Algorithm for Face Detection, Adapted for Platforms with Limited Resources

Key words: Face detection, skin detection, applications for people with disabilities

Abstract: Face detection at unknown illumination is a complex process which does not have a simple solution. The task is additionally complicated when the used hardware platform is with limited processor power and memory size. This paper concerns the algorithm realization of face detector that can be used for creation of Java applications for mobile terminals with built-in photo camera. The aim is creating of mass accessible application for people with impaired vision so that they can be informed (by Text To Speech conversion) about the availability and number of faces within the frame, obtained by the photo-camera.

Introduction

There are lot of techniques for face detection [5,22]. Part of them, working at snapshot level (color image), can be classified into four categories [19]:

1. Knowledge-based methods

These rule-based methods encode human knowledge of the relationship between facial features such as: eyes, nose, and mouth [9,18]. These methods cannot be used if image size is too small.

2. Feature-based methods

These methods aim to find structural features that are robust to drawbacks as: face pose and orientation, lighting conditions, presence of structural components such as moustaches and breads. To this category belong techniques based on texture and skin-color analysis.

The face texture can be used to separate face from other objects [1,11]. These techniques are used mainly as auxiliary because there may be other objects similar to human face texture. Besides they are sensitive to the objects illumination.

Human skin color has been used and proved to be an effective feature in face detection tasks [15,17]. These techniques determine where the color of each pixel describe human skin or not. Thus one can find the areas that can have faces. It is not difficult to detect a lot of faces within a image. These techniques are not very sensitive to human skin variations.

3. Template-matching methods

Several standard face patterns are manually parameterized by a function. Given an input image, the correlation value with the standard patterns are computed for the face contour, eyes, nose, and mouth [16,20]. These methods are not suitable for images with small size and cannot be realised in real-time on platforms with limited resources.

4. Appearance-based methods

The face model is learned from a set of training images which should capture the representative variability of facial appearance. This learned model is then used for face detection. To this category belong following techniques: Artificial Neural Networks [4,12], Baysian classifiers [13], Hidden Markov Models [10]. These techniques cannot be applied if there are several coinciding faces or if they are oriented to the camera which differs from the training one. They cannot be used directly at limited resource platforms.

Selection of a method for face detection

The selection of the face detection technique depends on performance of the hardware platform and application requirements:

- soft real time face detection;
- possibility to work with inexpensive camera (without white balance, auto focus, and flash);
- low resolution of analysed images;
- application must be able to work properly in common indoor and outdoor lighting conditions;
- application must be low sensitive to face pose and orientation;
- mass accessibility of the application.

Considering above requirements and restrictions the best choice for hardware platform are mobile terminals with build-in photo-camera and Java Virtual Machine (JVM).

The type of light source (natural, artificial) strongly influence the skin color. Since the application has to work at different object illumination the algorithm has to adapt to this parameter. This problem can be solved with white color balance. Unfortunately few mobile terminals allow program control of the white balance, flash and focusing.

The small size of the frame does not allow program realisation for white balance. The use of inexpensive CMOS CCD matrixes additionally impairs the algorithm properties since they describe the color shades incorrectly. The transfer function of CCD matrix has to be modelled in order to correct this color distortion, but in order to realise this it is necessary to have a lot of training data and enough computing power [2].

It is proposed to use coarse obtaining of regions with human skin color and to realise this on basis of analysis of pixels color and the geometric dimension characteristics to be used for face detection.. This technique has following advantages:

- possibility for realisation in soft real time with limited resource platform;
- partial independence from the face pose, orientation and scale;
- independence of the number of faces in the frame.

The basic problems which are solved at face detection task are as follows:

- Selection of model for numeric color representation (color space) that is partially invariant to the mentioned disturbing factors;
- Human skin color modelling;
- Classification of regions containing human skin.

Colorspace selection

An appropriate mathematical representation of color signal is needed in order to prepare digital image processing. Several color spaces [21] have been utilized to label pixels as skin or not:

1. *RGB* colorspace

RGB is one of the most widely used linear colorspaces for processing digital image data. The color id represented as a combination of three color rays: red (*R*), green (*G*), and blue (*B*). The correlation between color channels and mixing of luminance and chrominance make *RGB* colorspace not a good choice for face detection algorithms.

2. Normalized RGB (*NRGB*)

NRGB is a non-linear transformation of the *RGB* space. They are obtained by normalizing the color parameters of linear *RGB* model with linear intensity (*RGB* first norm):

$$I = R + G + B\,,$$
$$r = R/I\,, g = G/I\,, b = B/I\,. \tag{1}$$

This normalization reduce pixel brightness dependence.

3. Tint-Saturation-Luminance (TLS) colorspace

At this model T and S values are based on normalized *RGB* model, thus, it is expected to have a similar performance in intensity normalization, but more computational time.

