

Methods For The Automatic Alignment Of Colour Histograms

Christopher Rohan Senanayake

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Engineering Doctorate
of
University College London.

Department of Computer Science
University College London

2009

I, Christopher Rohan Senanayake, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

Colour provides important information in many image processing tasks such as object identification and tracking. Different images of the same object frequently yield different colour values due to undesired variations in lighting and the camera. In practice, controlling the source of these fluctuations is difficult, uneconomical or even impossible in a particular imaging environment. This thesis is concerned with the question of how to best align the corresponding clusters of colour histograms to reduce or remove the effect of these undesired variations.

We introduce feature based histogram alignment (FBHA) algorithms that enable flexible alignment transformations to be applied. The FBHA approach has three steps, 1) feature detection in the colour histograms, 2) feature association and 3) feature alignment. We investigate the choices for these three steps on two colour databases : 1) a structured and labeled database of RGB imagery acquired under controlled camera, lighting and object variation and 2) grey-level video streams from an industrial inspection application. The design and acquisition of the RGB image and grey-level video databases are a key contribution of the thesis. The databases are used to quantitatively compare the FBHA approach against existing methodologies and show it to be effective. FBHA is intended to provide a generic method for aligning colour histograms, it only uses information from the histograms and therefore ignores spatial information in the image. Spatial information and other context sensitive cues are deliberately avoided to maintain the generic nature of the algorithm; by ignoring some of this important information we gain useful insights into the performance limits of a colour alignment algorithm that works from the colour histogram alone, this helps understand the limits of a generic approach to colour alignment.

Acknowledgements

This thesis is dedicated to my wife, Maria Teresa Aguilera-Peon

I embarked on the EngD program as a *mature student*, it felt like a risk to leave paid employment in a respectable job at the BBC Research and Development facility in Kingswood Surrey. Friends ask me if I am glad that I undertook the program, I tell them that I can't imagine giving up the lessons that I have learnt and the new perspectives I hold. The EngD period has quite simply been the biggest period of personal growth and change in my life, during this time I married an amazing woman and we have a beautiful daughter together, Daniela. We are currently expecting our second child and I have accepted an exciting position in a great technology company in the United States. This thesis owes a debt to a number of people who have either directly or indirectly helped. I briefly summarize their contribution here.

My supervisor Dr. Daniel Alexander is a mentor and friend. His constant corrective guidance has been a key factor in overcoming many of the unforeseen obstacles that occurred during the research and writing up periods of this thesis. His willingness to meet regularly and freely have made significant challenges surmountable, the exploration of many blind alleys was halted early and new directions were defined during these meetings. I would recommend any prospective doctoral student who gets the opportunity to work with Danny to grab the opportunity with both hands.

This thesis was based around a problem at Buhler Sortex as defined by Dr. Gabriel Hamid. Dr. Hamid has given his time freely and has made himself available to explain and define problems relevant to the industrially related portions of this thesis. I am grateful for the joint grant received from the ESPRC and Buhler Sortex. My second supervisor Dr. Lewis Griffin and Dr. Simon Prince both made invaluable contributions during the first and second year reviews, their suggestions for approaching the research process were insightful and timely. Members of the UCL computer vision laboratories have provided a useful sounding board and source of ideas and references. Dr. Martin Lillholm deserves special mention, he joined the lab and I was lucky enough to sit next to him. His deep knowledge and sharp insight have helped immensely; Martin is one of those rare guys who seems to have the correct answer to any question you ask him. Other members such as Tony Shepherd, Alistair Moore and Bill Crum all responded enthusiastically to my interrogations of their knowledge of various techniques. Other people at UCL have proved invaluablely helpful at times, Romy Beattie in particular is a star. She resolved numerous EngD administrative issues that were getting pushed between people's desks until she got on the case.

Nobody has been more connected to the highs and lows of the doctoral process than my wife. She has worked hard, often in high pressure situations to support our family through this time. Without her hard work and support, it would have not been financially possible to finish this thesis. Maria-Teresa has constantly encouraged me and never considered that we give up, even when precarious situations arose. Her love and sacrifice have been a driving force to finish this thesis. Daniela has taught me something every step of the way, dealing with doctors, hospital visits and weeks off at home due to bugs and sickness while managing EngD deadlines has taught me patience and calm. Seeing her joyful smile has kept me focussed on the big picture.

My parents have been fantastic, they have supported and encouraged me every step of the way. Becoming a parent myself has given me an even greater appreciation of the continuous effort, support and love that they have given me throughout my entire life. I will never forget the encouraging words of my mother, who has always told me from a young age that “you can achieve anything you put your mind to”. My father is significantly responsible for getting me interested in science, technology and computers. His love of tinkering with things and browsing a thousand different subjects rubbed off on me. I remember our first computer, a Sinclair Spectrum, 48K with rubber keys and a cassette tape to load programs - just great.

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 20 |
| 1.1 | Colour consistency in Computer Vision | 20 |
| 1.2 | Colour consistency in the food processing industry | 22 |
| 1.3 | The automatic histogram alignment problem | 23 |
| 1.4 | Goals and Contributions | 23 |
| 1.5 | Thesis Plan | 25 |
| 2 | Literature Review | 26 |
| 2.1 | Colour fundamentals | 26 |
| 2.1.1 | The three elements of colour | 26 |
| 2.1.2 | A simple colour model | 32 |
| 2.1.3 | Colour ambiguities: Metamerism and Colour inconsistency | 32 |
| 2.1.4 | Physical basis for inconsistency | 33 |
| 2.1.5 | Colour Spaces | 36 |
| 2.2 | Colour Histograms | 38 |
| 2.3 | Discussion | 38 |
| 2.4 | A taxonomy of colour inconsistency correction methods | 40 |
| 2.5 | Transformation Methods | 48 |
| 2.5.1 | Transformations | 48 |
| 2.5.2 | Methods For Computing The Transformations | 50 |
| 2.6 | Motivation | 53 |
| 3 | Feature based histogram alignment | 55 |
| 3.1 | Feature based histogram alignment algorithm | 55 |
| 3.1.1 | Feature Detection | 56 |
| 3.1.2 | Feature Matching | 57 |
| 3.1.3 | Feature Alignment | 58 |
| 3.1.4 | Discussion | 58 |
| 3.2 | Qualitative Evaluation | 60 |
| 3.2.1 | 1D FBHA | 60 |
| 3.2.2 | 2D FBHA | 61 |

| | | |
|----------|---|------------|
| 3.2.3 | 3D FBHA | 62 |
| 3.2.4 | Shape preserving histogram transformations | 63 |
| 3.3 | Summary Conclusions | 64 |
| 4 | An image database for testing RGB colour alignment | 73 |
| 4.1 | Database design | 73 |
| 4.1.1 | Objects | 74 |
| 4.1.2 | Capture conditions | 74 |
| 4.1.3 | Object labeling | 77 |
| 4.1.4 | Image variation sets | 77 |
| 4.2 | Existing Colour Datasets | 77 |
| 4.3 | Histogram alignment metrics | 83 |
| 4.3.1 | Bin to Bin Measures | 83 |
| 4.3.2 | Cross-Bin Measures | 84 |
| 4.3.3 | Manually defined metrics | 84 |
| 4.3.4 | Empirical comparison of metrics | 85 |
| 4.3.5 | Discussion | 92 |
| 4.4 | Quantitative evaluation of RGB colour alignment | 92 |
| 4.4.1 | Experiment 1: Feature Based Alignment Hypothesis | 93 |
| 4.4.2 | Experiment 2: Closest Euclidean Feature Match Hypothesis | 105 |
| 4.4.3 | Experiment 3: FBHA comparison | 105 |
| 4.5 | Conclusions | 115 |
| 5 | Application of feature based histogram alignment to Buhler Sortex machines | 125 |
| 5.1 | The Buhler Sortex Z-series | 125 |
| 5.1.1 | Histogram alignment problem | 126 |
| 5.1.2 | Current Approach and Commercial Confidence | 127 |
| 5.1.3 | Motivation for improved calibration routines | 128 |
| 5.2 | Product colour inconsistency reduction | 129 |
| 5.2.1 | Non-segmentation alignment methods | 130 |
| 5.2.2 | Segmentation based alignment methods | 140 |
| 5.3 | Experimental Evaluation | 145 |
| 5.3.1 | Data | 145 |
| 5.3.2 | Qualitative evaluation of feature detection and association | 153 |
| 5.3.3 | Quantitative evaluation of colour inconsistency corrections | 154 |
| 5.4 | Summary Conclusions and Discussion | 158 |
| 6 | Conclusions and Further Work | 164 |
| 6.1 | Commercial relevance and contributions | 164 |
| 6.1.1 | Direct applicability of findings | 164 |

| | | |
|----------|--------------------------------|------------|
| 6.1.2 | Future applications | 164 |
| 6.2 | FBHA | 164 |
| 6.3 | Existing methods | 167 |
| 6.4 | Further work | 167 |
| 7 | Summary of Achievements | 170 |
| 8 | Glossary | 172 |
| 9 | Appendix | 173 |
| 9.1 | The Pseudoinverse | 173 |
| 9.2 | Ordering Results | 174 |

List of Figures

| | | |
|-----|--|----|
| 1.1 | A tomato with a black rotten patch on the left and a healthier looking tomato on the right. | 20 |
| 1.2 | Two scenes captured by an Aibo robot that have been segmented using the method of Rofer [1]. 1.2(a) shows a scene and 1.2(b) shows a segmented version of the image. Each unique colour in 1.2(b) indicates a unique class label. 1.2(c) and 1.2(d) show a different scene and its segmented version. In this image the scale of the ball and robot is larger when compared to 1.2(a). | 21 |
| 1.3 | Input, Accept and Reject Samples for White Long Grain Parboiled Rice using a Buhler Sortex Z-series machine. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission. | 22 |
| 2.1 | The electromagnetic spectrum describes radiation at different wavelengths. Visible light occupies a small part of the spectrum between 380nm and 780nm. Picture by L.Keiner (Reprinted with permission) | 27 |
| 2.2 | Spectral power distributions from GE Lighting [2]. 2.2(a) shows the spectral power distribution of outdoor daylight and 2.2(b) shows the spectral power distribution under Spx35 florescent lighting. | 28 |
| 2.3 | Schematic view of the human eye (Source: Creative Commons [3]). | 30 |
| 2.4 | Colour matching functions for the RGB and the CIE X,Y,Z primaries shown in 2.4(a) and 2.4(b) respectively. Red is used to indicate R and X, green indicates G and Y, blue indicates B and Z primaries. The curves indicate the relative amounts of the primary colours needed to match a test target colour with the indicated wavelength. | 31 |
| 2.5 | The pinhole camera (Source: Creative Commons [4]). | 32 |
| 2.6 | Two different arrangements for sampling different spectral ranges. 2.6(a) shows the single chip Bayer pattern array where a red, green or blue filter element is placed over each pixel. 2.6(b) shows a multi-chip arrangement that splits light with a prism and uses three separate imaging chips for the red, green and blue bands. Images reproduced from Dalsa [5]. | 35 |
| 2.7 | Spectral reflectance curves comparing two different sets of objects. 2.7(a) shows the reflectivity of Barley seeds (red) and Bark (green). 2.7(b) shows the reflectivity of Redwood (red) and a brown paper bag (green). All data is from Glassner [6]. | 39 |
| 2.8 | Introduced taxonomy of colour inconsistency correction methods. | 42 |

