

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
20 April 2006 (20.04.2006)

PCT

(10) International Publication Number
WO 2006/040311 A2

(51) International Patent Classification:
G06T 5/00 (2006.01)

(21) International Application Number:
PCT/EP2005/055142

(22) International Filing Date: 11 October 2005 (11.10.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
04292426.6 11 October 2004 (11.10.2004) EP

(71) Applicant (for all designated States except US): **THOMSON LICENSING** [FR/FR]; 46, Quai Alphonse le Gallo, F-92100 Boulogne-Billancourt (FR).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **WEITBRUCH, Sébastien** [FR/DE]; Chabeuilstr. 17, 78087 Mönchweiler (DE). **GRAS, Vincent** [FR/FR]; 40, chemin du Clairault, F-37270 Veretz (FR). **THEBAULT, Cédric** [FR/DE]; Oberestr. 8, 78050 Villingen-Schwenningen (DE).

(74) Agents: **LE DANTEC, Claude** et al.; Thomson, 46, Quai Alphonse Le Gallo, F-92100 Boulogne Billancourt (FR).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

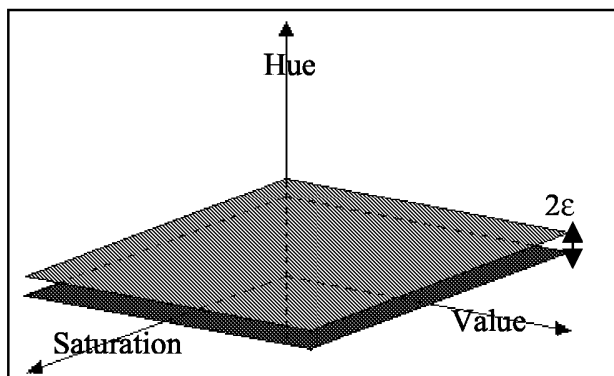
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD FOR DETECTING SKIN TONE PIXELS



(57) Abstract: The present invention relates to a method for detecting skin tone pixels in a picture and a method for processing video data for display on a display device using this detection method. They can be implemented in display device controlling the video levels to be displayed by a Pulse Width Modulation of the illumination time of the cells of the display device. According to the invention, the detection of skin tone pixels in a video picture is based on the hue, the saturation and the colour value of the pixels of the picture and in that the skin tone pixels belongs to a volume in the Hue Saturation Value (HSV) color space, called HSV space wherein the hue of the skin tone pixels is comprised between a minimum hue (H^-) and a maximum hue (H^+).

METHOD FOR DETECTING SKIN TONE PIXELS

The present invention relates to a method for detecting skin tone pixels in a picture and a method for processing video data for display on a display device using this detection method. They can be implemented in display device controlling the video levels to be displayed by a Pulse Width Modulation of the illumination time of the cells of the display device.

Background

Generally, Plasma Display Panel (PDP) utilizes a matrix array of discharge cells, which could only be "ON" or "OFF". Therefore, unlike a CRT or LCD in which gray levels are expressed by analog control of the light emission, a PDP controls gray level by a Pulse Width Modulation of each cell. This time-modulation will be integrated by the eye over a period corresponding to the eye time response. The more often a cell is switched on in a given time frame, the higher is its luminance (brightness). If the video levels are coded on 8 bits (225 levels per color, so 16.7 million colors), each video level can be represented by a combination of the 8 bits having the following weights: 1 - 2 - 4 - 8 - 16 - 32 - 64 - 128 .

To realize such a coding, the frame period can be divided in 8 lighting sub-periods (called sub-fields), each corresponding to a bit and a brightness level. The number of light pulses for the bit "2" is the double as for the bit "1"... With these 8 sub-periods, it is possible through combination to build the 256 gray levels. The eye of the observers will integrate over a frame period these sub-periods to catch the impression of the right gray level. Figure 1 shows such a decomposition.

The light emission pattern introduces new categories of image-quality degradation corresponding to disturbances of gray levels and colors. They will be defined as "dynamic false contour effect" since it corresponds to

disturbances of gray levels and colors in the form of an apparition of colored edges in the picture when an observation point on the display panel moves. Such failures on a picture lead to the impression of strong contours appearing on homogeneous area. The degradation is enhanced when the image has a smooth gradation (like skin...) and when the light-emission period exceeds several milliseconds. When an observation point (eye focus area) on the screen moves, the eye will follow this movement. Consequently, it will no more integrate the same cell over a frame (static integration) but it will integrate information coming from different cells located on the movement trajectory and it will mix all these light pulses together, which leads to a faulty signal information.

Basically, the false contour effect occurs when there is a transition from one level to another with a totally different code. So the first point is, from a code (with n sub-fields) which permits to achieve p gray levels (typically $p=256$), to select m gray levels (with $m < p$) among the 2^n possible sub-fields arrangements (when working at the encoding) or among the p gray levels (when working at the video level) so that close levels will have close sub-fields arrangements. The problem is to define what "close codes" means. Different definitions can be taken, but most of them will lead to the same results. The second point is to keep a maximum of levels in order to keep a good video quality. For this, the minimum of chosen levels is equal to twice the number of subfields. For all further examples, a 11 sub-fields mode with the following weights is used :

1 2 3 5 8 12 18 27 41 58 80

For these issues, the Gravity Center Coding (GCC) was introduced in EP 1 256 924. The content of this document is expressively incorporated by reference herewith. As seen previously, the human eye integrates the light emitted by Pulse Width Modulation. So if one considers all video levels encoded with a basic code, the time position of these video levels (the

center of gravity of the light) is not growing with the video level as shown in figure 2. The centre of gravity CG2 for a video level 2 is larger than the centre of gravity CG1 of video level 1. However, the centre of gravity CG3 of video level 3 is smaller than that of video level 2. It introduces false contour.

- 5 The center of gravity is defined as the center of gravity of the subfields 'on' weighted by their sustain weight :

$$CG(\text{code}) = \frac{\sum_{i=1}^n sfW_i * \delta_i(\text{code}) * sfCG_i}{\sum_{i=1}^n sfW_i * \delta_i(\text{code})}$$

where

- sfW_i is the subfield weight of i^{th} subfield,
- 10 - δ_i is equal to 1 if the i^{th} subfield is 'on' for the chosen code, 0 otherwise,
- $SfCG_i$ is the center of gravity of the i^{th} subfield, i.e. its time position, as shown in figure 3 for the first seven sub-fields.

- 15 So, the temporal centers of gravity of the 256 video levels for the 11 sub-fields code chosen here can be represented as shown in Figure 4. As it can be seen, this curve is not monotonous and presents a lot of jumps. These jumps correspond to false contour. According to GCC, these jumps are suppressed by selecting only some levels, for which the gravity center will
- 20 grow continuously with the video levels apart from exceptions in the low video level range up to a first predefined limit and/or in the high video level range from a second predefined limit on. This can be done by tracing a monotone curve without jumps on the previous graphic, and selecting the nearest point as shown in Figure 5. Thus, not all possible video levels are
- 25 used when employing GCC.

In the low video level region it should be avoided to select only levels with growing gravity centre because the number of possible levels is low and so if only growing gravity centre levels were selected, there would not be enough

levels to have a good video quality in the black levels since the human eye is very sensitive in the black levels. In addition the false contour in dark areas is negligible.

- 5 In the high level region, there is a decrease of the gravity centres, so there will be a decrease also in the chosen levels, but this is not important since the human eye is not sensitive in the high level. In these areas, the eye is not capable to distinguish different levels and the false contour level is negligible regarding the video level (the eye is only sensitive to relative amplitude if the Weber-Fechner law is considered). For these reasons, the monotony of the curve is necessary just for the video levels between 10% and 80% of the maximal video level.

- 15 In this case, for this example, 40 levels ($m=40$) are selected among the 256 possible. These 40 levels permit to keep a good video quality (gray-scale portrayal). This selection can be made when working at the video level, since only few levels (typically 256) are available. But when this selection is made at the encoding, there are 2^p different sub-fields arrangements, and so more levels can be selected as seen on figure 6, where each point corresponds to a sub-fields arrangement. There are different subfields arrangements giving a same video level. Furthermore, this method can be applied to different codings, like 100Hz for example without changes, giving also good results.

- 25 On one hand, the GCC concept enables a visible reduction of the false contour effect. On the other hand, it introduces noise in the picture in the form of dithering needed since less levels are available than required. The missing levels are then rendered by means of spatial and temporal mixing of available GCC levels. The false contour effect is an artefact that only appears on specific sequences (mostly visible on large skin area) whereas the introduced noise is visible all the time and can give an impression of noisy dis-
- 30

play. For that reason, it is important to use the GCC method only if there is a risk of false contour artefacts.

Document EP 1 376 521 introduces a solution for this based on a motion
5 detection enabling to switch ON or OFF the GCC depending on whether there is or not a lot of motion in the picture.

In a particular embodiment, the GCC concept can not be used in all areas of the picture. As the GCC concept is based on the fact that the GCC selected
10 levels are a subset of existing levels, if a part of the picture is using GCC whereas the rest is coded with all levels, it will not be possible for the viewer to see any differences except the fact that there will be some areas with more or less noise and more or less visible false contour. Generally, the frontiers between areas coded with GCC and standard areas are invisible.
15 This property of the GCC concept enables to define a texture-based concept. In this embodiment, the picture is segmented in two types of areas:
- areas where the false contour effect is not really disturbing for the viewer (landscape, grass, trees, buildings, water...), and
- areas where the false contour effect is really disturbing since unnatural
20 (large and homogeneous skin areas)

Two types of modes are then defined for these two types of areas:

- a false contour critical mode: GCC really optimized for false contour (more noise).
- 25 - a standard mode: NO GCC or GCC having a lot of selected levels (almost no noise).

Two types of modes are shown in Fig.7A and Fig.7B : one standard mode comprising 255 video levels and one false contour critical mode comprising 40 video levels.

The main issue is that the noise causes artefacts which in most times are visible and therefore it should be reduced to a minimum. On the other hand, the false contour effect is only visible on specific sequences and should only be tackled on these sequences and above all only on critical parts of such sequences. The aim of this improved concept is the segmentation of the picture in specific areas where the false contour effect will be tackled. Such kind of critical areas are skin tone areas.

A first important segmentation parameter is the colour itself. Different situations could happen according the picture is a black and white picture or a normal coloured picture:

- a) Black and white picture: it can be tested by using the YUV information (U and V being almost negligible for the whole picture). The false contour critical mode is activated for the whole picture. Indeed, it will be very disturbing on a black and white film to see coloured edges (false contour lines).
- b) Normal coloured picture:
 - If the current pixel is almost grey (Red, green and blue components are very similar), the area is sensitive to false contour. Indeed, false contour introduces coloured edges (blue, red ...), which are not awaited by the human eye on grey areas. In that case, the false contour critical mode is activated for the current pixel.
 - If the current pixel has a skin colour, the area is also sensitive to false contour..
 - In all other cases, the standard mode is activated.

Thus, a method for detecting of skin tone pixels is needed to operate the segmentation of the picture.

Invention

The invention proposes a method for detecting skin tone pixels in a video picture. It is based on the hue, the saturation and the colour value of the pixels of the picture wherein the skin tone pixels belongs to a volume in the Hue Saturation Value color space, called HSV space, the hue of the skin tone pixels being comprised between a minimum hue and a maximum hue.

In a preferred embodiment, the value of the skin tone pixels in said HSV space is also comprised between a minimum value and a maximum value the probability density of the value being equal to 1 between said minimum and maximum values. In the same manner, the saturation of the skin tone pixels in said HSV space is comprised between a minimum saturation and a maximum saturation, the probability density of the saturation being equal to 1 between said minimum and maximum saturations.

Advantageously, the method comprises a step for applying a resemblance function to the pixels of the picture, said resemblance function being based on the minimum and maximum saturations and minimum and maximum values and being maximal when the density probability of the hue, saturation and value is maximal and a step for comparing the results of said application to a resemblance threshold, the pixels whose resemblance function is greater than or equal to said resemblance threshold being considered as skin pixels.

In a variant, the method comprises the following steps :

- a) applying a resemblance function to the pixels of the picture, said resemblance function being based on said minimum and maximum saturations and said minimum and maximum values and being maximal when the density probability of the hue, saturation and value is maximal,

- b) comparing the results of said application to a first resemblance threshold, the pixels whose resemblance function result is greater than or equal to said first resemblance threshold being labelled by a first symbol,
- c) comparing the results of said application to a second resemblance threshold, the pixels whose resemblance function result is greater than or equal to said second resemblance threshold and having at least one neighbour pixel labelled by a first symbol being labelled by a second symbol, said second resemblance threshold being lower to said first resemblance threshold,
- d) when all pixels of the picture have been compared, replacing all second symbols by first symbols, and
- e) returning to step a), the number of iterations of said steps sequence being predefined, the pixels being labelled by a first symbol at the end of the last iteration being considered as skin pixels.

15

In another variant, the method comprises the following steps :

- a) applying a resemblance function to the pixels of the picture, said resemblance function being based on said minimum and maximum saturations and said minimum and maximum values and being maximal when the density probability of the hue, saturation and value is maximal,
- b) comparing the results of said application to a first resemblance threshold and labelling the pixels whose resemblance function result is greater than or equal to said first resemblance threshold by a symbol,
- c) comparing the results of said application to a second resemblance threshold and labelling by a same symbol the pixels having at least one neighbour pixel labelled by said symbol whose resemblance function result is greater than or equal to the second resemblance threshold and lower than or equal to said first resemblance threshold, said second resemblance threshold being lower to said first resemblance threshold, and

20

25

d) returning to step a), the number of iterations of said steps sequence being predefined, the pixels being labelled by the symbol at the end of the last iteration being considered as skin pixels.

