Color-spaces and color segmentation for real-time object recognition in robotic applications

Pedro M. R. Caleiro, António J. R. Neves and Armando J. Pinho

***Resumo* – O reconhecimento de objectos é uma parte importante numa grande gama de aplicações robóticas. O presente artigo[1] visa explorar as vantagens e desvantagens no uso de diferentes espaços de cor (diferentes representações digitais de cores) e diferentes métodos de identificar cores numa imagem com vista à detecção de objectos em aplicações robóticas. O assunto da supressão de sombras através da manipulação da informação de cor também é estudado.**

***Abstract* - Object recognition plays an important role in a vast diversity of robotic applications. This paper[1] addresses the vantages and disadvantages of using some known color spaces (digital color representations) and segmentation methods for the purpose of accurate detection of color-coded objects, which are objects that are represented by a given color. The robots are previously programmed to associate each color or combination of colors to the type of object to be detected. The subject of shadow suppression based solely on color manipulation will also be addressed.**

I. LUMINANCE & CHROMA

Usually, images are acquired from a robot, using a camera and processed in real-time. Such frames are sent in a commonly used color representation like RGB or YUV. The RGB color-space is an attempt to accurately represent the additive properties of the Red, Green and Blue components of light. A proper quantization of each of

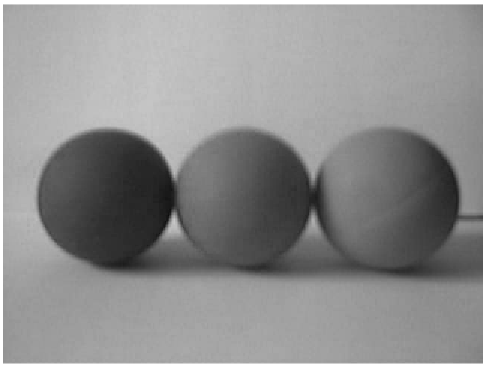

Fig 1 Captured frame in RGB representation. Three balls are displayed. Blue, red and green (left to right).

these three components is able to represent any color in

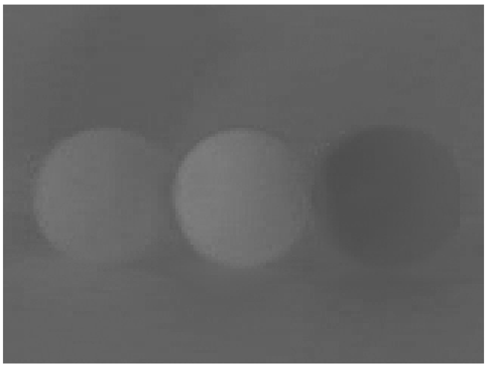

Fig 2 Same frame in YUV representation with an altered, constant Y component

the visible light spectrum. While this is a natural representation of color, it is not well suited for applications in which a distinction between color and how bright that color is are needed, like color quantization and object recognition. To overcome this limitation and take advantage of the chromatic perception of the human eye to the purpose of data compression, the YUV color representation was devised. The color of an object and how bright it is are three distinct components. Y is the luminance of a color, and UV are its two chromatic components. But the Y component is quantized according

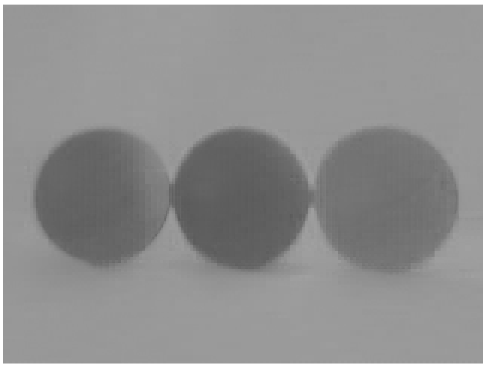

Fig 3 Captured frame in HSV representation, with constant luminance (value component)

to the color perception in the sensor cells in the retina of the human eye. If we look closely at the formula for its Y component (converted from the RGB space):

$$Y = 0.299 * R + 0.587 * G + 0.114 * B$$

we can clearly see that this component still contains chromatic information, with undesirable effects in accurate chromatic representation. Images I) and II) represent this assumption. Image I) shows the original frame in RGB.

