Evaluation of Colour Models for Computer Vision using Cluster Validation Techniques

David Budden, Shannon Fenn, Alexandre Mendes and Stephan Chalup

School of Electrical Engineering and Computer Science
Faculty of Engineering and Built Environment
The University of Newcastle, Callaghan, NSW, 2308, Australia.
{david.budden,shannon.fenn}@uon.edu.au
{alexandre.mendes,stephan.chalup}@newcastle.edu.au

Abstract. Computer vision systems frequently employ colour segmentation as a step of feature extraction. This is particularly crucial in an environment where important features are colour-coded, such as robot soccer. This paper describes a method for determining an appropriate colour model by measuring the compactness and separation of clusters produced by a k-means algorithm. *RGB*, *HSV*, YC_bC_r and *CIE L*a*b** colour models are assessed for a selection of artificial and real images, utilising an implementation of the Dunn's-based cluster validation index. The effectiveness of the method is assessed by qualitatively comparing the relative correctness of the segmentation to the results of the cluster validation. Results demonstrate there is a significant variation in segmentation quality among colour spaces, and that YC_bC_r is the best choice for the DARwIn-OP platform tested.

Keywords: Image segmentation, colour representations, colour space analysis, clustering, cluster validation

1 Introduction

In computer vision, a complete mapping from an arbitrary 3-component colour space C to an indexed colour space assigns a colour class label m_i for every possible point in that space [15]. If each channel is represented by an n-bit value, the mapping is defined as:

$$\begin{aligned} C = [0, 2^n - 1]^3 &\mapsto m_i \\ i &\in [0, k-1] \end{aligned} \tag{1}$$

where k represents the number of colours defined within the indexed space. In a colour space C, each pixel in an image is represented by a triplet with each value representing the contribution of each component to the overall colour of that pixel. Projecting the pixel values into the colour space constructed by the orthogonal component axes results in a projected colour space distribution of

the original image. Points within the projected distribution are typically clustered about centroids, which represent the predominant colours within the image. Thus, in an image where the colour clusters are compact and well separated, a clustering algorithm such as k-means [7, 11, 20] is able to automate the process of colour segmentation. This is particularly applicable in an environment where important features are uniquely coloured, such as robot soccer [14]. Where computational resources are limited, the colour segmentation process is performed off-line, with the resultant mapping represented in the form of a $2^n \times 2^n \times 2^n$ lookup table (LUT). This LUT can then be used for efficient, real-time colour classification.

Many techniques already exist for effective colour segmentation, including clustering [7, 20, 19]; mean-shift and mode finding [4, 17, 20]; and histogram thresholding [1]. Unfortunately, there are no general algorithms or colour models that are suited to all colour images [3]. Instead, this paper compares four common colour models: *RGB* [1–3, 7, 20]; *HSV* [1–3, 7, 12, 19, 20], YC_bC_r [1, 3, 12, 20] and *CIE L*a*b** [3, 7, 20]) and proposes a method to assess their suitability for unsupervised colour segmentation by measuring the compactness and separation of clusters produced by the k-means algorithm. Robot soccer is chosen as a suitable colour-coded environment for testing the procedure [14], but it is anticipated that the procedure could be extended to determine the optimal colour model in other equivalent scenarios.

This paper is organized as follows. In Section 2 we describe the colour models tested; in Section 4 we describe the test images; in Section 3 we describe the performance metrics used to validate the clusters obtained; in Section 5 we present the computational results; and finally, in Sections 6 and 7 we have a discussion of the results, followed by the conclusion.

2 Colour Models

2.1 RGB

The RGB colour model is a space in which each colour is represented by a combination of tristimuli R (red), G (green) and B (blue) [3]. Any colour can be created by exactly one combination of these three colour bases, which are defined according to their wavelength (700.0 nm for red, 546.1 nm for green and 435.8 nm for blue [20]). Although it is intuitive to represent colour in such a manner, RGB suffers from sensitivity to variations in illumination due to the high correlation of its three components [1–4, 19].