| | | |
|-----|---|----|
| 2.9 | A Macbeth chart, commonly used for colour calibration tasks. | 48 |
| 3.1 | Example of the local maxima in a one dimensional histogram. 3.1(a) shows a histogram obtained from the green channel of an image of skittles in the image database. All local maxima are shown as red dots, maxima in low level noise is highlighted as 1) and multiple local maxima on a cluster are highlighted as 2). 3.1(b) shows a zoomed view of the local maxima in the cluster labelled 2). | 65 |
| 3.2 | 3.2(a) shows a representation of the scale space of the histogram in Figure 3.1(a); horizontal slices indicate histograms blurred at increasing scale moving from bottom to top, denser regions of the scale space are rendered closer to white and local maxima at each scale are drawn as circles. The maxima form paths across scales, with paths from less significant peaks ending earlier in the scale space. 3.2(b) shows the persistent maxima using $T = 20$ | 66 |
| 3.3 | The source image in 3.3(a) and target image in 3.3(b) exhibit colour inconsistency. 3.3(c) shows the colours of the source image transformed using 1D FBHA with a linear transform in each channel. | 67 |
| 3.4 | Source and target histograms are shown as overlaid line plots in the top portion of each sub-figure. The red plots show the target histogram in the red channel 3.4(a), green channel 3.4(b) and blue channel 3.4(c). The corresponding blue plots show the source histograms. Detected features are marked on the source and target histograms with a star. The bottom portions of each sub-plot show an exploded view of the source and target histograms and the matched features. | 68 |
| 3.5 | Exploded view of the histograms of the corrected data plotted with the solid line and the target histogram plotted with the dotted line. The aligned features are shown using a line to connect the aligned feature and target feature. Subfigures 3.5(a), 3.5(b) and 3.5(c) show the red, green and blue channels respectively. | 69 |
| 3.6 | Image of plastic skittles. Figure 3.6(a) shows the source image and figure 3.6(b) shows the target image where a red light modifies the appearance of the skittles. Figure 3.6(c) shows the modified source image where 2D FBHA is used in the RG channels and 1D FBHA is used in the blue channel. | 70 |
| 3.7 | Sub-figure 3.7(a) shows the square root of RG histogram for the source image in Figure 3.6(a) and sub-figure 3.7(b) shows the square root of the RG histogram for the target image in Figure 3.6(b). Taking the square root of the histograms allows the shapes of local features at different scales to be observed more easily on a single plot. | 71 |

| | | |
|-----|---|----|
| 3.8 | Feature detection and matching for the source and target RG histograms shown in Figures 3.7(a) and 3.7(b). The source RG histogram in 3.7(a) is shown as an intensity plot in sub-figure 3.8(a), dense regions of the histogram are shown closer to white and less dense regions are shown closer to black. The target RG histogram in 3.7(b) is shown as an intensity plot in 3.8(b). Matched detected features are shown with a green cross, detected features that remain unmatched are shown with a red cross. A blue line is drawn on the target RG histogram from the matched feature on the target histogram to the position of the matched feature on the source histogram. | 72 |
| 4.1 | Two images of plastic toys on a grey cardboard background. In 4.1(a) the scene is lit using clear bulbs, in 4.1(b) a red bulb is placed above the scene. | 73 |
| 4.2 | Typical images from the four different object categories used in the colour alignment database; for each object category, objects are imaged under different scale, lighting and camera conditions. Image 4.2(a) shows a representative image from the set Red and Cyan paper set. Image 4.2(b) shows an image from the set of red, green and blue paper strips. Image 4.2(c) shows an image of the plastic skittles and ball on a grey background. Image 4.2(d) shows an image from the set of stuffed animals. | 75 |
| 4.3 | Organisation of the UCL colour variation database. The directory structure under each of the four object type folders is identical. The unique parts of the hierarchy are shown. At the lowest level of the hierarchy there are three folders corresponding to the different cameras; each camera directory contains five images corresponding to five different local lighting conditions. | 76 |
| 4.4 | Images from the four different image sets illustrating the different scale variations captured and categorised in the database. The red and cyan pieces of paper occupy roughly equal portions of the image in 4.4(a), the cyan paper occupies a larger portion of the image in 4.4(b). The red, green and blue strips are arranged to occupy approximately a third of the image each in 4.4(c); 4.4(d) shows a scale adjustment of the red and blue paper. Images 4.4(e) and 4.4(f) illustrate scale variation in the skittles set. Images 4.4(g) and 4.4(h) illustrate scale variation in the Teddy Bears set. | 78 |
| 4.5 | Locations and equipment used to create different lighting conditions. 4.5(a) shows the naturally lit lounge and 4.5(b) shows the office lit by florescent bulbs. 4.5(c) shows the bulbs and dimmer switch used create local lighting variation, 4.5(d) shows the light meter used to approximately monitor the ambient lighting conditions. | 79 |
| 4.6 | The three different cameras used to acquire the colour variation database. These are 4.6(a): a Nikon Coolpix 4600, 4.6(b): an Olympus Camedia C40 Zoom and 4.6(c): a FujiFilm FinePix 6900 Zoom. | 79 |
| 4.7 | Example masked regions from sample images from the four different object types. Images 4.7(a), 4.7(c), 4.7(e) and 4.7(g) show numbered distinct masked regions for the corresponding images in 4.7(b), 4.7(d), 4.7(f) and 4.7(h). | 80 |

| | | |
|------|--|-----|
| 4.8 | Two images from the UEA uncalibrated colour database. Both images are of wall paper under the same lighting and camera conditions. The data-base contains the same images taken under different combinations of lighting and camera changes. | 81 |
| 4.9 | Different objects in the SOIL database. | 81 |
| 4.10 | Sample images from the SFU database. | 83 |
| 4.11 | Distance metric comparison for the histogram comparisons described in Experiment 1. | 88 |
| 4.12 | Contour plots of source (blue) and target (red) Gaussian distributions for a target Gaussian translation of 5 on the x-axis. Target values of θ illustrated are 0 4.12(a), 20 4.12(b), 45 4.12(c) and 90 4.12(d). The complete sequence repeats these target cluster rotations at different translated positions. | 89 |
| 4.13 | Contour plots of source (blue) and target (red) bimodal distributions. In 4.13(a) the distributions are identical, 4.13(b) moves the 2nd target cluster by 10 along the x axis. 4.13(c) moves the 1st target cluster by 5 along the x-axis, keeping the second cluster displacement at 10. 4.13(d) aligns the first cluster components and displaces the second cluster by a distance of 20. The total distance between corresponding clusters is increasing across the sequence which allows the bias of metrics towards movement in the overlapping clusters to be investigated. | 89 |
| 4.14 | Distance metric comparison for the histogram comparisons described in Experiment 2. | 90 |
| 4.15 | Distance metric comparison Experiment 3. | 91 |
| 4.16 | Contour plots of source (blue) and target (red) bimodal distributions. For both histograms the first cluster has a weight of 0.7 and the second cluster has a weight of 0.3. 4.16(a) shows the histograms perfectly overlapping. In order to investigate the bias on metrics of the larger overlapping clusters the large and small clusters are individually translated (in 4.16(b) and 4.16(c)) before moving both clusters together 4.16(d). 4.16(b) moves the larger cluster, 4.16(c) moves the smaller cluster and 4.16(d) moves both clusters together. | 92 |
| 4.17 | Colour coding scheme to represent the different alignment transforms. | 97 |
| 4.18 | Ranked transformation methods with (C)(L-LI)(L-AL)(S) variation : 1) Red-cyan paper | 98 |
| 4.19 | Ranked transformation methods for all 1770 image pairs from : 1) Red-cyan paper 4.19(a), 2) Skittles 4.19(b), Teddy bears 4.19(c) and three paper strips 4.19(d). | 100 |
| 4.20 | Ranked transformation methods with (C)(L-LI)(L-AL)(S) variation : 1) Red-cyan paper 4.20(a), 2) Skittles 4.20(b), Teddy bears 4.20(c) and three paper strips 4.20(d). | 101 |
| 4.21 | Ranked transformation methods for image pairs with 0(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 4.21(a), 2) Skittles 4.21(b), Teddy bears 4.21(c) and three paper strips 4.21(d). | 102 |
| 4.22 | Ranked transformation methods for image pairs with 0(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.22(a), 2) Skittles 4.22(b), Teddy bears 4.22(c) and three paper strips 4.22(d). | 103 |

| | | |
|------|---|-----|
| 4.23 | Ranked transformation methods for image pairs with 00(L-AL)0 variation for: 1) Red-cyan paper 4.23(a), 2) Skittles 4.23(b), Teddy bears 4.23(c) and three paper strips 4.23(d). | 104 |
| 4.24 | Colour coded matrices indicating the transformations that ranked 1st 4.24(a), 2nd 4.24(b), 3rd 4.24(c), 4th 4.24(d), 5th 4.24(d) and 6th 4.24(f). There are 60 images in this set, a coloured entry in the i th row and j th column indicates the transform that mapped the i th image to the j th image in the set and gave an average mahalanobis score that ranked at the position represented by the matrix. The colour coding scheme is shown in Figure 4.17. | 106 |
| 4.25 | Example of false negative feature detection and incorrect matching. 4.26(a) shows source (blue plot) and target histograms (red plot) in the blue channel for the teddy bears data set. The location of the missing feature in the target histogram is highlighted by the black box. 4.26(b) shows the final matches produced by CEM. | 110 |
| 4.26 | Example of false positive feature detection and incorrect matching. The left most match is deemed to be incorrect. 4.26(a) shows the detected features and 4.26(b) shows the matches. | 111 |
| 4.27 | Example of false positive feature detection resulting in catastrophic matching failure. 4.27(a) shows the detected features and 4.27(b) shows the matches. | 112 |
| 4.28 | Example of correct feature detection, incorrect matching and a structural mismatch in 1 cluster. Source and target histograms from the green channels of images of the red, green and blue paper strips are shown in 4.28(a), the red plot is the target histogram and the blue plot is the source histogram. Detected features are shown as crosses. 4.28(b) shows an exploded view of the source and target histograms and the final correspondences generated by the CEM matching step. | 113 |
| 4.29 | Example of correct feature detection, incorrect matching and a structural mismatch in 2 clusters. Source and target histograms from the green channels of images of the red, green and blue paper strips are shown in 4.28(a), the red plot is the target histogram and the blue plot is the source histogram. Detected features are shown as crosses. 4.28(b) shows an exploded view of the source and target histograms and the final correspondences generated by the CEM matching step. | 114 |
| 4.30 | Rankings of FBHA and the competing methods evaluated in experiment 1 in section 4.4.1. Shows 0(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 4.30(a), 2) Skittles 4.30(b), 3) Teddy bears 4.30(c) and 4) three paper strips 4.30(d). | 116 |
| 4.31 | Rankings of FBHA and the competing methods evaluated in experiment 1 in section 4.4.1. Shows 0(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.31(a), 2) Skittles 4.31(b), 3) Teddy bears 4.31(c) and 4) three paper strips 4.31(d). | 117 |
| 4.32 | Rankings of FBHA and the competing methods evaluated in experiment 1 in section 4.4.1. Shows (C)(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.32(a), 2) Skittles 4.32(b), 3) Teddy bears 4.32(c) and 4) three paper strips 4.32(d). | 118 |