4. Hue-Saturation-Value (HSV) colorspace

The *HSV* belongs to user-orientated color spaces. The *HSV* parameters are better suited for human interaction than models introduced above, because they relate to the human perception of color: “hue” defines dominant color, “saturation” describes how pure the color, and “value” describes the luminance. Although the *HS*-plane for skin detection is used. Poor selectivity in H-plane worse the model characteristics.

5. R/G ratio colorspace

The R/G ratio was used as skin presence indicator, because the skin invariably contains a significant level of red color [6].

6. Perceptually motivated colorspaces

The price for better perceptual uniformity of the color model is complex colorspace transformation function from *RGB* model. The most popular perceptually motivated colorspaces are *CIELAB* and *CIELUV* [21]. Theoretically, color representation similar to the color sensitivity of human vision system should help to obtain high performance skin detection algorithm. The experiments show that this is not a true [14].

Considering the requirements mentioned about application and the defaults of the described colorspaces, *NRGB* has been chosen. The reasons for this are as follows:

- The camera receives the information directly in *RGB* format and it is not necessary to make program converting of model for each pixel of the frame;
- Weak sensitivity to the changes in illumination and position of the light source.

Rg-area is chosen, and *b* parameter is ignored. The reason for this is the weak sensitivity of CMOS CCD matrixes to the blue spectrum of the visible light. Irrespective of the human skin type (Caucasians, Africans, Afro-Americans and Asians), the distribution of its color in *rg*-area can be assumed as unimodal. This enables its simple mathematical description.

Human Skin Color modeling at unknown camera parameters

The mathematical description of regions containing human skin can be realized in several ways:

1. Non-parametric methods for human skin distribution modeling.
2. Parametric methods for human skin distribution modeling.
3. Using of rules for description of region shape.

The non-parametric methods aim at obtaining probability density function (*pdf*) for the color describing human skin without being necessary to obtain mathematical model. These classifiers give good results if there are no big differences between the training and testing material and therefore they are not applicable with the concrete task [7]. The non-parametric methods does not depend on the form of contour describing the distribution of human skin color, but the memory they require is not available for most of modern mobile terminals.

The parametric methods aim at reducing the size of used memory and therefore mathematical model is sought in order to describe human skin color distribution. In practice the modeling most often is realized by the mixed Gaussian model [8,17]. The parametric methods require very good processor computing power which still cannot be guaranteed by modern mobile terminals.

Considering the parametric and non-parametric methods defaults for sought color description we choose the form of the region containing human skin to be described approximately geometrically by a system of fuzzy rules. *rg*-area is used. The experiments as well as the color models such as *TSL* and *R/G* show, that the relation of red color to green color is important in defining of regions with human skin [14, 15]. Geometric description of *rg*-area contours is proposed by their adapting to the light intensity and *r/g* ratio.

Fig.1 illustrates the possible ranges of *r* and *g* parameters change at description of human skin in *rg*-area.

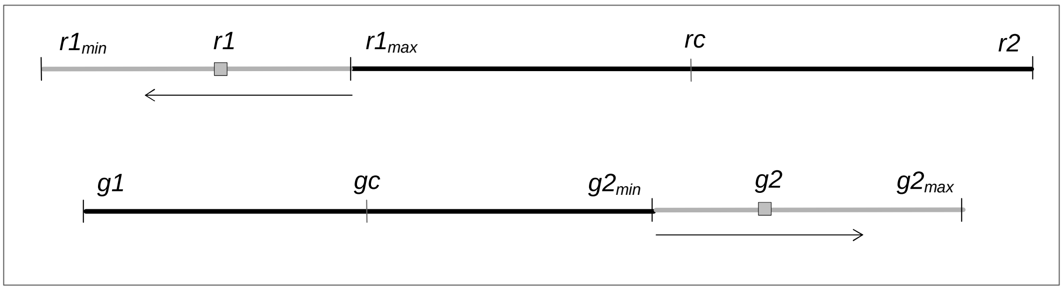


Fig.1. r and g ranges in *rg*-area

Following parameters are used:

- $r1$ and $r2$ are lower and upper limits of parameter r, where $r1 \in [r1_{\min}, r1_{\max}]$;
- $g1$ and $g2$ are lower and upper limits of parameter g, where $g2 \in [g2_{\min}, g2_{\max}]$;
- rc and gc are centers of segments $\overline{r1_{max} r2}$ and $\overline{g2_{min} g1}$,

$$rc = (r1_{max} + r2)/2 ,$$
$$gc = (g1 + g2_{min})/2 . \qquad (2)$$

The ranges $(r1_{max}, r2)$ and $(g1, g2_{min})$ define an area in *rg*-area for which it is certain they describe the human skin color under the condition the current values of *r* and *g* belong to the respective range.

It is experimentally found that as much as the current value of *g* tends to *gc*, as the lower limit of *r*-range ($r1$) is less and as much as *r* tends to *rc*, as the upper limit of *g*-range ($g2$) is bigger.