5 Advantageously, the method further comprises the following steps :

- generating K-1 derivate pictures $I^{(k)}$, $k \in [1, K-1]$, by subsampling said picture I corresponding to $I^{(0)}$, each derivate picture $I^{(k)}$ being the derivate picture of the picture $I^{(k-1)}$,
- applying a resemblance function to the pixels of all pictures $I^{(k)}$, $k \in [0, K-1]$,
- 10 - computing a multiscale resemblance value based on the results of the application of the resemblance function to the pixels of all pictures $I^{(k)}$, $k \in [0, K-1]$, and
- comparing the multiscale resemblance value to a third resemblance threshold, the pixels whose multiscale resemblance value is greater than or
- 15 equal to said resemblance threshold being considered as skin pixels.

For example, for a given pixel, the multiscale resemblance value is the sum of the results of the application of the resemblance function to the pixels of all pictures $I^{(k)}$, $k \in [0, K-1]$, each result for a given picture $I^{(k)}$ being weighted

20 by a coefficient related to said picture.

Advantageously, the method further comprises a step for determining in the picture homogeneous areas of a first type comprising only pixels considered as skin pixels and in that only the pixels of the homogeneous areas of first

25 type having a size bigger than a predetermined minimum size are kept as skin pixels.

Advantageously, the method further comprises a step for determining in the picture homogeneous areas of a second type comprising only pixels not con-

30 sidered as skin pixels and in that the pixels of the homogeneous areas of

second type having a size smaller than a predetermined minimum size are considered as skin pixels.

The invention relates also to a method for processing video data for display on a display device having a plurality of luminous elements corresponding to the pixels of a picture, wherein the time of a video frame or field is divided into a plurality of sub-fields during which the luminous elements can be activated for light emission in small pulses corresponding to a sub-field code word of n bits used for encoding the p possible video levels lighting a pixel, comprising the steps of:

- determining in a picture, an homogeneous area, called first part of the picture, having a size larger than a predetermined minimum size and a skin colour,
- encoding said first part of the picture using a first encoding method, wherein among the set of possible video levels for lighting a pixel, a sub-set of m video levels with $n < m < p$ is selected, which is used for light generation, said m values being selected according to the rule that the temporal centre of gravity for the light generation of the corresponding sub-field code words grows continuously with the video level, and
- encoding at least one second part different from said first part of the picture using a second encoding method different from said first encoding method, characterized in that the inventive skin tone detection method is used for determining said first part of the picture.

Drawings

Exemplary embodiments of the invention are illustrated in the drawings and are explained in more detail in the following description. The drawings showing in:

Fig. 1 the composition of a frame period for the binary code;

- Fig. 2 the centre of gravity of three video levels;
- Fig. 3 the centre of gravity of sub-fields;
- 5 Fig. 4 the temporal gravity centre depending on the video level;
- Fig. 5 chosen video levels for GCC;
- Fig. 6 the centre of gravity for different sub-field arrangements
10 for the video levels;
- Fig. 7A and 7B the centres of gravity of video levels in a standard mode
and a false contour critical mode;
- 15 Fig. 8 an hexagonal cone color model;
- Fig. 9 a repartition of skin pixels in projection on the plane
(value,saturation);
- 20 Fig. 10 the form of a skin color detection set;
- Fig. 11 the probability density of the hue;
- Fig. 12 skin pixels in the (saturation, hue) plane;
- 25 Fig. 13 a modified probability density of the saturation;
- Fig. 14 the topology of the hue domain;
- 30 Fig.15a to 15c the visualisation of a resemblance function;

- Fig. 16a a mosaic of 128 varied skin pixels;
- Fig. 16b the result of the application of the detection criterion on
the mosaic of fig. 16a;
- 5 Fig. 17a a photography comprising skin tone areas;
- Fig. 17b the result of the application of the detection criterion on
the photography of fig. 17a;
- 10 Fig. 18 a photography comprising skin tone areas;
- Fig. 19 the result of the application of a point to point detection
on the photography of fig. 18;
- 15 Fig. 20 the skin area frontier of the detection in fig. 19;
- Fig. 21a a skin area enhancement of the photography of fig. 18;
- 20 Fig. 21b the frontier of the enhanced skin area of fig. 21a;
- Fig. 22a and 22b the results of a point-to-point detection using different
parameters;
- 25 Fig. 23 the results of an hysteresis area increase on the photog-
raphy of fig. 18;
- Fig. 24a to 24b diagrams illustrating the hysteresis area increase proc-
30 ess;

	Fig.25	the first steps the hysteresis area increase process;
	Fig. 26	sub-samples of a 128x128 image;
5	Fig. 27	the results of the application of a function d on sub-samples of fig.26;
	Fig.28a and 28b	a combination of the images of fig.27 and the result of the segmentation process;
10	Fig. 29	the results of a multi-scale detection process;
	Fig.30a and 30b	a connexity illustration and the parameters used for form analysis;
15	Fig.31a and 31b	a 16x16 binary image and the labelling of the forms in the binary image;
	Fig. 32	a form labelling in four quadrants of the image;
20	Fig. 33	the results of a small area deleting and hole filling using form analysis;
	Fig. 34	a first implementation of the inventive method;
25	Fig. 35	a point-to-point detection hardware implementation;
	Fig. 36	an implementation of the image context integration;
30	Fig. 37	an implementation of the skin area processing.

Exemplary embodiments

1) Point-to-point color detection

5

According to the invention, the detection of skin tone is based on the color space HSV (Hue Saturation Value). The Hue information is used for skin color segmentation. A simple hexagonal cone model is chosen for computing the hue, the saturation and the value of color as shown in Figure 8. This model has simple formula and leads to satisfying results.

10

$$V = \max(R, G, B)$$

$$S = 1 - \min(R, G, B) / V \quad (1)$$

$$H = \begin{cases} \frac{G - B}{V - \min(R, G, B)} & \text{si } V = R \\ 2 + \frac{B - R}{V - \min(R, G, B)} & \text{si } V = G \\ 4 + \frac{R - G}{V - \min(R, G, B)} & \text{si } V = B \end{cases}$$

15

In order to cover a maximum of skin tones with the better accuracy, it is proposed to divide the detection in color types. The minimum of color types to be used is one (one general color tone) whereas the better results are obtained by using p different types of skin tones (European, African, Asian...).

20

One skin tone defines a simple detection set in the HSV space. According to a given tolerance coefficient ε this set is theoretically equal to a volume in the Hue Saturation Value (HSV) color space, called HSV space wherein the hue of the skin tone pixels is comprised between a minimum hue H^- and a maximum hue H^+ . In a HSV space where hue direction, value direction and saturation are three orthogonal directions, this set is a parallelepiped of width 2ε in hue direction as shown in figure 10. The restriction lies theoretically only on this parameter and not on the value and the saturation because these two color parameters do not help characterizing skin color.

25

The fact that the value does not constitute a pertinent parameter for skin detection is not surprising because this quantity represents the luminosity. In addition to that, the whole saturation domain [0, 1] can be covered with different skin pixels. An example of repartition of the saturation and the value is shown in figure 9 for different skin pixels.

So, if it is considered that the triplet (H, S, V) is a random variable (each skin pixel delivers a realization of this variable), its behavior for skin pixels is specified by the following assertions.

- i) V is distributed in an equiprobable way over the value range,
- ii) S is distributed in an equiprobable way over the saturation range,
- iii) H follows a normal law of mean m_H and variance σ_H ,
- iv) The hue H is statistically independent to the couple (V, S), that is to say for all (V, S) the probability density $p_{H|(V, S)}$ is equal to the marginal law p_H .

The figure 10 schematizes the detection set corresponding to assertions i) to iv) in the HSV space. Thanks to the description of the statistic behavior of the hue variable, the tolerance coefficient ε can be associated to the probability λ that a skin pixel does not lie in the detection set. The relation is then

$$P(H \in \{m_H - \varepsilon, m_H + \varepsilon\}) = 1 - \lambda.$$

where m_H is the mean value of the hue variable.

The graphic of figure 11 shows the model for the probability density (normal law) and the graphical relation between ε and λ . Furthermore, for a given set of skin pixels judged to be representing the same skin tone, the following formula determines statistically the mean m_H and the variance σ_H of the hue variable for N skin pixels:

$$\begin{cases} m_H = \frac{\sum_{i=1}^N H_i}{N} \\ \sigma_H = \sqrt{\frac{\sum_{i=1}^N (H_i - m_H)^2}{N}} \end{cases} \quad (2)$$

Figure 12 is a set of skin pixels measurements. First the distribution of hue
on this graphic matches the chosen model in point iii). Furthermore formula
(2) extracts from the measurements the variance and the mean of the hue.
Numerical values corresponding to figure 12 are $m_H = 0.39$ and $\sigma_H = 0.15$.

Skin detection requires practically additional restrictions. Without other
conditions the indetermination of the hue when the value or the saturation is
equal to zero is not properly taken into account. A specific decision in these
particular cases is not pertinent because the value and the saturation vary
continuously. The solution is to modify smoothly the properties of the couple
(S, V) so as to reduce strongly the probability at the edges of the color
domain. The simplest way to model this behavior is to introduce 4 new limit
values V_{min} , V_{max} , S_{min} , S_{max} and change assertions i) and ii) by the new fol-
lowing properties i') and i'')

i') V is a random variable of probability density p_V equal to 1 be-
tween V_{min} and V_{max} and going linearly to the value 0 respectively at the ex-
tremities of the value range.

i'') S is a random variable of probability density p_S equal to 1 be-
tween S_{min} and S_{max} and going linearly to the value 0 respectively at the ex-
tremities of the saturation range.

Figure 13 shows the form of the transformed probability density of the satu-
ration illustrating assertion i'').

The statistical description of the (H, S, V) behavior is useful to specify the
detection criterion of skin color for a point-to-point detection. The parameters
of this algorithm are the considered skin tone and a coefficient $\lambda \in [0, 1]$ giv-
ing the probability that a skin pixel is not detected. The specified skin tone

determines the appropriate statistical parameters m_H , σ_H , V_{min} , V_{max} , S_{min} , S_{max} . The detection criterion D is then :

$$D(H, S, V) = 1 \text{ if } (H, S, V) \in U_\lambda, \text{ and } 0 \text{ else}$$

where U_λ is the subset of the HSV space verifying $\lambda = P((H, S, V) \in U_\lambda)$.

Determining U_λ for each λ_i is unfortunately not easy and require further statistical estimation, notably the probability a priori that a pixel is a skin pixel. Let us rather complete the detection criterion while introducing a pertinent resemblance function δ taking its values in the range [0, 1] and being maximal when the density probability $p_{(H,S,V)}$ is maximal. The simplified version of the skin color detection becomes D':

$$D'(H, S, V) = 1 \text{ if } \delta(H, S, V) \geq \alpha, 0 \text{ else}$$

with

$$\delta(H, S, V) = (1 - d_H(H, m_H))(1 - k_S d(S, [S_{min}, S_{max}])(1 - k_V d(V, [V_{min}, V_{max}]))) \quad (3)$$

and where

- α is a resemblance threshold that belongs to the range [0, 1] and replaces the probability λ ,
- the function d_H is the distance in hue space. In the simple hexagonal cone model of the color, the hue varies between -1 and 5 and can go continuously from 5 to -1 (see figures 8 and 14); the pertinent distance over this set is then the distance over a circle.

$$\begin{aligned} d_H(H_1, H_2) &= (H_2 - H_1) / 3 && \text{if } H_1 \leq H_2 \leq H_1 + 3 \\ &= (H_1 - H_2 + 6) / 3 && \text{if } H_1 + 3 \leq H_2 \\ &= (H_1 - H_2) / 3 && \text{if } H_2 \leq H_1 \leq H_2 + 3 \\ &= (H_2 - H_1 + 6) / 3 && \text{if } H_2 + 3 \leq H_1 \end{aligned}$$

- the function d is the usual distance from a point to a set; k_S and k_V are coefficients introduced to guaranty that the terms $1 - k_S d(S, [S_{min}, S_{max}])$ and $1 - k_V d(V, [V_{min}, V_{max}])$ remain between 0 and 1; thus:

$$\begin{aligned} d(S, [S_{min}, S_{max}]) &= S_{min} - S && \text{if } S < S_{min}, \\ &= 0 && \text{if } S \in [S_{min}, S_{max}] \end{aligned}$$

$$= S - S_{\max} \quad \text{if } S > S_{\min}.$$

and

$$k_S = \max(S_{\min}, 1 - S_{\max})^{-1}$$

$$k_V = \max(V_{\min}, 255 - V_{\max})^{-1}.$$

5

Figures 15a to 15c help visualizing the resemblance function δ and interpreting graphically the modified detection criterion D' . The figure 15a is a mapping of the color when $V = 150$ and H and S vary in the full range $[-1, 5] \times [0, 255]$. Figure 15b is the application of resemblance function δ on this picture.

10 Finally, figure 15c shows a cut of the skin detection set $d^{-1}\{[\alpha, 1]\}$ in the plane $V = 150$ for $\alpha = 0.8$ after that the detection function D' has been applied. All pixels which have a resemblance function equal to or greater than α are considered as skin pixels and are identified by the hatched area of the figure 15c.

15

To verify practically the efficiency of the detection criterion D' , it is interesting to test it first with varied real skin pixels and then apply it on a real relevant picture. Figure 16a is a mosaic of 128 varied skin pixels. The right black and white picture (figure 16b) shows the application of the detection criterion D' on this mosaic. Black means that the pixel is detected as skin color and white means "not detected". The major party of the left image is black. Thus the detection criterion covers statistically well any skin pixel.

20

Finally, figure 17a, representing a photography, is a natural image used to test the criterion with the following parameters.

25

- $M_H = 0.39$
- $S_{\min} = 0.1, S_{\max} = 0.9, V_{\min} = 30, V_{\max} = 250$
- $\alpha = 0.9$

Figure 17b is the result of the detection where black means "detected".

30

2) Image context integration

The point-to-point detection algorithm implementing the above-developed criterion D' associates to an image a binary image of same size. For each pixel, the binary image is equal to the skin detection criterion applied to the corresponding pixel. It is interesting to remark that a point-to-point detection produces already structured extended areas. The accuracy is then good. Figure 18 that is an image extracted from the site of the French swimming federation, will be used as test image for segmentation algorithms.

Figure 19 shows the point-to-point skin color detection applied to the image of Figure 18. Up to now black means "detected" and white "not detected". The result localizes indubitably skin areas. There are unfortunately false detections and missings.