| | | |
|------|--|-----|
| 4.33 | Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 4.33(a) and 2) Skittles 4.33(b). | 119 |
| 4.34 | Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)0 variation for: 1) Teddy bears 4.34(a) and 2) three paper strips 4.34(b). | 120 |
| 4.35 | Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.35(a) and 2) Skittles 4.35(b). | 121 |
| 4.36 | Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)(S) variation for: 1) Teddy bears 4.36(a) and 2) three paper strips 4.36(b). | 122 |
| 4.37 | Normalised counts showing the number of times each transformation method performs best against the others with (C)(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.37(a) and 2) Skittles 4.37(b). | 123 |
| 4.38 | Normalised counts showing the number of times each transformation method performs best against the others with (C)(L-LI)(L-AL)(S) variation for: 1) Teddy bears 4.38(a) and 2) three paper strips 4.38(b). | 124 |
| 5.1 | The single chute monochromatic Buhler Sortex Z1 sorting machine. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission. | 126 |
| 5.2 | The three chute monochromatic Buhler Sortex Z+ sorting machine. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission. | 127 |
| 5.3 | Side view slice of Z series machine chute. Camera systems inspect both the front and rear of the rice stream falling down the chute. The information from the front and rear views is use to reject food produce from the stream using air ejectors. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission. | 128 |
| 5.4 | A 1024 by 1024 captured image of rice with approximately 3 percent defect. The i th column in the image represents 1024 sequential grey-level captures from the pixel in the CCD array at the i th column position. Picture ©copyright Buhler Sortex Ltd, 2008. | 129 |
| 5.5 | A portion of the capture stream that clearly shows the individual grey-levels that are recorded over a small spatial region and short time frame. Rows in the image are grey level values captured over time, columns indicate spatial position across the chute. Picture ©copyright Buhler Sortex Ltd, 2008. | 130 |
| 5.6 | Histograms of product, defect and background for the 1024 pixels across the view obtained by computing histograms for the grey level values observed in each pixel. The vertical axis indexes the 256 different grey levels and the horizontal axis indexes pixel position. The frequency count is displayed as a grey-value, where high frequencies are rendered close to white and lower frequencies are rendered closer to black. | 131 |

- 5.7 Log of the histograms in figure 5.6 with the defect portions of the histograms highlighted in red. This shows the distribution of the defect across the view despite the large scale variation between product, defect and background. 131
- 5.8 Three dimensional coloured height plot of the log histograms across the view displaying grey levels 100 to 200. 132
- 5.9 Plots of histograms from pixel 500 near the centre of the chute in green and from pixel 10 near the left edge of the chute in blue. 133
- 5.10 The logical flow of information from the front and rear cameras within a chute. Defect thresholding is performed on the front and rear views independently. The defect information is aggregated by a spatial processing module, the decision to fire the air ejector is base on the size of the detected defect and the machine settings. 134
- 5.11 Histogram of grey-level intensities from a single pixel in blue and its representation at medium and high levels of blurring (plotted in green and black respectively). 135
- 5.12 Grey level representation of the scale space of a grey-level histogram. Each row in the image represents the blurred values of the histogram at different scales, higher values are rendered closer to white and lower values are rendered closer to black. The blur scale index indexes the lowest scale at the bottom row to the highest scale at the top row. . . . 136
- 5.13 Contour plot of the scale space shown in figure 5.12. Local maxima at each scale are displayed using a circle. 136
- 5.14 Associated features across the view using the *AssociateToTargetPixel(Targetpixel)* method. Features indicated with a circle are associated together and features indicated with a square are associated together. Features from the target pixel are drawn in red. . . 137
- 5.15 Associated features across the view using the *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* method described in algorithm 7. Features indicated with a circle are associated together and features indicated with a square are associated together. The initial seed features are drawn in red, the algorithm associates the features drawn in blue to the target features in two passes. First the features on the left are associated by matching the features in a column to the matched features in the adjacent column, this process is repeated for the features on the right of the initial seed features. The features drawn in green are *EdgeT* pixels away from the side of the chute; the green features on the left side of the chute are target features for the features on the left side of the chute marked in pink, the green features on the right side of the chute perform the same purpose for the matched features on the right side of the chute. 137
- 5.16 Associated features across the view using *ThreeStageAssocAndFixup(F,Targetpixel,EdgeT)*. The method uses *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* shown in Figure 5.15 as a first step, then the algorithm scans for gaps in the features across the view. The black line shown indicates the detection of a gap and the interpolated line between the detected features. 138

| | | |
|------|--|-----|
| 5.17 | Thresholds computed within a single chute using average intensity statistics method. The green plot shows the mean background values, b_i (eqn: 5.8), computed by turning the feed off to inspect the background. The black plot is the mean value f_i in each pixel (eqn: 5.9) with the feed turned on. The blue plot is the threshold r_i , in each pixel computed with PercMean . The red plot is the threshold t_i in each pixel (eqn: 5.11), computed with DiffOffset | 143 |
| 5.18 | The red plot shows modified thresholds in each pixel using the ExtrapEdgeThresholds procedure described in Algorithm 8. The green plot shows the mean background values, b_i (eqn: 5.8), computed by turning the feed off to inspect the background. The blue plot is the threshold r_i , in each pixel computed with PercMean . Note that the extrapolated red DiffOffset lines cross the green plot on the right hand side. This is undesirable behaviour. | 144 |
| 5.19 | Persistent deep structure features and background segmentation thresholds computed using the DStructMidPoint method. Background features are plotted in green, the blue plot shows the product features. The red plot shows the background segmentation thresholds computed in equation 5.12 | 145 |
| 5.20 | Buhler Sortex Z1 sorting machine and PC based capture system. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission. | 146 |
| 5.21 | Buhler Sortex Z1 sorting machine and recirculation rig. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission. | 146 |
| 5.22 | The author operating the touch screen interface on the Buhler Sortex Z1. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission. | 147 |
| 5.23 | Camera and sorting electronics. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission. | 147 |
| 5.24 | Top chute to be filled with rice on the Buhler Sortex Z1. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission. | 148 |
| 5.25 | Bottom of chute on the Buhler Sortex Z1. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission. | 148 |
| 5.26 | Associated persistent features from the front view using an offset of 110 using <i>Associate-ToTargetPixel(Targetpixel)</i> in 5.26(a), <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> in 5.26(b) and <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> with gap filing in 5.26(c). | 149 |
| 5.27 | Associated persistent features from the rear view using an offset of 110 using <i>Associate-ToTargetPixel(Targetpixel)</i> in 5.27(a), <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> in 5.27(b) and <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> with gap filing in 5.27(c). | 150 |

| | | |
|------|---|-----|
| 5.28 | Associated persistent features from the front view using an offset of 120 using <i>Associate-ToTargetPixel(Targetpixel)</i> in 5.28(a), <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> in 5.28(b) and <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> with gap filing in 5.28(c). | 151 |
| 5.29 | Associated persistent features from the rear view using an offset of 120 using <i>Associate-ToTargetPixel(Targetpixel)</i> in 5.29(a), <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> in 5.29(b) and <i>ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)</i> with gap filing in 5.29(c). | 152 |
| 5.30 | The fifteen best performing within view transformation methods applied to the front 5.30(a) and rear 5.30(b) view data-sets with a calibration offset of 110. The scores indicate the variance of the product components of the histograms across the view after correction. A lower variance indicates better alignment. | 160 |
| 5.31 | The fifteen best performing within view transformation methods applied to the front 5.31(a) and rear 5.31(b) view datasets with a calibration offset of 120. | 161 |
| 5.32 | Variance of the product components of the histograms across the view after correction with moment based and feature based global correction transforms on data front the front 5.32(a) and rear 5.32(b) views with an offset setting of 110. | 162 |
| 5.33 | Variance of the product components of the histograms across the view after correction with moment based and feature based global correction transforms on data front the front 5.33(a) and rear 5.33(b) views with an offset setting of 120 | 163 |
| 9.1 | Ranked transformation methods for image pairs with 000(S) variation for: 1) Red-cyan paper 9.1(a) and 2) Skittles 9.1(b). | 174 |
| 9.2 | Ranked transformation methods for image pairs with 000(S) variation for: 1) Teddy bears 9.2(a) and 2) three paper strips 9.2(b). | 175 |
| 9.3 | Ranked transformation methods for image pairs with 0(L-LI)00 variation for: 1) Red-cyan paper 9.3(a), 2) Skittles 9.3(b), Teddy bears 9.3(c) and three paper strips 9.3(d). . . | 176 |
| 9.4 | Ranked transformation methods for image pairs with (C)000 variation for: 1) Red-cyan paper 9.4(a), 2) Skittles 9.4(b), Teddy bears 9.4(c) and three paper strips 9.4(d). | 177 |
| 9.5 | Ranked transformation methods for image pairs with 00(L-AL)(S) variation for: 1) Red-cyan paper 9.5(a), 2) Skittles 9.5(b), Teddy bears 9.5(c) and three paper strips 9.5(d). . . | 178 |
| 9.6 | Ranked transformation methods for image pairs with (C)(L-LI)00 variation for: 1) Red-cyan paper 9.6(a), 2) Skittles 9.6(b), Teddy bears 9.6(c) and three paper strips 9.6(d). . . | 179 |
| 9.7 | Ranked transformation methods for image pairs with (C)0(L-AL)0 variation for: 1) Red-cyan paper 9.7(a), 2) Skittles 9.7(b), Teddy bears 9.7(c) and three paper strips 9.7(d). . . | 180 |
| 9.8 | Ranked transformation methods for image pairs with (C)00(S) variation for: 1) Red-cyan paper 9.8(a), 2) Skittles 9.8(b), Teddy bears 9.8(c) and three paper strips 9.8(d). | 181 |
| 9.9 | Ranked transformation methods for image pairs with 0(L-LI)0(S) variation for: 1) Red-cyan paper 9.9(a), 2) Skittles 9.9(b), Teddy bears 9.9(c) and three paper strips 9.9(d). . . | 182 |

9.10 Ranked transformation methods for image pairs with (C)(L-LI)(L-AL)0 variation for: 1)
Red-cyan paper 9.10(a), 2) Skittles 9.10(b), Teddy bears 9.10(c) and three paper strips
9.10(d). 183

9.11 Ranked transformation methods for image pairs with (C)0(L-AL)(S) variation for: 1)
Red-cyan paper 9.11(a), 2) Skittles 9.11(b), Teddy bears 9.11(c) and three paper strips
9.11(d). 184

9.12 Ranked transformation methods for image pairs with (C)(L-LI)0(S) variation for: 1)
Red-cyan paper 9.12(a), 2) Skittles 9.12(b), Teddy bears 9.12(c) and three paper strips
9.12(d). 185

List of Tables

| | | |
|-----|--|----|
| 4.1 | The image variation sets in the UCLColvariation database. Each variation set refers to a subset of the image pairs for an object type in the database, the image pairs in the subset differ in experimental capture conditions as described. The short hand codes are used to refer to these image variation sets. | 82 |
| 4.2 | Cluster Positions for the target histogram. | 90 |
| 4.3 | Cluster Positions for the target histogram. | 91 |

Chapter 1

Introduction

1.1 Colour consistency in Computer Vision

Colour is an important source of information in computer vision systems. Objects with different material properties can be imaged as different colours. Common applications that use colour information are object segmentation, object tracking [7] and retrieving similar images from a database [8]. Figure 1.1 shows an image of a tomato with a rotten patch on the left and a healthy looking tomato on the right. Clearly, the tomato on the right is preferable to eat, a person can make this judgement quickly and effortlessly using colour information. Buhler Sortex [9], the industrial partner company for this project produce systems that use colour information to separate good and bad food [10]. The annual Robocup competition [11] is a soccer tournament for autonomous robots. Figure 1.2 shows images from the Aibo robot dog league. Colour information is used to distinguish between the ball, robots, the terrain and the goal.

