Variable Colour Depth Look-Up Table Based on Fuzzy Colour Processing

Heesang Shin and Napoleon H. Reyes

Institute of Information and Mathematical Sciences, Massey University, Auckland
New Zealand
shinheesang@gmail.com, n.h.reyes@massey.ac.nz
http://www.massey.ac.nz/~nhreyes

Abstract. This paper presents an application of a Fuzzy Colour Contrast Fusion (FFCF) algorithm for reduced colour depth representation of a colour image while maintaining efficient colour sensitivity that suffices for accurate real-time colour-based object recognition. We investigate the effects of applying fuzzy colour contrast rules to varying colour depth as we extract the optimal rule combination. The experiments were performed using the robot soccer game set-up with spatially varying illumination intensities on the scene. Interestingly, our results show that for all cases, variable colour depth reduction actually improve colour classification via a pie-slice technique, in a modified rg-chromaticity colour space. For 6 different colours, the algorithm was able to yield 6.5% higher overall accuracy with 92% reduction of LUT storage space than the full colour depth LUT.

1 Introduction

The compendium of algorithms presented in this paper work in cohesion, with the main purpose of more accurately classifying colour pixels depicting moving target objects under varying illumination intensities, while cutting-down memory space by finding the most efficient colour depth for each target colour. The approach described here is suitable for vision systems that employ region segmentation by colour. Such systems are deemed crucial in real-time mobile robot applications [1] like Robocup [2] and FIRA [3].

Altogether, the beauty of the techniques is that all possible colour combinations can be pre-processed, classified and stored into a variable colour-depth look-up table for real-time object tracking use. The system is calibrated by using exemplars of the colour objects in different illumination settings in an indoor environment. These exemplars are selected based on human visual judgment of which region is relatively dim, dark and bright. In turn, the gradient between these illumination conditions are automatically worked out by the algorithms. The algorithms work together to compensate for the effects of light intensity at every pixel of an object. That is, light intensity at each point in the image that is a product of the reflectance at the corresponding object point and the illumination intensity at the same location, apart from a constant factor contributed by the optical arrangement of the imaging device [4]. A fuzzy colour contrast fusion algorithm (FCCF) [5] is employed to perform colour correction via contrast enhance or degrade operations [5] on the RGB colour channels independently. This is handled by a synthesis of optimal colour contrast rules which are extracted automatically by ranking all 343 (7 operations per channel) possible rule combinations according to an algorithm presented in [6]. Previously, FCCF in full colour depth has been successfully applied to real-time colour-object recognition [5] [7], and also been tested previously to work with the YUV, HSI and rg-chromaticity spaces with improved colour classification results [7]. In this work we explore the efficacy of FCCF when the colour depth is reduced to save memory space for colour classification, allow for portability to vision systems with limited resources and parallel and distributed processing adaptability. We analyse the clustering of the combination of fuzzy colour contrast rules at differing amount of reductions in colour depth, in search for the optimal ones. The ultimate goal is to find the minimum colour depth for each of the colour channels, while maintaining at least the same level of accuracy as the full colour depth LUT. We performed the experiments on the same illumination set-up and target colours used in [6] for direct comparison, and the colour classification results attained by the variable colour depth LUT show superior performance.

2 The Algorithms

2.1 Colour Descriptors from rg-Chromaticity Colour Space

The colour descriptors were extracted from a modified rg-chromaticity colour space. The original rg-chromaticity space is already known to remove ambiguities due to illumination or surface pose[8]. On top of that, the colour descriptors we are using also lend themselves amenable for pie-slice colour classification [9].

rg-chromaticities: $r = \frac{R}{R+G+B}$, $g = \frac{G}{R+G+B}$

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

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

2.2 Fuzzy Colour Contrast Fusion (FCCF)

Chromaticity distortions are compensated for via a fusion of colour contrast operations employed through fuzzy rules. The inputs are the original colour tristimulus in RGB form, as well as the calculated rg-Hue and rg-Saturation values. With reference to the set of optimal colour contrast rules detailing the contrast operations, levels of operations (i.e. 1x, 2x, 3x), as well as the colour contrast constraint angles, the appropriate rules are applied independently on the colour channels. Consequently, FCCF produces the colour-corrected RGB values for more accurate colour classification.