Despite the imperfection, a point-to-point detection could be enough for encoding application. The following two remarks suppose nevertheless possible improvements:

- the binary image determines the way the encoding algorithm uses the two possible coding LUTs; it is clear that coding adjacent pixels with the same process is preferable as doing this with two different processes; for this reason the less important the frontier (set of pixels of value 1 that have at least one neighbor of value 0) is, the best the result will be; figure 20 shows the frontier of the segmentation area of the point-to-point detection algorithm; the frontier represents 3404 pixels; let us consider already a simpler segmentation image (figure 21a) without thinking how it has been obtained; the frontier evaluation of this image is illustrated by Figure 21b and comprises 1604 pixels; thus thanks image segmentation enhancement the size of the code switching is practically strongly reduced.

- the detected skin area will be encoded using the false contour optimized code; as this artifact occurs only in extended areas, small areas are useless for false contour optimization; in contrary of that, as the false con-

tour optimized code necessarily has worth dithering noise behavior, it would be even preferable to delete small areas from the skin detection image.

Obtaining less noisy results requires additional processing that fundamentally treats the image per region and not per pixel. The first improvement
 5 consists in a contextual area increasing. The second improvement is based on a multi scale analysis of the image and the last one is the use of form analysis for binary image manipulation for removing the pixels of the detected area that are not compatible with the assumed form of the detected
 10 area.

2.1 Hysterisis skin area increasing (first improvement)

Let us consider a color picture $I = (I(i, j))_{0 \leq i \leq m-1, 0 \leq j \leq n-1}$ of m lines and n columns.
 15 The criterion D' defines a binary skin detection image $D'(I)$. This image is obtained from the intermediary image $\Delta = \delta(I)$ with a resemblance threshold α . If the threshold α is too big ($\alpha = \alpha_{sup}$ Figure 22a), the detected areas do not cover well the skin areas, in other words, as far as the hue deviates the pixel will not be detected as a skin pixel. In Figure 22a, $m_H=0.39$, $S_{min}=0.1$,
 20 $S_{max}=0.9$, $V_{min}=30$, $V_{max}=250$ and $\alpha=0.9$. In contrary of that if α is too small ($\alpha = \alpha_{inf}$ in Figure 22b), the result will cover with a good accuracy the skin areas but will introduce undesirable residual areas too. The hysteresis area increasing allows covering skin area with a better accuracy without introducing additional residual areas while using effectively both resemblance thresholds
 25 α_{inf} and α_{sup} .

Figure 23 shows the result of the hysteresis algorithm. The inferior resemblance threshold is $\alpha_{inf} = 0.8$ and the superior resemblance threshold is $\alpha_{sup} = 0.9$. Some objects straightforwardly disappear compared to the result
 30 of the point-to-point detection parameterized with the resemblance threshold

$\alpha = 0.8$ whereas the detection of the real skin area detection remains unchanged.

The next example details the hysteresis increase principle through a didactic example. Let us consider a small image $\Delta = \delta(I)$ of 6×6 pixels (Figure 24a) containing four different resemblance values $R_1 < R_2 < R_3 < R_4$ such as $R_1 < \alpha_{inf} < R_2$ and $R_3 < \alpha_{sup} < R_4$. Let us also have a look to the results of the application of the detection criterion D' with the resemblance thresholds α_{sup} (Figure 24b) and α_{inf} (Figure 24c). Black means detected.

The elementary hysteresis extension consists in the following processing (steps 1 to 3). The needed data are the resemblance thresholds α_{inf} and α_{sup} (with $\alpha_{inf} \leq \alpha_{sup}$), the image Δ and two symbols σ and τ .

Step 1) Each pixel (i, j) in Δ is labelled with symbol σ when $D'(i, j) \geq \alpha_{sup}$.

Step 2) Each pixel (i, j) such as $\Delta(i, j) \geq \alpha_{inf}$ having a neighbor (i', j') labeled σ is labelled with the second symbol τ .

Step 3) When all pixels have been tested, all symbols τ are replaced with symbol σ and then step 1) is carried out again

An iteration number n_{iter} determines in addition to that the number of times the elementary extension has to be repeated. The result is obviously the set of the pixels labeled with symbol σ . Figure 25 shows the first 4 iterations of the elementary hysteresis extension. Here the process stationeries at the fourth iteration. The detected pixels at the end of the fourth iteration are a subset of the detected pixels of Figure 24c (upper left part of Figure 24c).

As a variant, The elementary hysteresis extension consists in the following processing (steps 1 to 3). The needed data are the resemblance thresholds α_{inf} and α_{sup} (with $\alpha_{inf} \leq \alpha_{sup}$), the image Δ and one symbols σ .

Step 1) Each pixel (i, j) in Δ is labelled with symbol σ when $D'(i, j) \geq \alpha_{sup}$.

Step 2) Each pixel (i, j) such as $\alpha_{\text{sup}} \geq \Delta(i, j) \geq \alpha_{\text{inf}}$ having a neighbor (i', j') labeled σ is also labelled with the symbol σ .

Step 3) When all pixels have been tested, returning to step 1) .

5 2.2 multi scale color analysis (second improvement)

The multi scale approach consists in extracting the information in all images resulting of a sub sampling of the first original image. Figure 26 illustrates the multi scale analysis: the original image $I = I^{(0)}$ (upper left corner) is consti-
10 tuted of 128×128 pixels. Thus it is possible to consider 7 ($128 = 2^7$) sub samples $I^{(1)}, \dots, I^{(7)}$ that are represented in the mosaic.

A multi scale approach is profitable when the information to be extracted in the image lies over all scales. Apparently the color is a local property be-
15 cause it is defined only by the three values R, G and B at each pixel. In this sense, a multi scale analysis is useless. On the other hand photography shows under multiple lighting a set of colored objects. These objects have a spatial extension and the color is partially bounded to these objects. As conclusion, in real images, color effectively has a spatial coherence. In other
20 words, color information of a pixel does not only lie at the considered pixel but in the neighborhood too.

The following intuitive idea takes already into account the influence of the neighborhood. At a given pixel (i, j) of the image I , instead of taking the con-
25 text free decision $D'(I(i, j))$ as already defined, let us compute the mean color $I_V(i, j)$

$$I_V(i, j) = \text{card}(V)^{-1} \times (\sum_{s \in V} I(s)),$$

where card is the number of elements over a rectangular neighborhood V and take the decision $D'_V(I)(i, j) = D'(I_V(i, j))$.

This first step is a particular case of the multiscale approach where only a single sub sampling is used. The next paragraph introduces the multi scale analysis in its general form.

5 An image $I = (I(i, j))_{0 \leq i \leq m-1, 0 \leq j \leq n-1}$ is analyzed. In this paragraph, it is simpler to handle a square image whose dimensions are powers of two (that is to say $\exists p \in \mathbb{N} / m = n = 2^p$). The series $(I^{(k)})_{k \geq 0}$ represents the different degree of sub sampling of the image I . Then $I^{(0)} = I$, $I^{(1)} = I'$ is the image of size $\frac{1}{2} m \times \frac{1}{2} n$ and at each pixel (i', j') of I' , $I(i', j')$ is equal to the mean of the image I at the
 10 four points $(2i', 2j')$, $(2i'+1, 2j')$, $(2i'+1, 2j'+1)$ and $(2i, 2j+1)$. The image I' can be considered as the derivative of the image I .

The relation bounding to successive images $I^{(k)}$ and $I^{(k+1)}$ is obviously the same as the one between I and I' , that is two say, for each k , $I^{(k+1)} = (I^{(k)})'$. The
 15 sub sampling can not continue infinitely because the resolution is obviously limited. Numerically, if $m = n = 2^p$, it is clear that the derivation ends at the degree p . In fact, the image $I^{(p-1)}$ has 4 pixels and image $I^{(p)}$ only one.

In addition to that, the derivate image I' can be interpreted as an image of
 20 same size as I but whose pixel $s' = (i', j')$ are four time bigger as the pixels of the original I . Then each pixel s' covers the four pixel $(2i', 2j')$, $(2i'+1, 2j')$, $(2i'+1, 2j'+1)$ and $(2i, 2j+1)$. Consequently for each pixel $s = (i, j)$ of the original image, it is possible to define a unique series $s^{(k)} = (i^{(k)}, j^{(k)})$, $k \in \{0, \dots, p\}$, such as each pixel $s^{(k)}$ covers its previous pixel $s^{(k-1)}$.

25 These definitions show the way the multi scale analysis extracts information in all scales in the image I . The purpose is to take at each pixel $s = (i, j)$ a decision $D'_i(s)$ that generalizes the point-to-point detection decision $D'(s)$ by using additional information in the image I . The resemblance function δ and
 30 the resemblance threshold α intervene this way in the multi scale decision. Let δ_α be a function taking its value in the range $[-1, 1]$ such as:

$$\delta(H, S, V) = 1 \Rightarrow \delta_\alpha(H, S, V) = 1$$

$$\delta(H, S, V) = \alpha \Rightarrow \delta_\alpha(H, S, V) = 0$$

$$\delta(H, S, V) = 0 \Rightarrow \delta_\alpha(H, S, V) = -1$$

5 There is no interest to take a complicate form. The chosen definition is simply a piecewise linear function f_α of δ that satisfies the three above conditions.

$$(4) \quad f_\alpha(\delta) = \delta_\alpha = \alpha^{-1}\delta - 1 \quad \text{if} \quad \delta \leq \alpha$$

$$10 \quad f_\alpha(\delta) = \delta_\alpha = (1 - \alpha)^{-1} (\delta - \alpha), \quad \text{else}$$

Let us now define recursively the multi-scale resemblance value $\rho_\alpha(I)(s)$ of the pixel s of the image I for the given resemblance threshold α . This definition introduces then not only the quantity $\rho_\alpha(I)(s)$ but all quantities $\rho_\alpha(I^{(k)})(s^{(k)})$ as the recursive following formula shows :

$$\begin{aligned} \rho_\alpha(I^{(p)})(s^{(p)}) &= \delta_\alpha(I^{(p)})(s^{(p)}) \\ \rho_\alpha(I^{(k)})(s^{(k)}) &= \delta_\alpha(I^{(k)})(s^{(k)}) + \omega_{k+1} \rho_\alpha(I^{(k+1)})(s^{(k+1)}) \quad \forall 0 \leq k \leq p-1 \end{aligned}$$

Figure 27 reuses the example of figure 26. Each image $I^{(k)}$ of figure 26 is used to compute the images $\delta(I^{(k)})$ with $0 \leq k \leq p$. In this case where the derivation goes until the degree $p = 7$ and where ω_k is identical for all picture $I^{(k)}$ and is equal to ω , the final result $\rho_\alpha(I)$ is :

$$\begin{aligned} \rho_\alpha(I) &= \rho_\alpha(I^{(0)}) = \sum_{k=0, \dots, 7} \omega^{-k} \delta_\alpha(I^{(k)}) \\ &= \sum_{k=0, \dots, 7} \omega^{-k} f_\alpha(\delta(I^{(k)})) \end{aligned}$$

25 This is a simple combination of these 8 images. This combination is represented by the image of Figure 28a, whereas the segmentation result with a resemblance threshold of the image $\rho_\alpha(I)$ equal to zero is shown in Figure 28b.

30 These formulas are enough to define precisely $\rho_\alpha(I)(s)$ because $\rho_\alpha(I)(s) = \rho_\alpha(I^{(0)})(s^{(0)})$. The introduction of the coefficients ω_k is clear. Each coefficient ω

has the function to limit the weight of previous term $\rho_\alpha(I^{(k+1)})(s^{(k+1)})$ in comparison to the term $\delta_\alpha(I^{(k)})(s^{(k)})$. Then it is obvious that ω_k has to be chosen between 0 and 1. The greater the coefficient ω is, the more the influence of the sub samples is important, that is to say the more the neighborhood influences the final result. The computed values $\rho_\alpha(I)(s)$ at each pixel of the image are evenly distributed around value 0. This is the purpose of the transformation from the couple (δ, α) to the single function δ_α . Then the resulting skin detection image is obtained by a threshold of image $\rho_\alpha(I)$ with a value β equal to 0.

10

The computation of $\rho_\alpha(I)$ followed by a comparison to a resemblance threshold equal to 0 leads to a new detection criterion that theoretically extracts the color information of all the sub samples $I^{(k)}$ of I . Practically it is preferable to stop the process to a sub sampling degree $q < p$. The coherence in the image lies effectively rarely over the entire image. Furthermore, if the iterative process begins with the image $I^{(q)}$, $q < p$, that is to say, $\rho_\alpha(I)$ will be:

15

$$\begin{aligned}\rho_\alpha(I^{(q)})(s^{(q)}) &= \delta_\alpha(I^{(q)})(s^{(q)}) \\ \rho_\alpha(I^{(k)})(s^{(k)}) &= \delta_\alpha(I^{(k)})(s^{(k)}) + \omega_{k+1} \rho_\alpha(I^{(k+1)})(s^{(k+1)}) \quad \forall 0 \leq k \leq q-1\end{aligned}$$

The detection at each pixel $s = (i, j)$ only depends on values $I(s'), s' \in s^{(q)}$ and not on the entire image I . Figure 29 is the multi scale detection algorithm applied to the test image with a degree $q = 7$. Compared to the basic point-to-point detection (criterion D'), the detection is more robust against missing detection inside skin areas and against false detection outside skin areas.

25

Finally for two pixels $s_1 = (i_1, j_1)$, $s_2 = (i_2, j_2)$, let note $r = \min \{0 \leq k \leq p; s_1^{(k)} = s_2^{(k)}\}$. Then from $k = p$ to $k = r$, $\rho_\alpha(I^{(k)})(s_1^{(k)}) = \rho_\alpha(I^{(k)})(s_2^{(k)})$. As consequence, the computation of image $\rho_\alpha(I)$ can be done at a very low cost. To make a comparison, up to a certain mask size, the mask linear filtering of image I requires more operations than the computation of $\rho_\alpha(I)$.