[1] This article is available in full (and in color) at http://clientes.netvisao.pt/~acondeca/

Image II) is the same frame represented in YUV, where the Y component in each pixel was set to a fixed constant value, so that only the differences in the UV chroma components are seen. There's a clear blurring in the color

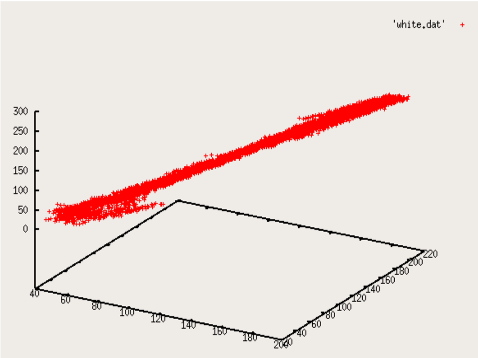

Fig.4 Linear distribution of various shades cast upon a white object, represented in the RGB space

of the balls which blend with the achromatic background. This fact makes objects difficult to identify using YUV representations. Other color representations with the same problem are YMP (a cylindrical distortion of YUV, where MP are the chromatic components in polar representation), YIQ and IHLS (discussed further in this article).

II. SATURATION & HUE

In an attempt to completely separate the luminance from the color components new color spaces were devised. The three I'll discuss in this paper are HLS, HSV and IHLS. In these three spaces, their luminance components correspond to a color in the achromatic RGB axis (R = G = B). This could be an advantage in color detection. As a demonstration, a few RGB samples were taken from a white object with shadows cast upon it. The result is the graphic observed in image IV), where we can see a clearly

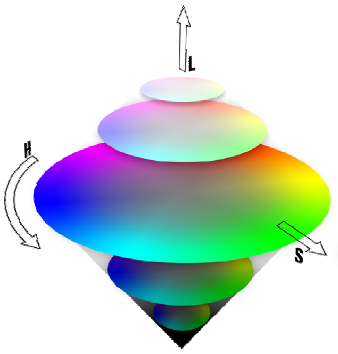

Fig 5. Bi-conic representation of the HLS color space.

linear variation from pure white to pure black, along the achromatic axis, which means that an achromatic object remains achromatic (considering optimal lighting conditions and white balance correction) no matter how bright it's being illuminated. The noise observed in the graphic is probably due to sensor noise in the camera and

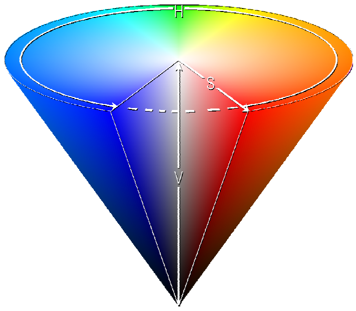

Fig. 6 Conic representation of the HSV color space.

a second uncontrolled light source. In image III) we can see the HSV representation of image I) with its V (value, which corresponds to luminance in HSV) modified to be constant along the frame. The effect observed in the YUV space, the blending of colors with the white background, doesn't occur. So, what are the differences between the

Fig. 7 Scene in RGB

HSV, HLS and IHLS which make them apart? If we look to images V) and VI) we wil l see a representation of the HLS and HSV color spaces, respectively. Due to their nature, there are some combinations of component representations which don't fall inside the spaces themselves. To overcome this

Fig. 8 Same scene as image VII) in IHLS

problem, they're usually projected into a cylinder. But this poses another problem, at their projected luminance extremes (low luminance in HSV, and both low and high in HLS), hue information becomes irrelevant, and worse, such colors could become highly saturated (a bluish white (low saturated blue) could become a deep blue (highly saturated blue) when converting from RGB to HLS and back, for example). Referring to picture IX) we can see that when the value (luminance) component of HSV is again altered to remain constant in the whole frame, dark areas in the original image VII), have random hue (due to noise) and become highly saturated. The same effect occurs in HLS, image X), but this time it also affects high