Adaptive correction of the values of $r1$ and $g2$ is proposed by taking into consideration of light intensity and *r/g* ratio. Let's assume *I* is the light linear intensity value:

$$I = R + G + B, \; I \in (I_{min}, I_{max}) . \qquad (3)$$

The color description of each pixel in the *rg*-area is as follows:

$$r = R/I, \; g = G/I . \qquad (4)$$

The following types of corrections have been assumed for the left bound of the *r*-area and the right bound of the *g*-area:

1. **Linear adaptation to intensity**

$$r1 = r1_{max} - \left(I - I_{min}\right) / Sr1, \tag{5a}$$

$$g2 = g2_{min} + \left(I - I_{min}\right) / Sg2, \tag{5b}$$

where:

$$Sr1 = \left(I_{max} - I_{min}\right) / \left(r1_{max} - r1_{min}\right), \tag{6a}$$

$$Sg2 = \left(I_{max} - I_{min}\right) / \left(g2_{max} - g2_{min}\right). \tag{6b}$$

If $I = I_{min}$, then $r1 = r1_{max}$ ($g2 = g2_{min}$) and ranges are not changed. Maximum expanding of the ranges ($r1 = r1_{min}$, $g2 = g2_{max}$) is obtained if $I = I_{max}$.

2. **Non-linear adaptation to intensity**

$$r1 = r1_{max} - \left(I - I_{min}\right)^2 / Sr1, \tag{7a}$$

$$g2 = g2_{min} + \left(I - I_{min}\right)^2 / Sg2, \tag{7b}$$

where:

$$Sr1 = \left(I_{max} - I_{min}\right)^2 / \left(r1_{max} - r1_{min}\right), \tag{8a}$$

$$Sg2 = \left(I_{max} - I_{min}\right)^2 / \left(g2_{max} - g2_{min}\right). \tag{8b}$$

3. **Adaptation to *r*/*g* ratio**

$$r1 = r1_{max} - \Delta g\left(r1_{max} - r1_{min}\right), \tag{9a}$$

$$g2 = g2_{min} + \Delta r\left(g2_{max} - g2_{min}\right), \tag{9b}$$

where:

$$\Delta g = 1 - \frac{abs(g - gc)}{g2_{min} - gc}, \tag{10a}$$

$$\Delta r = 1 - \frac{abs(r - rc)}{r2 - rc}. \tag{10b}$$

If $r = rc$ ($g = gc$), then $\Delta r = 1$ ($\Delta g = 1$) and maximum expansion of the ranges was received ($r1 = r1_{min}$, $g2 = g2_{max}$). If $r = r2$ or $r = r1_{max}$ ($g = g2_{min}$ or $g = g1$), then $\Delta r = 0$ ($\Delta g = 0$) and ranges are not changed.

4. **Adaptation to intensity and *r*/*g* ratio**

$$r1 = r1_{max} - \Delta g\left(I - I_{min}\right) / Sr1, \tag{11a}$$

$$g2 = g2_{min} + \Delta r\left(I - I_{min}\right) / Sg2, \tag{11b}$$

where Δg and Δr are obtained from Eq.(10a) and (10b), $Sr1$ and $Sg2$ - from Eq.(6a) and (6b).

Fig.2 illustrates graphical change in *r*-range with proposed correcting algorithms .

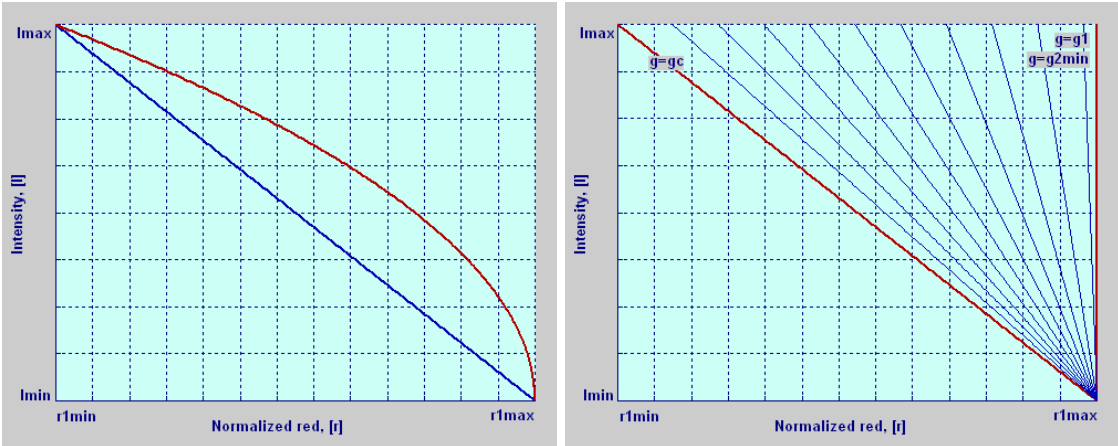


Fig.2. Adaptation of *r*-value to intensity (left) and to intensity and *r*/*g* ratio (right)

Fig.2 (left) shows the dependence of the lower limit of the *r*-range at different degree of intensity (linear and non-linear adaptation to intensity). The non-linear adaptation enables smoother expansion of range especially for the low intensity values. This prevents the wrong classification of pixels in the grayish regions. Fig.2 (right) illustrates the *r*-range lower limit change in connection with reporting of *I* and *g* values. The biggest expansion is obtained at value of *g* tending to *gc* and $I = I_{max}$.