30

2.3 Form analysis (third improvement)

Form analysis consists in separating areas in a binary image and associating to each form topologic characteristics such as the surface because in a digital world the surface of a form is identified to its number of pixels. Form analysis is unavoidable to modify powerfully a basic intermediate binary image. Point-to-point skin detection can not produce a proper binary image. For any chosen detection parameters, the result contains necessarily more or less undesirable residual areas. That the human perception can identify easily these areas does not mean that the subjacent processing is simple. In fact deleting or modifying specific areas requires a form analysis of the entire segmentation image.

This kind of treatment represents a significant cost but is interesting because as far as it is possible to identify each area of a binary image the field of possible operations becomes very large.

In a general manner, it consists in determining in the picture homogeneous areas of a first type comprising only pixels considered as skin pixels and keeping as skin pixels only the pixels of the homogeneous areas of first type having a size bigger than a predetermined minimum size. Furthermore, it also consists in determining in the picture homogeneous areas of a second type comprising only pixels not considered as skin pixels and considering as skin pixels the pixels of the homogeneous areas of second type having a size smaller than a predetermined minimum size are.

Mathematically the area analysis or form analysis consists in labeling related parts in a binary image. The connexity has now to be clarified. Some notations are useful to go further. Let $S = [0, m - 1] \times [0, n - 1]$ be the set of the pixels. Let $\{0, 1\}$ be the set of the values of a binary image. Then a binary image is an element $(I_s)_{s \in S}$ of $\{0, 1\}^S$. The connexity depends essentially on the chosen neighborhood system. The neighborhood system defines for

each pixel the set of pixels that are said to be its neighbor. One simple system is the 4-neighborhood. Each pixel has the same neighborhood type that can be defined as follows. For each pixel $s = (i, j)$ the neighborhood of s is:

$$V(s) = \{s, (i+1, j), (i, j+1), (i-1, j), (i, j-1)\} \cap S$$

5

A binary image is obviously divided into two basic classes, the class $I^{-1}(1) = \{s \in S; I(s) = 1\}$ and the class $I^{-1}(0) = \{s \in S; I(s) = 0\}$. The first class $I^{-1}(1)$ is said to be the set of the forms of image I . The second class $I^{-1}(0)$ is then the background. The connexity notion lies on the set of the forms. Two pixels s_1 and s_2 of $I^{-1}(1)$ are said to be connected in I if and only if there exists an integer $w \geq 1$ and a continuation $(s^{(k)})_{0 \leq k \leq w}$ such as

$$\begin{aligned} s^{(0)} &= s_1 \\ s^{(w)} &= s_2 \\ \forall 0 \leq k \leq w-1, s^{(k+1)} &\in V(s^{(k)}) \cap I^{-1}(1) \end{aligned} \tag{6}$$

15

The notation $s_1 @_I s_2$ indicates that $s_1, s_2 \in I^{-1}(1)$ are bounded with the relation (6). For any binary image, $@_I$ is an equivalence relation. Then it defines a partition of the set of the forms. Each form $F \subset I^{-1}(1)$ is defined as an equivalence class of the relation $@_I$. In other words, for any form F , there is a pixel s so that $F = \{s'; s @_I s'\}$.

20

The quick example of figure 30a illustrates the 4-neighborhood concept and the connexity. There are 3 pixels (respectively labeled red, yellow and green). The red one and the yellow one are connected and a white line between them is traced. In contrary of that it is impossible to find a way between the red one and the green one.

25

The purpose of the form analysis algorithm is to label each pixel of the image in accordance to the relation $@_I$. The effective resolution depends strongly to the considered neighborhood system. The 4-neighborhood has the interesting property to allow an easy step by step estimation of the con-

30

nexity relation step by step thanks to a recursive four parts cutting of the image. In other words it is sufficient to know the set of the forms in each quarter of an image I to know the forms of the entire image.

5 Here comes the description of the principle of the form analysis algorithm through a synthetic basic example. The data is a binary image. The purpose of the form analysis is to label each form and to gather for each form F a set of parameters. In the following example, the parameters are (see also illustration of figure 30b):

- 10 - the volume $V(F)$ that is the number of pixels of the form,
 - the upper left corner $ULC(F)$ which is the upper left corner of the enclosing rectangle,
 - the horizontal diameter $d_H(F)$ of the form,
 - the vertical diameter $d_V(F)$ of the form.

15

On Figure 31a is drawn a 16×16 binary image ($L = 16$). Each color of the neighbor picture (Figure 31b) represents a form of the binary one. There are 8 labels (Black, gray, red, green, orange, blue, pink, chestnut).

20 It is easy to deduce from the form analysis of each quadrant illustrated by Figure 32 the definitive result (that is to say picture of labels above). In figure 32, image quarter 1 is identified by symbol "•", image quarter 2 by "■", image quarter 3 by "\" and image quarter 4 by "/". The form analysis of each quadrant is synthesized by the four pictures of size 8×8 .

25

Furthermore, the next tables gather form attributes (volume, diameters, upper left corner). The first table (Table 1) gathers attributes of the forms of the entire image and the second table (Table 2) those of each quarter. A color label identifies a form. In table 1, the last row indicates for each label the list of the form of the 4 sub images belonging to it. For each quarter, the last line of table 2 indicates the enclosing label of the entire image.

Attribute / label	Black	Gray	Red	Green	Blue	Pink	Chest-nut
Volume	9	76	7	3	4	4	10
Horizontal diameter	3	12	4	2	3	3	7
Vertical Diameter	5	14	3	2	2	3	2
Upper left Corner	(11, 0)	(0,1)	(0, 10)	(7, 7)	(4, 0)	(12, 12)	(15, 6)
Sub labels	<i>Green 3 (that is to say green label of quadrant 3)</i>	<i>Gray 1 Red 1 Gray 2 Gray 3 Blue 4</i>		<i>Green 1 Orange 3 Red 4</i>	<i>Blue 1</i>	<i>Green 3</i>	<i>Red 3 Gray 4</i>

Table 1

Attribute	Label				
	Gray	Red	Blue	Green	Orange
Upper left (1)					
Volume	19	6	4	1	
Horizontal diameter	7	4	3	1	
Vertical Diameter	4	2	2	1	
Upper left Corner	(0, 1)	(6, 2)	(4, 0)	(7, 7)	
<i>Enclosing label</i>	<i>Gray</i>	<i>Gray</i>	<i>Blue</i>	<i>Green</i>	
Upper right (2)					
Volume	16	7	2		
Horizontal diameter	4	4	1		
Vertical Diameter	6	3	2		
Upper left Corner	(2, 0)	(0, 2)	(6, 7)		
<i>Enclosing label</i>	<i>Gray</i>	<i>Red</i>	<i>Orange</i>		
Down right (3)					
Volume	16	8	5	4	1
Horizontal diameter	5	5	2	3	1
Vertical Diameter	6	2	4	3	1
Upper left Corner	(0, 0)	(6, 0)	(0, 6)	(4, 4)	(0, 0)
<i>Enclosing label</i>	<i>Gray</i>	<i>Braun</i>	<i>Green</i>	<i>Pink</i>	
Down left (4)					
Volume	2	1	25	9	
Horizontal diameter	2	1	6	3	
Vertical Diameter	1	1	6	5	
Upper left Corner	(7, 6)	(0, 7)	(0, 2)	(3, 0)	
<i>Enclosing label</i>	<i>Chestnut</i>	<i>Green</i>	<i>Gray</i>	<i>Black</i>	

Table 2

All tabled attributes are adapted to the recursive form analysis. The following formulas (for volume, diameters and upper left corner) show how each parameter of the reunion of p forms inherits the values of each form:

- 5 □ Volume of the aggregation of p forms $F_1, \dots, F_p$:

$$V(F_1 \cup \dots \cup F_p) = \sum_k V(F_k)$$

- Upper left corner:

$$ULC(F_1 \cup \dots \cup F_p) = (\min_k((i_k + ULC_i(F_k))), \min_k((j_k + ULC_j(F_k))))$$

where

- 10 - (i_k, j_k) is the upper left corner of the enclosing quarter of form F_k ; if F_k belongs to the quarter 1 (respectively 2, 3, 4) $(i_k, j_k) = (0, 0)$ (respectively $(0, L/2)$, $(L/2, L/2)$, $(L/2, 0)$); and
 - $ULC = (ULC_i, ULC_j)$

- Horizontal and Vertical Diameter:

15 $(d_v(\cup_k F_k), d_h(\cup_k F_k)) = \max_k ((ULC'_i(F_k) + d_v(F_k), d_h(F_k))) - ULC(\cup_k F_k)$
 with $ULC'(F) = (i, j) + ULC(F)$

20 The above basic example shows the recursive functioning of the form analysis algorithm. Internally, the algorithm identifies the forms with labels and aggregates them step-by-step. These labels are not color but integers and the number of them can reach practically the number of pixels of the image. Practically, the labelling consists in associating to the image a field of integer $(E_s)_{s \in S}$ that verifies :

25 $I(s) = 0 \Leftrightarrow s \in I^{-1}(0)$
 $I(s) \neq 0 \text{ and } s @_l t \Leftrightarrow E(s) = E(t)$

30 The label field E allows easily saying if two odds pixels belong to the same form or not. Conversely, it is also interesting to be able to identify immediately each labeled form. The field E only does not allow this. It is necessary

to add a list $(L_e)_{e \in E(S)}$ indexed by the set of the label $E(S) = \{e_1, \dots, e_k\}$, such as for each $e \in E$,

$$L_e = \{s \in S; E(s) = e\} = E^{-1}(e).$$

5 The couple formed by the label field E and list L is the fundament of the form analysis. But not only the connexity can be easily recursively deduced. Then each label e (identifying each form) can be accompanied with a large family of form properties that have this interesting specificity. Including the attributes of the example, the following list of attributes can be this way estimated

10 :

- The volume V (number of pixels of a form);
- The horizontal and vertical diameters (width and height of the smallest including rectangle);
- The minimum in sense of the relation (particular point of the form frontier)

15

$$S(i, j) < (i', j') \text{ if and only if } i < i' \text{ or } i = i' \text{ and } j < j';$$

- The perimeter; and
- The inside of each form.

20 Some attributes like the two last ones constitute important additional data and are not necessarily useful for the future processing. Other attributes are simple integer coefficient and do not represent an important additional memory occupation.

25 The immediate result that the form analysis gives is obviously area erasing. The criterion for deleting area is mostly the volume of the area. If the volume is too small, the area is undesirable and can be removed. The holes in a form can be interpreted several manners. It can be a miss of the detection. It can be also a detail in a skin area that does not have the same color. In the first case, it is logical to fill the hole. In the second case, it is preferable too

30 to fill the hole in order to minimize the size of the entire frontier between the

forms (set $I^{-1}(1)$) and the background (set $I^{-1}(0)$). This is a second direct result of the form analysis.

Figure 33 shows the application of the form analysis implementing the small detection areas erasing and the hole filling. The threshold value is $\alpha = 0.8$, then the algorithm works on one of the above punctual detection results that have been reproduced in this specification.

3 Implementation of the inventive method

3.1. Global processing diagram

The following above developed processings can be implemented in a real time working system:

- Detection criterion D' ;
- Point-to-point detection implementing D' ;
- Multi scale color detection ;
- Hysteresis area increasing;
- Form analysis.

The global encoding concept can be synthetised by the diagram of Figure 34. A RGB image is encoded for a plasma screen display. The RGB video data are first transformed into HSV video data by transformation means 100. Skin tone areas are then detected in the HSV data by skin detection means 110. This detection is used for segmenting the picture to be encoded. The diagram shows the construction of p different segmentation images $I_1, I_2, \dots, I_p$, each image corresponding to one particular skin tone. The used criteria for building these binary images are respectively $D'_1, \dots, D'_p$, each one being similar to D' with particular parameters ($m_{H,i}, \sigma_{H,i}, S_{min,i}, S_{max,i}, V_{min,i}, V_{max,i}, i = 1, \dots, p$). The continuation of the process consists in computing the disjunction $I = I_1 \wedge \dots \wedge I_p$ of the p intermediate images Then for any pixel (i, j) , if $I(i, j) = 1$, one or more of the p images $I_1, I_2, \dots, I_p$, is equal to 1, then this

point belongs to a skin area, the first LUT has to be applied. Else, the point does not belong to a skin area, then the second LUT has to be applied. These skin detection means outputs for each of the picture to be displayed a bit indicating if this pixel is a skin tone pixel or not. This bit is then used by
5 encoding means 120 for encoding the RGB video data of the picture. There are two available encoding look-up tables, LUT2[1] and LUT2[2], for encoding these data. The look-up table LUT2[1] is optimized for false contour effect reduction. Due to the dithering, it is clear that this LUT can not have a good noise behavior. The second look-up table LUT2[2] uses less
10 dithering and then has a better noise behavior. The counterpart is nevertheless that it introduces false contour. Skin area detection is compatible to these both LUTs because it is critical for false contour effect. The data outputted by the encoding means are subfield data.

15 The diagram supposes a priori that all pixels of the RGB image are available for the computation of I. In the same way, all pixels of the binary image are eventually available for the encoding. Those points must be specified for hardware implementation. The skin detection improvements can be classified in two types.

- 20
- Image context integration: The detection criterion let intervene several neighbor pixels. It is exactly the multi scale analyse.
 - Image processing: Matches the segmentation image to encoding. It concerns the skin area hysteresis increasing and the form analysis.

25 **3.2 Point-to-point detection algorithm**

The point-to-point detection does not require specific memory at all. It is widely sufficient to do simultaneously with the subfield data the RGB to HSV transformation and the skin color detection. For each coming R, G, B data, it
30 applies the first or second LUT in accordance to the produced 1 bit skin detection data $D'(H, S, V)$ as shown by figure 34. The skin detection block

detects one skin tone and for each of them the following parameters can be used:

Skin / Parameters	Europe	Asia	Afrika	All
m_H	0.33	0.29	0.43	0.39
σ_H	0.92	0.90	0.90	0.85
S_{min}, S_{max}	0.2, 0.8	0.2, 0.8	0.2, 0.8	0.2, 0.8
V_{min}, V_{max}	40, 255	40, 255	30, 255	30, 255

This table distinguishes 3 color types (European, Asian and Afrikan skin tone) for which statistical results have been computed. However, if just one skin tone ($p = 1$ in Figure 34) is used, as the hue is neighbor from one color type to another, it is possible to cover all skin tones (however with a lower efficiency than with $p > 1$) using parameters of the column "all".