A major problem for all colour computer vision systems is that the recorded colour of an object varies when camera and lighting conditions change. Colour consistency occurs when an object or objects with the same material properties are imaged to give the same recorded colour values irrespective of any different lighting and camera conditions that may be present. Colour consistency of objects in an uncontrolled scene can be improved by introducing careful selection of the lighting and camera conditions;



Figure 1.1: A tomato with a black rotten patch on the left and a healthier looking tomato on the right.

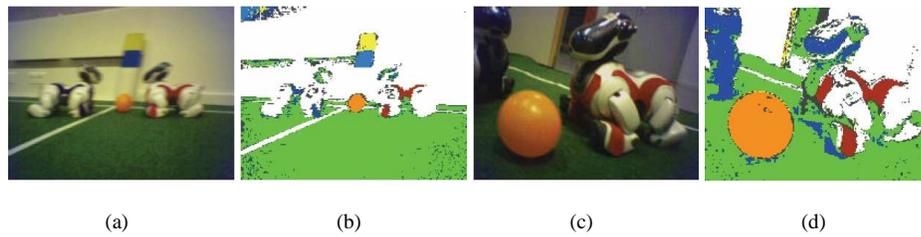


Figure 1.2: Two scenes captured by an Aibo robot that have been segmented using the method of Rofer [1]. 1.2(a) shows a scene and 1.2(b) shows a segmented version of the image. Each unique colour in 1.2(b) indicates a unique class label. 1.2(c) and 1.2(d) show a different scene and its segmented version. In this image the scale of the ball and robot is larger when compared to 1.2(a).

however, it rapidly becomes impractical or even impossible to produce incremental improvements using this approach because lighting environments and cameras have inherent variabilities that are difficult to minimize due to the limits of manufacturing technology. Colour consistency should not be confused with colour constancy, the term colour constancy is used to refer to methods that reduce object colour variation due to lighting effects only. High colour consistency means that colour variation can be attributed to the object material properties and not the properties of the camera or light source. This is significant in the Robocup application when tracking the ball and goal position; regions of the ball and terrain in Figure 1.2(d) have been misclassified, misclassifications occur frequently when the lighting or camera conditions are highly variable. Currently, Robocup soccer matches are played indoors as the variability due to outdoor lighting fluctuations is considered too great [12]. In addition, a large amount of time is needed to set up the colour thresholds for each Aibo every time lighting conditions change [13]. Increasing the degree of colour consistency improves the tracking process and makes object segmentation thresholds more reliable.

Improved colour consistency reduces the camera and lighting variations. Computer vision applications that would benefit from this are:

1. **Colour object segmentation**, where data has been captured with lighting and/or camera variation.
2. **Image lookup from a database**, where images were captured under different lighting conditions and/or camera conditions
3. **Colour object tracking** when subsequent frames vary due to camera and/or lighting effects.
4. **Robust colour models**: object variation can be compactly modelled when camera and lighting variation is reduced. If these effects can be minimized, object colour models constructed under one set of lighting and camera conditions could be more easily deployed under a different set of conditions.

This thesis argues that colour consistency can be improved in a range of applications by summarizing colour data-sets with histograms and then finding transformations that align the histograms to minimize the differences due to camera and lighting variations. Histograms can be computed from single images,



Figure 1.3: Input, Accept and Reject Samples for White Long Grain Parboiled Rice using a Buhler Sortex Z-series machine. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission.

image frames in a video sequence or from portions of a video stream over time. Colour data acquired from a range of different situations can be summarized using colour histograms, this means that generic methods to align colour histograms could have significant and wide reaching impact.

1.2 Colour consistency in the food processing industry

Colour sorting machines are a vital part of the modern food processing and production process; these machines are used to sort products such as rice and coffee into accept and reject categories as seen in Figure 1.3. Buhler Sortex is a leading global supplier of optical sorting machines in over 100 countries; they have more than 20,000 installations around the world and are continually striving to improve the quality of the sorting process. Sorted foodstuffs have a significantly higher economic value than unsorted foodstuffs. Buhler Sortex and their customers are keen to gain economic advantage through optimization of the sorting process.

Product classification performance varies within and between machines; this variation exists despite the fact that machines have been engineered to provide a highly controlled inspection environment. Buhler Sortex engineers and scientists have determined that a key factor affecting classification algorithms is the variability in the recorded colours of product being sorted. Current techniques for minimizing this undesired variation involve a large degree of hand tuning using interactive tools; performing this tuning requires a high degree of skill. Buhler Sortex are continually looking to improve the classification performance and reduce the set-up time of their machines. The economic and environmental impact of even a small performance improvement is extremely significant due to the large volumes of food produce pro-

cessed. A typical machine sorts approximately nine tonnes per hour and works for approximately 6000 hours per year. Therefore, a performance improvement that saves just 0.5 percent of this product volume will yield an extra 270 tonnes of product per machine per year. The application of generic and automatic colour space alignment algorithms on Buhler Sortex machines would mean that the manual tuning step could potentially be eliminated or at least minimized. If Buhler Sortex machines operate at higher levels of performance then less product will be wasted and higher quality sorted produce will result.

A further goal of improved calibration routines on Buhler Sortex machines is to build colour distributions for specific products such as *basmati rice* or *kenyan coffee* that can be calibrated across different machines. Improved calibration methods will enable these colour distributions to be applied in practice since the calibration mapping within and between machines will be better understood. Deployment of colour models for food products would reduce set-up times and costs. At present, engineers may be required to fly to foreign sites to perform detailed calibration and set up procedures; colour distributions that can be automatically calibrated for different machines would greatly reduce these costs.

1.3 The automatic histogram alignment problem

Colour histograms of objects with the same material properties that are imaged under different camera and lighting conditions exhibit differences due to colour inconsistencies. This thesis hypothesises that meaningful structure can be extracted from histograms and alignment transforms can be found that align the structure of the histograms to increase colour consistency. The term automatic histogram alignment problem is used in this thesis to describe the process of finding alignment transforms from colour histograms without manually defined labels. The applications in Section 1.1 and the Buhler Sortex problem described in 1.2 would all benefit from robust and generic solutions to the automatic histogram alignment problem.

The next chapter reviews existing methods for solving the colour inconsistency problem and highlights that many solutions are only applicable to a specific problem domain. Domain specific solutions are important as they teach us how to produce high performance systems in a pragmatic way; however, it is difficult to apply methods from one problem domain to another. Generic histogram alignment algorithms with known performance characteristics would allow vision systems that handle colour inconsistency to be built using standardized colour consistency modules; prior knowledge and related constraints can then be integrated into the problem as required. This thesis develops generic histogram alignment methods and tests the methods on an RGB colour image database and grey-level video stream data.

1.4 Goals and Contributions

The goal of this thesis is to create unsupervised alignment algorithms that can align the corresponding clusters in colour histograms.

The aim is for these algorithms to have general relevance in computer vision and to find direct application to the problems faced by Buhler Sortex. This section summarizes the original contributions of this thesis developed in the pursuit of this goal.

1. Design and capture of structured databases for the study of colour space alignment.

- A structured, labelled RGB colour database.

The image database systematically introduces scale, camera, local and ambient lighting variation for four sets of simple objects. Different histogram alignment methods can be tested on image sets where the class of physical variation leading to observed colour variation is known. The hand labeled object annotations allow robust quantitative evaluation of aligned histograms.

- Buhler Sortex grey-level video streamed data.

A new real time capture system has been developed to record video data from a Z1 Buhler Sortex food processing system. The real time nature of this data allows existing and new real time colour calibration methods to be studied in a manner that has direct relevance to the real-time machine behaviour.

2. Algorithms to solve the colour space alignment problem.

- A three-step feature based histogram alignment (FBHA) framework is introduced. The three steps are 1) unsupervised colour space feature detection, 2) feature association and 3) feature alignment. There are a variety of choices at each step, these choices are evaluated in isolation using the RGB image database. Examination of each component in isolation in addition to the aggregate performance provides insight into the challenges and advantages of the FBHA approach.
- Scale space techniques are utilised as a robust way of extracting peaks from noisy histograms. Matching these peaks leads to the discovery that structural mismatches between corresponding clusters commonly occur in data from commodity cameras.

3. Colour space alignment metrics and evaluation methodology.

- A labeled histogram metric for comparing multi-modal distributions is introduced, it limits the bias towards the largest and overlapping clusters. This means that it considers the alignment of each cluster to be of equal importance.
- A method for ranking FBHA and existing colour inconsistency removal techniques is introduced. The variability of transform performance is found to be high, the method uses bootstrap confidence tests to establish a ranking that accounts for outlier behavior.
- A method to compare colour histogram corrections on Buhler Sortex data is introduced. Corrections are ranked by the residual colour variation of the corrected acceptable product, good scores indicate correction methods that are worthy of further investigation in a sorting setup.

4. Empirical analysis.

- A new ranking of existing colour inconsistency removal methods and transforms is developed. To the author's knowledge, no comparable ranking exists in the literature. The ranking helps system designers pick an appropriate method or transform when constructing a computer vision system. The analysis highlights the high performance of point alignment transforms. An important discovery is that the performance of commonly used transforms and methods is highly sensitive to small variations in the data acquisition conditions.
- Different combinations of system components and transformations are tested and ranked using Buhler Sortex data. The ranking is novel and leads to insights concerning the importance of different processing steps and transform selection.

1.5 Thesis Plan

- This chapter has introduced the automatic colour histogram alignment task and related it to the highly relevant problem of colour consistency.
- **Chapter 2** reviews related work and constructs a taxonomy of colour inconsistency correction methods. Background methods that directly support the development of ideas in subsequent chapters are specified.
- **Chapter 3** introduces the FBHA methods and qualitatively evaluates their behavior.
- **Chapter 4** introduces an RGB image database containing examples of colour inconsistency. The different sources of colour inconsistency in the database are described. FBHA methods are quantitatively compared to a set of reference methods using the database.
- **Chapter 5** studies the Buhler Sortex machine and applies the FBHA methodology to grey-level video stream data.
- **Chapter 6** discusses the commercial impact of this work and the value added to the industrial sponsor.
- **Chapter 7** draws relevant conclusions from this work and suggests future research directions.
- **Chapter 8** summarises the achievements of this thesis.

Chapter 2

Literature Review

This chapter reviews important background material that allows solutions to the automatic histogram alignment problem to be developed and placed in context. First, the basics of colour vision are introduced; the aim is to explain how colour phenomena occur in a wide range of physical environments. Second, existing colour inconsistency methods are reviewed and organized so that their key operational points can be understood. Finally, a compilation of mathematical transforms is specified. The transforms are used by existing methods and the specifications are drawn upon at later points in the thesis.

2.1 Colour fundamentals

This section introduces basic concepts of colour vision. The relationship between colour, light, material properties and camera sensors is described. Adelson's plenoptic function [14] summarizes the light sampling process that produces colour data. The plenoptic function is used in this review to describe different common sampling schemes; it provides a common basis for thinking about different physical capture conditions. Next, the ambiguous mappings between an object's material properties and observed colour are discussed; the different types of ambiguity are listed and examples of when they occur are given. The section concludes with a description of how colour is represented using colour spaces and the purpose of these spaces.

2.1.1 The three elements of colour

Colour requires three elements:

1. light,
2. interaction of the light and objects in the scene,
3. the capture of light at a sensor.

The following subsections explain these three elements in terms of their physical principles. The perception of colour by humans depends on the physics of the world and what happens in the eye and the brain. Human sensations of colour involve the photo-chemical processes in the eye combined with psychological processes in the brain. In computer vision, colour images are represented as numbers and depend on the physical world and the physics of cameras. This section introduces these concepts in some detail.