Algorithm realisation

The face detecting algorithm has three stages:

1. **Preprocessing of image in the frame:**
   1.1. Noise reduction by means of 3x3 mean filter;
   1.2. Conversion of image in grey-scale by obtaining of pixel non-linear intensity.
   1.3. Contrast improvement (3x3 gradient based operator [2]) and edge extraction (3x3 Sobel operator).
2. **Searching of pixels with human skin color:**
   2.1. Obtaining of *RGB* current pixel values.
   2.2. Normalization of obtained values by (1).
   2.3. Prepare feature vector: (*I*, *r*, *g*).
   2.4. Correction of *r*- and *g*-ranges, if it is necessary:
   - If $r \notin [r1_{max}, r2]$ and if $g \in [g1, g2_{min}]$, then follows a correction of the lower limit of *r*-range (*r*1);
   - If $g \notin [g1, g2_{min}]$ and if $r \in [r1_{max}, r2]$, then follows a correction of the upper limit of *g*-range (*g*2).

   2.5. Classification. If $r \in [r1, r2]$ and $g \in [g1, g2]$, it is assumed that the current pixel describes the human skin color. Degree of pixel importance is set within [0.1, 1] range depending on the distance of the values of *r* and *g* parameters from *rc* and *gc* respectively.

   2.6. The pixels which are outside *r*- and *g*- ranges, but within the limits of 10% from end points (for both ranges), are set significance within (0, 0.1) range. The remaining pixels are considered not describing human skin.
   2.7. Steps from 2.1 to 2.6 are repeated for all pixels within the frame.

3. Face detection

3.1. Coarse search of skin areas without using adapting techniques. Setting of significance for each pixel.

3.2. Fine correction of skin areas with adapting techniques use.

3.3. Obtaining of regions which can contain faces.

If skin is not found by steps 3.1 and 3.2 or its quality is below certain threshold Th1 ($Th1$=Number of pixels with skin color / Number of all pixels), the regions are formed by color segmentation with filters for red, orange and magenta color. This is necessary because if the light is weak and there is no flash, step 2 of the algorithm shall not generate correct result.

Regions, corresponding to the geometric dimensions of human face are sought. In most cases it is not possible to find eyes and mouth positions which are important for the human face detection because the image resolution is small. Therefore only the form of each region is analysed. For this purpose the centre of gravity for each region is obtained and the relation of height (H) to width (W) H/W is found. If this relation is characterized with less than $Th2$ percents of the golden ratio (1.618) and contains pixels with general significance above 70% it is assumed that the region describes human face.

3.4. Dividing of regions with common borders. This is realized on the basis of: color segmentation of image, edge information, obtained at image preprocessing stage and degree of significance of pixels, describing human skin. The algorithm used generates approximately the same result with “watershed segmentation” algorithm [3], but it is 5.6 times faster.

3.5. Classification. The parameters of ellipse describing most precise the region are found as follows: radiuses (a, b) and inclination towards the vertical (θ). The final decision is based on value of parameter $k1$:

$$k_1 = \left(\sum S_{in} - \sum S_{out}\right) / \sum S \ , \qquad (12)$$

where:

- $\sum S_{in}$ denotes the general significance of pixels in the ellipse;
- $\sum S_{out}$ denotes the general significance of pixels out of the ellipse;
- $\sum S$ denotes the general significance of pixels in the region.

If $k_1 > Th3$, then we assume that selected region contains a face.

Experimental Results

The values of the parameters used for the realization of the experimental part are described in Table 1.

Table 1. Values of used parameters

Parameter	Value *
I_{min}	[150-280] (depends on mean intensity in the frame)
I_{max}	700
$r1_{min}$	335
$r1_{max}$	400
$r2$	555
$g1$	230
$g2_{min}$	335
$g2_{max}$	365

*Th*1	2%
*Th*2	20%
*Th*3	80%
θ	[$0^o \pm 15^o$]
r-range correction	Eq. (9, 10)
g- range correction	Eq. (5, 6)

* To use integer arithmetic all *I*, *r* and *g* values are multiplied by 1000.