Figure 35 is a more complete diagram showing all the processings applied to the RGB video data. Notably, It shows that, before being encoded, the video data are gamma corrected (block 10), transformed before dithering (block 20) and then dithered (block 30). The block 20 uses two other look-up tables LUT1[1] and LUT1[2] for implementing GCC coding. The 1 bit skin detection data is used for selecting either LUT1[1] for block 20 and LUT2[1] for block 120 or LUT1[2] for block 20 and LUT2[2] for block 120.

3.3 Image context integration for skin area detection

The multi scale analysis requires a memorisation of a rectangular region enclosing the pixel for which the skin detection is applied. The diagram of figure 36 shows an example of the implementation of the detection algorithm where the considered rectangular region is the rectangle of size 3×7 . As the diagram implies it, a minimum of 4 line memories 131 to 134 allows a processing over this region. If the device works with clock T, the first line

memory 131 receives at the cadence T the computed HSV values of incoming RGB values of image I . Each time the first line memory 131 is filled, that is to say each $n \times T$ (n is the number of pixels per line in the image), the oldest memory line (which ?) is removed, the three following
 5 ones 132 to 134 are translated so that the data used by the skin detection means 110 are updated. Each T the skin detection block 110 can read in the three memorised lines 132 to 134 the 21 ($=3 \times 7$) HSV required values for the detection. In the general case the mask has p lines and q columns. Then, the number of required line memories is $p + 1$ and in each of the p last
 10 lines, the skin detection block reads q HSV values. The processing finally requires a delay of p lines, that is to say $p \times n \times T$.

At the edges of the image (in the 3×7 size mask case, edges are the first line and last line, the 2 first and 2 last rows), the detection still works by
 15 repeating the necessary number of times the extreme lines and rows.

3.4 Skin area processing

The hysteresis region increasing works as mentioned before on the entire image $\delta(I)$ whereas the form analysis of the image $D(I)$ or $D'(I)$ needs to have
 20 the entire binary image during the processing. Consequently, it is necessary to add frame memories to implement these two processes.

Figure 37 represents the necessary line memories for a multi scale analysis. The region detection is generalized to p lines $\times$ q rows. As in the previous
 25 case, the detection algorithm delivers at the cadence T either a 8 bit data corresponding to the value $\delta(H, S, V)$ or simply a 1 bit data corresponding to the detection $D(H, S, V)$ ($D(H, S, V) = \delta(H, S, V) \geq \alpha$). For the hysteresis area increasing the 8 bit data is required, for the form analysis, only the 1 bit detection value is useful. These data must be stored at the cadence T in a
 30 first temporary frame memory 141. Each time this frame memory 141 is filled (that is to say each $m \times n \times T$), the entire content ($m \times n \times 8$ bits in hysteresis

increasing case or $m \times n \times 1$ bits in form analysis case) is copied in a second memory 142 that do not change during $m \times n \times T$. The area increasing or the form analysis block 150 can then work on this memory. The result ($m \times n$ bits that is to say a binary image) is stored in a third memory 143. This
5 memory is read at the cadence T for the correct encoding of each pixel. Finally a delay of one frame (that is to say $m \times n \times T$) is necessary so that the image processing works with the right skin detection image.

CLAIMS

1. Method for detecting skin tone pixels in a video picture, characterized in
5 that it is based on the hue, the saturation and the colour value of the pixels
of the picture and in that the skin tone pixels belongs to a volume in the Hue
Saturation Value (HSV) color space, called HSV space wherein the hue of
the skin tone pixels is comprised between a minimum hue (H^-) and a
maximum hue (H^+).
- 10 2. Method according to claim 1, wherein
- the value (V) of the skin tone pixels in said HSV space is comprised be-
tween a minimum value ($V_{\min}$) and a maximum value ($V_{\max}$), the probability
density of the value (V) being equal to 1 between said minimum and maxi-
15 mum values ($V_{\min}, V_{\max}$),
 - the saturation (S) of the skin tone pixels in said HSV space is comprised
between a minimum saturation ($S_{\min}$) and a maximum saturation ($S_{\max}$), the
probability density of the saturation (V) being equal to 1 between said mini-
mum and maximum values ($S_{\min}, S_{\max}$).
- 20 3. Method according to claim 2, characterized in that it comprises a step for
applying a resemblance function () to the pixels of the picture, said resem-
blance function based on the values $V_{\min}$, $V_{\max}$, $S_{\min}$, $S_{\max}$ and being maximal
when the density probability ($p_{(H,S,V)}$) of the hue, saturation and value is
25 maximal and a step for comparing the results of said application to a resem-
blance threshold (), the pixels whose resemblance function is greater than
or equal to said resemblance threshold () being considered as skin pixels.
- 30 4. Method according to claim 2, characterized in that it comprises the follow-
ing steps :

- a) applying a resemblance function () to the pixels of the picture, said resemblance function being based on said minimum and maximum saturations and said minimum and maximum values and being maximal when the density probability ($p_{(H,S,V)}$) of the hue, saturation and value is maximal,
- 5 b) comparing the results of said application to a first resemblance threshold (θ_{sup}), the pixels whose resemblance function result is greater than or equal to said first resemblance threshold being labelled by a first symbol (),
- c) comparing the results of said application to a second resemblance threshold (θ_{inf}), the pixels whose resemblance function result is greater than or
- 10 equal to said second resemblance threshold and having at least one neighbour pixel labelled by a first symbol () being labelled by a second symbol (), said second resemblance threshold (θ_{inf}) being lower to said first resemblance threshold (θ_{sup}),
- d) when all pixels of the picture have been compared, replacing all second
- 15 symbols () by first symbols (), and
- e) returning to step a), the number of iterations (n_{iter}) of said steps sequence being predefined, the pixels being labelled by a first symbol () at the end of the last iteration being considered as skin pixels.

20 5. Method according to claim 2, characterized in that it comprises the following steps :

- a) applying a resemblance function () to the pixels of the picture, said resemblance function being based on said minimum and maximum saturations and said minimum and maximum values and being maximal when the density probability ($p_{(H,S,V)}$) of the hue, saturation and value is maximal,
- 25 b) comparing the results of said application to a first resemblance threshold (θ_{sup}) and labelling the pixels whose resemblance function result is greater than or equal to said first resemblance threshold by a symbol (),
- c) comparing the results of said application to a second resemblance threshold (θ_{inf}) and labelling by a same symbol () the pixels having at least one
- 30 neighbour pixel labelled by said symbol () whose resemblance function

result is greater than or equal to the second resemblance threshold and lower than or equal to said first resemblance threshold, said second resemblance threshold (θ_{inf}) being lower to said first resemblance threshold (θ_{sup}), and

- 5 d) returning to step a), the number of iterations (n_{iter}) of said steps sequence being predefined, the pixels being labelled by said symbol (γ) at the end of the last iteration being considered as skin pixels.

6. Method according to claim 4 or 5, characterized in that it further comprises
10 the following steps :

- generating K-1 derivate pictures $I^{(k)}$, $k \in [1, K-1]$, by subsampling said picture I corresponding to $I^{(0)}$, each derivate picture $I^{(k)}$ being the derivate picture of the picture $I^{(k-1)}$,
- applying a resemblance function (ρ_α) to the pixels of all pictures $I^{(k)}$, $k \in [0, K-1]$,
15
- computing a multiscale resemblance value (ρ_α) based on the results of the application of the resemblance function (ρ_α) to the pixels of all pictures $I^{(k)}$, $k \in [0, K-1]$, and
- comparing the multiscale resemblance value (ρ_α) to a third resemblance
20 threshold (β), the pixels whose multiscale resemblance value is greater than or equal to said resemblance threshold (β) being considered as skin pixels.

7. Method according to Claim 6, characterized in that, for a given pixel, the multiscale resemblance value (ρ_α) is the sum of the results of the application
25 of the resemblance function (ρ_α) to the pixels of all pictures $I^{(k)}$, $k \in [0, K-1]$, each result for a given picture $I^{(k)}$ being weighted by a coefficient (w_k) related to said picture.

8. Method according to one of the preceding claims, characterized in that it
30 further comprises a step for determining in the picture homogeneous areas of a first type comprising only pixels considered as skin pixels and in that

only the pixels of the homogeneous areas of first type having a size bigger than a predetermined minimum size are kept as skin pixels.

9. Method according to one of the preceding claims, characterized in that it
5 further comprises a step for determining in the picture homogeneous areas of a second type comprising only pixels not considered as skin pixels and in that the pixels of the homogeneous areas of second type having a size smaller than a predetermined minimum size are considered as skin pixels.

10. Method for processing video data for display on a display device having
10 a plurality of luminous elements corresponding to the pixels of a picture, wherein the time of a video frame or field is divided into a plurality of sub-fields during which the luminous elements can be activated for light emission in small pulses corresponding to a sub-field code word of n bits used for en-
15 coding the p possible video levels lighting a pixel, comprising the steps of:
- determining in a picture, an homogeneous area, called first part of the picture, having a size larger than a predetermined minimum size and a skin colour,
- encoding said first part of the picture using a first encoding method,
20 wherein among the set of possible video levels for lighting a pixel, a sub-set of m video levels with $n < m < p$ is selected, which is used for light generation, said m values being selected according to the rule that the temporal centre of gravity (CG1, CG2, CG3) for the light generation of the corresponding sub-field code words grows continuously with the video level, and
25 - encoding at least one second part different from said first part of the picture using a second encoding method different from said first encoding method, characterized in that the method of any one of claims 1 to 9 is used for determining said first part of the picture.