Light

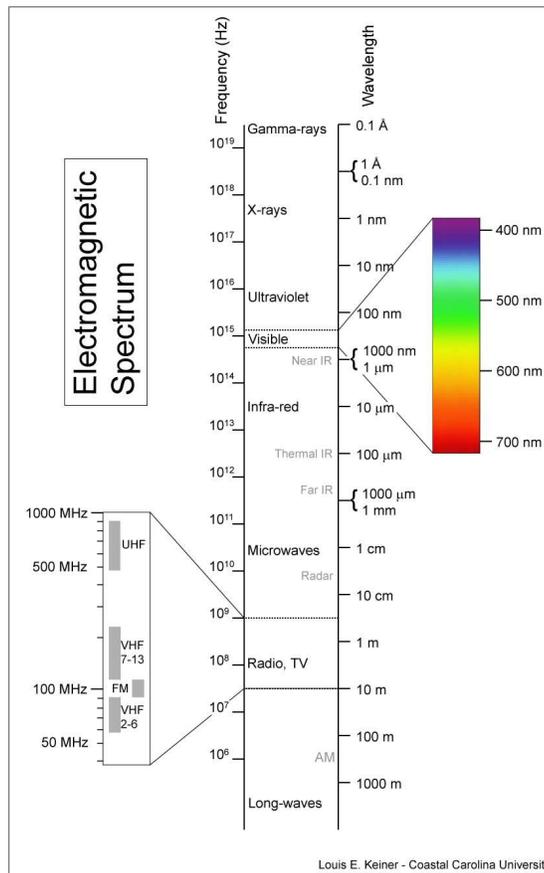


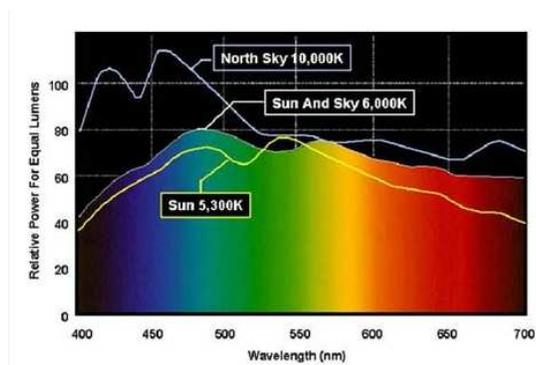
Figure 2.1: The electromagnetic spectrum describes radiation at different wavelengths. Visible light occupies a small part of the spectrum between 380nm and 780nm. Picture by L.Keiner (Reprinted with permission)

Visible light is electromagnetic radiation between the wavelengths of 380nm to 780nm. Infra-red and ultra-violet light exist below and above this range respectively. Light and other forms of radiation exist on a continuous spectrum of wavelengths as illustrated by Figure 2.1. Light is composed of particles called photons, each photon carries a definite amount of energy. Light is remarkable as it exhibits the properties of both a wave and a particle, this behaviour is known as wave-particle duality [15].

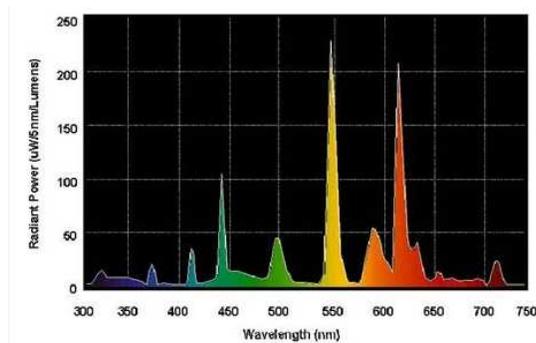
A light source emits photons in different directions. Radiometry deals with the measurement of light and the distribution of light in space is measured using radiance. Radiance is the power (energy per unit time) travelling at some point in a specified direction, per unit area perpendicular to the direction of travel, per unit solid angle. Its units are watts per square meter per steradian [16]. It is common to study the property of light by ignoring the dependency on solid angle and plotting the spectral exitance in watts per square meter against wavelength. Figure 2.2(a) shows a spectral power distribution outdoors and 2.2(b) shows a spectral power distribution measured under a florescent light. Measurements are obtained with a photometer, the vertical scales in these plots are normalized to lumens. Lumens is a transformed representation of actual radiance that is used to measure the perceived power of light [2]. It

is immediately noticeable how different these two distributions are. Spectral power distributions change for different lighting conditions.

The spectral power distribution is a summary description. Light interacts with a physical environment in complex ways. Adelson *et al.* [14] introduced the plenoptic function as an idealized concept to describe the incident light (*i.e incoming radiance*) at every point in space. The value $P(\theta, \phi, \lambda, t, V_x, V_y, V_z)$ of the plenoptic function, P , is the intensity of the light with wavelength, λ at position, V_x, V_y, V_z at time, t in direction, θ, ϕ . The function describes the incoming radiance at every point V_x, V_y, V_z of an idealized eye along a ray direction specified in spherical coordinates parameterized by θ, ϕ , for every wavelength λ at every time t . Adelson *et al.* state that the plenoptic function allows specification of a colour holographic movie: “A true holographic movie would allow reconstruction of every possible view, at every moment, from every position, at every wavelength, within the bounds of the space-time wavelength region under consideration. The plenoptic function is equivalent to this complete holographic representation of the visual world”. The plenoptic function is important as it allows one to abstractly examine the structure of the information that is available to the observer by visual means. In computer vision or computer graphics, the plenoptic function is naturally parameterized in terms of (x, y) spatial coordinates on the image plane to give the value $P(x, y, \lambda, t, V_x, V_y, V_z)$.



(a)



(b)

Figure 2.2: Spectral power distributions from GE Lighting [2]. 2.2(a) shows the spectral power distribution of outdoor daylight and 2.2(b) shows the spectral power distribution under Spx35 florescent lighting.

The interaction of light with objects

When light hits an object it is modified by the interaction with the object's material structure. Berns [17] lists the following different interactions:

- Transmission through a transparent or translucent object.
- Reflection from a specular or matt object.
- Absorption of light by an object.
- Scattering of light through a material.
- Fluorescence. Fluorescent materials absorb and then re-emit lights at different wavelengths.

In practice, a combination of these effects may occur for a given interaction between light and object. Models that consider all the effects would be prohibitively complex, it is common to introduce simplifying assumptions and propose models for limited classes of objects. The bidirectional reflectance distribution function (BRDF) [17] describes how incoming irradiance coming from different directions is reflected in different directions. The function can be used to describe interactions with diffuse, specular or diffuse and specular interactions. In computer graphics, the BRDF of a material such as skin [18] can be measured and then utilized in the production of realistic looking rendered images.

In medical imaging, current research in optical tomography seeks to reconstruct images of the brain by shining infra-red light on the brain and inferring internal brain structure from light readings that have passed through the head [19]. These methods must account for the complex manner in which light is scattered. To summarize, models of light and material interaction exist for a range of different materials. However, the models are frequently complex.

The observer

The capture of the incoming distribution of light by a photosensitive observer is the final element necessary to describe colour. The human eye is a natural starting point for discussion. Figure 2.3 shows a schematic view of the eye. Light enters the eye by passing through the cornea and lens and an image is formed on the retina. The retina is comprised of rod and cone cells, the rods are responsible for detecting brightness and the cones are responsible for colour vision. The majority of normal people possess three types of cone cell. The three different cone cells are each excited in a different manner by different wavelength ranges of light.

Guild [20] conducted colour matching experiments to investigate trichromacy and to determine the response functions of the three types of cone cell. Participants were presented with a split visual field, the left side was illuminated by a monochromatic light and the right hand side was illuminated by a mixture of red, green and blue monochromatic lights. The participant's task was to mix the red, green and blue (RGB) lights until the colour appearance of the right side of the field matched the left hand side. The task was repeated for different illuminants on the left hand side of the visual field. Figure 2.4(a) shows the RGB tristimulus curves that result from this experiment, these curves are the best fit curves to the mixing results obtained from all participants. The RGB matching functions can be negative, this is because it is

not possible to match all colours by mixing the RGB primaries. In order to effect certain matches, Guild moved a primary light source from the mixing side of the field and to the test light side. This situation is modelled by subtractive matching and results in the negative portions of the matching function. The CIE 34 XYZ matching functions are a basis transformation of the RGB primary matching functions so that X,Y and Z are positive everywhere. The updated CIE 63 XYZ standard is shown in Figure 2.4(b). The CIE 34 standard presented the test and mixture field using two degrees of visual angle, whereas the CIE 64 standard used ten degrees [21]. Understanding the original colour matching experiments is important because most modern cameras and colour spaces are based upon the findings of these original matching experiments.

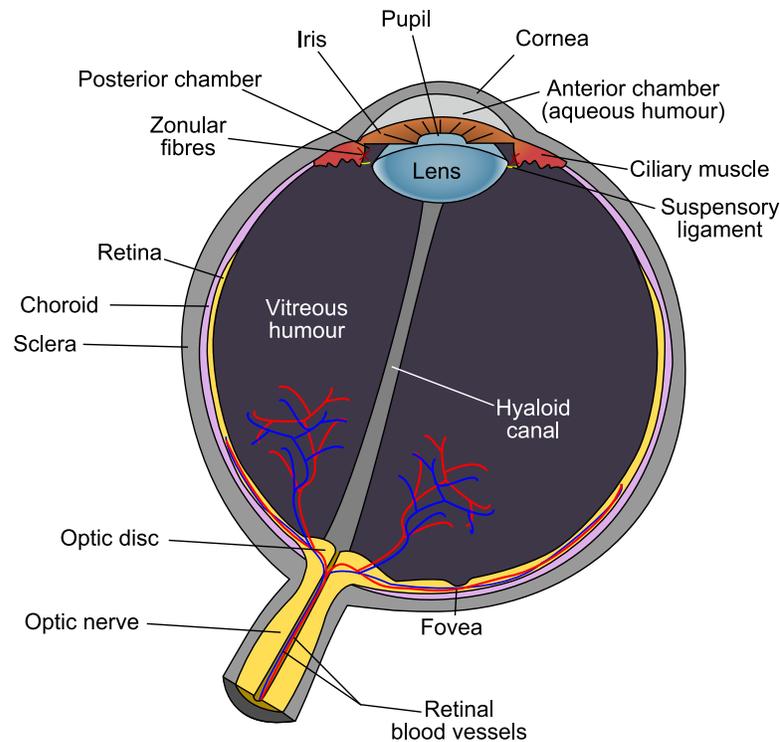
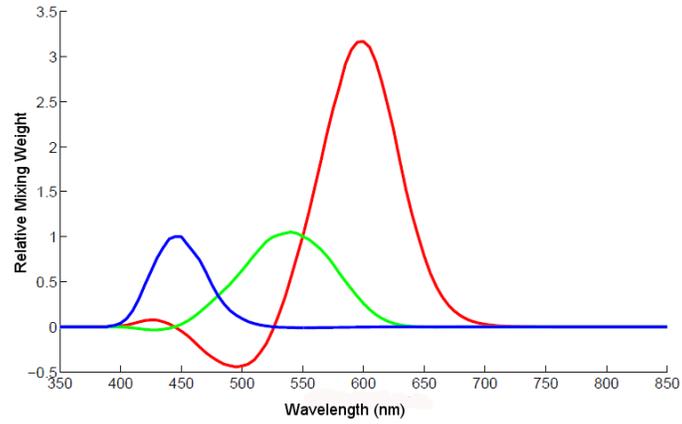
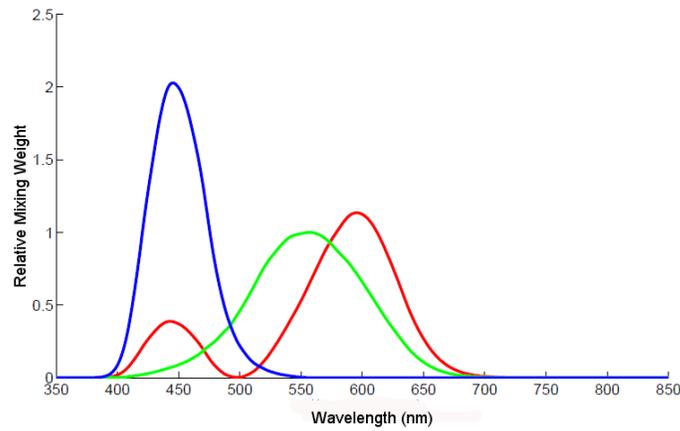


Figure 2.3: Schematic view of the human eye (Source: Creative Commons [3]).

