Colour spaces

This appendix describes the colour spaces that are most commonly used for brightness normalisation in order to detect skin colour. A large set of skin and background samples is used to show their spread in the different colour spaces.

A.1 The RGB colour space

Extensive experiments in the human visual system have showed that the cones — sensors in the eye responsible for colour vision — can be divided into three principal sensing categories, corresponding roughly to red (R), green (G) and blue (B) [WS00]. Therefore, colours are seen as combinations of these so-called primary colours [GW00]. For this reason, most of the cameras and emissive colour displays represent pixels as a triple of intensities of the primary colours in the RGB colour space: \([R, G, B] \in \mathbb{R}^3\). This is also the reason why the RGB space is very commonly used by the computer graphics and image processing community.

A disadvantage of the RGB representation is that the channels are very correlated, as all of them include a representation of brightness. This is illustrated in Figure A.1 and A.2 in which the brightness information can be recognised from R, G and B channels shown separately.

True colour 24 bits RGB images have the triple \([R, G, B]\) represented by 256 discrete values (ranging from 0 to 255) [Jac01], thus the range of RGB colour values forms the cube of \((2^8)^3\) possible values as shown in Figure A.3. The high correlation between lightness and RGB channels can be noted by the line of the grey values, where \(R = G = B\). In fact, if the corresponding elements in two points, \([R_1, G_1, B_1]\)
A.2 The CIE chromatic space

The CIE chromatic space is a standard proposed in 1931 by the Commission Internationale de l’Eclairage – the International Commission on Illumination. Some modifications have been proposed later, but this
section is restricted to the 1931 standard. It has been used in several colour processing tasks [GW00] and it is used to define the colour gamut, i.e., the range of possible colour values that a device can represent.

This two dimensional space has the $x$ and $y$ axes respectively defined by the pure chromatic colours red and green $(r, g)$, defined by this normalisation process:

$$
    r = \frac{R}{R+G+B} \\
    g = \frac{G}{R+G+B}
$$

which is, in fact, a $\mathbb{R}^3 \rightarrow \mathbb{R}^2$ map. Pure blue $(b)$ is redundant after the normalisation because $r+g+b = 1$ [WS00].

The use of this colour space for skin detection has became popular specially after the work on face tracking developed at SCS, Carnegie Mellon University [YW96, YLW98b].

### A.3 The perceptual colour space

The perceptual colour spaces were designed by Smith in [Smi78] in order to provide a more “intuitive” way of describing colours and lightness. Three quantities are used to define them: hue, saturation and brightness. Brightness embodies the achromatic notion of intensity. Hue is an attribute associated with the dominant wavelength in a mixture of light waves. It represents colour as perceived by an observer. Thus, when we call an object blue, yellow or red, we are specifying its hue. Saturation refers to the relative purity or the amount of white light (or grey of equal intensity) mixed with a hue. Primary colours (pure red, green and blue) are fully saturated, whereas colours such as pink (red and white) and lavender (violet and white) are less saturated. The degree of saturation is inversely proportional to the amount of white light added [GW00].

Basically, there are two distinct perceptual colour spaces: HSL (hue, saturation, lightness); and
A.3 The perceptual colour space

HSV (hue, saturation, value). Both are defined with polar coordinate systems. HSV is represented by a hexcone where Hue is the angle around the vertical axis, S is the distance from the central axis and V is the distance along the vertical axis. Primary and secondary pure colours are fully saturated (S = 1). As illustrated in Figure A.4, starting from $H = 0^\circ$ (which represents pure red), a secondary or primary colour is located at each $60^\circ$ of hue. Complementary colours are $180^\circ$ opposite one another measured by H. Colours along the vertical axis have zero saturation, i.e., grey scale values. Note that when $S = 0$, the value of H is irrelevant [Jac01], [Smi78].

HSL colour space is a double hexcone and can be thought of as a deformation of the HSV space. The distinction between HSV and HSL lays in the representation of brightness information, which determines the distribution and dynamic range of both the brightness (L or V) and saturation (S). In practice, the HSL colour space is best for grey level image processing and also for representing objects in such a way that colour images can be distinguished even in monochrome images (e.g. to show colour cartoons on black-and-white TV receivers), whereas the HSV image space is a better representation for colour processing [Jac01].
A.4 The YUV and YCbCr colour spaces

As described in [RMG98], [AP96], and [ZYW00], on performing skin detection, the brightness channel is discarded and the HS space is used instead. Therefore, there is no significant difference between HSV and HSL in this application [Bow99].

![Images of HSV channels](image)

Figure A.5: HSV channels from Figure A.1(a) shown separately.