Fig 9 Same scene as image VII) in HSV

luminance areas. IHLS which was an improvement over HLS proposed by Allan Hanbury and Jean Serra [1], has the same luminance component problem as YUV, as stated before, and when the luminance of the colors is changed to a constant value, same values fall outside of its representation range (not all combinations of component values in IHLS have a meaning when converted to another color space). When compared, image 8), represented in HLS with a constant luminance component, has a clear

Fig. 10 Same scene as image VII) in HLS

hue and saturation distortion (noise) when compared to the original image VII) in RGB. This effect is less prominent in image IX, represented in HSV, which appears to have this distortion only in low luminance (corresponding to value in HSV) areas. In IHLS VIII), chroma and luminance are not distinctly separated as expected. But HSV has a drawback, since it considers white as an independent color, and not an effect of luminance, different lightning conditions and incorrect camera white balance could severely distort chroma (HS) information as observed in the cloth above the table which has become violet. That could be solved with a proper adaptive white balance technique, but then again, that would distort hue even more at low light conditions. Alexandre R.J. François and Gérard G. Medioni suggest [2] simply discarding hue information below a certain luminance threshold using an algorithm which adjusts the threshold from frame to frame to determine this value.

III. OBJECT REFLECTIVITY AND PIXEL COLOR DISTORTION

The color of an object is an inherent property, consisting in the reflection of light coming from one or multiple light sources to the receiver sensors (in a camera or in the human eyes for example). Two objects with the same color may reflect each of the different primary colors (Red, green and blue) in different proportions despite seeming to possess the same color to the naked eye. Again referring to image III) you may see the colors of the balls reflected on the white ground and red light emitted from the red ball to the blue one and retransmitted to the camera (which

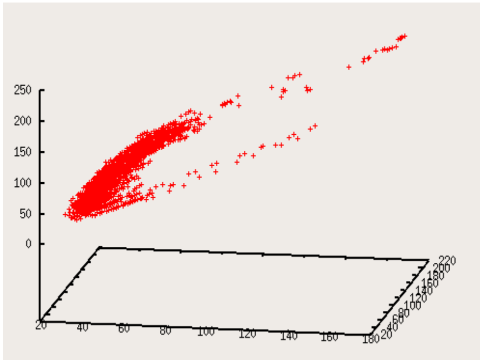

Fig. 11 Pixel information of the blue ball represented in RGB.

indicates that the blue ball is able to reflect red light, thus the color of the ball is independent of its ability to reflect other colors), which perceives it as magenta, which is composed of both blue and red. Under optimal conditions (only one pure white light source and a pure blue object, ignoring white balance) the graphic represented in XI) should be a straight line along the blue component plane, and should linearly and abruptly converge to the achromatic axis. So doesn't happen in the graphic (it should converge to R = G = B = 0 and R = G = B = 255) because of specular ambient light existing in the environment on which the data was taken from. The two points at which the color converges to the achromatic axis (or ambient light conditions, second light source or white balance under an uncontrolled environment) are variable from object to object (tested with different objects possessing the same color). The ability to identify the conversion point at high luminance levels is also variable from object to object. It can't be identified in the graphic due to the low reflection properties of the ball itself. The low luminance point at which the chroma information becomes irrelevant is easily observable. Considering the same information in HSV representation, the ball's hue and saturation would be fairly accurate with the exception of low light incidence zones (thus the two seemingly parallel lines in the graphic, an achromatic one and a blue hue one). The ability of an object to reflect a given color component, could also be related to the angle of light incidence and angle between the object and the sensor. To make an analogy, imagine looking at a lake. If we looked from above, we could probably see the lake's "true" color (if the water is not crystalline clear), but if we looked from a near horizontal angle at it, we could see trees or the sky reflected on its surface and, thus, not its color.

Color reflection in objects is hard to correct without resorting to shape and/or movement recognition, which is not in the scope of this article, and is very difficult to correct properly using only color recognition. An option would be to use stereo vision (two parallel cameras) to try to distinguish depth and correlate the information from both cameras in order to produce a reflection free image.

