Noname manuscript No.
(will be inserted by the editor)

Finding Near Optimum Colour Classifiers

Genetic Algorithm-Assisted Fuzzy Colour Contrast Fusion using Variable Colour Depth

Heesang Shin · Napoleon H. Reyes

Received: date / Accepted: date

Abstract This paper presents a complete self-calibrating illumination intensity-invariant colour classification system. We extend a novel fuzzy colour processing technique called Fuzzy Colour Contrast Fusion (FCCF) by combining it with a Heuristic-assisted Genetic Algorithm (HAGA) for automatic fine-tuning of colour descriptors. Furthermore, we have improved FCCF's efficiency by processing colour channels at varying colour depths in search for the optimal ones. In line with this, we introduce a reduced colour depth representation of a colour image while maintaining efficient colour sensitivity that suffices for accurate real-time colour-based object recognition. We call the algorithm Variable Colour Depth (VCD) and we propose a technique for building and searching a VCD look-up table (LUT). The first part of this work investigates the effects of applying fuzzy colour contrast rules to varying colour depths as we extract the optimal rule combination for any given target colour exposed under changing illumination intensities. The second part introduces the HAGA-based parameter-optimisation for automatically constructing accurate colour classifiers. Our results show that for all cases, the VCD algorithm, combined with HAGA for parameter optimisation improve colour classification via a pie-slice colour classifier.For 6 different target colours, the hybrid algorithm was able to yield 17.63% higher overall accuracy as compared to the pure fuzzy approach. Furthermore, it was able to reduce LUT storage space by 78.06% as compared to the full-colour depth LUT.

Heesang Shin
Massey University
Institute of Information & Mathematical Sciences
Private Bag 102 904, North Shore City 0745
Auckland, New Zealand
Tel.: +64 9 414 0800 extn 9257
Fax: +64 9 441 8181
E-mail: h.shin@massey.ac.nz

Napoleon H. Reyes, Ph.D.
Massey University
Institute of Information & Mathematical Sciences
Private Bag 102 904, North Shore City 0745
Auckland, New Zealand
Tel.: +64 9 414 0800 extn 9512
E-mail: n.h.reyes@massey.ac.nz

Keywords Fuzzy Colour Contrast Fusion · Variable Colour Depth · Fuzzy-Genetic · Look-up Table

1 Introduction

Colours depicting a moving object, as captured by a camera change under spatially varying illuminations. In addition, confounding effects due to the spectral reflectance characteristic of the object, the spectral power distribution of the illuminant [1] and sensitivity of the camera make the colour classification task very difficult [2]. On the contrary, our model, the human visual system is able to recognise the colours of objects irrespective of the light used to illuminate them. This ability is called colour constancy and is a result of human evolution that adjusts the white balance dynamically as lighting changes [3]. Unlike other visual properties like shape and size, colours are considered to be view invariant and largely independent of resolution [4]. This makes colours a highly valuable property for object recognition, but colours are only stable under constant illumination. This raises difficulty and opportunity for computer vision research, and in particular, colour classification. The colour classification process we refer to here pertains to correctly distinguishing colours belonging to the same object as it moves across the scene, with the presence of other similar colours. Although there are many researches about colour classification, only few researches successfully classify colours under varying illuminations [2, 5–10].

FCCF(Fuzzy Colour Contrast Fusion) is one of the promising algorithms that offers a solution to classifying colours in real-time, under spatially varying illuminations and presence of other similar colours that should be effectively distinguished [10]. FCCF embodies colour constancy by employing colour contrast operations on each colour channel independently using fuzzy logic. However, FCCF requires a multitude of colour classification parameters for each target colour object. These parameters are divided into two parts, the pie-slice decision region (classification angles and radii), and the colour contrast operation rules. Finding the pie-slice classification region originally depends on manual calibration and is a painstaking process. In [6], a motion-based colour learning technique was proposed to extract the pie-slice classification region, but the extraction of the optimal colour contrast operation rules was performed only through a brute-force search method. What is desirable is an automatic colour classification system that searches for the pie-slice classification region and colour contrast rules more effectively. The system should learn the illumination conditions of the scene and classify the target colours without any further human intervention. Furthermore, improvement of the algorithm's accuracy and reduction of memory requirements are worth researching about. These desirable features are studied and dealt with in this paper.

2 Related Works

We have reviewed related literatures in the fields of colour extraction, segmentation, classification and object recognition from 1990 onwards. Prior to 1990, colour object recognition was not a very viable task due to the prevailing hardware limitations during those times: slow CPU and small memory capacity, and the unavailability of fast digital image acquisition systems. As a consequence, object recognition researches were mostly limited in the gray scale level. From 1990, the number of researches in the field of colour object recognition has increased exponentially as the computer processing power leaped almost every year.

Many techniques for object recognition has taken the colour-based approach. Of these approaches, colour histograms emerged to be popular because it is simple to construct, fast, and it is view and shape invariant. Swain and Ballard used colour histograms to extract features in the scene with controlled lighting conditions in 1990 [4]. One of the promising techniques proposed in this work is the Variable Colour Depth colour representation, along with the VCD LUT. So far, to the best of our knowledge, there is only one existing algorithm by Stachowicz and Lemke, that adheres to the same idea of using colour depth reduction for improved colour discriminability [11]. They proposed an image identification technique using a simplified colour histogram in 2000 [11]. They also introduced a colour depth reduction technique that is used to construct a simplified colour histogram. However, their proposed approach is too simplistic, using only 3-bits for colour pixel classification. The algorithm fails for colour object recognition of objects lacking colour diversity. In addition, the presence of similar colours were not investigated as well. In a similar fashion, Browning and Veloso in 2005 also used colour histograms for classifying colour objects, but in an outdoor environment with promising results. Their approach employs an adaptive thresholding technique, and claims that the algorithm works even when there are changes in the illumination. Nevertheless, the presence of similar colours were never taken into account [5]. To speed-up the process of real-time colour classification, Kim and Chung used a look-up table approach in 1999 [12]. However, this entails a burden of calibrating the classifier manually. Moreover, simplification techniques through scaling were also introduced to speed-up the colour segmentation task. Dong et al. showed fast image segmentation using the K-means clustering algorithm with a layered pyramidal structure in 2006 [13]. The proposed technique worked fast even for very large images because their algorithms approximates the centre of the colour cluster from a $\frac{1}{16}$th scaled version of the original image. On the other hand, techniques based prior knowledge of the scene's geometry were employed to aid the colour training task. Heinemann et al. in 2007 [7] proposed a colour training algorithm for the RoboCup [14], four-legged (AIBO) league that matches an acquired colour information with prior knowledge of the geometry of the playing field. More recently, knowledge of the spatial relationship between colour classes in a colour space was utilised to develop an adaptive colour classification technique that works even when there are sudden illumination changes [8]. However, spatial illumination variations in the field and discrimination between similar colours were not taken into account. In addition, hardware-assisted adaptive illumination invariant techniques were introduced. Takahashi et al. in 2008 [15] employed a mechanical PID control to automatically adjust camera parameters such as the iris and the gain to adapt to illumination changes in the target environ-

ment. A reference red ring around the lens was used to determine when and how much adjustments for the iris and gain parameters need to be performed. Also, Hayashi and Fujiyoshi in 2008 [9] derived a colour-illuminance model that shows changes in RGB colour values under different illuminance settings with fixed white balance. The proposed research claims that conversion between an RGB colour value obtained from one iris parameter (F-number) to other RGB colour values corresponding to a different iris parameter is possible. However, the proposed system is dependent on an illuminance meter attached to the object to estimate the luminance level. Lastly, fuzzy colour processing algorithms are employed rather sparsely. Many of the existing techniques employ fuzzifications of the colour classes to solve ambiguity issues. On the other hand, in this research, the fuzzy techniques are mainly employed for colour corrections, to compensate for the illumination effects. To mention some related works on fuzzy approaches, Kashanipour et al. proposed a colour classification technique using fuzzy rule-sets operating in the HSI colour space and optimised with particle swarm optimisation technique in 2008 [16].
In another research, Hilderbrand and Fathi in 2004 [17] analysed welding spots through colour inspection and shape estimations. The proposed research employed fuzzy logic that enabled the use of linguistic terms rather than numeric values to describe the quality of welding spots. The colour space used is called HSI-colour space, with the I-value set to a constant value of 1.0. A colour classifier consisting of eight cylindrical coordinates from the HS-colour space represents the parameters of the fuzzy sets.