2.2 HSV

The HSV colour model separates information regarding the chrominance and intensity values of a colour by projecting the RGB colour space onto a non-linear chroma value H (hue), a radial saturation percentage S (saturation) and a luminance-inspired value V (value) [3, 7, 20]. The HSV colour model is frequently used for colour segmentation as the individual colour components are

independent of the image brightness [3, 17, 19], thus resulting in more uniform clusters for similar colour values [1, 17]. Among the disadvantages, the H component is an angular value and thus wraps around from 2π to zero, potentially resulting in the splitting of clusters to opposite ends of the hue axis. Furthermore, the nonlinear transformation results in high susceptibility to noise for low values of V [3].

2.3 $\mathbf{YC_bC_r}$

Similarly to HSV, $\mathrm{YC}_b\mathrm{C}_r$ separates chrominance information into two channels C_b (blue chroma) and C_r (red chroma), and intensity into a third channel Y (luma) [3, 20]. The $\mathrm{YC}_b\mathrm{C}_r$ colour space can be obtained applying the following linear transformation to the RGB space:

$$\begin{bmatrix} Y' \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{bmatrix} \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}, \tag{2}$$

where R', G' and B' are 8-bit gamma-compressed colour components [20]. Although much of the component correlation found in RGB is removed, it still exists in part due to the linear nature of the transformation [3].

2.4 CIE L*a*b*

The CIE (Commission International de l'Eclairage) XYZ colour system, similarly to RGB, defines three tristimulis X, Y and Z, which can be combined to create any colour. Any colour can be created by linear combination of positive quantities of each component [3, 7, 20] (unlike RGB, which for instance, requires a negative amount of red light to be added to obtain certain colours in the blue-green range [20]). The CIE XYZ colour space can be obtained by applying the following linear transformation to the RGB space [20]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3}$$

The CIE L*a*b* (CIELAB) colour space applies nonlinear transformations to CIE XYZ to more accurately reflect the logarithmic manner in how humans perceive differences in chromaticity and luminance [3, 7, 20]. For a nominal white value (X_n, Y_n, Z_n) (chosen as the CIE D65 standard $(0.9642, 1, 0.8249)$), the L^* (lightness) component is defined as [20]:

$$L^* = 116 f\left(\frac{Y}{Y_n}\right), \quad f(t) = \begin{cases} t^{1/3} & \text{, if } t > \delta^3 \\ t/(3\delta^2) + 2\delta/3 & \text{, otherwise} \end{cases} \tag{4}$$

where $\delta = 6/29$, resulting in a value in the range $[0, 100]$. Similarly, a^* and b^* are defined as [20]:

$$a^* = 500\left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right] \quad \text{and} \quad b^* = 200\left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right] \tag{5}$$

3 Performance Metrics

The performance of a clustering algorithms is reflected in the quality of the resulting clusters. Normally, clusters are evaluated by measuring their compactness (intracluster similarity) and separation (intercluster dissimilarly) from other clusters [10]. Many performance metrics have been suggested for assessing the validity of cluster partitions [8], including the *Silhouette method* [16, 18], *Dunn's based index* [6, 8, 16], *Davies-Bouldin index* [5, 8, 16] and *C-index* [13, 8, 16]. In this paper, the Dunn's index was chosen as most suitable for validating cluster partitions within the colour spaces described in Section 1, as it has low computational complexity and does not rely on presumptions regarding the shape or relative sizes of clusters [8].

3.1 Dunn's Index

For any partition $X = X_1 \cup \ldots X_i \cup \ldots X_k$, with k clusters, where X_i represents the i^{th} cluster of such a partition, the Dunn's index, D, is defined as [6, 8, 16]:

$$D(X) = \min_{1\leq i\leq k}\left\{\min_{\substack{1\leq j\leq k\\ j\neq i}}\left\{\frac{\delta(X_i, X_j)}{\max_{1\leq l\leq k}\{\Delta(X_l)\}}\right\}\right\} \tag{6}$$

where $\delta(X_i, X_j)$ is the intercluster distance between clusters i and j; and $\Delta(X_l)$ is the intracluster distance for cluster l. The higher is the value of $D(X)$, the better is the clustering. The aim of this measure is to maximise the minimum intercluster distance $\delta(X_i, X_j)$, whilst minimising the maximum intracluster distance. In this paper, we use k-means for varying values of k. The optimal number of clusters, k^*, is that which maximises D. As Dunn's index is calculated using only two distances it is particularly susceptible to outliers [8], thus is is important that the distance functions $\delta(X_i, X_j)$ and $\Delta(X_l)$ be chosen in such a way as to minimise the effect of chromatic noise.

