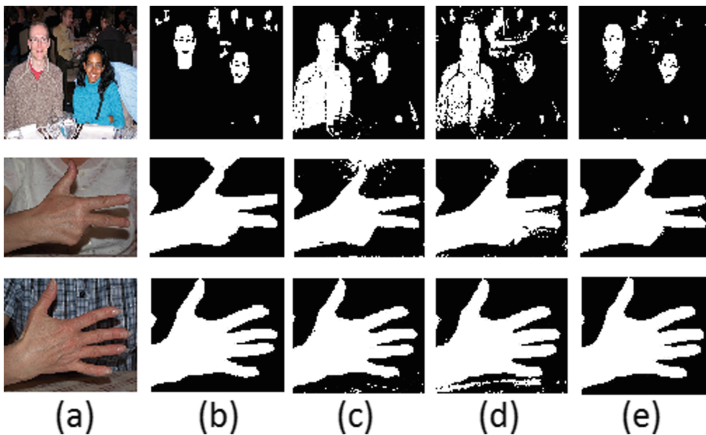


Fig. 5. Skin detections results: (a) input images; (b) skin mask; (c) RGB thresholding - S. Kolkur et al.; (j) proposed method M1; (k) proposed method M2.

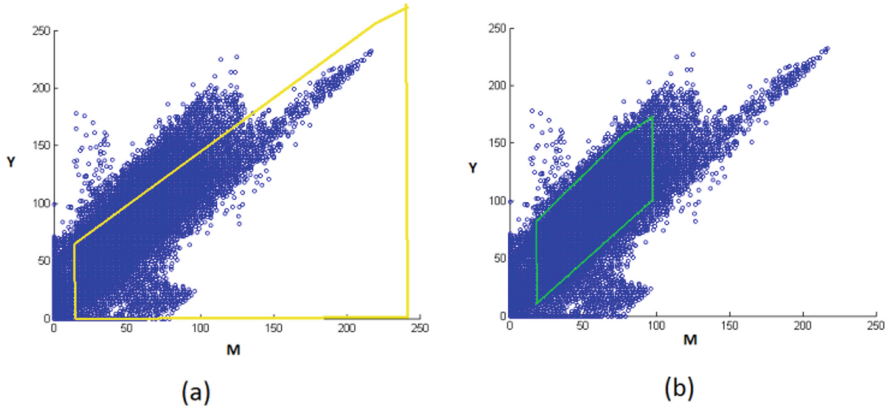


Fig. 6. M-Y subspace: region covered by thresholds (a) M1 model; (b) M2 model.

From Fig. 5 we can notice a difference in the results of $M1$ and $M2$. In model $M2$ the distinction between skin and non-skin regions is nearly perfect, contrary to model $M1$ where in addition to skin area parts of clothes were also detected incorrectly as skin. $M1$ threshold covers a large part of the MY subspace (see Fig. 6(a)) and encompasses the skin-colors clusters and the other colors. On the other hand, $M2$ threshold (see Fig. 6(b)) covers only the colors belonging to the skin-colors cluster. This explains the inequality in the two models results and proves that $M2$ is more performing. Figure 5 shows also that the results obtained by model $M2$ were better than those obtained by the method proposed in [19], which validates our choice of the color space and the threshold model.

Afterwards, our method was compared to other explicit methods and we have selected the most widely used in the literature: the methods proposed in Sobotka and pitas [1], Chai and Ngan [2], Gomez and Morales [3], Hsu et al. [4] Kovac et al. [6], Brancati et al. [18] and the RGB threshold used in Kolkurl et al. [19]. This choice is made because we wanted to test it against methods of the same class *i.e.* rule-based methods that use color as single attribute.

The different methods were tested on two data sets that are publicly available: The database for hand gesture recognition HGR1 [20], which contains 899 hand images, taken in different light conditions and containing images with skin-like objects. And the Human skin detection dataset (Pratheepan data set) [21], which contains 78 skin images containing several images with complex background. The evaluation was done on a simple PC, equipped with an *IntelCorei3 – 3217U* CPU at 1.8 GHz and with 4 GB RAM using MatLab R2013a. For a 640×480 resolution image the execution time of our method is 245 ms, which is very satisfying considering the configuration of the machine used.