3 Problem Domain

An object wearing a multitude of solid colour patches is recognised and tracked-down in real-time. These collection of colours, when projected on a colour space would tend to form clusters and thereby distinguishable from one another. However, the colour clusterings are unstable even when the object being tracked down is under a fixed illumination because quantum electrical effects in the camera sensor chip easily distorts the colours captured and cannot be prevented. Even worse, the colour clusters drift in the colour space due to spatially varying illuminations [2, 5–10]. Figure 1 illustrates a snapshot of solid colour patches under spatially varying illuminations. The image was taken approximately 8 feet off the ground using an overhead camera on top of the robot soccer [?] playing field. It is worth noting that there are green and yellow colour patches in the scene. However, these colours are very difficult to distinguish from the image due to very strong bright white light illumination at the centre of the field.

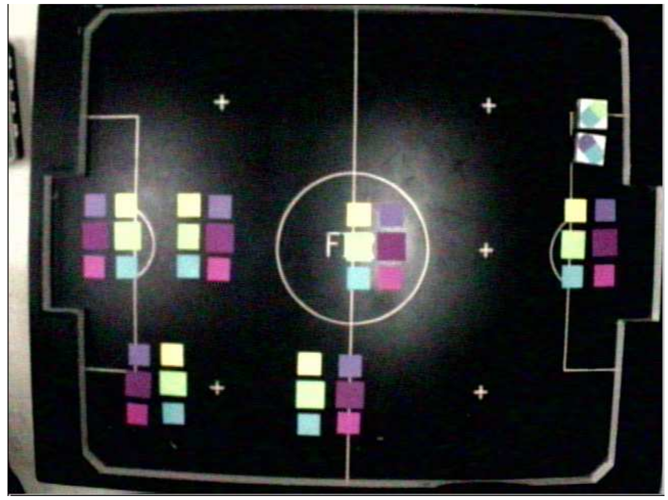

Fig. 1 Test Image with 40 Solid Colour Patches in 6 Different Colours Under Spatially Varying Illumination [2].

4 The Algorithms

4.1 Colour Space and the Pie-Slice Decision Region

Although the algorithms presented here works for any colour spaces, we have particularly selected the modified rg-chromaticity space for all the experiments performed herewith as it is already embodying some illumination invariance property [10]. Each captured colour pixel from a camera is converted to its corresponding rg-Hue and rg-Saturation colour components in the modified rg-chromaticity colour space. Equation 1 shows the conversion formulae:

$$r = R/(R+G+B)$$

$$g = G/(R+G+B)$$

$$rg - Saturation = \sqrt{(r-0.333)^2 + (g-0.333)^2}$$

$$rg - Hue = \tan^{-1}\frac{(g-0.333)}{(r-0.333)} \qquad (1)$$

rg-Hue and rg-Saturation values are used as angle and radius to refer to a point in a modified rg-chromaticity colour space where the origin lies at the coordinate (0.333, 0.333). This also corresponds to the white light point. Figure 2 shows the modified rg-chromaticity colour space, with the origin's shifted position.

A colour pixel is classified according to its position

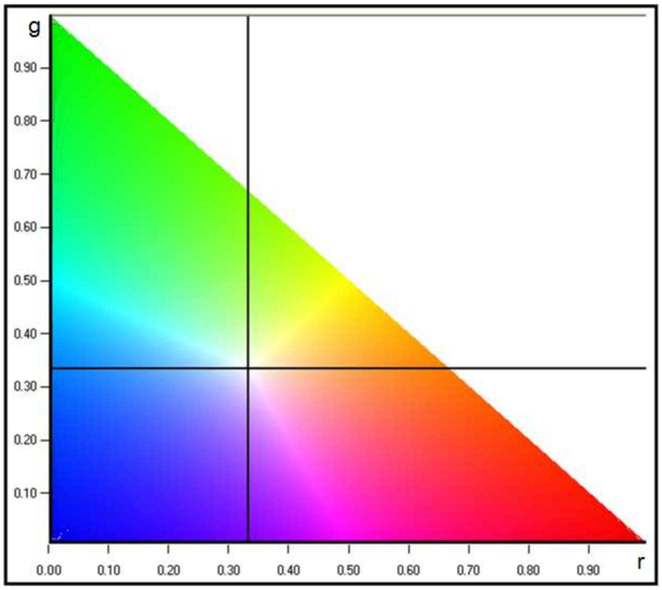

Fig. 2 rg-chromaticity Colour Space with Origin Shift Position [2].

in the colour space. A pie-slice decision region characterised by bounding angles and bounding radii defines a colour class (e.g. pink, orange, violet, etc.). The angle corresponds to the hue, while the radius tells about the saturation or purity of the colour. Figure 3 shows the pie-slice colour decision region.

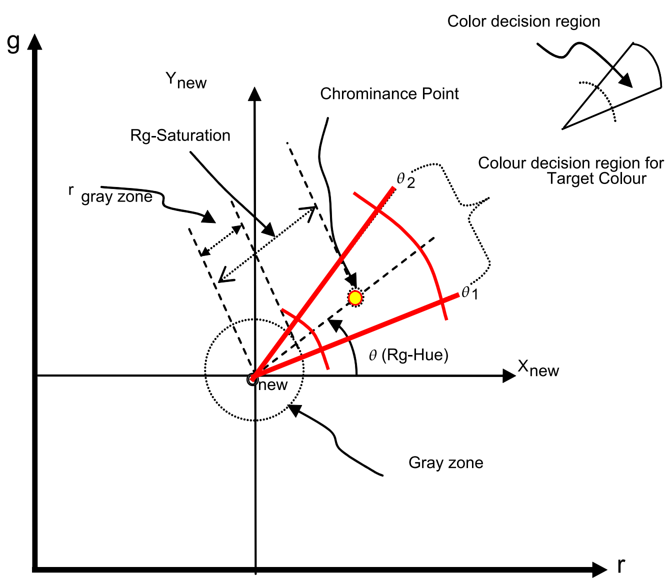

Fig. 3 Pie-Slice Colour Decision Region in the Modified rg-Colour Space [2].

4.2 Fuzzy Colour Contrast Fusion

In 2004 FCCF was introduced as a generic colour processing algorithm for stabilising the formation of colour clusters in a colour space for more accurate colour classification [2]. It is used in conjunction with a pie-slice decision region and employs colour contrast rules to compensate for the effects of spatially varying illumination in the scene. FCCF has continued to evolve since its inception. Originally, the algorithm requires only 9 parameters (i.e. min and max angles, and min and max radius for the pie-slice decision region, min and max angles for the colour contrast angles, and the colour contrast operators for the red, green and blue channels. In 2005 [10], FCCF was tested successfully on various colour spaces. In 2006 [18], FCCF was combined with Adaboost training for automatic colour classification that does not rely on the pie-slice colour classifier. Most recently, in 2008 [6], FCCF was fused with a colour learning algorithm that utilises successive frames containing the object being tracked were used for automatic calibration of the angles used for the pie-slice classifier. In addition, a colour classification scoring system was proposed and was tested to successfully extract the best colour contrast rule combination. However, in all these approaches, the calibration of all the parameters is not performed fully automatically.

Two complementary colour contrast operators (colour Enhance and colour Degrade) comprise the main mechanisms of FCCF. In the contrast enhance operation, any signal greater than or equal to the threshold (0.5) will be amplified, while signals less than the threshold will be attenuated. On the other hand, the contrast degrade operation performs the opposite. It pulls any given signal closer towards the threshold setting. FCCF applies the colour contrast operators independently on each colour component, and at different levels. Equation 2 shows the colour contrast enhance (or sigmoid/logistic function) operator, while Equation 3 shows the colour contrast degrade (or the logit function).

$$\alpha = \begin{cases} 2{\mu_\alpha}^2(y) & 0 \le \mu_\alpha(y) < 0.5 \\ 1 - 2[1 - \mu_\alpha(y)]^2 & 0.5 \le \mu_\alpha(y) \le 1 \end{cases} \tag{2}$$

$$\alpha = \begin{cases} 0.5 - 2(1 - [\mu_\alpha(y) + 0.5]^2) & 0 \le \mu_\alpha(y) < 0.5 \\ 0.5 + 2[\mu_\alpha(y) - 0.5]^2 & 0.5 \le \mu_\alpha(y) \le 1 \end{cases} \tag{3}$$

Successive applications of the contrast enhancement or degradation cause more adverse effects as shown in figures 4 and 5. The new RGB values produced after the application of the colour contrast operations are called *new* R, *new* G, and *new* B. These new RGB

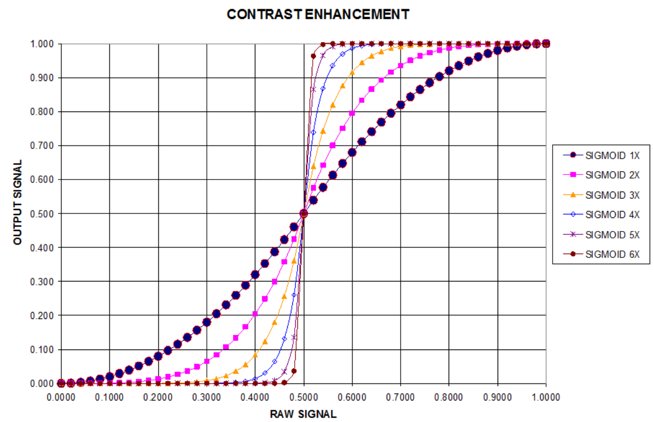

Fig. 4 Colour Contrast Enhance Operator (Sigmoid/Logistic Function) [2].