Two types of experiments are made:

- off-line mode – program realization on Matlab™;
- on-line mode – Java application for mobile terminals (midlet).

Off-line experiments

Fig.3 illustrates the results of the algorithm for detecting of human skin regions with or without adaptive setting of *rg*-regions.

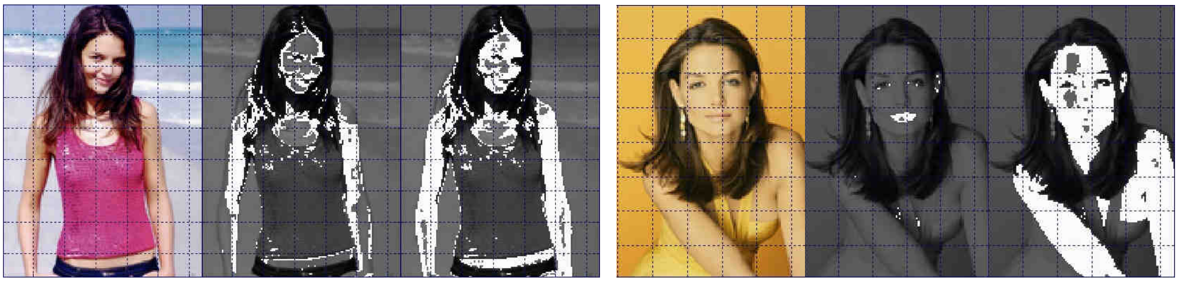


Fig. 3. Skin detection - from left to right: test image, without adaptation, with adaptation

The algorithm for face detecting relies mainly on the quality characteristics of algorithm for detecting of human skin regions. The possibility for correct detection of regions with skin is improved average by 35 % with the use of algorithm adaptive version.

Fig.4 illustrates the function of algorithm adaptive version with external pictures at sunny weather.

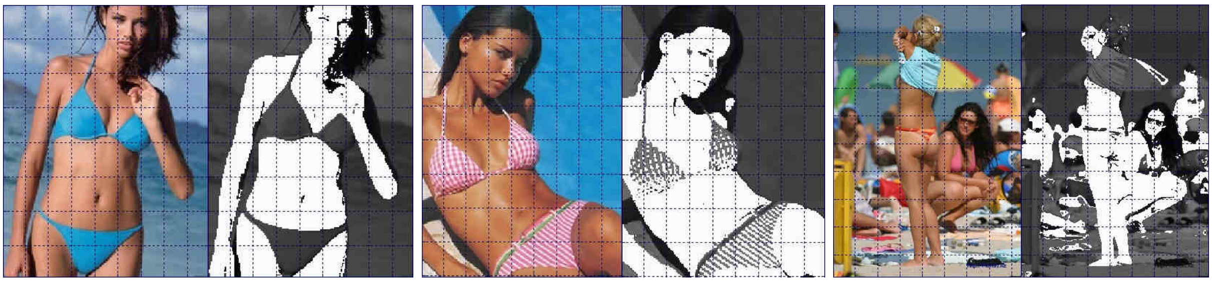


Fig. 4. Adaptive skin detection – sunny weather

The algorithm adaptive version for detecting of skin is invariant to human race as shown on Fig.5.

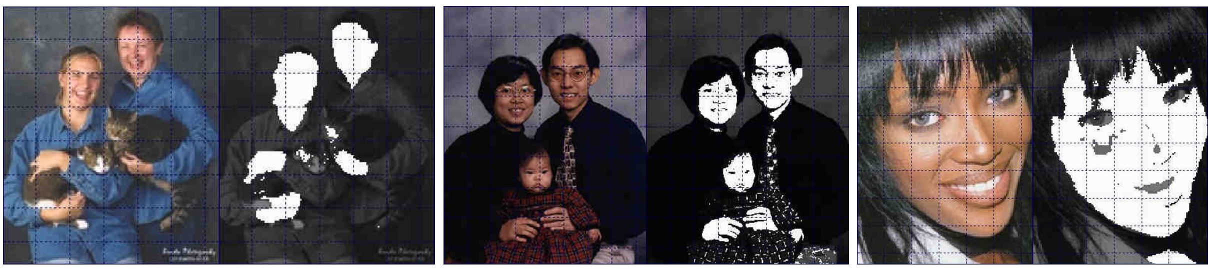


Fig. 5. Adaptive skin detection – different human race

The algorithm qualitative characteristics are also kept when there are a lot of faces in the frame. This is shown in Fig.6. All regions containing skin are detected. Two of the formed regions do not contain skin (coat and object with human skin color). When there are such errors the module for check of region geometrical dimensions is counted.