3.2 Methods of Calculating Distances

As mentioned before, the Dunn's Index requires the calculation of distances between objects to be clustered. Among the possible distance metrics, we opted for the *Euclidean distance*, which was used in all experiments presented in this study. In addition, intercluster and intracluster distances can be calculated in multiple ways. They are addressed next.

Intercluster Distances: Several methods have been suggested for calculating the intercluster distance between clusters X_i and X_j [10, 16]. Below we show a list of distances and how they are calculated:

- *Single linkage*: Shortest distance between any two objects, each belonging to a separate cluster X_i and X_j.
- *Complete linkage*: Greatest distance between any two objects, each belonging to a separate cluster X_i and X_j.
- *Average linkage*: Average distance between all pairs of points belonging to separate clusters X_i and X_j.
- *Centroid linkage*: Distance between the centroids of clusters X_i and X_j.
- *Average to centroids*: Average distance between the centroid of cluster X_i and all the points belonging to cluster X_j. This was the intercluster distance chosen for assessing image clustering performance, as it represents a good trade-off between outlier sensitivity and computational complexity.

For two clusters X_i and X_j belonging to partition X, the average to centroids linkage, $\delta(X_i, X_j)$, is defined as:

$$\delta(X_i, X_j) = \frac{1}{|X_i| + |X_j|} \left(\sum_{x \in X_i} d(x, C_{X_j}) + \sum_{y \in X_j} d(y, C_{X_i}) \right), \tag{7}$$

where the cluster centroids, C_{X_i} and C_{X_j}, are defined as:

$$C_{X_i} = \frac{1}{|X_i|} \sum_{x \in X_i} x \qquad C_{X_j} = \frac{1}{|X_j|} \sum_{y \in X_j} y \tag{8}$$

Intracluster Distances: As with the intercluster distances, there exists several methods of calculating the intracluster distance of a cluster X_i, including [16]:

- *Complete diameter*: Greatest distance between any two points belonging to the cluster.
- *Average diameter*: Average distance between all pairs of points belonging to the cluster.
- *Centroid diameter*: Average distance between the cluster centroid and all points belonging to that cluster. This distance was chosen for assessing image clustering performance, again for presenting a good trade-off between outlier sensitivity and computational complexity.

For a cluster X_i belonging to partition X, the centroid diameter, $\Delta(X_i)$, is defined as:

$$\Delta(X_i) = 2 \left(\frac{\sum_{x \in X_i} d(x, C_{X_i})}{|X_i|} \right), \tag{9}$$

where the cluster centroid C_{X_i} is defined as in Eq. 8.

4 Test Images

This paper uses a series of images, some artificially generated and some captured using the DARwIn-OP [9] robot vision system. Robot soccer [14] was chosen as an example of a colour-coded environment, where important features to be identified are assigned different colours. As not all features will be present in every vision frame, two images were selected to represent common soccer scenarios. The first image (see Table 1, bottom-right corner) has the camera pointed towards the ground, and contains only the field, field lines and the ball. Therefore, only three main colours are present. The second image (see Table 2, bottom-right corner) demonstrates the camera pointed toward the goal, and adds a robot and goalposts to the image. The number of main colours thus arises to five.

To understand how the system reacts to the presence of noise in an "ideal" image, a set of artificial data was generated for each image. They consist of homogeneous blocks of colours representative of those of the image features at varying levels of Gaussian noise (10%, 20% and 30%, see tables 1 and 2). As clustering is performed within each colour space (see Section 2), the results are independent of the position of each pixel in the image, thus the Gaussian noise should have an effect similar to the introduction of other coloured objects into actual vision frames, as well as variance in lighting conditions (real-world examples of such "noise" would be a cable extended over the field or a spectator watching the match, captured by the camera).

5 Experimental Results

The experimental results are illustrated by two groups of images, with 3 and 5 colours that ideally, should be identified and clustered correctly. k-means clustering was applied to all test images described in Section 4 in each of colour spaces described in Section 2. The range for the number of clusters for each image was chosen to be no less than the number of features identifiable in the images (e.g. field, goal, ball, etc). Thus, the values of k tested were chosen as $k = \{3, 4, \ldots, 10\}$ for the images in Table 1, and $k = \{5, 6, \ldots, 12\}$ for those in Table 2.