Some qualitative results of the different methods are illustrated in Fig. 7 for the two datasets and in Fig. 8 for real images.



Fig. 7. Skin detection results in the two used datasets: HGR1 and Pratheepan; (a) input image; (b) skin mask; (c) Sobottka and Pitas; (d) Chai and Ngan; (e) Hsu et al.; (f) Gomez et Morales; (g) Kovac et al.; (h) Brancatti et al.; (i) RGB thresholding - S. Kolkur et al.; (j) proposed method M1; (k) proposed method M2.

Figure 7 and Fig. 8 shows that our method (model *M2*) qualitatively surpasses the other methods. In fact, in the other methods we have either strict rules that by willing to eliminate all non-skin pixels do not entirely detect the skin regions. Or tolerant rules that incorrectly classify non-skin regions while trying to ensure perfect skin regions detection. In our method we tried to have a fair compromise between the two. We can also notice that the other methods are irregular, as they do not provide good results for all images.



Fig. 8. Skin detection results on real images; (a) input image; (b) Sobottka and Pitas; (c) Chai and Ngan; (d) Hsu et al.; (e) Gomez et Morales; (f) Kovac et al.; (g) Brancatti et al.; (h) RGB thresholding - S. Kolkur et al.; (i) proposed method M1; (j) proposed method M2.

To make a quantitative evaluation of the proposed method and thus test its effectiveness we used the following measurements: Precision, Recall, Specificity, F-measure and D_{prs} . These measures are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

$$Specificity = \frac{TN}{TN + FP} \tag{13}$$

$$F - measure = \frac{2 \times Precision \times Recall}{Recall + Precision} \tag{14}$$

$$D_{prs} = \sqrt{(1 - Precision)^2 + (1 - Recall)^2 + (1 - Specificity)^2} \tag{15}$$

Where TP is the number of pixels correctly classified as skin pixels, TN the number of pixels correctly classified as non-skin pixels, FP the number of pixels wrongly classified as skin pixels, and as FN the number of pixels wrongly classified as non-skin pixels. Among the five measurements considered, we note that the F-measurement takes into account the Precision and the Recall. The D_{prs} , which represents the Euclidean distance between the resulting image and

the Skin mask, takes into account: Precision, Recall and Specificity which represents a general measurement of the effectiveness of the method. Hence, these two criteria are the most important and representative of the quality of skin detection methods. We note that the values of these five measurements are to be maximized, except for the D_{prs} , which is to be minimized. The values of the measurements for the different tested methods are presented in Table 1. As far as our method is concerned, we notice that the values of the comparison measurements differ according to the thresholds considered. Our objective was to find a compromise allowing to have the highest possible TPR and TNR and at the same time the lowest possible FPR and FNR which allows us to have the best measurements values. So in Table 1 we present the results obtained by using the thresholds mentioned in the algorithm, with the two models $M1$ and $M2$.

Table 1 confirms quantitatively that model $M2$ is better than model $M1$. Indeed, although model $M1$ provides very good results on the HGR1 database with even the best F-measure of all methods. The results of model $M1$ were not as good when tested on the Pratheepan database. So we have chosen model $M2$ which was satisfying on both dataset for the comparison with the other methods.

Also from Table 1 we notice that our method (model $M2$) outperforms the other methods with which it was compared. Indeed, in terms of the two most important measures: F-measure and D_{prs} our method achieves the best result whether on HGR1 or Pratheepan datasets. Concerning the other measurements; our method obtains the best Precision in the two bases HGR1 and Pratheepan. Our method obtains the second best Recall on Pratheepan and the fourth best *Recall* on HGR1 with a difference of only 0.05 from the first. Finally, the Specificity of our method is the best on HGR1 and fourth on the Pratheepan dataset, with a considerable difference from the first. Thus contrary to the other methods which have inconsistent results on the two data sets, our method is always the best or at least among the best. And it is only for the last case that our method is a little far from the best ones. We note that we obtained during the experimentation of the Pratheepan database a Specificity value 0.1 higher than the one presented in Table 1 but with a drop in Precision and F-measure. This led us to keep the threshold presented in this paper, but also because we didn't want to take different thresholds for the two data bases.