The pixels in a frame itself could be distorted by the compression applied to the data sent by the camera. This could result in blocky images (especially in the JPEG, Motion JPEG and MPEG compression standards), especially in low detail areas. If the requested resolution at which the camera was ordered to send the frame is lower than its sensor array, the camera could interpolate several pixels and transmit it as one, which would result in blurred edges in high contrast adjacent areas and misrepresented colors, which in turn could cause problems when trying to segment the objects. This could be a problem for object segmentation, but that shall be discussed ahead in more detail. There could be other causes (noise for example, damaged sensors, light refraction in the object, ...) interfering with the true perception of a pixel color by the sensor. This distortion effect is more accentuated in adjacent pixels. Since there could be something influencing the pixels' color which is not discernible in the scene captured by the camera, or too much information to process, a statistical model could be used to eliminate or at least reduce this effect.

Like color reflection, color aberration introduced by noise, malfunctioning sensors and color compression and specially interpolation by the camera is a problem that interferes with accurate color segmentation. Interpolation of color information by the camera leads to problems in segmentation since two objects (with distinct colors) could have some pixel color values shared . Information from previous frames and/or an appropriate filter could be used to reduce these color aberrations. Stereo vision could again be used to good effect.

IV. COMPONENT BASED SHADOW SUPPRESSION

This technique consists in ignoring information in one of the color components (luminance for example) and trying to segment the objects only using the remaining two components (chroma components for example). If the chroma of an object was constant regardless of shadows (low light) cast upon it, it could be possible to ignore luminance for example and discard shadow information (shadow suppression). As seen before, transmission of chroma information from object to object and loss of color information due to high reflectivity of chromatic distortion in either very high or very low luminance areas makes this feat unfeasible without proper image preprocessing, since the color of an object, and therefore the shadows cast upon it, are not constant and are in turn influenced by the environment and camera parameters, mainly white balance correction. It could still be possible to achieve, though, with proper image preprocessing, but this would probably imply some shape based object detection.

V. COLOR SEGMENTATION METHODS

Segmentation consists in detecting an object based on prior information about the same object. This paper focuses on segmenting objects by color information only, as there are other methods (shape, movement, etc...). Two simple methods used to store color are using a color list (a list of values corresponding to the colors we wish to detect) and defining a color range (a geometrical representation of the color information. Eg: "all colors between values A and B"). Image XII) depicts a scene where the red ball (the one partially hidden by the black segmentation mask) was selected using a color list. It's observable that the orange ball was also detected (black pixels) even though that was not intended to happen. We may conclude that either the red ball, the orange ball, or both, have their chroma information distorted due to the nature of the HSV space itself or the environment. This loss of chroma information usually happens when there isn't enough light illuminating the object. The same

Fig 12 Chroma based segmentation using a color list in a HSV frame

Fig. 13 Same segmentation method with automatic camera white balance

phenomenon occurs in HLS space, with the aggravation of having chroma information distorted in highly illuminated objects. Referring to the same image as an example, it becomes clear that the orange ball is reflecting less green light to the sensors in an area of its surface which is poorly lighted, which is in turn detected as red. Using a color range to select the balls could aggravate this problem even further, especially in a color space where chroma and luminance are not clearly separated. Proper white balance, though, reduces the distortion slightly as seen in image XIII), but one can't always be sure that the white balance correction, even if calibrated manually, is accurate during the entire life span of the robotic application (due to varying lighting conditions in the environment). Even if we were to attach a white reference point (a piece of white paper for example) to the camera's vision field, we could not be sure that the same lighting conditions would be accurate to the entire scene (due to different light sources and object reflections).

A method which could be used for segmentation would consist in defining a detection region, inside the color space, formed from a pixel color list. If two regions intersect, (two object colors to be segmented), the application inside the robot could assume the pixels inside the intersecting regions as either non valid data, and

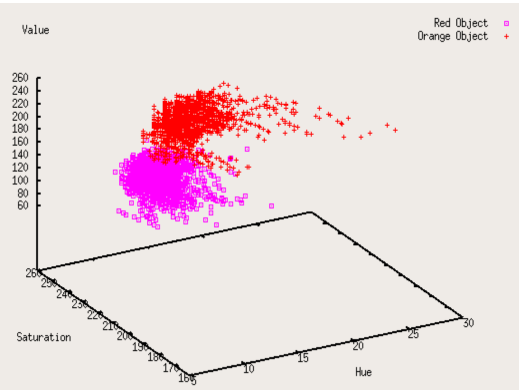

Fig 14 Graphic representing pixel color lists (in HSV space) takes from two distinct objects. Region intersection is observed.