The Dunn's index was applied as a method of evaluating the performance of the clustering in each colour space by considering the compactness and separation of resultant clusters, as described in Section 3. The process of clustering and validation for each image, colour space and k value was repeated 100 times to allow for variations in performance due to the random initial placement of the cluster centroids in the k-means method. The figures in the tables below are the average values over those 100 runs. The number between brackets we present the standard deviation of each 100 runs. The best result for each image is evidenced in boldface, along with its standard deviation.

Table 1 demonstrates that for a small number of predominant colours and a low level of chromatic noise, clusters within the RGB colour space are overall more compact and separated. Those clusters correctly classified the three colours.

Table 1. Cluster quality evaluation using the Dunn's index for the images with 3 dominant colours. The four colour spaces were tested using four images - 3 artificially created, with chromatic noise levels of 10%, 20% and 30%, and one original image, obtained through the DARwIn-OP [9] vision system. The figures represent the average of 100 runs of the k-means algorithm; the standard deviation is between brackets. The RGB colour space presented a better performance for low chromatic noise levels, whereas YC_bC_r performs better with higher noise levels, as with the original image.

3 colours - 10% noise

Colour space	Number of clusters (k)									
	3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
RGB	**3.66** (0.00)	0.50 (0.00)	0.23 (0.11)	0.16 (0.11)	0.11 (0.08)	0.10 (0.06)	0.09 (0.02)	0.08 (0.03)
HSV	0.15 (0.00)	0.38 (0.06)	0.12 (0.03)	0.21 (0.01)	0.50 (0.04)	0.24 (0.02)	0.23 (0.04)	0.15 (0.08)
YC_bC_r	3.00 (0.00)	0.56 (0.00)	0.39 (0.26)	0.40 (0.10)	0.28 (0.06)	0.20 (0.10)	0.16 (0.11)	0.11 (0.09)
CIELAB	1.11 (0.00)	0.25 (0.00)	0.18 (0.03)	0.21 (0.00)	0.23 (0.03)	0.29 (0.06)	0.22 (0.09)	0.19 (0.04)

3 colours - 10% noise

3 colours - 20% noise

Colour space	Number of clusters (k)									
	3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
RGB	**1.01** (0.00)	0.59 (0.00)	0.30 (0.01)	0.32 (0.02)	0.34 (0.09)	0.32 (0.09)	0.30 (0.04)	0.29 (0.01)
HSV	0.34 (0.00)	0.17 (0.01)	0.31 (0.08)	0.22 (0.03)	0.22 (0.00)	0.23 (0.00)	0.18 (0.06)	0.11 (0.02)
YC_bC_r	0.83 (0.00)	0.72 (0.00)	0.35 (0.00)	0.55 (0.00)	0.34 (0.02)	0.34 (0.07)	0.34 (0.05)	0.32 (0.03)
CIELAB	0.32 (0.00)	0.35 (0.00)	0.19 (0.00)	0.23 (0.00)	0.29 (0.00)	0.32 (0.02)	0.19 (0.02)	0.15 (0.00)

3 colours - 20% noise

3 colours - 30% noise

Colour space	Number of clusters (k)									
	3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
RGB	0.52 (0.00)	0.58 (0.00)	0.32 (0.00)	0.45 (0.00)	0.40 (0.05)	0.38 (0.02)	0.37 (0.02)	0.35 (0.02)
HSV	0.33 (0.00)	0.38 (0.00)	0.18 (0.01)	0.20 (0.01)	0.21 (0.01)	0.13 (0.01)	0.14 (0.03)	0.16 (0.03)
YC_bC_r	0.47 (0.00)	**0.75** (0.00)	0.42 (0.00)	0.53 (0.00)	0.30 (0.01)	0.32 (0.02)	0.35 (0.02)	0.41 (0.03)
CIELAB	0.26 (0.00)	0.23 (0.01)	0.24 (0.00)	0.18 (0.00)	0.18 (0.01)	0.17 (0.01)	0.19 (0.00)	0.16 (0.03)