1/14

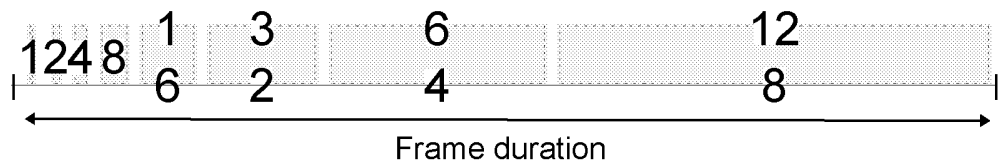


FIG.1

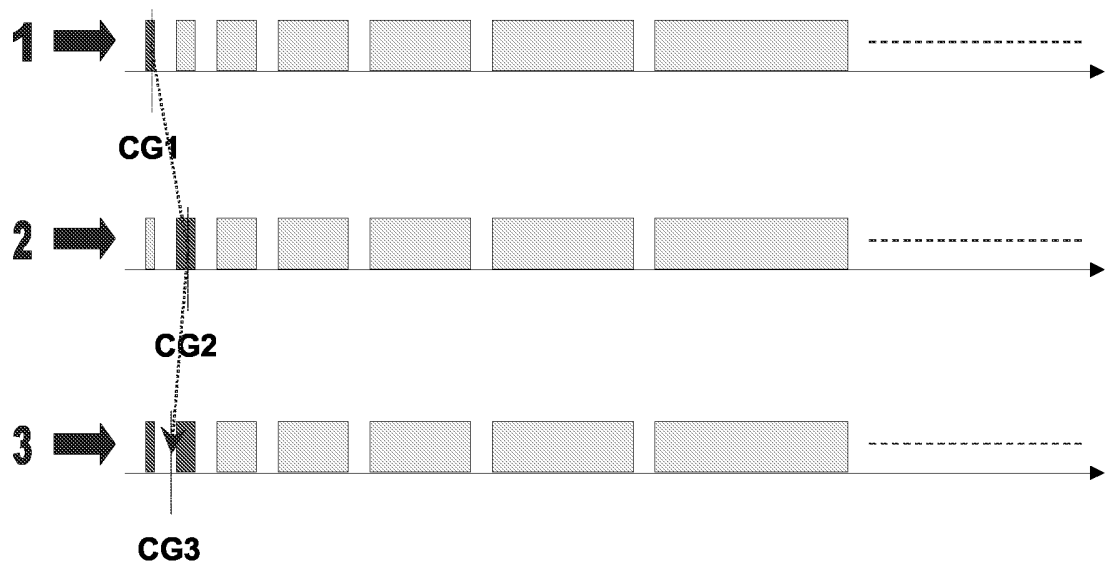


FIG.2

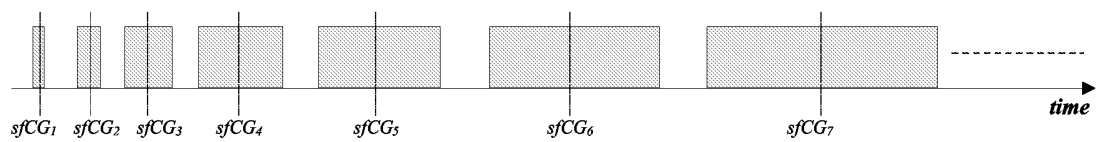


FIG.3

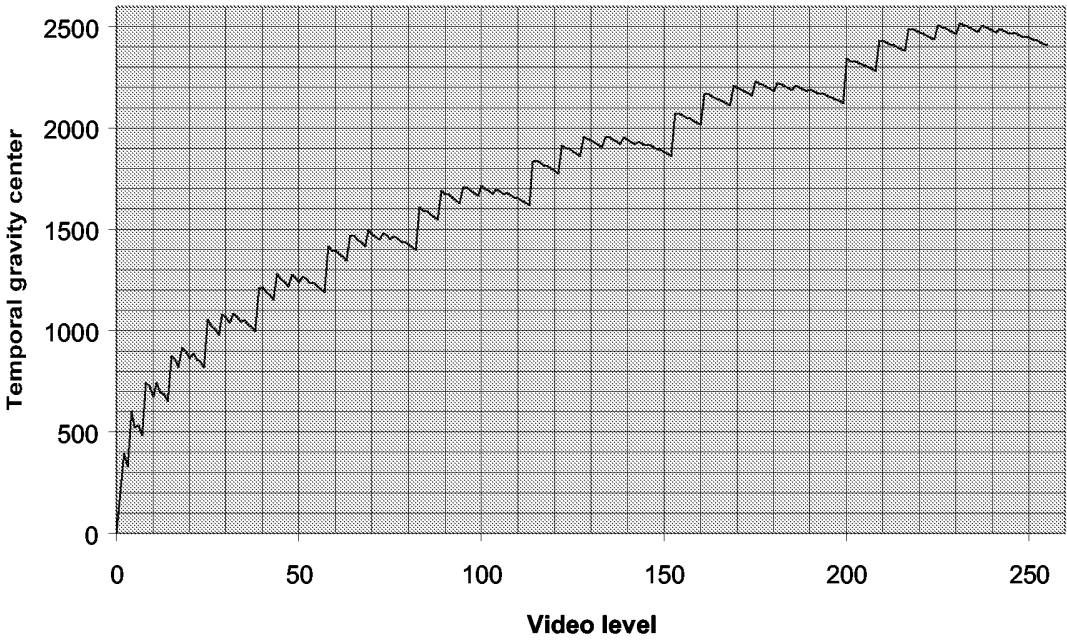


FIG.4

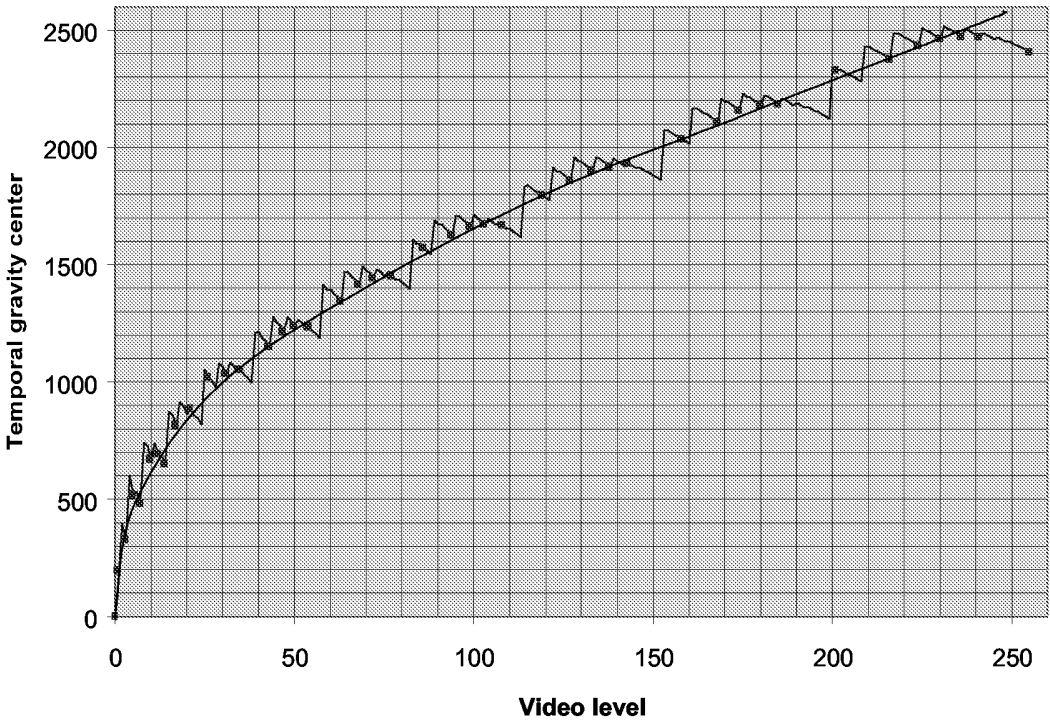


FIG.5

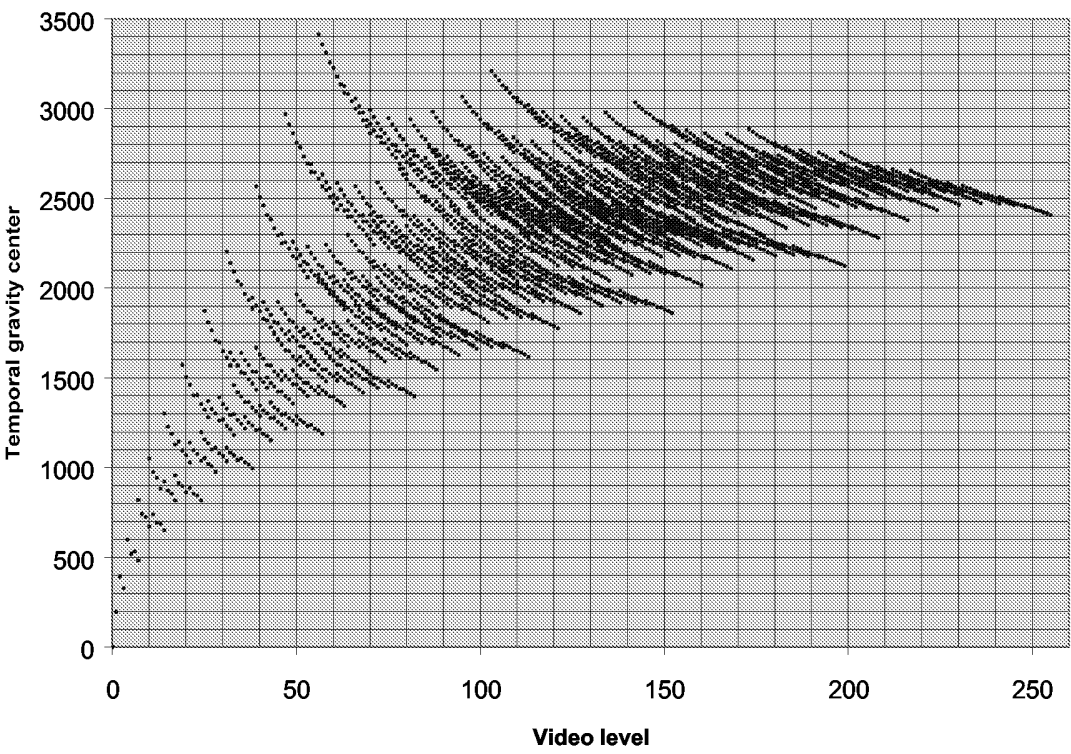


FIG.6

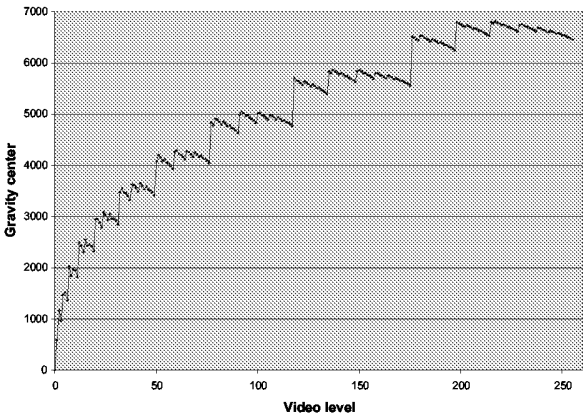


FIG.7A

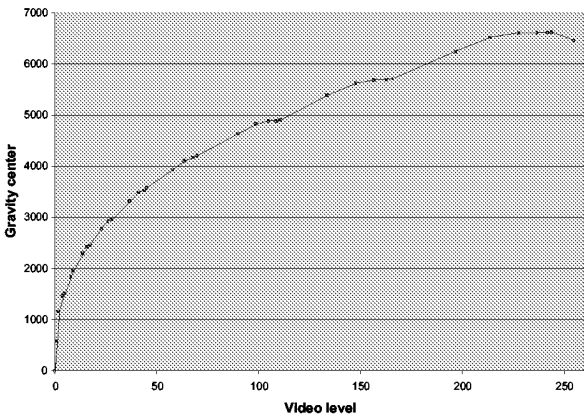


FIG.7B

4/14

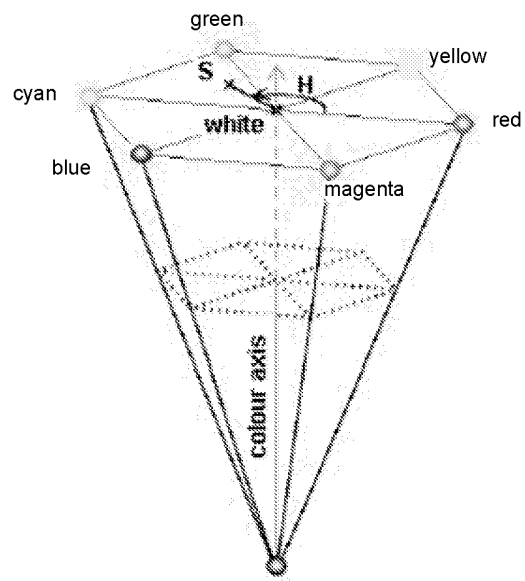


FIG.8

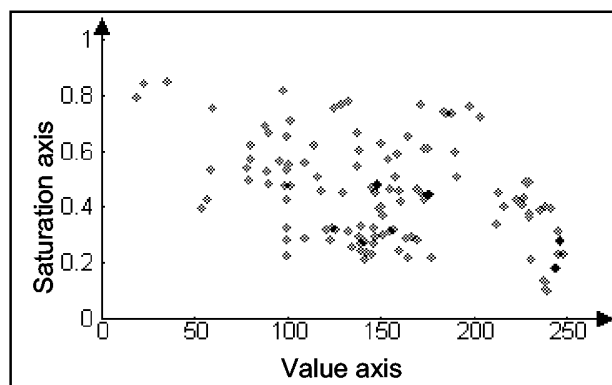


FIG.9

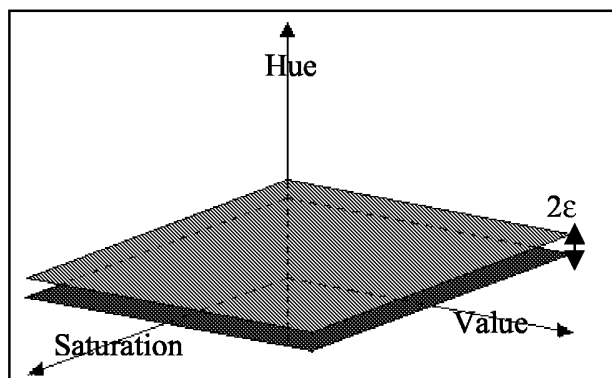


FIG.10

5/14

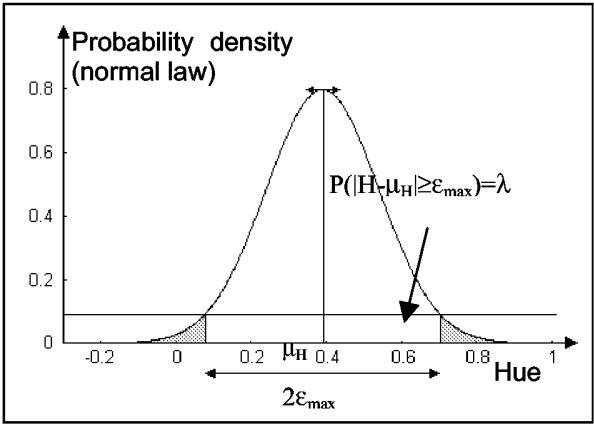


FIG.11

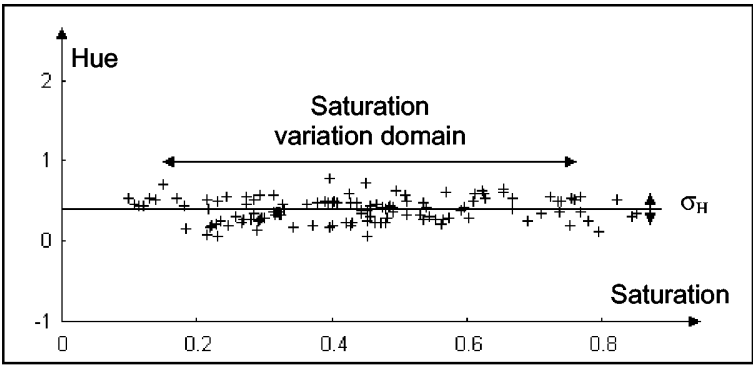


FIG.12

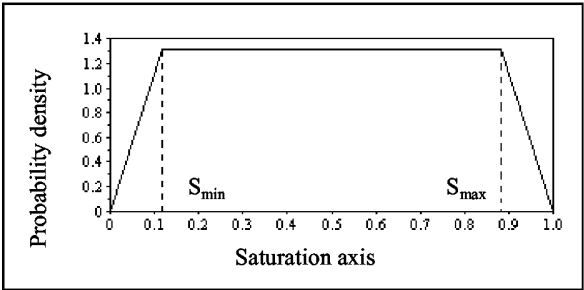


FIG.13

6/14