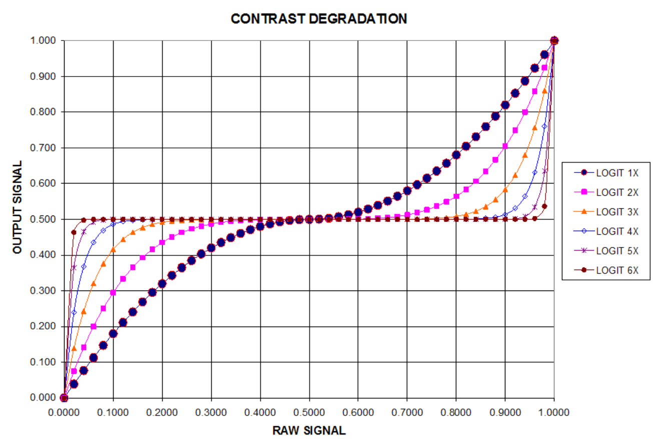

Fig. 5 Colour Contrast Degrade Operator (Logit Function) [2].

components are then converted to new rg-Hue and new rg-Saturation values to see whether this new point in the modified rg-chromaticity colour space is within the pie-slice decision region of the target colour. Figure 6 illustrates the overall colour classification algorithm described.

Using FCCF in a real-time environment requires a

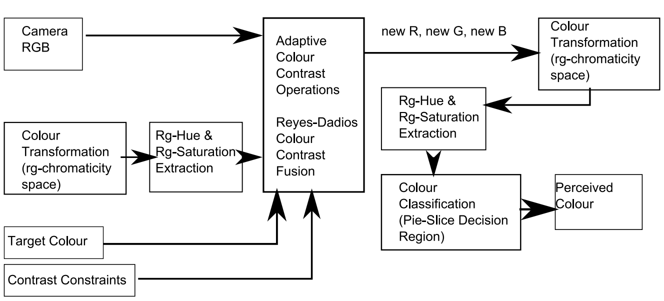

Fig. 6 Colour Contrast and Classification System Architecture [2].

pre-calculated look-up table (LUT). As described previously, the application of contrast operators and classification of colours requires a substantial effort. Therefore, pre-classifying all the possible colours in the colour space and storing the results into a LUT is very practical. In a real-time environment, an acquired RGB value simply indexed using an LUT to find which target colour the pixel belongs to. Algorithm 1 describes the complete LUT building process.

Algorithm 1: Look-Up Table Building Algorithm

for *Every possible Red values* **do**
  for *Every possible Green values* **do**
    for *Every possible Blue values* **do**
      if *colour value is classified as colour* t **then**
        $L = (R \ll (depth*2)) + (G \ll depth) + B;$
        $LUT[L] = t;$

depth is the colour depth values of each colour channel in source image.

4.3 Variable Colour Depth

4.3.1 Variable Colour Depth

When computers access and manipulate a colour explicitly, the colour space model representation must be in binary form. The most common defining characteristic is called colour depth. Colour depth represents the total number of bits assigned to represent a colour in a given colour space. Commonly, the colour depth is divided evenly for each colour space component and each component's normalised value is converted to fit in a given number of bits. Bits representation is usually in integer form and the fractional parts are either lost, or rounded up or down when converting from a higher to lower resolution and vice-versa. In RGB colour space, the common colour depth setting is 24-bits, which consists of 8-bits per colour component. This represents the 256 shades of red, green and blue that totals 16.7 million colours when combined. Each bit of colour depth increment yields double the precision of a component. The corollary is also true, each bit of colour depth decrement loses half its precision. Although higher colour depth represents more colours, it needs more memory and, demands higher computational efforts, and does not guarantee to yield better outcome.

Variable colour depth is a non-conventional approach to representing colour for digital image processing. In the RGB colour space, three components are usually

Table 1 Sample Variable Colour Depth Representations of the Normalised Colour Component Values 0.8 Red, 0.5 Green, and 1.0 Blue

Colour Depth				Binarized Representation			Nomarlised Value		
Total Bits	Red	Green	Blue	Red	Green	Blue	Red	Green	Blue
24	8	8	8	11001100	01111111	11111111	0.8	0.498039	1
22	8	7	7	11001100	0111111	1111111	0.8	0.496063	1
22	8	6	8	11001100	011111	11111111	0.8	0.492063	1
21	5	8	8	11000	01111111	11111111	0.774194	0.498039	1
21	6	7	8	110010	0111111	11111111	0.793651	0.496063	1
21	7	7	7	1100101	0111111	1111111	0.795276	0.496063	1
18	7	6	5	1100101	011111	11111	0.795276	0.492063	1
18	6	6	6	110010	011111	111111	0.793651	0.492063	1
18	8	5	5	11001100	01111	11111	0.8	0.483871	1
18	6	6	6	110010	011111	111111	0.793651	0.492063	1
17	7	5	5	1100101	01111	11111	0.795276	0.483871	1
17	5	6	6	11000	011111	111111	0.774194	0.492063	1
15	5	5	5	11000	01111	11111	0.774194	0.483871	1

represented using equal magnitudes (i.e. 0..255), and therefore, represented using equal number of bits. In the Variable Colour Depth representation, each colour component could be represented using varying number of bits. For example, a colour depth of 6-bits for red, 8-bits for green, and 8-bits for blue in the RGB colour space means that the red component have a quarter of resolution less than the green and blue components. However, it does not mean that the red component would only represent a quarter of the possible values covered by the other colour components but its colour gradient is only less smoother than other components. Table 1 shows the bit representation of Variable Colour Depth and the corresponding values of the colour components. Figure 7 illustrates the normalised output values from different colour depths.

When a colour pixel is represented as a combination

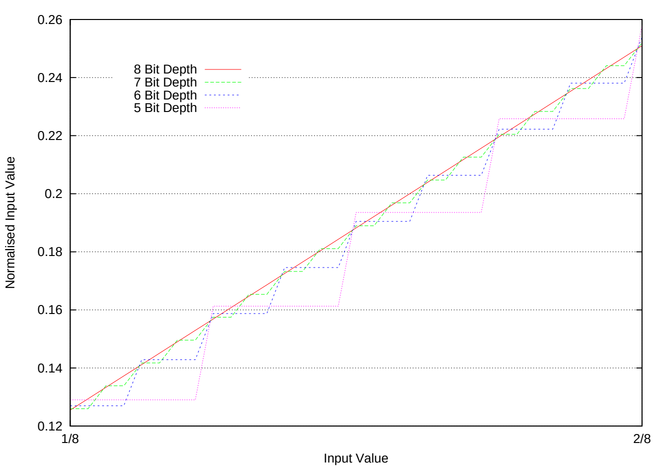

Fig. 7 Enlarged Section of Normalised Input Values From Various Colour Depth Input Values From 1/8 to 2/8

of colour components in the digital colour space, each component holds a value in some memory storage. In addition, each component reflects the component's level of contribution to the composition of the colour, as well as it's influence on the pixel's position in the colour space. The size of memory storage determines the number of possible levels between the minimum and the maximum that can be assigned to each component. We use the term 'colour component depth' to count the number of bits required to hold any given component. On the other hand, the term 'colour depth' is used to express the total number of bits to represent all of the components of a single colour in any given colour space. When the colour depth changes, it is common to adjust each colour component depth uniformly in the colour space altogether. When the colour depth decreases, the colour information is simplified and memory requirement is reduced. The simplified colour information affects the intended or unintended results on the image quality, such as loss of colour shades or distinguishing artifacts. Figure 8 illustrates the effects of colour depth reduction. The details of the clouds clearly show the loss of colour shades as a result of colour depth reduction.