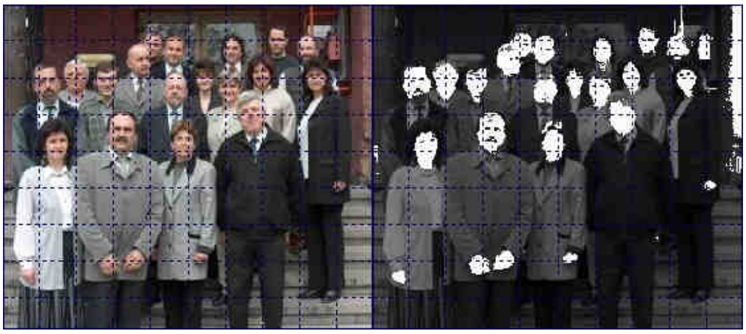


Fig. 6. Adaptive skin detection – several faces in the frame (source http://kst.tugab.bg)

The possibility for correct detection of skin regions is reduced with 14.5% for frames with disturbed white balance. Fig.7 illustrates the result of portrait picture processing with frames with disturbed white color balance.

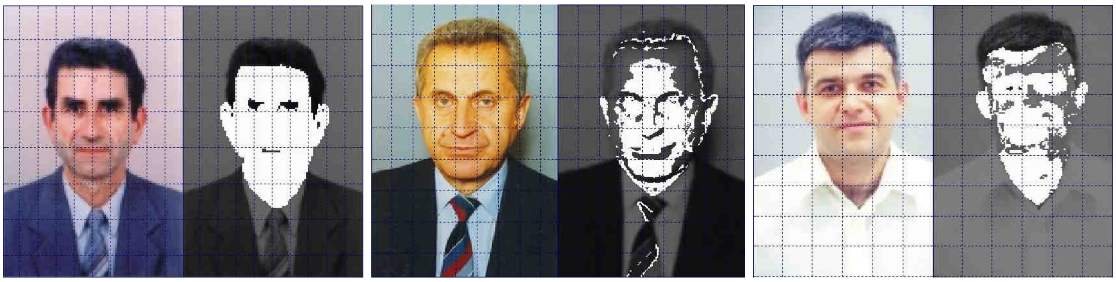


Fig. 7. Adaptive skin detection – disturbed white balance (source http://www.tugab.bg)

The main sources of errors are objects with the color of the human skin (clothes, hair, wooden objects) within the frame as shown in Fig.8. In these cases it is relied on elimination of regions that do not contain faces by geometric dimensions module.

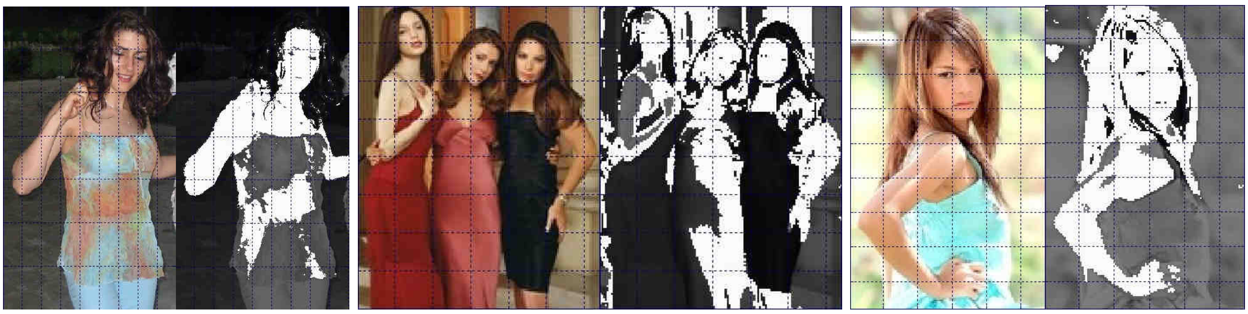


Fig. 8. Adaptive skin detection – false positive errors

On-line experiments

One hundred video frames (160x120 pixels) are used for on-line testing:

- 50 frames in room (20 at external lighting, 10 at luminescent lighting, 10 at incandescent lamps, and 10 by TV set);
- 50 frames in the open (25 at sunny weather and 25 at cloudy weather).

Mobile terminal requirements are as follows:

- Build-in photo camera;

- Java Virtual Machine (JVM) with MIDP 2.0 profile and CLDC 1.1 configuration;
- Mobile Media API (MMAPI) [23].
- Advanced Multi-Media Supplement API (AMMS) [24] (optional).

We use mobile phone Nokia 6630 to realise all on-line experiments. In Table 2 the performance of Nokia 6630 JVM (Monty VM) is presented.

Table 2. Nokia 6630 JVM benchmark

Operation	Operand type	Performance, KOps/s *
Add/Subtract	integer	71,470
	float	1,847
Multiply	integer	55,601
	float	2,227
Divide	integer	1,155
	float	957
Sine/Cosine	double	24

* Kilo operations per second

To improve face detection time only integer and double operations are used. There are no changes in application performance because of floating point to integer conversion (error 10^{-6}). The mean time intervals required for realisation of different algorithm stages are described in Table 3.