3 colours - 30% noise

3 colours - Original image

Colour space	Number of clusters (k)									
	3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
RGB	0.11 (0.00)	0.11 (0.01)	0.09 (0.00)	0.04 (0.00)	0.05 (0.01)	0.05 (0.01)	0.05 (0.01)	0.04 (0.01)
HSV	0.11 (0.00)	0.08 (0.00)	0.06 (0.00)	0.04 (0.00)	0.04 (0.00)	0.05 (0.01)	0.05 (0.00)	0.05 (0.01)
YC_bC_r	0.11 (0.00)	**0.12** (0.00)	0.12 (0.01)	0.05 (0.01)	0.05 (0.01)	0.05 (0.01)	0.04 (0.01)	0.04 (0.01)
CIELAB	0.10 (0.00)	0.11 (0.00)	0.07 (0.00)	0.07 (0.02)	0.07 (0.02)	0.04 (0.01)	0.04 (0.01)	0.03 (0.00)

3 colours - Original

Table 2. Cluster quality evaluation using the Dunn's index for the images with 5 dominant colours. In this set of tests, the YC_bC_r colour space performed better than the other three, irrespectively of the image type and noise level. In addition, the optimal Dunn's index value was obtained with $k = 5$ on all four images. RGB maintained a good overall performance, but in this case, YC_bC_r was consistently better.

5 colours - 10% noise

Colour space	Number of clusters (k)							
	5	**6**	**7**	**8**	**9**	**10**	**11**	**12**
RGB	1.86 (0.00)	0.53 (0.00)	0.74 (0.24)	0.32 (0.10)	0.29 (0.07)	0.27 (0.06)	0.26 (0.05)	0.26 (0.03)
HSV	0.21 (0.00)	0.21 (0.01)	0.45 (0.08)	0.13 (0.02)	0.20 (0.06)	0.17 (0.08)	0.10 (0.06)	0.06 (0.02)
YC_bC_r	**2.23 (0.00)**	0.60 (0.00)	1.07 (0.25)	0.48 (0.15)	0.33 (0.08)	0.29 (0.05)	0.29 (0.05)	0.27 (0.05)
CIELAB	0.24 (0.03)	0.33 (0.02)	0.26 (0.16)	0.17 (0.06)	0.18 (0.04)	0.19 (0.03)	0.18 (0.04)	0.19 (0.05)

5 colours - 10% noise

5 colours - 20% noise

Colour space	Number of clusters (k)							
	5	**6**	**7**	**8**	**9**	**10**	**11**	**12**
RGB	0.96 (0.00)	0.53 (0.01)	0.33 (0.03)	0.33 (0.02)	0.33 (0.07)	0.30 (0.05)	0.29 (0.04)	0.28 (0.03)
HSV	0.28 (0.04)	0.24 (0.02)	0.19 (0.00)	0.23 (0.03)	0.22 (0.03)	0.22 (0.03)	0.16 (0.05)	0.13 (0.04)
YC_bC_r	**1.22 (0.00)**	0.63 (0.00)	0.43 (0.14)	0.50 (0.07)	0.33 (0.01)	0.37 (0.02)	0.38 (0.04)	0.33 (0.06)
CIELAB	0.20 (0.00)	0.26 (0.00)	0.24 (0.10)	0.21 (0.05)	0.22 (0.05)	0.27 (0.00)	0.27 (0.02)	0.30 (0.01)

5 colours - 20% noise

5 colours - 30% noise

Colour space	Number of clusters (k)							
	5	**6**	**7**	**8**	**9**	**10**	**11**	**12**
RGB	0.97 (0.00)	0.47 (0.07)	0.44 (0.03)	0.41 (0.04)	0.40 (0.04)	0.38 (0.06)	0.37 (0.05)	0.36 (0.05)
HSV	0.30 (0.00)	0.24 (0.02)	0.26 (0.03)	0.38 (0.10)	0.32 (0.04)	0.22 (0.01)	0.21 (0.01)	0.20 (0.01)
YC_bC_r	**0.99 (0.00)**	0.67 (0.00)	0.84 (0.08)	0.43 (0.08)	0.42 (0.05)	0.41 (0.04)	0.40 (0.04)	0.40 (0.04)
CIELAB	0.34 (0.00)	0.29 (0.04)	0.32 (0.00)	0.33 (0.00)	0.31 (0.02)	0.38 (0.04)	0.44 (0.02)	0.36 (0.03)