In general, the reduction of colour depth degrades the

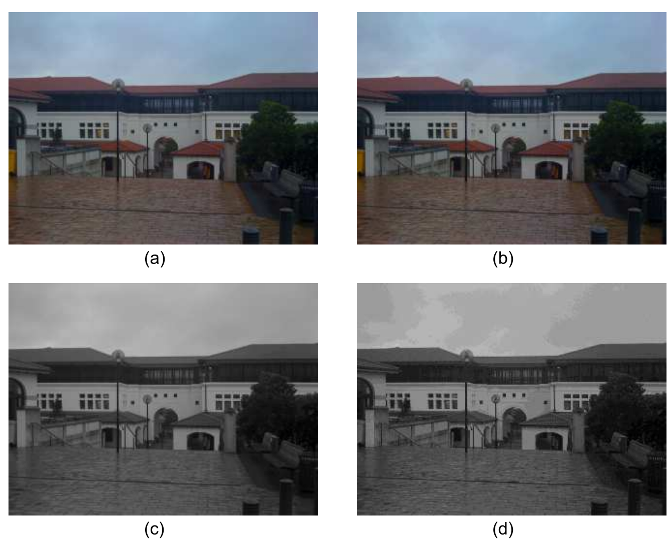

Fig. 8 Examples of Colour Depth Reduction; (a) Original 24-bit RGB Image; (b) Reduced 15-bit RGB Image from (a); (c) 8-bit Gray-Scale Image from (a); (d) 4-bit Gray-Scale Image from (c).

image quality. As a result, similar colours are merged into a single colour as shown in figure 8, from (d) the windows in the left hand side were merged together and they are no longer distinguishable from each other. However reducing the colour depth from 24-bit RGB colour depth to 15-bit RGB colour depth may not affect the human visual experience substantially. For example, figure 9 shows an image with the original 24-bit RGB colour depth, and the reduced 16-bit and 15-bit RGB colour depths. Although the 24-bit RGB colour depth image used more than ten times the number of colours used in other images, the visual differences are hardly noticeable. It particulary worked well when the 24-bit RGB colour depth was converted to 16-bit RGB colour depth which consisted of 5-bits for red and blue, and 6-bits the for green component. The main reason behind this is that the human vision system is particulary more sensitive to medium visible wavelengths (yellow-green) than other visible wavelengths [19].

Colour information reduction has been applied by many researches already, and the main impetus generally is to get rid of colour information that will not compromise the quality of the image. YUV for instance usually preserves only 1 value of U and V for every 4 pixels in an image, and this is usually specified as 4-1-1 (sometimes 4-2-2 in other systems). In this research, we are not reducing the colour depth of the image for storage purposes, but we are reducing the colour depth of the image for analysing it; that is, for colour classification. To the best of our knowledge, there is only one similar attempt adhering to the same idea. In [11], a colour depth reduction technique specifically for colour classification is proposed. Their technique however was only partially successful when tested on simple flag identification, stamp identification and landscape classification. Only 3 bits were used to classify a colour pixel and histograms for 8 predefined colour classes (i.e. red, green, blue, cyan, magenta, yellow, white and black) are generated afterwards. Due to lack of colour diversity of some objects being classified, the algorithm may fail as it also does not take into account spatial information.

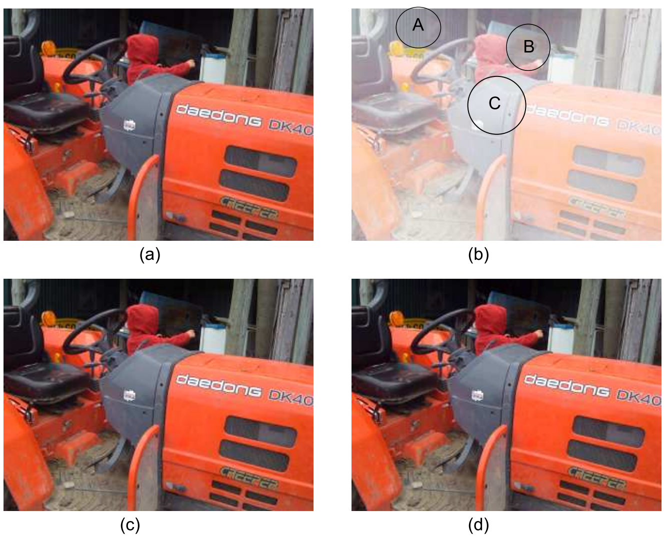

Fig. 9 Examples of Colour Depth Reduction; (a) Original 24-bit RGB Image, 55,880 Colours were Used; (b) Areas Where Colour Differences are Visible; (c) 16-bit RGB Image, 4,391 Colours were Used; (d) 15-bit RGB Image 2,814 Colours were Used.

The algorithms that we employ aim at increasing colour discriminability of the target objects, especially for cases where there are similar colours present in the scene and they need to be classified accurately. For real-time execution, a special Variable Colour Depth LUT is utilised. As a consequence, with the reduced colour depth, the proposed VCD LUT also improves storage efficiency as it requires significantly lesser storage space.

In a reduced colour component depth representation, adaptively varying the colour depths could actually enhance the discriminating features of colours. As an example, as shown in Figure 10, we have represented the red component using only 3-bits, while using 8-bits for both the blue and the green components. It is evident from the graph that reducing the bit representation for the red component allows for better discriminability for those regions in the colour space where all possible shades of red are present. The distinguishable horizontal segmentations appeared in the middle of the colour space where red component ranges from its minimum level up to its maximum level. Thus, for colour classification of target colours that are mostly comprised of the red component (e.g. pink and violet), reducing the red component could actually help improve colour classification accuracy. This is evident in figure 10.

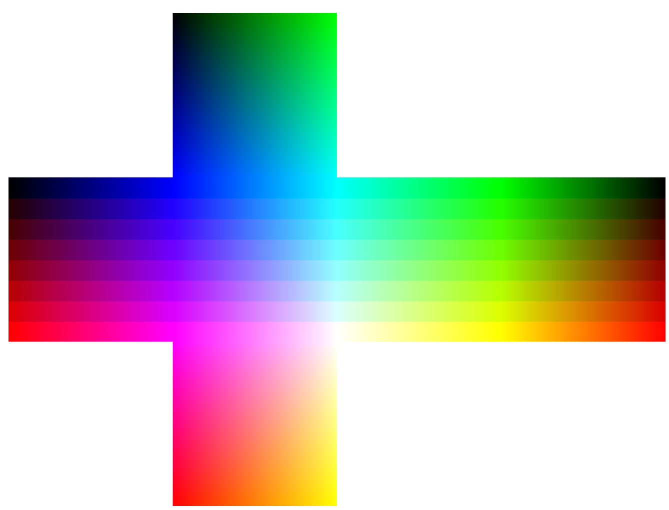

Fig. 10 RGB Colour Space in Variable Colour Depth of 3-8-8.

4.3.2 Variable Colour Depth LUT (VCD LUT)

A look-up table (LUT) is an array which holds reference values in a pre-defined order. In FCCF, a look-up table is used for fast colour classification. By accessing a look-up table, and indexing it with a given RGB value, the colour classification is determined. In the look-up table building process, a look-up table is constructed by using all possible colour values that could be represented in the colour space. In the RGB colour space, each RGB combination is assigned one of the possible pre-determined colour classes. Therefore, given a colour pixel value (e.g. in RGB), the table is used to determine its colour class. A single look-up table covers the whole colour space, with an assigned colour classifier for each point in the space. An advantage of the standard indexed LUT is that classification can be performed very fast. The Variable Colour Depth LUT (VCD LUT) differs from the standard indexed LUT as it allows for varying bit numbers in representing each colour component (e.g. 3-bit for red, 8-bits for green and 8-bits for blue). This requires a separate LUT for each pre-defined colour class, but even so, the size of each VCD LUT is very small as it only requires to hold a truth value for referencing a colour value. Altogether, a collection of VCD LUT tables would still be smaller than one standard indexed LUT. To illustrate this, figure 11 shows some comparisons between standard indexed LUT and VCD LUT in terms of LUT structure and memory requirement.

Each entry in the VCD LUT requires only a single bit, whereas the standard indexed LUT requires a collection of bits that is sufficient to represent the entire number of colour classes (i.e. a byte for less than 256 colour classes). VCD LUT is suitable for the Variable Colour Depth technique because each LUT is constructed adaptively with varying bit number requirements for each colour component. The colour depth requirements are optimised for each colour classifier. The VCD LUT also renders itself suitable to parallel processing by having exclusive VCD LUT per colour class that could be assigned to an independent process.

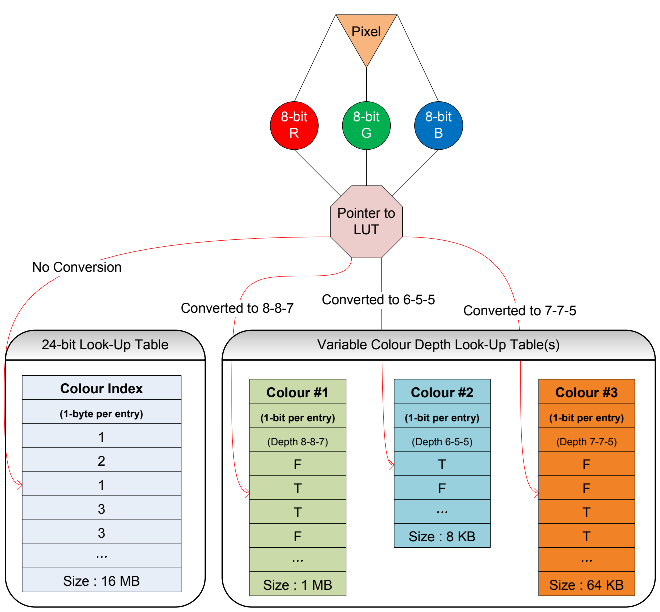

Fig. 11 Comparisons between Standard Indexed LUT and VCD LUT.