simply ignore them, or it could assume both colors as being present in the same frame pixel. Assuming the pixels as having two (or more) colors instead of ignoring them has the advantage of not discarding data which could prove valuable in correctly detecting the object, especially when light is low or shadows are present in the object, since low light leads to intersecting regions (in HSV & HLS representations). Other approach in treating region intersections would consist in estimating the true region the pixel belongs to by studying the area surrounding it, but that method would fall into the realm of shape recognition. Such segmentation region intersections easily occur in low value component colors in HSV representation (image XIV tries to demonstrate this intersection effect) and low and high luminance colors in HLS representation. These regions or pixel lists could also be corrected from frame to frame to compensate for incorrect white balance or varying illumination conditions.

If such segmentation method is not adequate or desirable for the task at hand, a statistical model based on the entire frame's pixel values could be devised to crop hue and saturation information in HSV or HLS spaces below or above a certain threshold and simply ignore chroma information beyond these thresholds. That would prevent incorrect detection of chroma.

VI. CONCLUSIONS

With the lack of two cameras for stereo vision it's very difficult to determine depth from a single frame, so either a way to correct pixel color distortion caused through reflection, compression and noise should be devised based solely on the information contained in a single frame or another camera should be used. Such preprocessing methods would improve upon the ability to later segment colors accurately.

The HSV color space seems to be the best color space to use in robotic applications since it clearly separates light and chromatic information and it is the one which least distorts the latter.

Using color regions instead of pixel color lists is preferable for segmentation purposes, because regions compensate slightly for lighting condition changes while the former do not. White balance clearly influences accurate color detection, specially in low light conditions where segmentation regions usually have a tendency to intersect in the HSV space.

Shadow suppression based only on color information is very difficult to achieve. The ability to execute it depends heavily on lighting conditions and white balance parameters.

It becomes clear that moving into shape recognition is probably the next best step if better object detection and shadow suppression techniques are required.

REMARKS

These results were obtained using a Logitech Webcam Pro 4000 under uncontrolled lighting conditions, and may or not be accurate under a different setting. All settings in the camera were set to neutral, and white balance was disabled (except where stated otherwise) so that it wouldn't influence results. Assume color space component quantization ranging from 0 to 255.

ACKNOWLEDGEMENT

I, Pedro Caleiro, would like to thank professor Armando Pinho and Eng. António Neves for their guidance and support in the writing of this article.

REFERENCES

[1] Allan Hanbury and Jean Serra - “A 3D-polar Coordinate Colour Representation Suitable for Image Analysis”.

[2] Alexandre R.J. François and Gérard G. Medioni - “Adaptive Color Background Modeling for Real-Time Segmentation of Video Streams”.

[3] Philipp Blauensteiner, Horst Wildenauer, Allan Hanbury and Martin Kampel - “On Colour Spaces for Change Detection and Shadow Suppression”.

[4] Allan Hanbury - “The Taming of the Hue, Saturation and Brightness Colour Space”.

[5] Allan Hanbury and Beatriz Marcotegui - “Waterfall Segmentation of Complex Scenes”.

[6] Allan Hanbury and Jean Serra - “Color Image Analysis in 3D-polar Coordinates”.

[7] Branislav Micusík and Allan Hanbury - “Supervised texture Detection in Images”.

[8] Branislav Micusík and Allan Hanbury - “Automatic Image Segmentation by Positioning a Seed”.

[9] Darrin Cardani - “Adventures in HSV Space”.

[10] Jesús Angulo and Jean Serra - “Color Segmentation by Ordered Mergings”.

[11] Shamik Sural, Gang Qian and Sakti Pramanik - “Segmentation and Histogram Generation Using the HSV Color Space for Image Retrieval”.

[12] Yuchun Fang and Tieniu Tan - “A Novel Adaptive Colour Segmentation Algorithm and Its Application to Skin Detection”.