Figure A.5 shows the H, S and V channels obtained from image from Figure A.1(a).

A.4 The YUV and YCbCr colour spaces

The YUV image space was created in order to make colour television broadcasts backwards-compatible with black and white TV receivers. The colour signal also needed to conserve bandwidth because three channels of RGB data would not fit into the limited broadcast signal bandwidth. The Y channel describes Luma, the range of value between dark and light, which is the signal shown in black and white televisions. The U and V chrominance channels subtract the Luminance values from Red (U) and Blue (V) to represent the colour only information (without brightness) [Mal02]. Thence, the basic conversion equation from RGB to YUV is:

\[
\begin{align*}
Y &= 0.3R + 0.6G + 0.1B \\
U &= B - Y \\
V &= R - Y
\end{align*}
\] (A.3)

The coefficients used to obtain luma are the same as those used for the NTSC standard conversion from RGB to grey level images [Poy96]. These coefficients are based on psychovisual experiments that estimated the proportion of red, green and blue that we perceive. It is shown that approximately 65% of all the cones in the human eye are sensitive to green light, 33% are sensitive to red light and only about 2% are sensitive to blue, but the blue cones are the most sensitive [WS00].
The YCbCr colour space was developed as part of ITU-R BT.301 during the development of a world-wide digital component video standard. This colour space was extensively used in the development of the JPEG standard, and was used for skin colour detection by several research projects, including the Pfinder [WADP97].

As shown in equation A.4, YCbCr is a scaled and zero-shifted version of the YUV, so that the chrominance values are always positive [PM93]:

\[
Cb = \frac{U}{2} + 0.5, \quad Cr = \frac{V}{1.6} + 0.5,
\]

for \( U \) ranging between \([-0.9, 0.9]\) and \( V \) ranging between \([-0.7, 0.7]\), which are the ranges obtained from the conversion from RGB \( \in [0, 1] \). So the range of \(Cb\) and \(Cr\) are \((0.05, 0.95)\) \((0.06, 0.94)\), respectively. For digital 8-bits values of \(U\) and \(V\), a 128 shift is employed, rather than 0.5.

Figure A.6 shows the RGB colour cube in the YCbCr colour space. It shows that not all the possible values in the triple \([Y, Cb, Cr]\) represent possible RGB colours. Therefore, special care must be taken to about overflow or underflow in RGB, when converted from YCbCr. Brightness normalisation is done by discarding the Y channel.

Figure A.6: The RGB colour cube in the YCbCr colour space, using 8 bit representation of values.
A.5 Visualising the colour spaces

To illustrate the effect of brightness normalisation using the above colour spaces, each method has been applied to the image of Figure A.1(a). An intermediate grey level (127) was chosen and the resulting images are shown in Figure A.7. Note that skin areas appear uniform and that shading information is lost for all the three methods.

(a) (b) (c)

CIE Pure Colours HSV CbCr

Figure A.7: Resulting images after brightness normalisation of the image in Figure A.1(a) using the CIE colour space (a), the HSV without the V channel (b), and the YCbCr without the Y channel (c).

In order to illustrate the distribution of skin samples in the colour spaces, a database with images from 141 different people was used. This database is composed by hand images grabbed from seven volunteers, and the AR face detection database of the University of Purdue [MB98]. The Purdue database contains 134 people (men and women) from several ethnic groups. Background samples were obtained from the background regions in the images (e.g. people’s clothing and other objects) and other images grabbed in the laboratory, as shown by some samples in Figure A.8. The camera used in the acquisition had the automatic colour and brightness balance.

Skin and background regions of the images in this database were manually selected in order to obtain the data set. After the training process, more than 0.5 million samples of skin and more than 1.2 million samples of background were obtained. Figure A.9 shows the skin and background samples in the RGB colour space.

Figure A.10(a) shows the plot of only skin samples in the RGB colour space. Note that the samples are more spread in the direction of the brightness variation. The directions of global variation of the sample data can be evaluated by performing Principal Component Analysis in this space [DCJ01, Mar02].
The eigenvector of the covariance matrix of the samples which is associated to the largest eigenvalue is oriented according to the largest variation of the data set. The second eigenvectors points to the direction that is perpendicular to the first, and has the second largest variation of the data, and so on. The eigenvectors of the skin colour database are shown in Figure A.10(b). The angle between the first eigenvector and the vector that points to the direction of the brightness variation is only 3.35 degrees. This confirms that it is necessary to use a normalised colour space or remove brightness information in order to get a more compact cluster of skin samples.