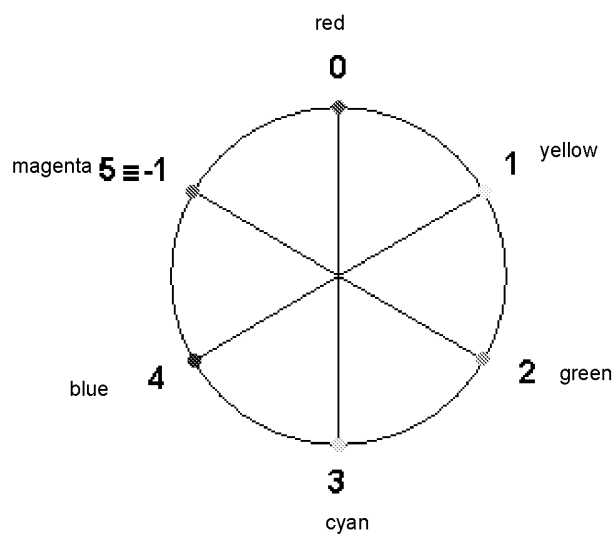


FIG.14

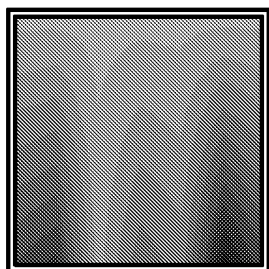


FIG.15a

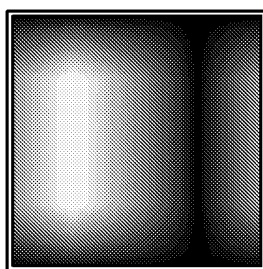


FIG.15b

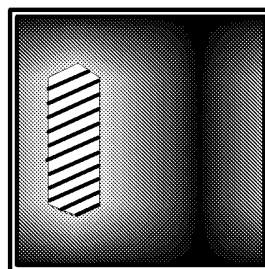


FIG.15c

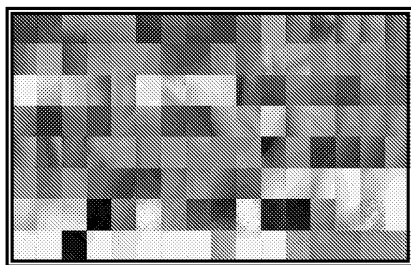


FIG.16a

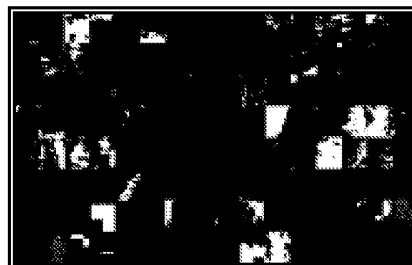


FIG.16b

7/14



FIG.17a

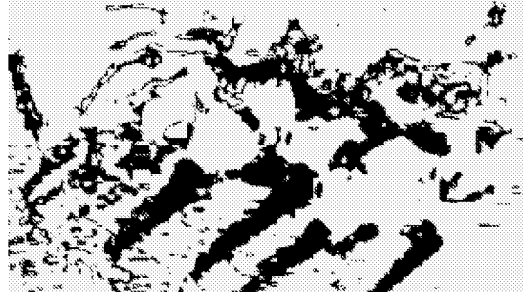


FIG.17b

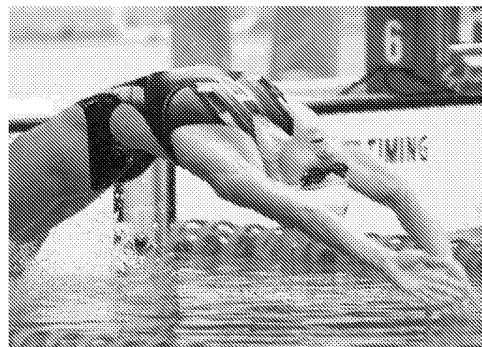


FIG.18

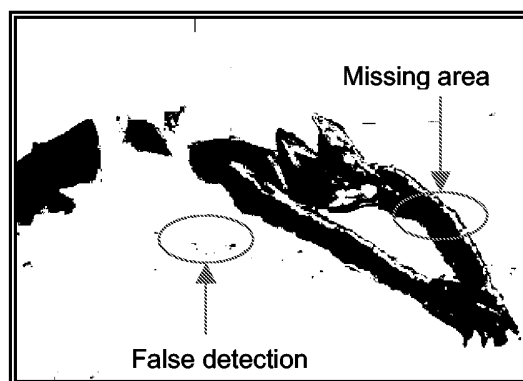


FIG.19

8/14

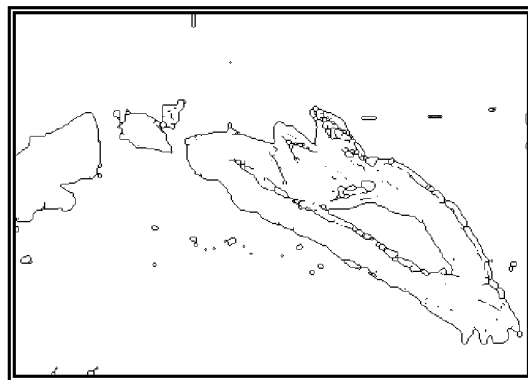


FIG. 20

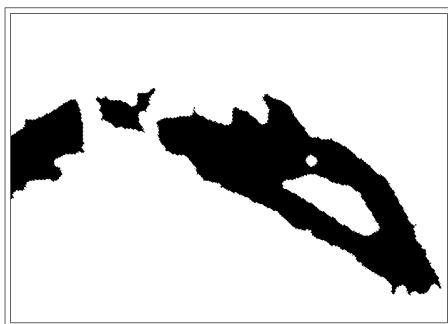


FIG. 21a

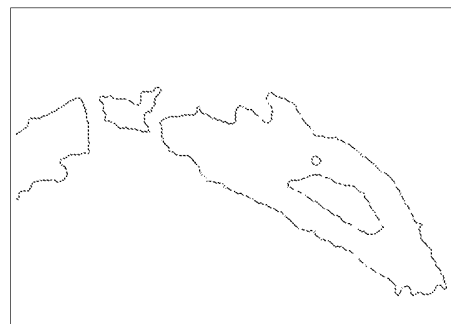


FIG. 21b



FIG. 22a



FIG. 22b



FIG.23

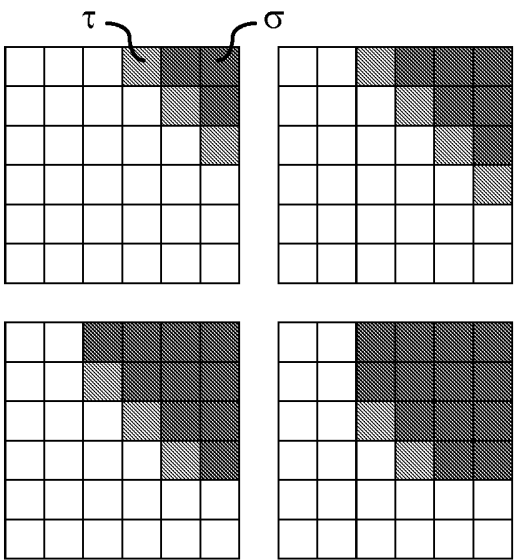
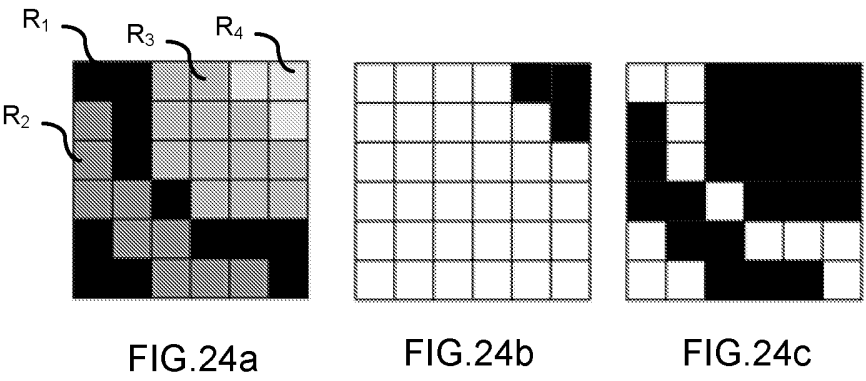