Table 2 Comparisons of Colour Classification Result between Indexed and VCD LUT

Colour Attributes	Standard Indexed LUT	VCD LUT		
Red/Green/Blue	Name of Colour	Blue	Violet	Pink
69/5/225	Blue	TRUE	FALSE	FALSE
103/4/217	Blue	TRUE	TRUE	FALSE
133/3/215	Violet	FALSE	TRUE	FALSE
156/1/209	Violet	FALSE	TRUE	TRUE
174/2/207	Pink	FALSE	TRUE	TRUE
196/2/201	Pink	FALSE	FALSE	TRUE
232/1/195	Pink	FALSE	FALSE	TRUE
255/255/255	Undefined	FALSE	FALSE	FALSE

4.3.3 Colour Ambiguity and LUT

When a colour value is classified according to multiple classifiers, there is always colour classification ambiguity. Figure 12 shows a situation where colour ambiguity arises due to multiple classifications. The ambiguity may resolved by classifying using neighbouring colour values as cue when ambiguity exists. In the standard indexed LUT however, indications of ambiguities in the colour classification is lost during the construction of LUT because only a single entry of colour classification value is possible. On the other hand, in the VCD LUT, the evidence of ambiguity is indirectly available as multiple LUTs are utilised, indicating different colour classifications for the same colour value. Table 2 illustrates a situation when colour classification ambiguity is present. It also shows how colour classification ambiguity is treated in both standard indexed LUT and VCD LUTs.

Fig. 12 Shades of Colour between Blue and Pink.

4.4 VCD LUT Building Algorithm

Algorithm 2 builds an LUT for each target colour t, scanning every possible colour values. If the colour value is classified as a target colour, a bit is set in the LUT at a calculated location to indicate membership to that target colour.

Algorithm 2: Variable Colour Depth LUT Build Algorithm

foreach $t \leftarrow$ *every target n colours* **do**
  for $R \leftarrow 0$ **to** $2^{\mathrm{ddrn}} - 1$ **do** `// Every possible Red values`
    for $G \leftarrow 0$ **to** $2^{\mathrm{ddgn}} - 1$ **do** `// Every possible Green values`
      for $B \leftarrow 0$ **to** $2^{\mathrm{ddbn}} - 1$ **do** `// Every possible Blue values`
        if *colour value is classified as colour t* **then**
          $L = (R \ll (ddgn+ddbn)) + (G \ll ddbn) + B$;
          $LB = L \gg (b\ \log_2)$;
          $Lb = 1 \ll (L \bmod b)$;
          $LUT[t][LB] = LUT[t][LB] \bigcup Lb$;

dsr, dsg, dsb are the colour depth values of each colour channel in source image
$ddrn$, $ddgn$ and $ddbn$ are the colour depth values of each colour channel in each target colour LUT
LB is an index to the LUT that corresponds to the target colour
b is the size of data type of LUT optimised for the system architecture (e.g. 8 for byte-aligned, and 16 for word-aligned)

4.5 VCD LUT Query Algorithm

Given the source colour value of a pixel, Algorithm 3 searches the LUTs of each possible target colours t to classify its colour. The corresponding LUT location for each target colour depth is calculated and a bit mask AND operation is used to extract the target query bit. Note that the LUT location calculation requires shift-left as well as shift-right operations in order to discard excessive bits in the source colour value.

4.6 General Variable Colour Depth - FCCF System Architecture

Figure 13 illustrates how the VCD LUT is constructed per colour classifier. All possible colour values in a given

Algorithm 3: Variable Colour Depth LUT Query Algorithm

foreach $t \leftarrow$ *every target n colours* **do**
  R = Red component value of target pixel;
  G = Green component value of target pixel;
  B = Blue component value of target pixel;
  $L = ((R \gg (dsr\text{-}ddrn)) \ll (ddgn+ddbn)) + ((G \gg (dsg\text{-}ddgn)) \ll ddbn) + (B \gg (dsb\text{-}ddbn))$;
  $LB = L \gg (b\ \log_2)$;
  $Lb = 1 \ll (L \bmod b)$;
  if $(\mathrm{LUT[t][LB]} \bigcap \mathrm{Lb}) \neq 0$ **then**
    Given pixel is qualified for target colour t

Variable Colour Depth is tested in the FCCF process to construct the VCD LUT.
Figure 14 shows how the VCD LUT is used in a real-time environment. The acquired colour pixel in the scene is converted into a separate Variable Colour Depth representation for each colour classifier. Next, each colour classifier accesses its own VCD LUT to determine the pixel's colour class. If there is only a single target colour object to track, then only one colour classifier is required to test, along with its own VCD LUT.

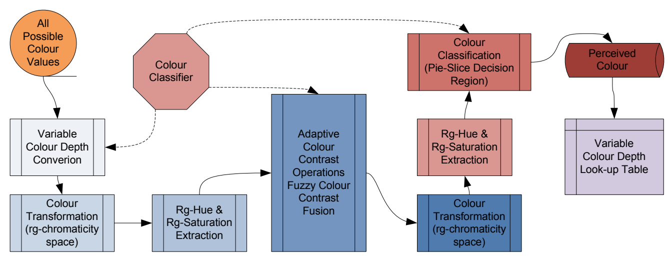

Fig. 13 Variable Colour Depth Look-Up Table Construction Architecture

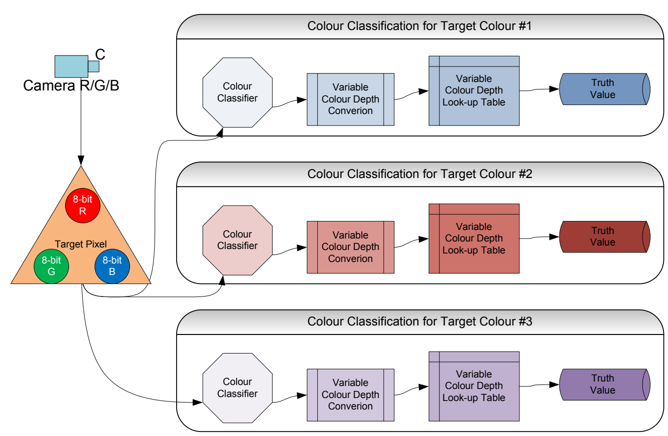

Fig. 14 Variable Colour Depth Look-Up Table for Real-Time Processing

5 Fuzzy-Genetic Colour Classifier Search

5.1 Motivation

Genetic Algorithm is considered to be a non-exhaustive search technique suitable for finding optimal or near optimal solutions for any given problem domain. The Fuzzy-Genetic Colour Calibration experiments designed in this research aims to find optimal parameter sets for accurate colour classification. However, the search space to be explored in finding an optimal colour classifier in FCCF is vast due to the real number valued-parameters of classification, such as contrast angles and radii to mention a few. Although the radii may be discretised to define the search space, angles are difficult to quantise due to the inherent characteristic of the arc length that is a product of angle and radius as shown in figure 15 and equation 4. Therefore, when the angles are discretised, for example, covering an area represented as A, b, and c in figure 15, the gradations, as a result of discretisation will eventually cause some inaccuracy. The inaccuracy of angle representations are represented as gray areas in b and c, and they also represent the region of errors. When the angle inaccuracy increases, this consequently enlarges the region of errors significantly. On the other hand, when the radius inaccuracy increases, the region of errors grows proportionally, but with lesser effect than the increase in angle inaccuracy.

$$L = \theta r \tag{4}$$

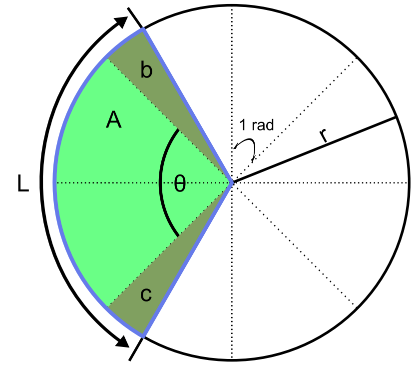

Fig. 15 The Shaded Pie-Slice, Covered by Arc L Represents The Area of Interest. Due to the Discretisation of the Angles, the Area of Interest Could Only be Approximated by a Smaller Pie-Slice, Covered by Angle θ. The Dotted Lines Indicate the Discretisation of Angles.