In computer vision, a camera is used to capture colour images. Both a camera or human eye sample the plenoptic function spatially, over different wavelength bands and over time. In a colour camera, a number of optical filters are used to extract information from the different wavelength bands. The ideal theoretical camera is a pin-hole camera shown in figure 2.5. An image is formed on the rear plane by placing the pin-hole aperture at V_x, V_y, V_z , this arrangement samples the plenoptic function at this point. Real cameras deviate from the ideal pin hole camera because of optical, electronic and manufacturing limitations. Section 2.1.4 explains how these deviations cause colour inconsistencies that exist in all real cameras. Section 2.1.5 discusses how RGB cameras may be suboptimal for many machine vision tasks and proposes alternative choices.



(a)



(b)

Figure 2.4: Colour matching functions for the RGB and the CIE X,Y,Z primaries shown in 2.4(a) and 2.4(b) respectively. Red is used to indicate R and X, green indicates G and Y, blue indicates B and Z primaries. The curves indicate the relative amounts of the primary colours needed to match a test target colour with the indicated wavelength.

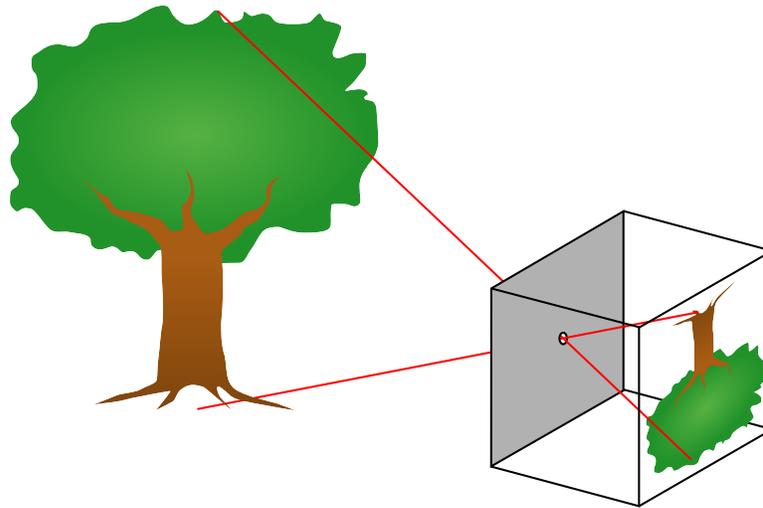


Figure 2.5: The pinhole camera (Source: Creative Commons [4]).

2.1.2 A simple colour model

The CIE defined a multiplicative model of colour that is commonly used because of its simplicity [16](pg. 115). The model is most appropriate for Lambertian surfaces, it states that the grey-level intensity in the i th channel is

$$q_i = \int_{\omega} E(\lambda)S(\lambda)Q_i(\lambda)d\lambda, i = 1, \dots, m. \quad (2.1)$$

The scene is illuminated by a single light characterized by the spectral power distribution $E(\lambda)$, this specifies how much energy the light source emits at each wavelength λ . The surface reflectance $S(\lambda)$ of the imaged object is the proportion of light incident on the surface reflected at each wavelength and the spectral sensitivity function, $Q_i(\lambda)$, is the sensitivity of the i th channel to light at each wavelength of the spectrum. The integral is over the range ω of wavelengths of light.

2.1.3 Colour ambiguities: Metamerism and Colour inconsistency

The mappings from an object's spectral reflectance function $S(\lambda)$ to observed colours are not unique. This is perhaps the most important idea in colour vision because of the consequences of two examples of these non-unique mappings: Metamerism and Colour inconsistency. Observed colour is dependent on the three elements of the colour triplet, the Lambertian model is used here to describe the different ambiguities.

Metamerism

Metamerism occurs when imaged objects with different material properties have the same recorded colour [17]; in the Lambertian model, material properties are described by the reflectance function $S(\lambda)$. Metamerism enables important applications such as printed colour pictures and colour television. The colours of objects in a picture or on a television screen appear to match their real world counterparts. This is an example of metamerism, the material properties of the ink of the paper, or the phosphors on a cathode ray television are different to the material properties of the displayed objects and yet the colours appear to match. Without metamerism it would not be possible to recreate images that accurately match

what we see in the outside world! In computer vision, metamerism is treated as a problem to be avoided or minimized if possible; this is because a common use of colour is to uniquely identify objects with the same material properties.

The two kinds of metamerism are:

1. Illuminant metamerism: Colours of different objects match under one lighting condition and mismatch under another.
2. Observer metamerism: Colours of different objects match under one observer (camera) condition and mismatch under another.

Colour inconsistency

Colour inconsistency occurs when a single object gives different recorded colours. The two types of colour inconsistency are:

1. Illuminant colour inconsistency: An object gives two different colours under two different illumination conditions.
2. Observer colour inconsistency: An object gives two different colours under two different observer (camera) conditions.

In practice, combinations of illuminant and observer conditions are possible for both metamerism and colour inconsistency.

2.1.4 Physical basis for inconsistency

This section explains how observer colour inconsistency and illuminant colour inconsistency are caused by physical conditions during image capture. An understanding of when these inconsistencies occur is important when designing computer vision systems.

Observer colour inconsistencies

Cameras used in computer vision require optical lens systems to focus light and electronic sensors to capture the light and convert it into numerical signals. This section describes details of modern camera design that suffer from colour inconsistency.

The optical system

Two notable sources of colour inconsistency in the optical system are vignetting and chromatic aberration. Vignetting is the fall off in intensity due to the geometry of a thin lens. The pixels at the edge of a CCD array observe darker colours than those at the centre, this is a problem when segmenting a colour object using the same colour thresholds in all pixels. The Aibo robot dogs used in Robocup suffer from significant vignetting, segmentation performance is improved across the visual field by calibration of the vignetting effect [22]. Horn [23] proposed a basic model for the behaviour of the vignetting fall off function. The basic model is often insufficient because it does not account for further imperfections in real lenses [24], also the vignetting function changes as a function of aperture and zoom. This makes photometric calibration of systems with variable aperture and zoom difficult [24]. Vignetting is

also a significant source of colour inconsistency in panoramic photography where multiple images with overlapping content are stitched into a larger image [25][26][27].

Chromatic aberrations occur because different wavelengths of light are refracted through a medium at different angles. In photographic lenses, this can result in different wavelengths being brought to focus at different points. Lens designers undergo great efforts to reduce this effect and using a high quality lens is the best way to combat this aberration. The two kinds of chromatic aberration are longitudinal chromatic aberration which shows up as the inability of a lens to focus different colours on the same focal plane and transverse chromatic aberration which can be observed as fringing at areas of high spatial detail. In practice, chromatic aberration will result as a combination of both the longitudinal and transverse effects. The effect can be seen by imaging a grid of black lines on a white background, inspection of the red, green and blue intensity profiles from vertical lines near the centre and the edge will show misalignment if chromatic aberration is present, see Willson [24] for more details of this method.

The sensor system

Light passes through the optical system and is focused on the image plane. Incident light on the imaged plane can be sampled using a charged coupled device (CCD) or complementary-metal-oxide-semiconductor (CMOS) arrays. Both types of sensor accumulate signal charge in each pixel proportional to the local illumination intensity, the charge is then converted to an output signal. With CCD arrays the camera circuitry is separate from the imaging chip, CMOS arrays convert charge to voltage on the chip in each pixel. Each sensor type has advantages in different situations, CMOS sensors are rugged and offer superior integration. CCDs offer superior images quality and flexibility [28]. In both types of sensor, manufacturing tolerances mean that there are physical differences between individual pixel sites. The combined effect of these physical and electronic differences mean that colour inconsistencies exist between different pixels on the same sensor and between different sensors. Further comparison of CCD and CMOS can be found in Janesick [29]. For an in depth study of sources of noise in electronic cameras see Kamberova [30].

Coloured lens filters are used to sample different spectral wavelengths when using CCD or CMOS chips. There are two common sampling arrangements: 1) the Bayer single chip arrangement [31] shown in Figure 2.6(a), and 2) the prism based multi-chip arrangement shown in Figure 2.6(b). The Bayer pattern places red, green and blue optical filters over the sensor elements to approximate the relative distributions of red, green and blue sensitive cone cells in the eye. There are more red and green filters because the human eye is less sensitive to the blue channel, most common cameras use this construction. Different spectral bands are sampled at different spatial positions, so the missing samples in each spectral band must be reconstructed; demosaicing algorithms [32] deal with the optimal manner to perform this reconstruction. The spatial sampling limitations of the Bayer pattern can be avoided with a more expensive and complex multi-chip design shown in 2.6(b). The multi-chip arrangement uses a prism to split light into the red, green and blue bands which are each fully sampled in the spatial domain using independent arrays of sensors. It is important to be aware of the different characteristics of sensor systems as they sample the plenoptic function in different ways and have different noise characteristics. This

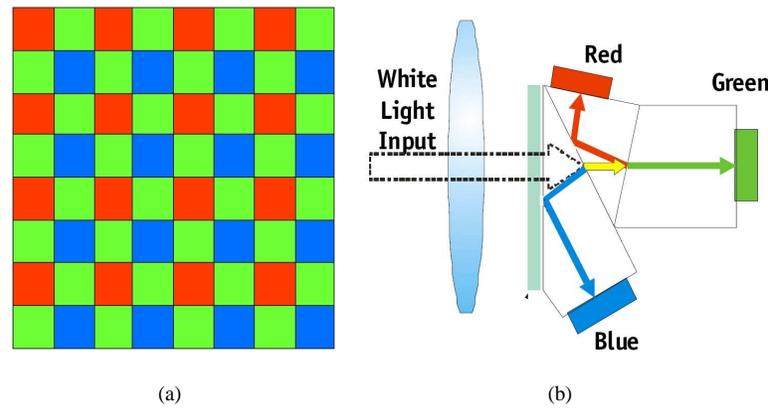


Figure 2.6: Two different arrangements for sampling different spectral ranges. 2.6(a) shows the single chip Bayer pattern array where a red, green or blue filter element is placed over each pixel. 2.6(b) shows a multi-chip arrangement that splits light with a prism and uses three separate imaging chips for the red, green and blue bands. Images reproduced from Dalsa [5].

means that different colour inconsistencies are to be expected.