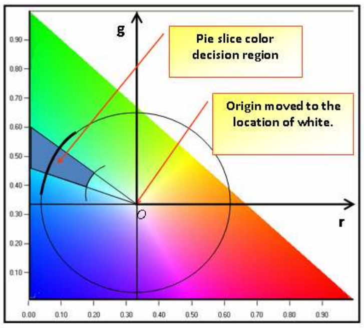

Fig. 1. Modified rg-chromaticity space and the pie-slice decision region

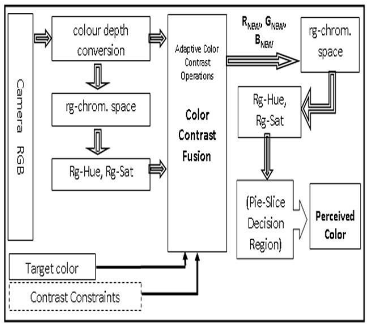

Fig. 2. FCCF with Reduced Colour Depth

Example 1. **Sample colour contrast rule**
If(pixel depicts LIGHTBLUE according to colour contrast constraint angles) Then apply colour contrast enhance on the Red channel 2 times, apply colour contrast degrade on the Green channel and apply colour contrast enhance on the Blue channel 3 times.

The fuzzy colour contrast rules employ contrast enhance and contrast degrade operations on the colour channels independently through Equations (1) and (2).

Contrast Enhance Operator:

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

Contrast Degrade Operator:

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

For each colour channel, there are 3 possible manipulations: enhance, degrade or no operation. Moreover, enhance and degrade operations could also be applied multiple times; that is, 1, 2, or 3 times. Therefore, for each colour channel, there are 7 possible combinations of operations. Altogether, for three colour channels, there are 343 possible rules. Note that the optimal rules for each target colour are extracted automatically through a scoring algorithm described in [6].

The following algorithms describe how to construct and search a variable colour depth Look-Up-Table.

Algorithm 1: Variable Colour Depth LUT build algorithm

```
foreach t ← every target n colours do
    for R ← 0 to 2^ddrn − 1 do                    // Every possible Red values
        for G ← 0 to 2^ddgn − 1 do                // Every possible Green values
            for B ← 0 to 2^ddbn − 1 do            // Every possible Blue values
                if colour value is classified as colour t then
                    L = (R≪(ddgn+ddbn))+(G≪ddbn)+B;
                    LB = L ≫ (b log2);
                    Lb = 1 ≪ (L mod b);
                    LUT[t][LB] = LUT[t][LB] ⋃ 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)

2.3 Variable Colour Depth LUT Build and Search

After the optimum set of colour contrast rules, pie-slice decision region parameters and optimum colour depth for each channel has been determined, the variable colour depth LUT is generated. Algorithm 1 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. Given the source colour value of a pixel, Algorithm 2 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.
There's a possibility that a single pixel could be classified as multiple colours, as returned by each LUT. As illustrated in Fig. 4, this could actually help resolve ambiguities when the object is under extreme illumination condition, the pixel may be classified as either pink, or blue or violet, but not yellow or green. With reference to the neighboring cluster of pixels, the pixel gets the same classification as the cluster. Moreover, the algorithms lend themselves amenable for parallel and distributed implementations since each LUT could be generated and searched independently of the others.

3 Experiments

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,

Algorithm 2: Variable Colour Depth LUT query algorithm