The colour classifier requires a large number of parameters to calibrate. This leads to a lot of difficulty in generating an optimal colour classifier automatically. There are 6 real number-valued parameters (classification angles, contrast angles and classification radii), 3 sets of classification operations and 3 sets of Variable Colour Depth subranges which all affects the result of colour classification independently. Genetic Algorithm offers to find a solution from the search space by performing mutation and crossover operations on the chromosomes. The algorithm may end up with a non-optimal solution. However, it is highly likely to return a more accurate set of colour classifier parameters than the manually calibrated ones.
Furthermore, it is also possible to feed a previously discovered solution set to the Genetic Algorithm repeatedly to allow it to evolve closer to the optimal colour classifier, however due to the size and complexity of the colour parameter optimisation problem, this approach is not exactly suitable to the problem at hand. It was empirically observed in this research that starting anew with a different random seed returns better results when the previous results was not satisfactory.

5.2 General Architecture

Figure 16 illustrates the overall architecture of Fuzzy-Genetic colour classifier search architecture. The Genetic Algorithm parameters and reference image are fed into the Genetic Algorithm to generate a colour classifier.

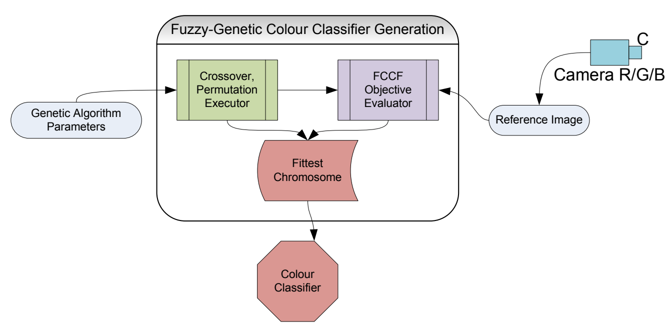

Fig. 16 Fuzzy-Genetic Colour Classifier Search Architecture

5.3 Chromosome Design

The chromosome defines the search space of the Genetic Algorithm. As discussed in the earlier sections, there is a total of 12 different parameters to construct a colour classifier; thus, there are also 12 different parameters to optimise. We designed the chromosome with a total size of 108 bits (figure 17). The chromosome design is

largely divided into two sections. The front 60 bits correspond to angles and radii, while the last 48 bits correspond to contrast rules and colour depth values. We divided it into two because the latter 48 bits could be disabled when using the guided search strategy, which is explained shortly thereafter.
For the 60 bits front part of the chromosome, 4 angles and 2 radii are assigned each with a 10-bit range sub-chromosome. Each sub-chromosome could represent 2^{10} values, which ranges from 0 to 1 representing the radius. This is about 0.001 incremental steps for the radius. If the angle parameter is using the full 0 to 360 range, the increments are about 0.35 degree. However, this can be sliced more narrowly if we limit the search range for the angles. In the experiments, we limited the search range for the angle up to 180 degrees. This allows incremental steps of about 0.176 degree.
The last 48 bits of the chromosome, divides into a length of 8-bits for representing the integral values of the contrast rules and colour depth. Since 8 bits are somewhat larger than the required 7 states of colour contrast rules and 4 levels of colour depths, it allows for larger variances of crossover and mutation operations.

Parameter	Range	Length	Incremental Steps
Min Angle	Pivot -30° to -90°	10 bits	0.058 ~ 0.176
Max Angle	Pivot +30° to +90°	10 bits	0.058 ~ 0.176
Min Radius	0 ~ 1	10 bits	0.001
Max Radius	0 ~ 1	10 bits	0.001
Min Contrast Angle	Pivot -30° to -90°	10 bits	0.058 ~ 0.176
Max Contrast Angle	Pivot +30° to +90°	10 bits	0.058 ~ 0.176
Red Contrast Rule	-3.00 to 3.99	8 bits	0.027
Green Contrast Rule	-3.00 to 3.99	8 bits	0.027
Blue Contrast Rule	-3.00 to 3.99	8 bits	0.027
Red Colour Depth	5 to 8.99	8 bits	0.015
Green Colour Depth	5 to 8.99	8 bits	0.015
Blue Colour Depth	5 to 8.99	8 bits	0.015

Fig. 17 Chromosome Design

5.4 Fitness Function

The fitness function, also known as the objective function gives fitness values that represent the ranks of chromosomes evaluated during the optimisation process. The fitness of the chromosomes tells exactly how close is the generated solution to the goal is. The fitness function used for the Fuzzy-Genetic colour classifier search employs the colour classification scoring formula proposed in [6] (Algorithm 4) The scoring function awards 1.0 for a perfect colour classification and is totally independent of the structure and/or number of colour classifier parameters. This is a very desirable feature for a fitness function.

6 Experiments and Analysis

The experiments were performed on the same robot soccer test bed used in [6] for comparison purposes. However, the calibration set up is non-typical, as it is plagued with spatially varying illumination intensities, with 6 target colours, represented by 40 colour patches, strategically positioned to be exposed under different illumination conditions (i.e. dim, dark, bright). The focus of the experiments is to compare colour classification results when the full colour depth (24 bits) is used vs. Variable Colour Depth with Genetic Algorithm-calibrated colour classification parameters.

6.1 Assessment Method

The classification performance is gauged based on a scoring formula proposed in [6]. The formula takes into account the number of true positives, false positives, as well as the area of the target colour objects, and has proven to identify the superior colour contrast rule combination. Colour Contrast Rule Extraction (CCRE) (Algorithm 4) describes how the scoring formula is constructed.

Algorithm 4: *CCRE(image, targetbounds)*, Scoring Formula [6].

1. For each target object calculate an individual score: $score_i = \frac{hits_i}{area_i}$
   if $hits_i < \frac{1}{n}\, area_i$ then $score_i = 0$; where $n = 4$ (empirically found)
2. Calculate average score:
   $avescore = \frac{\sum_{i=1}^{ntargets} score_i}{ntargets}$; where: $ntargets$ is the number of targets.
3. Calculate a general score:
   $genscore = \frac{Totalhits}{Totalhits+Totalmisses}$
4. Final score:
   $finalscore = (0.6\, avescore) + (0.4\, genscore)$
5. Adjust score to account for misclassifications:
   $if(Totalhits > 0)$
   $finalscore = finalscore - (\frac{Totalmisses}{Totalhits})$

6.2 Variable Colour Depth with CCRE

This experiment tests the effectiveness of using VCD by applying an extended version of CCRE which includes a new parameter (i.e. variable colour depth) for evaluating the performance. If the experiments give us more accurate classification results from a colour depth representation less than the full 24-bit colour depth, then that would suggest that some colour components are less important than others for classifying target colours. Consequently, this would also prove that FCCF have the capability of compensating for loss of colour component resolution.

6.2.1 Search Strategy

In order to find alternative colour depth values, a brute force search method was employed. Each candidate target colour classification holds the base parameters (i.e. angles, radii) retrieved from the previous results [6]. These parameters were kept constant, while the colour contrast rules and colour depth values were permutated to find the most accurate colour classification parameters. We limit the colour depth search space to only 64 possibilities. In effect, we considered only from 5-bits to 8-bits, per colour component. It is deemed that colour depth representations less than 5-bits per component wouldn't provide enough resolution for effective segmentation. It is also too costly to search to consider all 8^3 possibilities. For each target colour depth, there are 343 different colour contrast rules to test; therefore, there will be 21,952 colour classification tests required per target colour.

6.2.2 Colour Classification Results of Full 24-bit Colour Depth vs. Variable Colour Depth

We employed the same colour classification definition for the 6 target colours tested in [6] for direct comparison of algorithm performances. The previous research used the full 24-bit colour depth, and utilised an algorithm for automatic extraction of the colour contrast rules. Table 4 shows comparisons between the best scores from the previous research and this research.
As observed from the table, it is clear that the application of the Variable Colour Depth approach resulted to better scores than the full 24-bit colour depth in all 6 target colour objects. It is evident that the misclassifications have been significantly reduced down for all target colours.

6.2.3 Colour Contrast Rule Clustering

The clustering of the best set of colour contrast rules for each possible colour depth value, and for each target colour is shown in figure 18. The colour depth values were varied from 5 to 8, considering a total of 64 possible permutations inspected for each target colour. The actual numeric figures are presented in Table 3.
On the other hand, figure 19 shows a mapping of the best colour contrast rule combinations for the optimal colour depth values for each target colour. It also reflects the number of occurrences of the same colour contrast rule combination for each target colour. For example, *Light Blue (0/3/1)*59* from figure 19 shows indicates colour contrast rule of: no contrast on red, enhance three times on green and enhance once on blue component as, best colour contrast rule for 59 colour depth combinations out of 64. As indicated in the figures, the optimal colour contrast rule combinations for Light Blue, Pink and Purple adhere with the majority of colour contrast rule combinations found at different colour depth values. There are some observable patterns in the rule combinations as well. For instance, Green always requires the colour contrast degrade operation on the Blue channel. (Table 3) In contrast, there is no observable pattern on Yellow's contrast rule combinations; they are scattered all over the rule space.