Finally, from Fig. 7, Fig. 8 and Table 1 we can observe that model $M1$ provide very good results in HGR1 data set, but less good results in Pratheepan data set. Contrary to $M2$ model which is effective on both bases. This shows that the CMYK color space is the most suitable color space for rule-based skin detection methods provided that we choose the right threshold.

Table 1. Qualitative comparison between the selected rule-based methods and the proposed method

Measurement	Method	HGR1	Pratheepan
Precision	Sobotka and pitas	0.6477	0.4186
	Chai and Ngan	0.7731	0.3852
	Hsu et al.	0.7322	0.5215
	Gomez and Morales	0.8664	0.4121
	Kovac et al.	0.8335	0.5849
	Brancatti et al.	0.8714	0.5513
	Kolkurl et al. (RGB threshold)	0.8373	0.6060
	Proposed method M1	0.8807	0.5061
	Proposed method M2	0.9103	0.6596
Recall	Sobotka and pitas	0.8577	0.7808
	Chai and Ngan	0.9599	0.9494
	Hsu et al.	0.9229	0.9009
	Gomez and Morales	0.8149	0.7032
	Kovac et al.	0.8561	0.6258
	Brancatti et al.	0.9062	0.8199
	Kolkurl et al. (RGB threshold)	0.8601	0.9189
	Proposed method M1	0.9489	0.9022
	Proposed method M2	0.9065	0.9214
Specificity	Sobotka and pitas	0.5759	0.6844
	Chai and Ngan	0.7336	0.5781
	Hsu et al.	0.6928	0.7702
	Gomez and Morales	0.9236	0.6595
	Kovac et al.	0.8758	0.8955
	Brancatti et al.	0.8916	0.8230
	Kolkurl et al. (RGB threshold)	0.8777	0.6298
	Proposed method M1	0.8870	0.5948
	Proposed method M2	0.9287	0.7027
F-Measure	Sobotka and pitas	0.7381	0.5450
	Chai and Ngan	0.8565	0.5480
	Hsu et al.	0.8166	0.5585
	Gomez and Morales	0.8399	0.5197
	Kovac et al.	0.8447	0.6047
	Brancatti et al.	0.8885	0.6592
	Kolkurl et al. (RGB threshold)	0.8485	0.7304
	Proposed method M1	0.9135	0.6291
	Proposed method M2	0.9081	0.7688
D_{prs}	Sobotka and pitas	0.5693	0.7020
	Chai and Ngan	0.3521	0.7508
	Hsu et al.	0.4147	0.6414
	Gomez and Morales	0.2406	0.7413
	Kovac et al.	0.2525	0.5780
	Brancatti et al.	0.1924	0.5149
	Kolkurl et al. (RGB threshold)	0.2469	0.5466
	Proposed method M1	0.1720	0.6609
	Proposed method M2	0.1482	0.4587

5 Conclusion

In this paper we have presented a new rule-based skin detection method in the CMYK color space. Our method uses thresholds which are based on the relation between CMYK color component in order to recognize skin pixels. The proposed method has been tested on two skin image datasets and provided highly satisfying results when compared to other widely used explicit methods.

In the future works we will focus in improving the results especially the specificity. Mainly by combining thresholding models, since we have obtained better results in some cases using different thresholds. We also aim to add a more efficient decision method and combine the texture information to our color-based skin-detection model.

Acknowledgement. We would like to express our special thanks of gratitude to the Directorate General for Scientific Research and Technological Development (DGRSDT), for the support of this work under the grant number C0662300.