Table 3. Algorithm performance

Stage	Time, ms
1. Get snapshot	4
2. Pre-processing	83
3. Skin recognition	185
4. Face detection	108
Total time:	**380**

Fig.9 and Fig.10 show the results of algorithm adaptive version. The results of Fig.10 are obtained with excluded color segmentation. In this case at low level of illumination it cannot be relied only on the algorithm for searching of pixels with skin color.

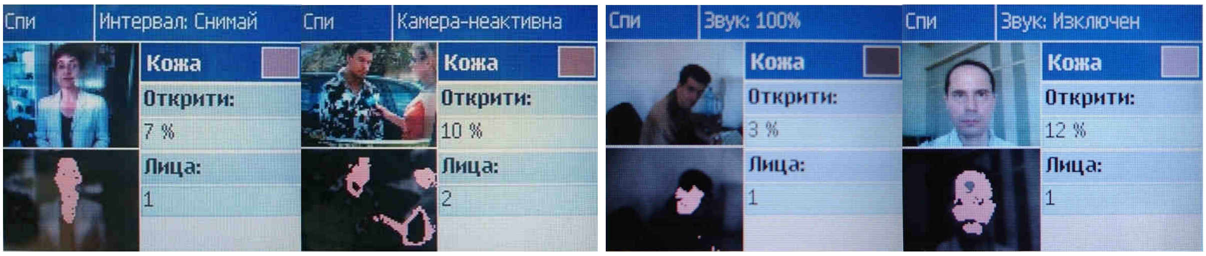
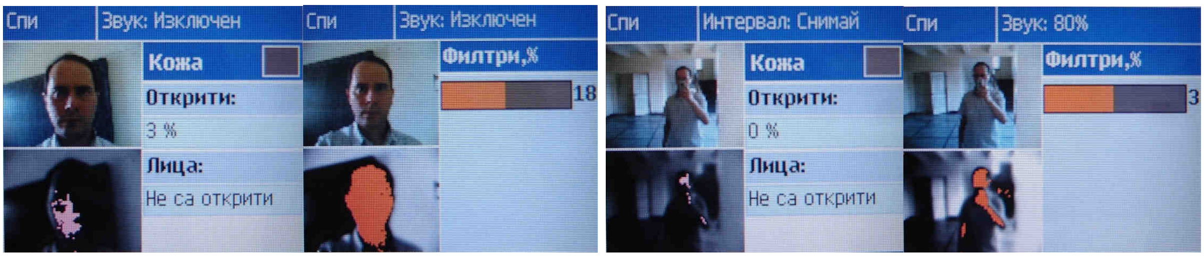


Fig.9. On-line face detection – TV set and indoor images



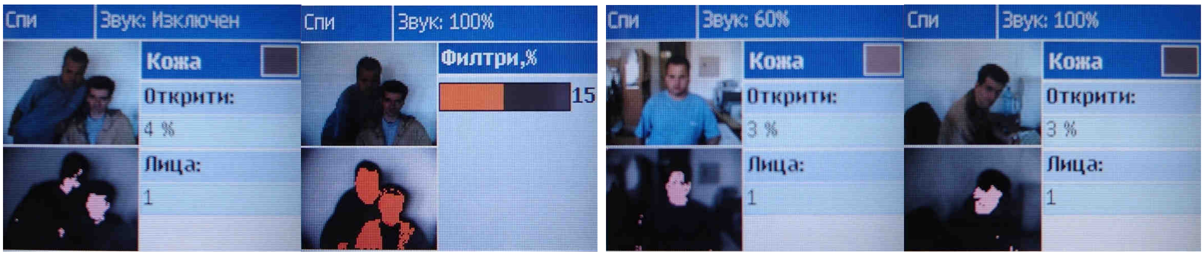


Fig.10. On-line face detection – low level of intensity

The possibility for correct detection with mobile terminals that do not support white balance is 86%. The detection errors are due mainly to the availability of objects with color similar to the human skin and people whose skin is too white. When white balance and auto-focusing are used (AMMS support required) application performance is 94%. Even if white balance is used, more faces are detected in 4 of the tested 100 frames (false positive errors) and face is not detected in 2 frames (false negative errors).

Conclusion

The paper presents an algorithm for face detection that can be used for realisation of applications for platforms with limited resources. The main advantages of proposed algorithm are:

1. We can observe that the relationship between the *r* and *g* ratio is important for skin recognition task. A new adaptive technique for correction of the skin color region limits in *rg*-area is proposed. The adaptation is based on light intensity and *r*/*g* ratio (Eq.5-Eq.11).

2. Color segmentation with filters for red, orange and magenta color is used to obtain correction of the skin regions when light intensity is low. This technique allows to save algorithm performance in poor light conditions, as shown on Fig.10.