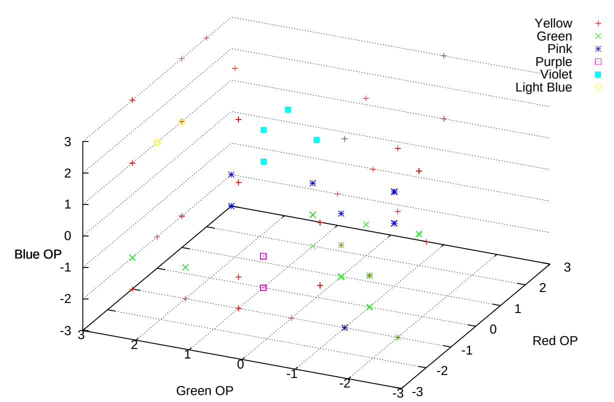

Fig. 18 Mapping of all the Best Colour Contrast Rule Combinations for all Colour Depth Values and for each Target Colour

6.2.4 Colour Pixel Clustering

Figure 20 shows 2 sets of data collected from 8 Light Blue objects under different illumination intensities. These data plots represent the colour pixels of the objects in

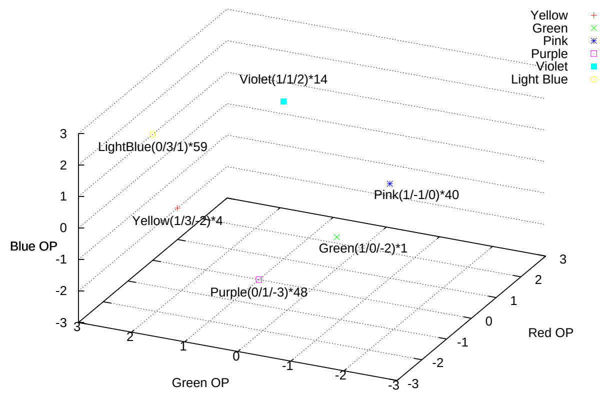

Fig. 19 Mapping of the Best Colour Contrast Rule Combinations for the Optimal Colour Depths for each Target Colour. Positive Number Indicates Contrast Enhancement and Level of Contrast Application; 0 for No Operation, while a Negative Number Denotes Contrast Degradation. * n Indicates Number of Occurrences.

Table 3 Colour Contrast Rule Distribution in 64 Possible Permutations

Levels		Yellow Red	Yellow Green	Yellow Blue	Green Red	Green Green	Green Blue	Pink Red	Pink Green	Pink Blue
	-3	0	1	7	0	0	29	0	0	9
Degrade	-2	9	7	11	0	1	35	0	0	2
	-1	29	21	4	11	23	0	4	52	13
Neutral	0	4	3	4	13	21	0	0	2	40
	1	10	14	19	19	9	0	50	3	0
Enhance	2	6	2	1	21	6	0	3	0	0
	3	6	16	18	0	4	0	7	7	0
Levels		**Purple Red**	**Purple Green**	**Purple Blue**	**Violet Red**	**Violet Green**	**Violet Blue**	**Light Blue Red**	**Light Blue Green**	**Light Blue Blue**
	-3	0	0	48	0	0	0	0	0	0
Degrade	-2	0	0	16	0	0	0	0	0	0
	-1	0	0	0	0	0	0	0	0	0
Neutral	0	64	0	0	50	23	0	59	0	0
	1	0	64	0	14	41	26	5	0	64
Enhance	2	0	0	0	0	0	38	0	0	0
	3	0	0	0	0	0	0	0	64	0

the rg-Hue vs. rg-Saturation chart. The first set was generated using a colour depth of 8-8-8 bits (8-bits for the red component, 8-bits for the green component and 8-bits for the blue component) denoted by '+', while the second set was generated using a colour depth of 5-5-6 bits denoted by 'x'. Most of the pixels are clustered within the minimum and maximum pie-slice decision angles of 137.124 to 162.792, and radii between 0 to 0.1. It can also be observed that the lower colour depth pixels denoted by 'x' relatively spread evenly across the bounding angle's due to loss of colour resolution.

Figure 21 is closely related to Fig. 20 as it shows the clustering of pixels of the same target objects in Fig. 20 enlarged at pie-slice decision angles, with the same illumination intensities and colour depths, except that the FCCF algorithm was applied. 2 sets of data were collected. The first set was generated using a colour depth of 8-8-8 bits (denoted by '+', while the second set was generated using a colour depth of 5-5-6 bits denoted by 'x'.

For the 2 data sets, the following colour contrast rules were applied: Red channel: no operation; Green channel: enhance 3 times; Blue channel: enhance 1 time. When FCCF was applied, it can be observed that the colour pixels close to the maximum angle, 162.792 were pulled inside the pie-slice decision region and were spread toward covering a broader radius. In effect, the lower colour depth pixels are now clustered with some regularity.

Further experiments show evidences that FCCF improves colour classification of other target colours by influencing the formation of the colour pixels within the confines of the pie-slice decision region.

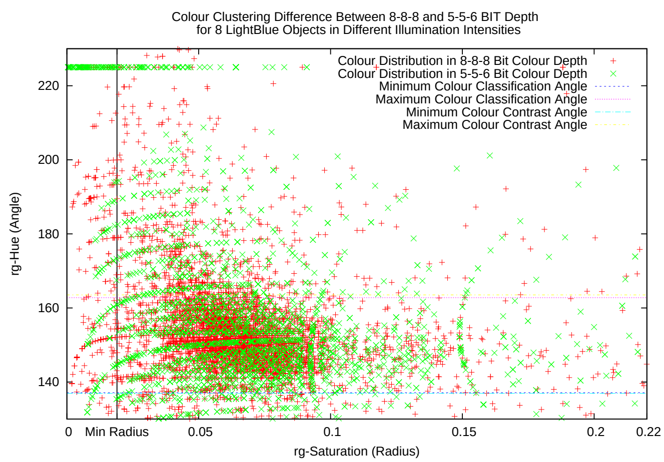

Fig. 20 Colour Pixel Clustering on rg-Hue / rg-Saturation Chart for Light Blue Objects

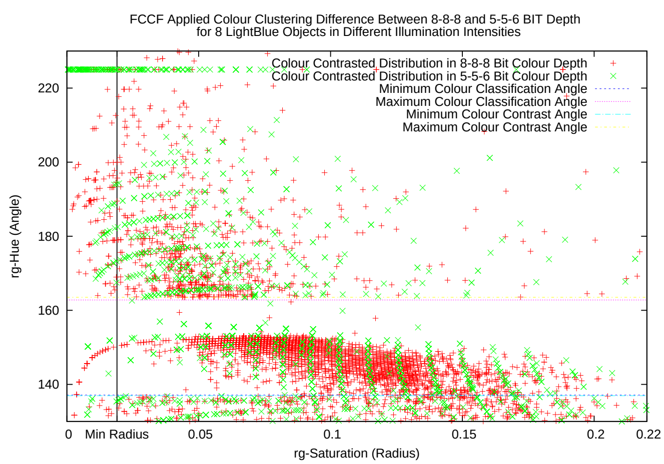

Fig. 21 Colour Pixel Clustering on rg-Hue / rg-Saturation Chart for Light Blue Objects with FCCF

6.3 Fuzzy-Genetic Colour Calibration

6.3.1 Angle Calculation and Range Limiting

The search space for both the pie-slice decision angles and contrast angles ranges from 0 to 360 degrees. However, it is very unlikely that the colour to be classified requires larger than 180 degrees of angle in the pie-slice decision range. The angle range is either supplied by previously discovered solution, or decided upon the extracted minimum and maximum angles taken from the actual colour pixels of the target objects. A tolerance between 30 to 60 degrees is applied to widen the search range, with the limit of less than 180 degrees for the scope of searching.

7 Synthesis

Table 4 summarizes the results of comparisons between three architectures: (1) CCRE [6], the brute-force approach to finding the best colour contrast rules, in the full 24-bit colour depth, (2) VCD + CCRE, the proposed brute-force colour depth and colour contrast rules extraction and (3) Fuzzy-Genetic + VCD, the proposed GA-assisted FCCF using VCD. The results show that for all target colours, the Fuzzy-Genetic + VCD returns the most accurate classifier. In addition, it comes second in terms of the least amount of memory storage space requirements.