```
foreach t ← 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≫(dsr-ddrn))≪(ddgn+ddbn))+((G≫(dsg-
    ddgn))≪ddbn)+(B≫(dsb-ddbn));
    LB = L ≫ (b log2);
    Lb = 1 ≪ (L mod b);
    if (LUT[t][LB] ⋂ Lb) ≠ 0 then
        Given pixel is qualified for target colour t
```

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 LUT (24 bits) is used vs. variable colour depth LUTs. 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.

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

We employed the same colour classification definition for the 6 target object colours tested in [6] for direct comparison of algorithm performances. The previous research used a 24-bit colour depth LUT for each target colour, and utilised an algorithm for automatic extraction of the angles and radii values, and colour contrast rules. Table 1 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 LUT in all 6 target object colours. It is evident that the misclassifications have been significantly reduced down to at least 50% for all target colours.

3.2 Optimised Colour Contrast Rules

The clustering of the best set of colour contrast rules for each possible colour depth value, and for each target colour is shown in Fig. 3. 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 1.

On the other hand, Fig. 5 shows a mapping of the best colour contrast rule combinations for the optimal colour depth values for each target colour. It also

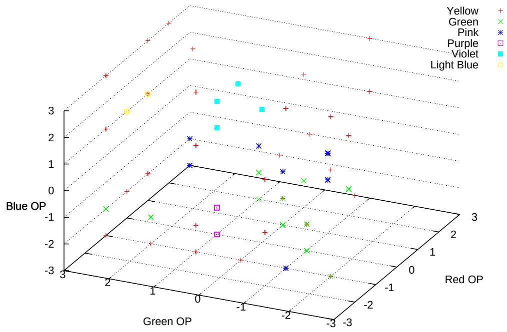

Fig. 3. Mapping of all the Best Colour Contrast Rule Combinations for all Colour Depth Values and for each Target Colour.

reflects the number of occurrences of the same colour contrast rule combination for each target colour. As indicated in the figures, the optimal colour contrast rule combinations for LightBlue, 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 color contrast degrade operation on the Blue channel. (Table 1) In contrast, there is no observable pattern on Yellow's contrast rule combinations; they are scattered all over the rule space. The scattering of the rules is due to the fact that the decision region for Yellow is defined with very narrow bounding angles. Consequently, these only proves that FCCF effectively employs colour contrast rules to compensate for the escaping pixels from the narrow pie-slice decision region for Yellow.

3.3 Colour Contrast Rules and Scores

Table 1 details the results of the experiments on optimised colour contrast rule extraction at varying colour depth. The table reflects the scores garnered by the rule combinations in classifying the 6 target colours (represented by 40 colour patches). The table indicates the colour depth and colour contrast operation used for each of the colour channel, the performance of the rule combination in terms of the number of hits, misclassifications, storage space requirement, improvement over the full-colour-depth LUT and rule combination's relative ranking. The best results show that FCCF increased colour classification even when there were lost bits in the colour depth.

3.4 Reductions in Memory Usage

As depicted in Table 1, the memory storage requirement for the reduced colour depth LUT optimized for colour classification varies between 16KB and 512KB.

Fig. 4. Colour Gradation From Blue to Pink

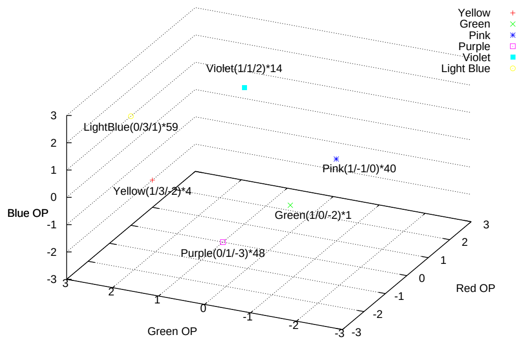

Fig. 5. Mapping of the Best Colour Contrast Rule Combinations for the Optimal Colour Depths for each Target Colour. Positive number indicates contrast enhancement and level of contrast application; 0 for No Operation, while a negative number denotes contrast degradation. * n indicates number of occurrences.

Altogether, for the classification of all 6 target colours, the total memory requirement for the variable colour depth LUT is only 984KB. In contrast, a 24-bit colour depth LUT requires 12MB for all 6 colours.

4 Conclusions and Future Work