Illuminant colour inconsistencies

Section 2.1.1 introduced the idea that different light sources have considerably different spectral power distributions and that imaging objects under different lights leads to colour inconsistencies. Objects imaged outside during changing atmospheric conditions or inside using different lights will be illuminated with lights that have different spectral power distributions. Additionally, it is practically impossible to construct an environment that illuminates objects in a perfectly constant manner at different positions in the scene. When using bulbs, different amounts of light are radiated in different directions and the characteristics of a bulb can change over time as it ages.

A further important consideration is the geometric relationship between the light source, the imaged object and the camera. For specular objects such as metals, the light leaving an object at a given point on the surface will vary significantly as the light source is moved. For diffuse objects such as matte paints that scatter light in all directions, movement of the light source produces a much smaller change in outgoing illumination from the surface patch. Imaging an object with a complicated shape compounds the complexity of the matter further, even if the object is made from a homogeneous material. This is because the local surface normal varies at different positions on the surface and the local surface reflectivity is dependent on the local normal. In the theoretical case, an ideal camera in a perfectly constant light field could be used to sample the plenoptic function by taking images of an object from all camera positions and viewing angles. This would give images that have different colour values that are due to the BRDF and other material properties of the object and the object geometry alone. This is the minimum possible colour variation that can be theoretically expected with an idealised lighting and camera setup when observing an object from different positions within the environment. In practice, the observed colour variation will be much greater due to the camera and lighting inconsistencies detailed. A pragmatic approach to using colour is necessary in machine vision, practical steps to minimize colour

variation are : 1) Controlling the illumination as much as is possible, 2) Limiting the range of different geometric relationships than can exist between the lights, object and camera and 3) Using the best quality cameras given the budget. The cameras, lenses and filters should be chosen by considering how the camera inconsistencies will affect the recorded colours. Trade offs between different camera choices should be made to maximize application performance.

2.1.5 Colour Spaces

A colour space is a co-ordinate system that allows all colours relevant to a domain of interest to be described. Many different colour spaces exist that have been developed for a variety of reasons. This section summarises prominent colour spaces derived from the RGB tristimulus experiments and contrasts these with non-RGB models. An N-dimensional colour space is formed by sampling N different ranges of the light spectrum and combining the resulting colour values into an N dimensional vector. Each resulting dimension is called a colour channel. Section 2.1.1 introduced tristimulus theory which motivates the use of three overlapping ranges in wavelength called the red, green and blue colour bands. Most colour cameras generate RGB data as the images are intended for human viewing. However, the automatic histogram alignment problem is independent of a particular colour space and so wavelength sampling selection is discussed in a general way. Finally, a procedure for designing a colour space for object recognition is discussed and the relative merits and disadvantages of this approach are discussed.

RGB derived models

The RGB colour space represents colours as a mixture of red, green and blue, this section details common colour spaces based on this representation.

Perceptually based systems

There are two problems with the RGB colour space: 1) the RGB axes do not correspond to intuitive notions of colour and 2) a constant Euclidean distance between different colours in RGB space does not correspond to a constant perceptual difference between the points.

Perceptually based colour spaces aim to alleviate these problems. The Hue, Saturation, Value (HSV) colour space forms a cone along the white black axis. Hue corresponds to the chromatic notion of colour, saturation is the distance from the axis and value is the brightness. HSV and a similar space HSL are transformations of the device dependent RGB space. This means that the HSV and HSL spaces provide greater intuition than RGB but differ for each device. These spaces find use in photo-editing and drawing software.

The CIE XYZ tristimulus functions plotted in figure 2.4(b) lead to so called chromaticity coordinates, these are the normalized tristimulus values:

$$x = \frac{X}{X + Y + Z}, y = \frac{Y}{X + Y + Z}, z = \frac{Z}{X + Y + Z}.$$

The x-y plane describes the chromatic variation of the colours and z represents lightness. The x-y plane plot is known as the chromaticity diagram and represents all possible human perceivable colours. The CIE XYZ colour space is not perceptually uniform, this means that different perceptual differences result from colours a constant distance apart. The field of colour vision has sought to develop perceptually

uniform systems, although none is perfect. The 1976 uv space [17] (pgs 63-65) was developed to improve the chromatic uniformity, its coordinates relate to XYZ according to:

$$u' = \frac{4x}{-2 + 12y + 3}, v' = \frac{9y}{-2x + 12y + 3}.$$

One problem with the $u'v'$ space is that it is not perceptually uniform in lightness. The CIELAB addresses this problem and has correlates for lightness, chroma and hue. CIELAB is considered to be the simplest colour appearance model (CAM). A CAM provides mathematical formulae to transform physical measurements of the stimulus and viewing environment into correlates of perceptual attributes of colour [33]. Most CAMs have a corresponding chromatic-adaptation transform (CAT); the CAT is a method for transforming the CAM of a scene acquired under a test illuminant so that the scene colours match those under a reference illuminant. The combination of CAMs and CAT seek to model the human colour constancy mechanism that enables people to perceive an object to be the same colour under different illuminants. Research has led to CAMs that predict known psycho-physical effects more accurately. For example, CIECAM97s is a CAM that predicts a number of human colour appearance phenomena such as chromatic adaptation.

Care should be taken when using colour spaces designed to model human vision processes in a machine vision setup because human colour vision processes are often not directly comparable to the processes in a particular machine vision setup.

Colour mixing models The need to mix appropriate proportions of inks to print colour images has led to the CMYK system. CMYK is a subtractive mixing system that expresses how an RGB colour can be created by mixing the appropriate amount of cyan, magenta, yellow and black inks on white paper.

Non RGB models

For most humans the notion of colour is synonymous with the RGB model. When analysing how to best use colour in a machine vision application, it is important to realise that human colour is a product of evolution and is in no sense the *correct* model. Birds, lizards, turtles and many fish have four types of cone cells and most mammals only have two types; Birds also see close to the ultra-violet band [34]. Each individual species has evolved a visual system specific to the environmental challenges and needs that it faces. Machine and computer vision problems also arise that are not best suited to the RGB camera, Grey-level inspection is important in industrial inspection [35] because of its simplicity and robustness. Hyperspectral imaging combines usage of the visible bands with infra-red and ultra-violet bands as required. For example, visible and IR bands can be used to extract information from airborne imagery of vegetation [36].

How to design a colour space

A key concern in computer vision is how to capture colours that discriminately identify different objects in a scene; this section discusses the principles of how to select spectral wavelength bands to achieve this goal. In practical terms, these principles guide the selection of colour lens filters so that discriminative colours are obtained. It is important to realise that the RGB colour space and its derivatives are prevalent in the literature because of the pervasiveness of modern RGB cameras, RGB cameras have been

developed to produce images that match human perception. Many computer vision applications do not require a visual output to be presented to a human and therefore may be better served by a different set of sampling wavelengths.

The best wavelength ranges to sample can be determined by considering the spectral reflectance of the objects being inspected. The spectral reflectance of an object varies with wavelength, and spectral reflectance curves have been prepared in the laboratory using a spectrophotometer for a range of objects by Glassner [6]. As an example, Figure 2.7(a) shows the spectral reflectance curves for barley seeds and bark. The curves are well separated across the full range of illuminant wavelengths from 400-690 nm, this means that a grey-level system in these ranges could be used to successfully distinguish barley seeds from tree bark. Figure 2.7(b) shows the spectral reflectance curves for redwood and a brown paper bag. At around 650 nm the curves cross, this means that using colours that sample within a narrow band around 650 nm would be ineffective at discriminating between the brown bag and better discrimination between the bag and redwood can be achieved by sampling colours between 400 and 550 nm.

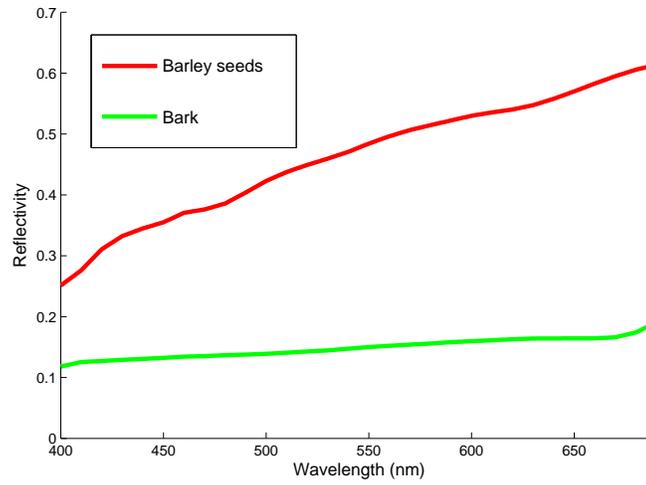
These examples illustrate that the best colour ranges for discrimination can be discerned by considering the spectral reflectances of the objects in question. If there is a distinct difference in reflectivity within a single band of wavelengths then this monochromatic range of wavelengths may be used. Monochromatic sorting removes dark rotten peanuts and dark defects from rice [10] (*pgs. 117-136*). When it is not possible to find a single region of the spectrum where the acceptable and defect food produce are separated then more colour channels are required. This is the reason that Buhler Sortex bi-chromatic machines are used to sort coffee, bi-chromatic systems are more complex to produce due to the duplication of optical and detection components, light-splitting devices and more complex signal processing [10].

2.2 Colour Histograms

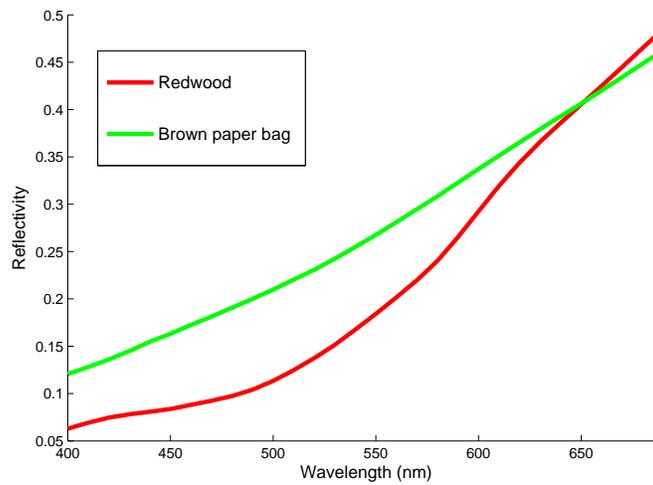
A colour histogram counts the number of times that each possible colour value occurs, colour values are represented as N-dimensional vectors. RGB histograms have been used in image database retrieval [8] and head tracking [37]. The combination of colour histograms and a robust comparison metric can be used to perform colour comparisons that are reasonably robust to mild fluctuations in lighting and object pose. Common reasons for utilising colour histogram comparisons is their robustness to geometric variation of the scene and viewpoint.

2.3 Discussion

The concepts introduced in this chapter allow the automatic histogram alignment problem introduced in section 1.3 to be motivated with further precision. The histogram alignment problem poses colour inconsistency removal as a histogram alignment task, this assumes that colour inconsistencies between colour data captured under different experimental conditions can be removed by aligning similar structures in histograms. This review highlights illuminant and observer colour inconsistencies, in practice both the illuminant and capture conditions are likely to vary together. The plenoptic function provides a general way to describe colour image formation, it can describe single image capture, video and different colour



(a)



(b)

Figure 2.7: Spectral reflectance curves comparing two different sets of objects. 2.7(a) shows the reflectivity of Barley seeds (red) and Bark (green). 2.7(b) shows the reflectivity of Redwood (red) and a brown paper bag (green). All data is from Glassner [6].

space representations.