8 Conclusions

We have presented a new approach to improving colour discriminability down to the bit level (colour depth). We have introduced the Variable Colour Depth algorithm, along with accompanying techniques for building and searching a VCD LUT. We have fused the VCD algorithm with FCCF and extended CCRE. We tested it against the FCCF and CCRE combination [6] to prove its efficiency. The results of experiments show that there is an increases of 6.9% in terms of overall colour classification accuracy and reduced memory space requirement by 91.99%. Lastly, we incorporated the HAGA algorithm for automating fully the calibrating parameters, and improving the overall colour classification accuracy by 17.63%.

References

1. Chalupa, L.M., Werner, J.S., eds.: The visual neurosciences. Volume 2. MIT Press (2004)
2. Reyes, N.H., Dadios, P.E.: Dynamic color object recognition using fuzzy logic. Journal of Advanced Computational Intelligence and Intelligent Informatics **8** (2004) 29–38
3. Ebner, M.: Color Consancy. Wiley (2007)
4. Swain, M., Ballard, D.: Indexing via color histograms. Computer Vision, 1990. Proceedings, Third International Conference on (Dec 1990) 390–393
5. Browning, B., Veloso, M.: Real-time, adaptive color-based robot vision. In: Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on. (2005) 3871–3876
6. Playne, D.P., Mehta, V.D., Reyes, N.H., Barczak, A.L.C.: Hybrid fuzzy colour processing and learning. In: ICONIP 2007, Part II, LNCS 4985. (2008) 386 – 395
7. Heinemann, P., Sehnke, F., Streichert, F., Zell, A.: Towards a calibration-free robot: The act algorithm for automatic online color training. In Lakemeyer, G and Sklar, E and Sorrenti, DG and Takahashi, T, ed.: RoboCup 2006: Robot Soccer World Cup X. Volume 4434 of Lecture Notes In Artificial Intelligence., Heidelberger Platz 3, D-14197 Berlin, Germany, Springer-Verlag Berlin (2007) 363–370 10Th International Robocup Symposium, Bremen, Germany, Jun 19-20, 2006.
8. Guerrero, P., Ruiz-del Solar, J., Fredes, J., Palma-Amestoy, R.: Automatic on-line color calibration using class-relative color spaces. In Visser, U And Ribeiro, F And Ohashi, T And Dellaert, F, ed.: Robocup 2007: Robot Soccer World Cup Xi. Volume 5001 of Lecture Notes In Computer Science., Heidelberger Platz 3, D-14197 Berlin, Germany, Springer-Verlag Berlin (2008) 246–253 11th RoboCup International Symposium, Atlanta, GA, JUL 09-10, 2007.
9. Hayashi, Y., Fujiyoshi, H.: Mean-shift-based color tracking in illuminance change. In Visser, U And Ribeiro, F And Ohashi, T And Dellaert, F, ed.: Robocup 2007: Robot Soccer World Cup Xi. Volume 5001 of Lecture Notes In Computer Science., Heidelberger Platz 3, D-14197 Berlin, Germany, Springer-Verlag Berlin (2008) 302–311 11th RoboCup International Symposium, Atlanta, GA, JUL 09-10, 2007.
10. Reyes, N.H., Messom, C.: Identifying colour objects with fuzzy colour contrast fusion. In: 3rd International Conference on Computational Intelligence, Robotics and Autonomous Systems, and FIRA RoboWorld Congress. (2005)
11. Stachowicz, M.S., Lemke, D.: Color recognition. In: Information Technology Interfaces, 2000. ITI 2000. Proceedings of the 22nd International Conference on. (2000) 329–334
12. Kim, D.Y., Park, H.K., Chung, M.J.: A robust and fast color-extracting using a look up table. In: Proceedings of the Fourth International Symposium on Artificial Life and Robotics. (1999) 650–653
13. Dong, L., Ogunbona, P., Li, W., Yu, G., Fan, L., Zheng, G.: A fast algorithm for color image segmentation. In: Proceedings of the First International Conference on Innovative Computing, Information and Control-Volume 2. (2006) 685–688
14. Kim, J.H., Seow, K.T.: Soccer Robotics. Springer (2004)
15. Takahashi, Y., Nowak, W., Wisspeintner, T.: Adaptive recognition of color-coded objects in indoor and outdoor environments. In Visser, U And Ribeiro, F And Ohashi, T And Dellaert, F, ed.: Robocup 2007: Robot Soccer World Cup Xi. Volume 5001 of Lecture Notes In Computer Science., Heidelberger Platz 3, D-14197 Berlin, Germany, Springer-Verlag Berlin (2008) 65–76 11th RoboCup International Symposium, Atlanta, GA, JUL 09-10, 2007.
16. Kashanipour, A., Milani, N.S., Kashanipour, A.R., Eghrary, H.H.: Robust color classification using fuzzy rule-based particle swarm optimization. In Li, D And Deng, G, ed.: Cisp 2008: First International Congress On Image And Signal Processing, Vol 2, Proceedings, 10662 Los Vaqueros Circle, Po

Table 4 Colour Classification Results in Comparisons

Result Set	Colour	Depth [R][G][B]	Angle		Radius		Contrast Angle		Contrast Rule [R][G][B]	Score	Hits	Misses	LUT Size	Score Improvement
CCRE		[8][8][8]	43.99	46.48	0.00	1.00	41.83	47.81	[3][1][-2]	0.648530	2104	68	2048KB	
VCD + CCRE	Yellow	[7][8][6]	43.99	46.48	0.00	1.00	41.83	47.81	[1][3][-2]	0.655979	2261	96	256KB	1.15%
Fuzzy-Genetic		[7][6][8]	32.18	52.42	0.03	0.60	15.70	67.14	[-2][-1][-1]	0.675945	2355	89	256KB	4.23%
CCRE		[8][8][8]	45.58	96.66	0.05	1.00	45.29	96.06	[0][-1][-3]	0.552422	3313	383	2048KB	
VCD + CCRE	Green	[6][5][8]	45.58	96.66	0.05	1.00	45.29	96.06	[1][0][-2]	0.639059	3137	127	64KB	15.68%
Fuzzy-Genetic		[8][8][5]	44.53	101.23	0.04	0.12	37.95	121.46	[0][-1][-2]	0.695366	3637	128	256KB	25.88%
CCRE		[8][8][8]	314.42	327.28	0.15	1.00	275.26	331.88	[1][-1][0]	0.586446	1714	99	2048KB	
VCD + CCRE	Pink	[7][8][7]	314.42	327.28	0.15	1.00	275.26	331.88	[1][-1][0]	0.622773	1679	46	512KB	6.19%
Fuzzy-Genetic		[8][8][7]	283.44	335.43	0.16	0.65	280.51	335.37	[2][-2][0]	0.705861	2223	50	1024KB	20.36%
CCRE		[8][8][8]	286.52	307.86	0.13	1.00	285.01	308.56	[0][1][-3]	0.572888	2777	314	2048KB	
VCD + CCRE	Purple	[6][7][7]	286.52	307.86	0.13	1.00	285.01	308.56	[0][1][-3]	0.576178	2782	309	128KB	0.57%
Fuzzy-Genetic		[5][6][5]	275.92	315.25	0.10	0.21	275.46	315.18	[0][0][-3]	0.624469	3246	373	8KB	9.00%
CCRE		[8][8][8]	232.34	282.28	0.04	1.00	228.31	293.36	[1][1][2]	0.526654	2535	497	2048KB	
VCD + CCRE	Violet	[5][7][5]	232.34	282.28	0.04	1.00	228.31	293.36	[0][1][1]	0.602979	1802	101	16KB	14.49%
Fuzzy-Genetic		[7][8][8]	215.80	262.42	0.07	0.51	215.52	289.69	[0][0][2]	0.72671	2312	43	1024KB	37.99%
CCRE		[8][8][8]	137.12	162.79	0.02	1.00	136.94	163.51	[0][3][1]	0.668808	2758	68	2048KB	
VCD + CCRE	Light Blue	[5][5][6]	137.12	162.79	0.02	1.00	136.94	163.51	[0][3][1]	0.690887	2786	30	8KB	3.30%
Fuzzy-Genetic		[8][6][6]	142.97	189.69	0.07	0.17	130.30	189.92	[-1][0][1]	0.724412	3016	17	128KB	8.31%

Box 3014, Los Alamitos, Ca 90720-1264 Usa, Tianjin Univ Technol, Ieee Computer Soc (2008) 110–114 1St International Congress On Image And Signal Processing, Sanya, Peoples R China, May 27-30, 2008.

17. Hildebrand, L., Fathi, M.: Knowledge-based fuzzy color processing. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on **34**(4) (Nov. 2004) 499–505
18. Reyes, N.H., Barczak, A.L., Messom, C.: Fast colour classification for real-time colour object identification: Adaboost training of classifiers. In: In Proceedings of the Third International Conference on Automonous Robots and Agents. (2006) 611–616
19. Holst, G.C.: CCD Arrays, cameras, and displays. JCD Publisising and Spie Optical Engineering Press (1996)