3. A technique for separating of skin regions with common borders are proposed. It is based on 3 types of information: color segmentation, edges and pixel significance and it is 5.6 time faster than “watershed segmentation” algorithm.

4. Only integer arithmetic can be used for algorithm realisation (Table 3).

All these signal processing techniques could be executed in soft real time.

Future work will be focused on the comparison between deferent linear and nonlinear adaptation techniques (Eq.5–Eq.11). The database with images, obtained from different mobile phones with building camera at various lighting conditions, face pose and rotation, will be prepared. This database will allow us to compare proposed algorithm with similar face detection techniques.

References

1. Augusteijn, M.F., T.L.Skujca, Identification of Human Faces through Texture-based Feature Recognition and Neural Network Technology, Proc. IEEE Conf. Neural Networks, pp.392-398, 1993.
2. Cardei, V., B. Funt, and K. Barnard, White point estimation for Ucalibrated Images, Proc. 7th Color Imaging Conference (CIC'99). Imaging Science and Technology Society, November 1999.
3. Collins,T., Image-J, Image Processing and Analysis in Java, Toronto Western Research Institute, 2004.

4. Feraud,R., et al., A Fast and Accurate Face Detector Based on Neural Network, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.23, pp.42-53, 2001.
5. Hjelmas,E., Face Detection: A Survey, Computer Vision and Image Understanding, Vol. 83, pp.236-274, 2001.
6. Jayaram,S., et al., Effect of Colorspace Transformation, the Illuminance component, and Color Modelling on Skin Detection, IEEE Conf. on Computer Vision and Pattern Recognition, 2004.
7. Jones,M.J., J.M.Rehg, Statistical Color Models with Application to Skin Detection, Cambridge Research Laboratory, CLR 98/11, 1998.
8. Kawato,S., J.Ohya, Automatic Skin-color Distribution Extraction for face Detection and Tracking, 5th. Int. Conf. on Signal Processing, Voll.II, pp.1415-1418, 2000.
9. Kotropoulos,C., I.Pitas, Rule-based Face Detection in Frontal View, Proc. Int. Conf. Acoustic, Speech and Signal Processing, Vol.4, pp.2537-2540, 1997.
10. Nefian,A.V., M.H.H. III, Face Detection and recognition using Hidden Markov Models, Proc. IEEE Int. Conf. Image Processing, Vol.1, pp.141-145, 1998.
11. Oai,Y., Y.Nakano, Face-Texture Model based on SGLD Matrices, Proc. 1st Int. Workshop Automatic Face and Gesture Recognition, pp.238-242, 1995.
12. Rowley,H., S.Baluja, T.Kanade, Neural Network-based Face Detection, IEEE Trans. on Pattern Analysis and machine Intelligence, Vol.20, No.1, pp.23-38, 1998.
13. Schneiderman,H., T.Kanade, Probabilistic Modeling of Local Appearance and Spatial relationship for object recognition, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp.45-51, 1998.
14. Shin, M.C., K.I. Chang, L.V. Tsap, "Does Colorspace Transformation Make Any Difference on Skin Detection?" IEEE Workshop on Applications of Computer Vision, pp. 275-279, Orlando, FL, 2002.
15. Sorino M., et al., Using the Skin Locus to cope with Changing Illumination Condition in Color-based Face Tracking, Proc. IEEE Nordic Signal Processing Symposium (NORSIG 2000), pp.383-386, 2000.
16. Tsukamoto,A., C.-W.Lee, S.Tsuji, detection and tracking of Human Face with Synthesized Templates, Proc. 1st Asian Conf. Computer Vision, pp.183-186, 1993.
17. Vezhnevets V., Sazonov V., Andreeva A., A Survey on Pixel-Based Skin Color Detection Techniques. Proc. Graphicon-2003, pp. 85-92, Moscow, Russia, 2003.
18. Yang, G., T.S.Huang, Human Face Detection in Complex Background, Pattern Recognition, Vol.27, No.1, pp.53-63, 1994.
19. Yang,M.-H., D.Kriegman, N.Ahuja, Detecting Faces in Images: A Survey, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.24, No.1, pp.34-58, 2002.
20. Yuille,A., P.Halliman, D.Cohen, Feature Extraction from Faces using Deformable Templates, Int. J. Computer Vision, Vol.8, No.2, pp.99-111, 1992.
21. Zarit,D.B., B.J.Super, F.K.Quek, Comparison of Five Color Models in Skin Pixel Classification, Int. Workshop on Recognition, Analysis, and Tracking of Faces and gestures in real-Time Systems, pp.58-63, Greece, 1999.
22. Zhao,W., R.Chellappa, A.Rosenfeld, Face Recognition: A Literature survey, ACM Computing Surveys, Vol. 35, pp.339-458, 2003.
23. http://jcp.org/en/jsr/detail?id=135
24. http://jcp.org/en/jsr/detail?id=234