5 colours - 30% noise

5 colours - Original image

Colour space	Number of clusters (k)							
	5	**6**	**7**	**8**	**9**	**10**	**11**	**12**
RGB	0.22 (0.04)	0.10 (0.04)	0.07 (0.02)	0.06 (0.01)	0.06 (0.01)	0.07 (0.02)	0.07 (0.03)	0.07 (0.03)
HSV	0.19 (0.00)	0.12 (0.01)	0.12 (0.01)	0.13 (0.04)	0.07 (0.02)	0.06 (0.02)	0.06 (0.03)	0.05 (0.03)
YC_bC_r	**0.25 (0.12)**	0.06 (0.00)	0.06 (0.01)	0.06 (0.02)	0.07 (0.02)	0.07 (0.03)	0.08 (0.03)	0.07 (0.03)
CIELAB	0.07 (0.02)	0.06 (0.02)	0.05 (0.01)	0.05 (0.01)	0.05 (0.02)	0.05 (0.01)	0.05 (0.02)	0.06 (0.03)

5 colours - Original

Additional chromatic noise results in better clustering within the YC_bC_r colour space and for a larger k value, equal to 4. The fourth cluster was used to classify the black colour. Also important to mention is the low standard deviation values obtained, which means that either all 100 k-means runs converged to the same local minimum; or to several local minima, but with similar Dunn's index values.

Table 2 shows the results for a larger number of predominant colours. In that case, YC_bC_r outperforms RGB, HSV and CIELAB colour spaces in all images, irrespective of chromatic noise level. RGB still has very robust results, but YC_bC_r is numerically better. The standard deviation values are considerably higher than for three predominant colours. Particularly for the original image, the standard deviation for YC_bC_r (k=5) is very high. After a thorough analysis, we observed the presence of four different local minima, with Dunn's indexes of 0.32 (obtained 29 times), 0.31 (35 times), 0.09 (10 times) and 0.04 (26 times). Even though YC_bC_r appears to be the best colour space, further tests with a larger number of real images should be conducted in the future to determine whether this result was just an artifact.

Analysing Tables 1 and 2 as a whole, we can conclude that RGB and YC_bC_r present consistently better results than HSV and CIELAB, irrespective of the the image tested. Apart from the numerical results, we present a qualitative assessment of the performance of the colour segmentation for each image using YC_bC_r, as it had a better performance with the original images, which represent the soccer game scenario (see Section 6).

All computational tests were implemented using MATLAB and the DARwIn-OP vision system uses a 2MP HD USB camera [9].

6 Discussion

The results in Section 5 demonstrate that for images with either more than three predominant features or greater than 20% chromatic noise, k-means produces clusters with maximum compactness and separation when performed in the YC_bC_r colour space. However, for this to be an accurate measure of the colour segmentation performance, it is crucial that the clusters correspond with the actual set of feature colours. In order to verify how representative the clusters found by the k-means are of the real colours in the image, we show a series of images in Figure 1. As an example, the image in Table 2 contains five uniquely coloured features - the green field, white lines, black robot, yellow goalposts and red ball. A correct segmentation should therefore map the pixels comprising each of these features to a unique colour label.

The pixels in the images in Figure 1b-e were painted with the centroid values of the clusters that they were assigned to. However, given that we executed 100 runs of the k-means procedure for each configuration of image and colour space, we had to determine, among all different solutions, which set of centroids was the most representative. After analysing the results for a given configuration, it was determined that the number of different solutions was never greater than five - i.e. the same sets of centroids were reached several times in those 100 runs.