FIG.25

10/14

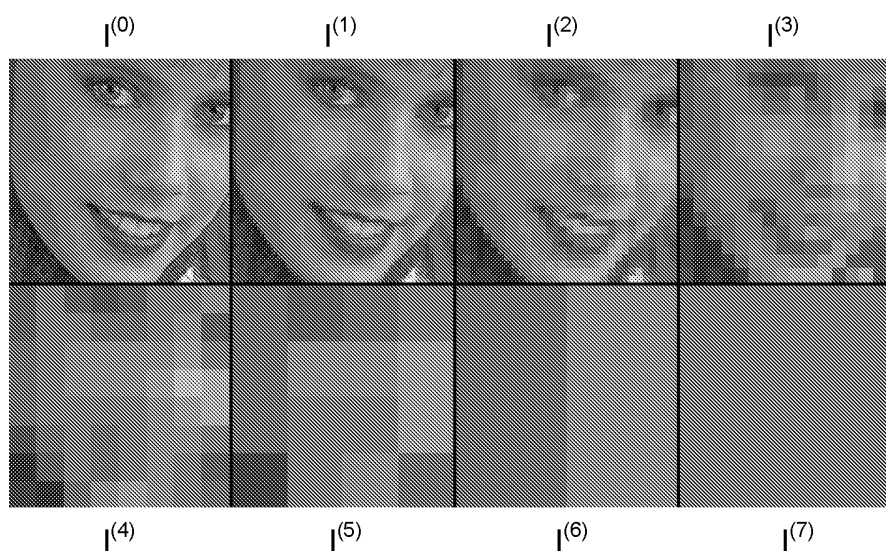


FIG. 26

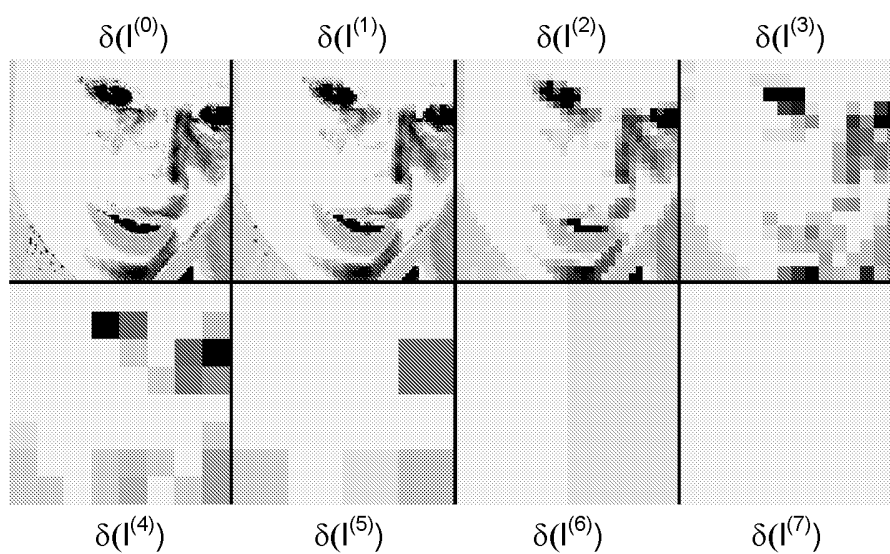


FIG. 27

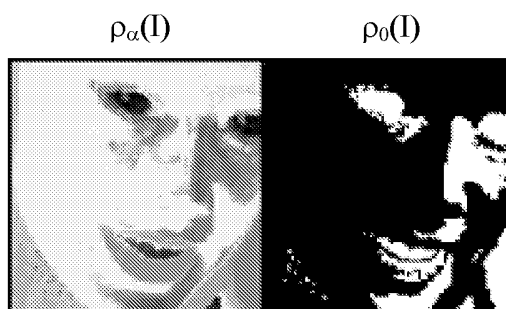


FIG. 28a

FIG. 28b

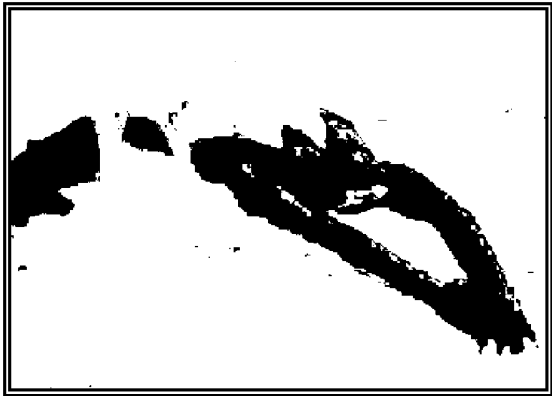


FIG.29

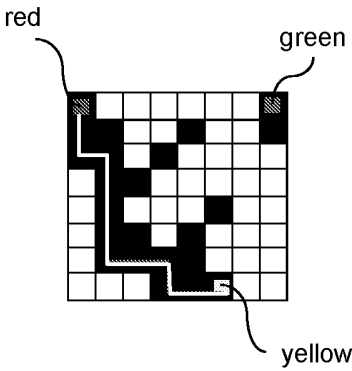


FIG.30a

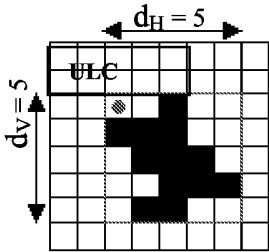


FIG.30b

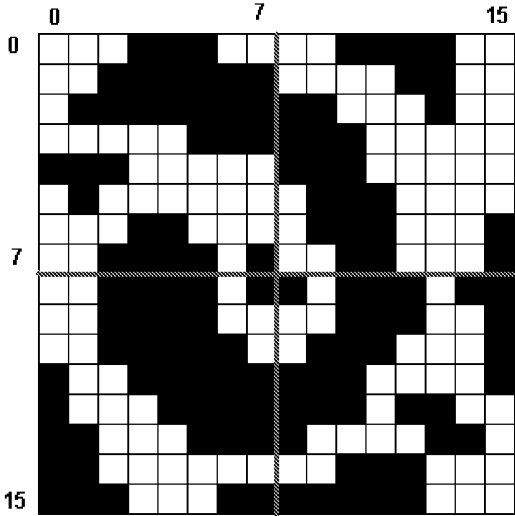


FIG.31a

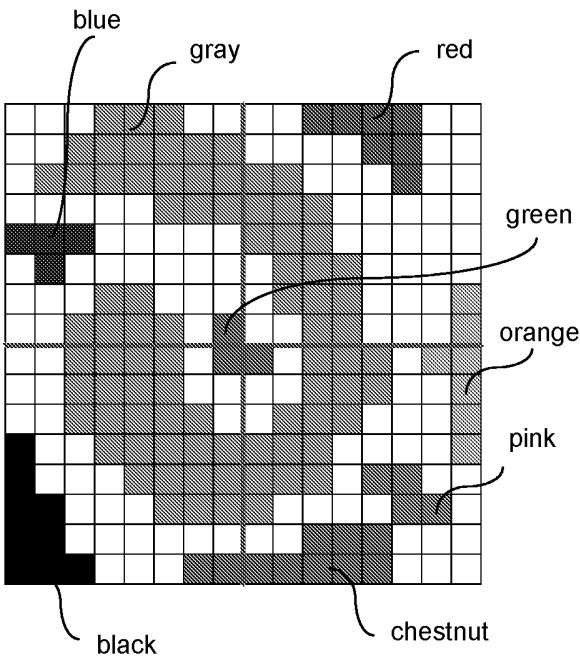


FIG.31b

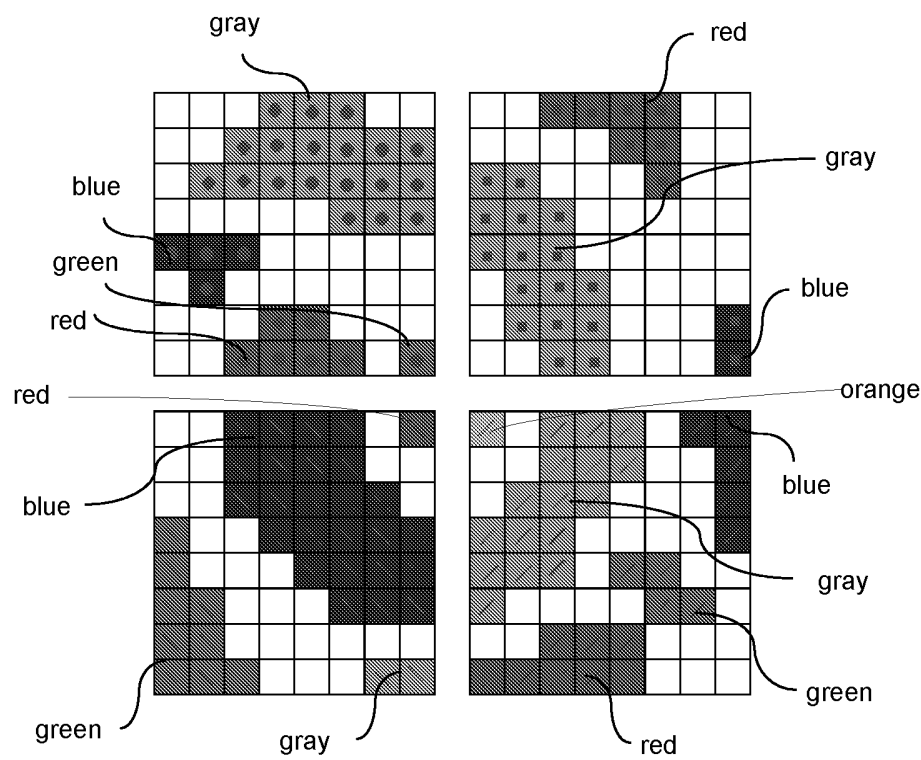


FIG.32

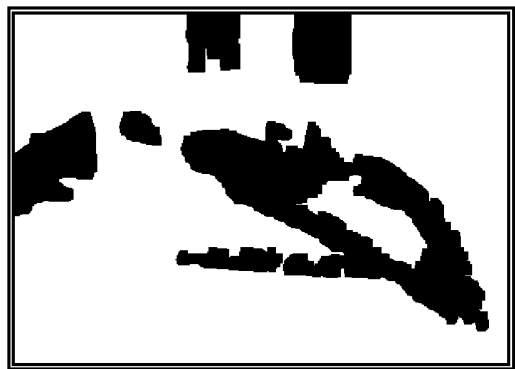


FIG.33

13/14

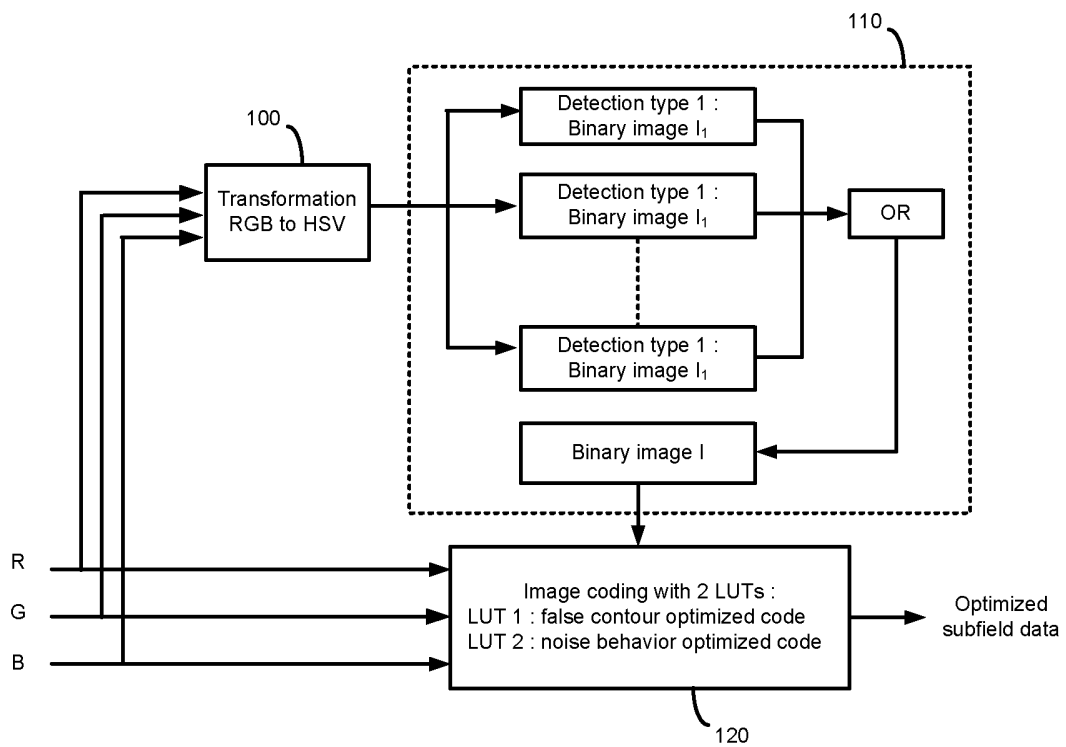


FIG.34

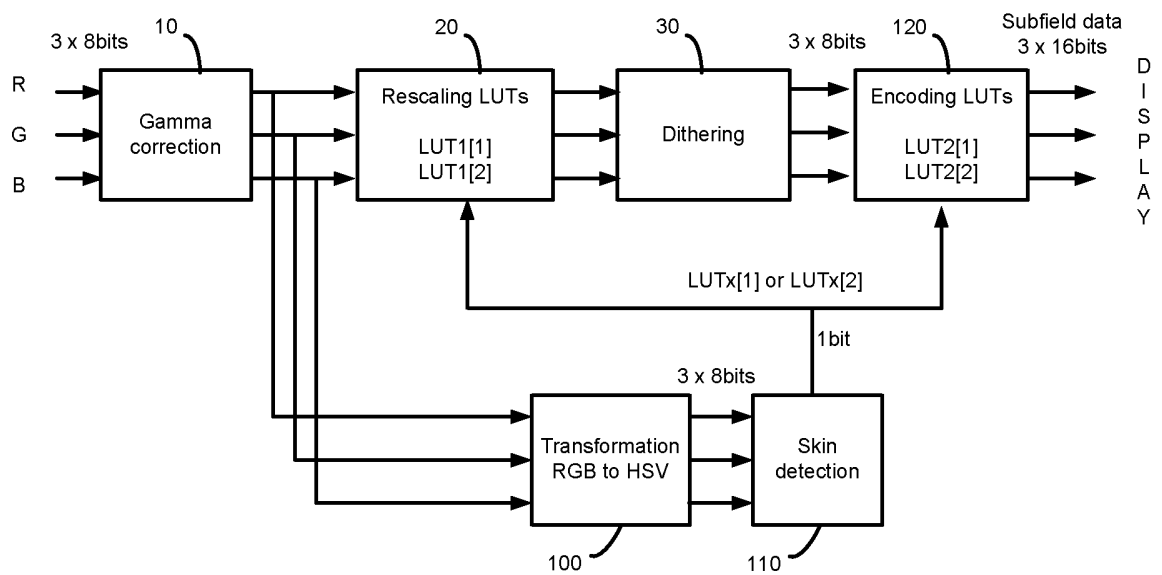


FIG.35

14/14

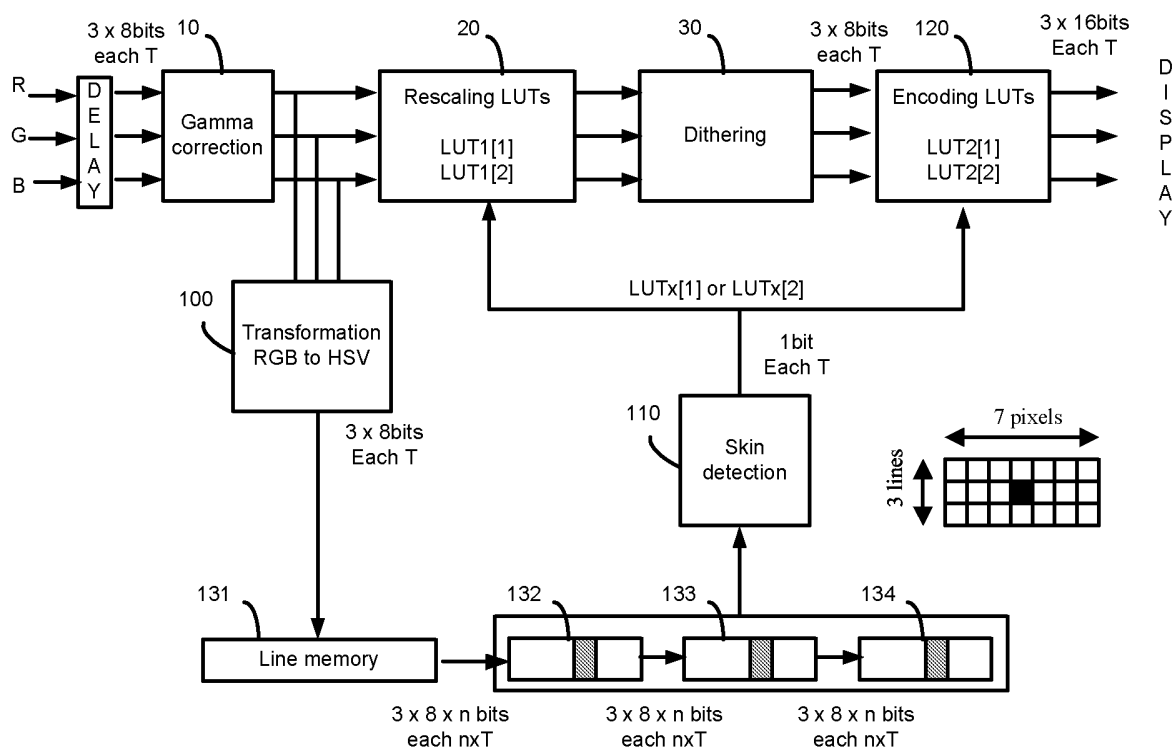


FIG.36

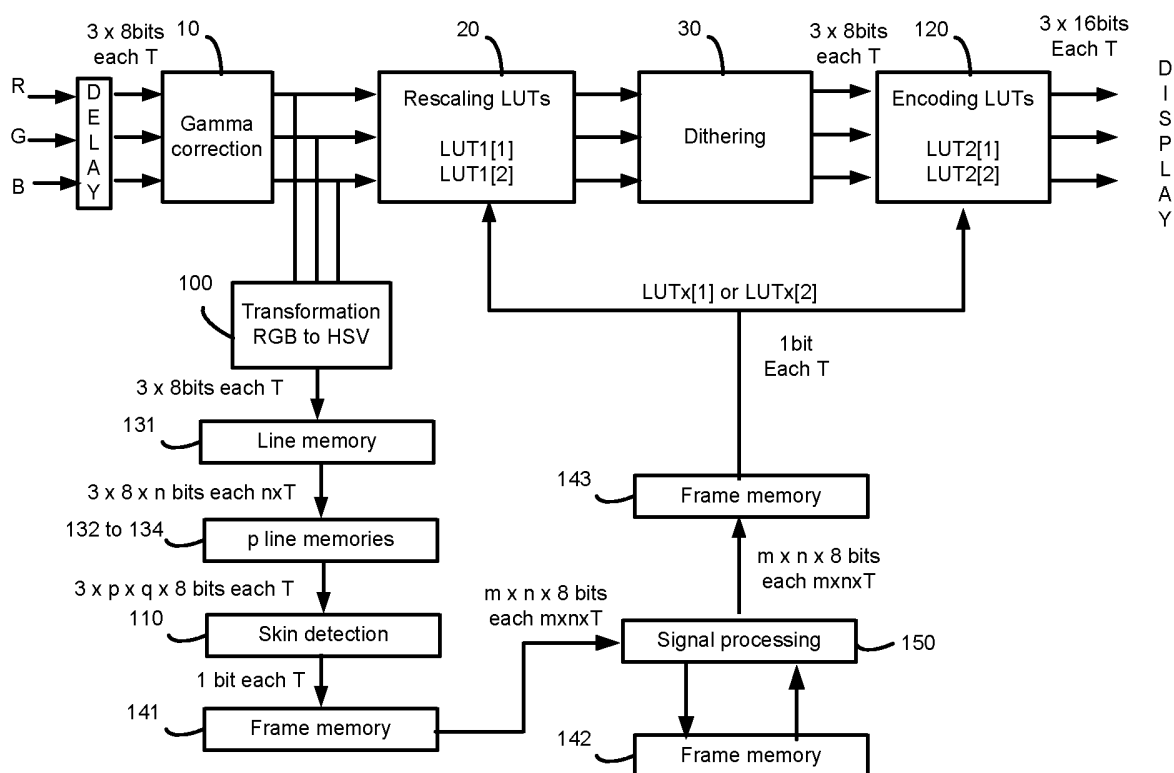


FIG.37