References

1. Sobottka, K., Pitas, I.: A novel method for automatic face segmentation, facial feature extraction and tracking. *Sig. Process. Image Commun.* **12**(3), 263–281 (1998)
2. Chai, D., Ngan, K.N.: Face segmentation using skin-color map in videophone applications. *IEEE Trans. Circ. Syst. Video Technol.* **9**(4), 551–564 (1999)
3. Gomez, G., Morales, E.: Automatic feature construction and a simple rule induction algorithm for skin detection. In: *Proceedings of Workshop on Machine Learning in Computer Vision*, pp. 31–38 (2002)
4. Hsu, R.-L., Abdel-Mottaleb, M., Jain, A.K.: Face detection in color images. *IEEE Trans. Patt. Anal. Mach. Intell.* **24**(5), 696–706 (2002)
5. Jones, M.J., Rehg, J.M.: Statistical color models with application to skin detection. *Int. J. Comput. Vis.* **46**(1), 81–96 (2002)
6. Kovac, J., Peer, P., Solina, F.: Human skin color clustering for face detection. In: *EUROCON. IEEE* (2003)
7. Kakamanu, P., Makrogiannis, S., Bourbakis, N.: A survey of skin-color modeling and detection methods. *Patt. Recogn.* **40**, 1106–1122 (2007)
8. Lee, Y.J., Lee, D.H.: Research on detecting face and hands for motion-based game using web camera. In: *Proceedings of the 2008 International Conference on Security Technology*, vol. 4, pp. 7–12. *SecTech* (2008). <https://doi.org/10.1109/SecTech.2008.14>
9. Cheddad, A., Condell, J., Curran, K., Kevitt, P.M.: A skin tone detection algorithm for an adaptive approach to steganography. *Sig. Process.* **89**(12), 2465–2478 (2009)
10. Zhang, Z.Z., Shi, Y.X.: Skin color detecting unite YCgCb color space with YCgCr color space. In: *Proceedings of 2009 International Conference on Image Analysis and Signal Processing, IASP 2009*, pp. 221–225 (2009)
11. Zaidan, A., Ahmed, N., Karim, H.A., Alam, G.M., Zaidan, B.: Increase reliability for skin detector using backpropagation neural network and heuristic rules based on YCbCr. *Sci. Res. Essays* **5**(19), 2931–2946 (2010)
12. Guo, J.-M., Liu, Y.-F., Chang, C.-H., Nguyen, H.-S.: Improved hand tracking system. *IEEE Trans. Circ. Syst. Video Technol.* **22**(5), 693–701 (2012)

13. Prema, C., Manimegalai, D.: Survey on skin tone detection using color spaces. *Int. J. Appl. Inf. Syst. (IJAIS)* **2**(2), 18–26 (2012). ISSN: 2249-0868
14. Prema, C., Manimegalai, D.: A novel skin tone detection using hybrid approach by new color space. *Int. J. Comput. Appl.*, 975–8887 (2012)
15. Yogarajah, P., Condell, J., Curran, K., McKeivitt, P., Cheddad, A.: A dynamic threshold approach for skin tone detection in colour images. *Int. J. Biom.* **4**(1), 38–55 (2012)
16. Kawulok, M., Nalepa, J., Kawulok, J.: Skin detection and segmentation in color images. In: *Advances in Low-Level Color Image Processing. Lecture Notes in Computational Vision and Biomechanics*, vol. 11, pp. 329–366. Springer, Heidelberg (2014)
17. Al-Mohair, H.K., Saleh, J.M., Suandi, S.A.: Hybrid human skin detection using neural network and k-means clustering technique. *Appl. Soft Comput.* **33**, 337–347 (2015)
18. Brancati, N., De Pietro, G., Frucci, M., Gallo, L.: Human skin detection through correlation rules between the YCb and YCr subspaces based on dynamic color clustering. *Comput. Vis. Image Underst.* **155**, 33–42 (2017)
19. Kolkurl, S., Kalbande, D., Shimpi, P., Bapat, C., Jatakia, J.: Human skin detection using RGB, HSV and YCbCr color models (2017). <https://doi.org/10.2991/iccasp-16.2017.51>
20. Database for hand gesture recognition. <http://sun.aei.polsl.pl/~mkawulok/gestures/>
21. Human skin detection dataset. http://web.fshtm.um.edu.my/~cschan/downloads_skin_dataset.html
22. CMYK history, 2 May 2016. <https://www.clubink.ca/blog/print/history-behind-cmyk-colour-model/>. Accessed 17 Feb 2020
23. RGB to CMYK conversion, 1 February 2020. <https://www.101computing.net/cmyk-to-rgb-conversion-algorithm/>. Accessed 9 Feb 2020