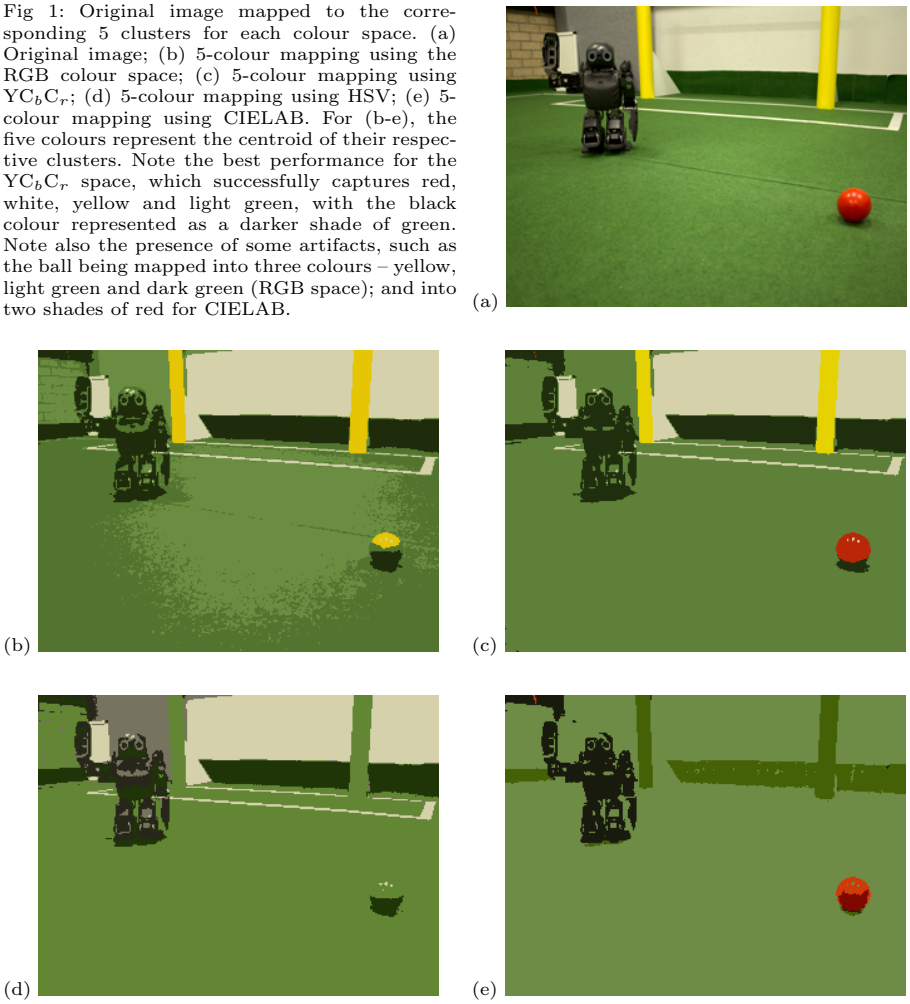

Fig 1: Original image mapped to the corresponding 5 clusters for each colour space. (a) Original image; (b) 5-colour mapping using the RGB colour space; (c) 5-colour mapping using YC_bC_r; (d) 5-colour mapping using HSV; (e) 5-colour mapping using CIELAB. For (b-e), the five colours represent the centroid of their respective clusters. Note the best performance for the YC_bC_r space, which successfully captures red, white, yellow and light green, with the black colour represented as a darker shade of green. Note also the presence of some artifacts, such as the ball being mapped into three colours – yellow, light green and dark green (RGB space); and into two shades of red for CIELAB.

With that in mind, the centroid values used in each configuration depicted in Figure 1b-e are the solutions that *occurred most frequently*.

Table 2 suggests the YC_bC_r colour space allows better clustering performance than the other three colour spaces for Figure 1a. That assertion is confirmed by qualitatively assessing the result of applying k-means determine the main colours in the image (for $k = 5$). Figure 1b shows that clustering in RGB failed to correctly label the red ball, which contains pixels that belong to three different clusters (yellow and two shades of green). In addition, it associated the field and the robot with three separate clusters, at different levels of green. In contrast, clustering in YC_bC_r (Figure 1c) labeled each feature correctly. The only issue in that image was that the robot was not clustered as either black or gray, which

would be the ideal situation. However, even though the centroid of that cluster has a dark green shade, the clustering is clearly better than the other four, and managed to identify the main features in the image. Figure 1d depicts the result for HSV; it fails to identify the goal post, assigning to it the same colour of the field. In addition, the ball presents two shades of green. Finally, in Figure 1e we show the result for CIELAB. It identifies the ball and the robot, but nothing else. The field has the same colour as the wall, and even the filed lines, which are easily identified by all methods, are missing in the picture. The result of those tests indicate that both sets of images, with 3 and 5 colours, support cluster validation as an effective method for assessing the appropriateness of a colour model for segmentation.