The FCCF algorithm proves to be effective in compensating for the effects of reduction in colour depth by influencing the formation of colour clusters depicting colour objects within their pie-slice decision region in the colour space. This is particularly useful for colour classification. The experiments' results show that different colour contrast rule combinations were derived for different colour depth configurations. The best colour classification scores all came from the utilisation of variable colour depth LUTs, and therefore with less precise colour values. These colour depth values varied between each colour channel. In general, it can be deduced that the lower colour-depth values eliminated selectively some colour information that paved the way for more effective colour discrimination. This resulted to better scores, either by achieving higher true positives in the colour classification process, or reducing false positives. For all 6 target colour objects, memory consumption is reduced from 12MB down to only 984KB, or 92% reduction from the full 24 bit colour space. While achieving reduced memory consumption, on the average, the colour classification scores for all 6 colours increased from 0.592625 up to 0.631309, or with 6.5% overall improvement. All colour classification scores improved for each target colour, as compared to using the full 24-bit colour space. For future works, we intend to fuse Genetic Algorithms with FCCF in automatically extracting the optimal colour descriptors,

Table 1. Colour Contrast Rules and Scores

Colour	Angle Min, Max	Depth R , G ,B	Contrast Rule R ,G ,B	Score	Hits	Misses	LUT Size	Improvement Rate	Rank
		8, 8, 8	3, 1, -2	0.648530	2104	68	2048KB	0%	3
Yellow	43.992	7, 8, 6	1, 3, -2	0.655979	2261	96	256KB	1.149%	1
	46.476	6, 8, 6	-1, 1, -3	0.609815	2258	172	128KB	-5.97%	4
		7, 8, 8	1, 3, -2	0.655374	2261	97	1024KB	1.055%	2
		8, 8, 8	0, -1, -3	0.552422	3313	383	2048KB	0%	4
Green	45.576	6, 5, 8	1, 0, -2	0.639059	3137	127	64KB	15.683%	1
	96.66	6, 8, 6	2, 1, -2	0.587168	3266	288	128KB	6.29%	3
		7, 8, 8	0, -1, -3	0.615805	3239	206	1024KB	11.474%	2
		8, 8, 8	1, -1, 0	0.586446	1714	99	2048KB	0%	4
Pink	314.424	7, 8, 7	1, -1, 0	0.622773	1679	46	512KB	6.194%	1
	327.276	6, 8, 6	1, -1, 0	0.603303	1623	58	128KB	2.874%	3
		7, 8, 8	1, -1, 0	0.616230	1684	55	1024KB	5.079%	2
		8, 8, 8	0, 1, -3	0.572888	2777	314	2048KB	0%	2
Purple	286.524	6, 7, 7	0, 1, -3	0.576178	2782	309	128KB	0.574%	1
	307.859	6, 8, 6	0, 1, -3	0.565163	2729	313	128KB	-1.348%	4
		7, 8, 8	0, 1, -3	0.572094	2773	314	1024KB	-0.139%	3
		8, 8, 8	1, 1, 2	0.526654	2535	497	2048KB	0%	4
Violet	232.344	5, 7, 5	0, 1, 1	0.602979	1802	101	16KB	14.492%	1
	282.276	6, 8, 6	1, 1, 2	0.545970	2502	442	128KB	3.668%	2
		7, 8, 8	0, 0, 2	0.529645	2400	425	1024KB	0.568%	3
		8, 8, 8	0, 3, 1	0.668808	2758	68	2048KB	0%	4
Light Blue	137.124	5, 5, 6	0, 3, 1	0.690887	2786	30	8KB	3.301%	1
	162.792	6, 8, 6	0, 3, 1	0.671966	2703	48	128KB	0.472%	2
		7, 8, 8	0, 3, 1	0.671255	2758	63	1024KB	0.366%	3

fuzzy rules and colour depth values for each colour channel to further improve the colour calibration process.

References

1. Bruce, J., Balch, T., Veloso, M.: Fast and inexpensive color image segmentation for interactive robots. In: In Proceedings of IROS-2000. (2000) 2061–2066
2. Kitano, H., M Asada, Y.K., Noda, I., Osawa, E., Matsubara, H.: Robocup: A challenging problem for ai. AI Magazine **18** (1997) 73–85
3. Kim, J.H., Seow, K.T.: Soccer Robotics. Springer (2004)
4. Horn, B.K.P.: Determining lightness from an image. Computer Graphics and Image Processing **3** (1974) 277–299
5. Reyes, N.H., Dadios, P.E.: Dynamic color object recognition using fuzzy logic. Journal of Advanced Computational Intelligence and Intelligent Informatics **8** 29–38
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. 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)
8. Finlayson, G.D., Schiele, B., Crowley, J.L.: ”using color for image indexing. In: Proceedings of Challenge of Image Retrieval. (1998)
9. Thomas, P., Stonier, R., Wolfs, P.: Robustness of color detection for robot soccer. In: Proceedings of the Seventh International Conference on Control, Automation, Robotics and Vision. (2002) 1245–1249