To illustrate the compression of the skin colour cluster in normalised colour spaces, Figures A.11, A.12, A.13 show the skin and background samples in the CIE, HSV and YUV chromatic spaces, respectively.

In comparison to the plots of the skin and background samples in the full colour spaces, the plots in normalised planes illustrate that such projections lead to lower dimensional spaces with more compact skin colour samples, improving the separability between them and background samples.
A.5 Visualising the colour spaces

Figure A.9: Colour samples in the RGB space: skin (red circles) and background (blue crosses) samples plot together.

Figure A.10: Variation of skin samples: (a) Skin colour samples in the RGB space; (b) Eigenvectors of the skin samples in the RGB space in their respective mean position. The first, the second and the third eigenvectors are indicated by a star, a square and a circle in its end, respectively. The dashed line indicates the grey level (brightness) direction.
A.5 Visualising the colour spaces

Figure A.11: Skin (red circles) and background (blue crosses) in the CIE chromatic space.

Figure A.12: Skin (red circles) and background (blue crosses) shown in the HSV colour space (a); and their projection into the HS plane (b).
Figure A.13: Skin (red circles) and background (blue crosses) shown in the YCbCr colour space (a); and their projection into the CbCr plane (b).
This appendix complements information of Chapter 6 about the adjoint transformation used by Drummond and Cipolla.

Consider two frames $a$ and $b$ where points are related by the homogeneous transformation

$$
X^b = T^b_a X^a = \left( \begin{array}{cc} R^b_a & t_{ab} \\ 0^T & 1 \end{array} \right) X^a.
$$

To derive the effect of changing frame on the screw vector, $\alpha$, consider writing the scene velocity in frame $b$ in two different ways

$$
\begin{pmatrix} \dot{X}^b \\ 0 \end{pmatrix} = \sum_i \alpha^b_i G_i T^b_a \begin{pmatrix} X^a \\ 1 \end{pmatrix} = T^b_a \sum_i \alpha^a_i G_i \begin{pmatrix} X^a \\ 1 \end{pmatrix}
$$

indicating that

$$
\sum_i \alpha^b_i G_i = T^b_a \sum_i \alpha^a_i G_i T^a_b.
$$

Recalling that $\alpha = (\omega^T v^T)^T$, and the earlier expressions for $G_i$, Eq. (B.3) is just

$$
\begin{pmatrix} [\omega]_x \\ v^b \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} R^b_a \\ t_{ab} \\ 0^T \\ 1 \end{pmatrix} \begin{pmatrix} [\omega]_x \\ v^a \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} R^a_b \\ -R^a_b t_{ab} \\ 0^T \\ 1 \end{pmatrix},
$$

where $[\omega]_x$ is the antisymmetric matrix such that $[\omega]_x r = \omega \times r$. Hence

$$
[\omega^b]_x = R^b_a [\omega^a]_x R^a_b
$$

which is equivalent to

$$
\omega^b = R^b_a \omega^a.
$$
Also
\[
v^b = -R^b_a[\omega^a] \times R^b_a t_{ab} + R^b_a v^a = [t_{ab}] \times R^b_a \omega^a + R^b_a v^a.
\] (B.7)

The relationship between the screws is defined as
\[
\alpha^b = \text{Ad}(T^b_a) \alpha^a
\] (B.8)

and hence using Eqs. (B.5, B.7) the adjoint is given by
\[
\text{Ad}(T^b_a) = \begin{pmatrix}
R^b_a & 0 \\
[t_{ab}] \times R^b_a & R^b_a
\end{pmatrix}
\] (B.9)

Eq. (B.9) agrees with Drummond and Cipolla, given that they recover \(\alpha = (v^\top \omega^\top)^\top\). Though a minus sign appears missing from the definition of their antisymmetric matrix \([t]_{\wedge}\).

As the measurement vector \(d\) is an invariant, \(F^b \alpha^b = F^a \alpha^a\) and so
\[
F^b = F^a \text{Ad}(T^b_a)^{-1}
\] (B.10)

which gives
\[
C^b = F^b \top F^b = \text{Ad}(T^b_a)^{-\top} C^a \text{Ad}(T^b_a)^{-1}
\] (B.11)

This differs from the equivalent in Drummond and Cipolla, a difference which may arise because equation (29) in ref [DC02] states \(T^b_a G_i(T^b_a)^{-1} = \sum_j \text{Ad}(T^b_a)_{ij} G_j\), in contradiction with the later (agreed) statement in equation (32) in ref [DC02] that \(\alpha^b = \text{Ad}(T^b_a)\alpha^a\).