7 Conclusion

This work analysed the use of different colour spaces in the quality of colour classification in a computer vision system. The application of a Dunn's index cluster validation technique demonstrated that the k-means algorithm produced clusters of maximum compactness and separation in the $\mathrm{YC}_b\mathrm{C}_r$ colour space, for images with five uniquely coloured features. RGB, HSV and CIELAB yielded poorer results independent of the level of chromatic noise present. For images with three uniquely coloured features, $\mathrm{YC}_b\mathrm{C}_r$ yielded the best results for higher levels of chromatic noise, whereas RGB had a better performance for low levels of noise.

The quantitative results were also qualitatively assessed by visualising the colour segmentation for each colour model. In that case, the qualitative analysis of the $\mathrm{YC}_b\mathrm{C}_r$ results confirmed the numerical results. Correct segmentation corresponded with higher values of the Dunn's-based index, thus supporting cluster validation as an effective technique of assessing the performance of a colour model for segmentation within a colour-coded environment.

Future research will focus on the automatic determination of optimal k values with the goal of achieving real-time classification of colours using the DARwIn-OP platform.

Acknowledgement

David Budden and Shannon Fenn would like to thank the University of Newcastle's Faculty of Engineering and Built Environment for the support provided by their Summer Research Scholarship program.

References

1. Bruce, J., Balch, T., Veloso, M.: Fast and inexpensive color image segmentation for interactive robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2061–2066. IEEE (2000)

2. Brusey, J., Padgham, L.: Techniques for obtaining robust, real-time, colour-based vision for robotics. In: Lecture Notes in Computer Science. LNAI 1856, pp. 63–73. Springer, USA (2000)
3. Cheng, H.D., Jiang, X.H., Sun, Y., Wang, J.: Color image segmentation: Advances and prospects. Pattern Recognition 34, 2259–2281 (2001)
4. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 603–619 (2002)
5. Davies, D.L., Bouldin, D.W.: A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence 1, 224–227 (1979)
6. Dunn, J.C.: Well-separated clusters and optimal fuzzy partitions. Journal of Cybernetics 4, 95–104 (1974)
7. Forsyth, D.A., Ponce, J.: Computer vision: A modern approach. Prentice Hall Professional Technical Reference, Canada (2005)
8. Gunter, S., Bunke, H.: Validation indices for graph clustering. Pattern Recognition Letters 24, 1107–1113 (2003)
9. Ha, I., Tamura, Y., Asama, H., Han, J., Hong, D.W.: Development of open humanoid platform DARwIn-OP. In: Proceedings of SICE Annual Conference, pp. 2178–2181. IEEE (2011)
10. Hakidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. Journal of Intelligent Information Systems 17, 107–145 (2001)
11. Hartigan, J.A., Wong, M.A.: Algorithm AS136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) 28, 100–108 (1979)
12. Henderson, N., King, R., Chalup, S.: An automated colour calibration system using multivariate gaussian mixtures to segment HSI colour space. In: Australasian Conference on Robotics and Automation (ACRA) (2008)
13. Hubert, L., Schultz, J.: Quadratic assignment as a general data analysis strategy. British Journal of Mathematical and Statistical Psychology 29, 190–241 (1976)
14. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E.: Robocup: The robot world cup initiative. In: Proceedings of the first international conference on Autonomous agents, pp. 340–347. ACM New York, USA (1997)
15. M. Sridharan, P.S.: Real-time vision on a mobile robot platform. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2148–2153. IEEE (2005)
16. N. Bolshakova, F.A.: Cluster validation techniques for genome expression data. Signal Processing 83, 825–833 (2003)
17. Park, J.H., Lee, G.S., Park, S.Y.: Color image segmentation using adaptive mean shift and statistical model-based methods. Computers and Mathematics with Applications 57, 970–980 (2009)
18. Rousseeuw, P.J.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20, 53–65 (1987)
19. Suran, S., Qian, G., Pramanik, S.: Segmentation and histogram generation using the HSV color space for image retrieval. In: International Conference on Image Processing, pp. 589–592. IEEE (2002)
20. Szeliski, R.: Computer Vision: Algorithms and applications. Springer, USA (2010)