The histogram alignment approach is attractive as it does not require explicit physical modelling of the illuminant and camera complexities. A transform that aligns colour inconsistent histograms is an implicit model for the colour inconsistencies. Using specific prior knowledge in a constrained situation is always more likely to yield more reliable algorithms, however it is believed that a generic approach is of great value. Ultimately, it is expected that knowledge of the best generic algorithms and application specific algorithms will greatly enhance the flexibility and power of the computer vision designer's toolbox.

Examples of different colour inconsistencies that can be evaluated within a common histogram alignment scheme are:

- Inconsistency between images: A single image samples the plenoptic function at a given instant in time t (*assuming all pixels are captured at exactly the same time*). A second image of the same scene under the same exact camera conditions but different lighting conditions samples the plenoptic function at a different time. Histograms can be computed for the colour data from each image and comparisons made.
- Inconsistencies between portions of a video stream: A video stream captures multiple colour values from the same pixels over time, the colour values from a single or multiple pixels can be represented as a histogram. Comparison between different video stream histograms compares the colour inconsistencies that exist between the different capture conditions.
- Different colour spaces: The fact that different colour spaces schemes are important has been highlighted. Different spectral sampling arrangements sample different ranges of λ in the plenoptic function. Generic histogram alignment methods that work for histograms of different dimensions would be useful because data from a particular colour space can be simply histogrammed and passed to the histogram alignment algorithm to perform colour inconsistency removal.

The flexibility of the histogram alignment approach is that the same methods can be applied to these and different scenarios as desired. Of course, the performance of a histogram alignment approach must be validated independently using appropriate data.

2.4 A taxonomy of colour inconsistency correction methods

This section presents a taxonomic organisation of existing literature on colour inconsistency correction in computer vision. The main aim of the taxonomy is to understand the relevant advantages and disadvantages of these methods.

The goal of colour inconsistency correction methods is to adjust the colour of some or all of a set of colour data points so that ambiguous mappings from an object's material properties to observed colour are removed. Metamerism cannot be removed by transformation of the data points using information in the colour data-set alone, instead metamerism should be controlled by controlling the lighting and camera set up; the method of Sanders [38] dynamically adjusts camera settings to minimize metamerism

to improve an object recognition application. In the absence of metamerism, colour inconsistency can be removed by alignment of the colour data-points.

The methods introduced in this thesis aim to reduce colour inconsistencies between different colour data-points where a histogram can be computed for each set of data-points. This class of problems is termed *between set alignment of colour data points*, and is emphasised in this taxonomy. For completeness, a class of popular colour inconsistency corrections that correct data-points within a single set are discussed briefly, these are termed *within set alignment of colour data point* methods. The development of histogram alignment algorithms is motivated by the wide range of *between set* problems. The taxonomy views all methods in the *between set* class in terms of the colour data-point sets to be aligned. Even when colour histograms are not directly used in a colour inconsistency correction method, it is useful to think about what happens to the colour histograms of the colour data-point sets to be aligned. Colour histogram alignment algorithms must do two things: 1) Identify salient features or class labels of the histograms to be aligned, and 2) Apply appropriate transforms to align the corresponding features or labels. This taxonomy organizes the *between set* methods according to how the features or class labels are obtained, this reveals how prior knowledge is embedded into a method and therefore how applicable it is in other domains. If all colour data-points had correct and unambiguous material class labels then the colour inconsistency correction problem would reduce to finding the best alignment transforms. In reality, labelled data are rarely available and so methods to extract features and labels from the colour histograms is critical. Ultimately, better labelling of the colour data allows more powerful alignment transformations to be applied. A graphical overview of the taxonomy is shown in Figure 2.8. The categories of the taxonomy are:

- Within set alignment of colour data points

A number of notable colour inconsistency corrections fall into the *within set* category. Vignetting removal methods seek corrections for the light attenuation that occurs near the edges of the image. It is common to take training images of constantly illuminated objects with homogenous material properties. Given these images, it is assumed that vignetting is due to lens aberrations which can be corrected. The GermanTeam Robocup entry [22] and the anti-vignetting method of Yu [39] both take this general approach. The GermanTeam method finds a spatially dependent correction in each colour band in YUV space to correct vignetting. Yu's method handles noisy reference images using a wavelet de-noising method and finds the parameters of a vignetting model to perform the correction. Zheng's method for vignetting correction [40] computes the parameters of a vignetting correction model from a single arbitrary image. It repeatedly segments the image into homogeneous intensity regions and then uses the regions to estimate a vignetting function, the procedure is iterated until convergence. Zheng's approach highlights the inter-connection between segmentation and colour inconsistency correction; the performance of the segmentation procedure and the vignetting correction are coupled. These vignetting correction methods are informative, but not generally applicable to other colour inconsistency correction problems. The correction methods and models used are specific to the vignetting problem.

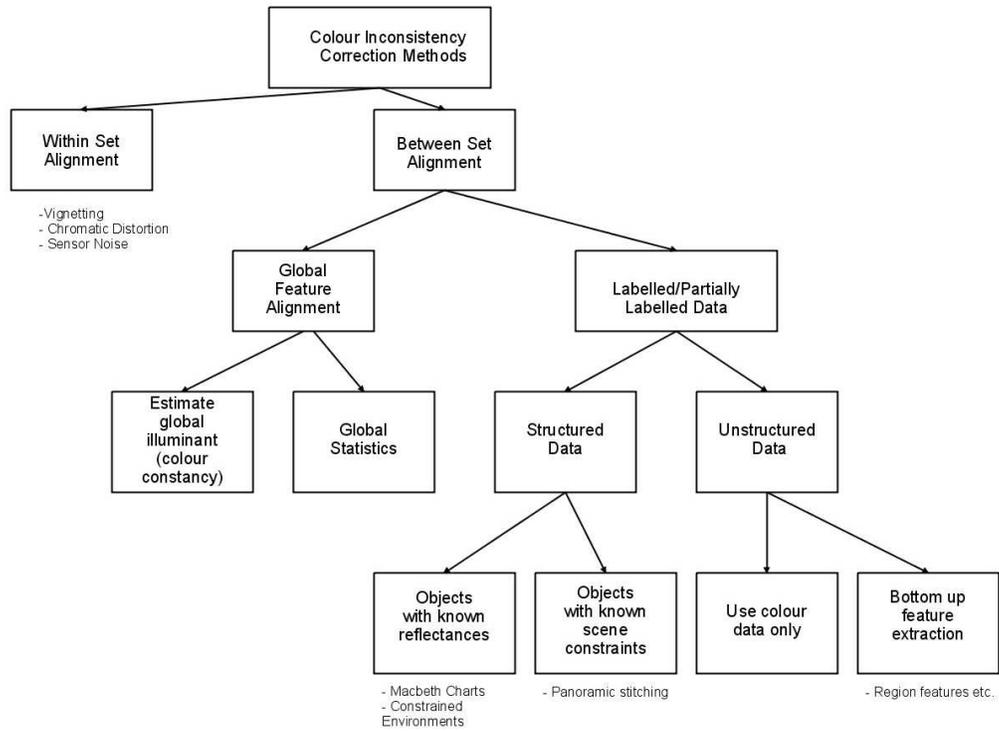


Figure 2.8: Introduced taxonomy of colour inconsistency correction methods.

Examples of other within-set corrections are chromatic aberration and sensor noise removal. Chromatic aberrations can be removed using an active vision system that dynamically brings different colour bands into focus to find a correction [24]. This method requires specialised hardware and specific test image patterns that make the method generally difficult to apply. Sensor noise is minimized [41][42] by acquiring dark images with the lens cap on. These noise minimization methods acquire a uniform reference field from a single camera view to align the colour response of the individual pixels.

- **Between set alignment of colour data points**

Methods in this category can be thought of as histogram alignment methods although they may not act explicitly on histograms. Colour data points are grouped into sets and a histogram is computed for each set. Relationships between the sets are used to align colour responses. A set of colour values is obtained by sampling the plenoptic function. Different applications use different sampling schemes; two different examples are 1) the colours acquired from a single pixel video sensor over time and 2) a single frame in a video sequence. Naturally, it is only sensible to compare histograms obtained from colour data sets that have been sampled in a similar manner.

An application may align a pair of histograms (e.g. two images of a scene taken with different lighting) and others may require alignment of a larger number of histograms (such as subsequent images in a video sequence). When it is known which histograms are more similar the problem is called an *ordered set histogram alignment problem*. For example, during a video capture of rice falling down a chute it is known that histograms from pixels close together are more similar than from pixels that are far apart. In other problems, no knowledge of the ordering is known in advance and this is called the *unordered set histogram alignment problem*. For example, reducing the colour inconsistencies between similar objects in randomly chosen image pairs from an image database.

- **Global feature alignment**

- * Utilize knowledge of colour formation

Colour constancy is a heavily researched area that aims to recover the scene illuminant of an image. Colour constancy estimates the scene illuminant and finds a mapping transform to a common (*canonical*) illuminant. Scene descriptions that are transformed to the canonical illuminant are considered to be illuminant invariant. These approaches can be divided into statistics based and physics based approaches [43]. The physics based methods build on models of material properties such as the dichromatic reflection model [44] and statistical methods correlate colours in the scene with statistical knowledge of the spectral power distribution of common lights and material properties of common surfaces. The initial motivation for research into the area of colour constancy was provided by the ability of humans to recognize colours of objects constantly under different lights. The Retinex model [45] has a basis in human perceptual modeling. Ciocca *et*

al. [46] have evaluated Retinex for preprocessing images to reduce dependency on illumination variation in an image retrieval task. The algorithm assumes constant scene illumination and objects with Lambertian material reflectance properties. A comprehensive review of computational colour constancy algorithms is provided by Barnard *et al.* [47]. They identify algorithms that utilize increasingly stringent assumptions about the nature of the light and scene. The problem with the computational colour constancy approach is that good results can only be obtained on highly constrained imagery. Finlayson [48] has shown that existing methods are not good enough to facilitate colour object recognition across a change in illumination during a database retrieval task; he also notes that no existing method accounts for device independence. This final point is critical as it means that colour constancy methods are not generally applicable across different uncalibrated cameras. Colour constancy methods do not take account of the different sources of colour inconsistency introduced by camera variations. Calibrating cameras to a common reference colour space requires detailed inspection of imaging charts and the use of involved procedures such as Barnard *et al.* [47]. In practice it is not possible to calibrate the response of all cameras in this way.

- * Use global properties of distribution.

The Von Kries transform is a multiplicative adjustment of the means in each channel [49] and the Grey-world transformation shifts a colour distribution so the mean colour is grey. These simple transforms were originally introduced as models of human colour constancy, the problem with these simple transforms is that they can perform well for some classes of images and poorly in others [50]. The colour transfer method of Reinhard *et al.* [51] transforms the colours of a source image to be perceptually similar to the colours of a target image. The method transforms from RGB to a de-correlated perceptual colour space [52], the mean and variance are aligned in this space before transforming back to the original RGB space. The method performs the alignment in a perceptually based space so that alignment along the axes of the space corresponds to improving the matching perceptual factors. The authors claim to use a device independent colour transform but their transform simply matches a single white point. True device independent mapping requires further characterisation of the cameras, the implication is that this method will perform quite differently across different uncalibrated cameras.

The method of Xiao and Ma [53] has similar aims to that of Reinhard [51] but seeks a transformation in the RGB space. The method performs two separate SVD decomposition of the RGB covariance matrices of the source and target distributions. The principal axes are assumed to correspond according to their ordered variance which is given by ordering the principal axes according to the size of their corresponding Eigenvalues. The corresponding axes are then used to find a transformation that is composed of a shift,

rotation and scaling. The method is known to fail on highly multi-modal imagery as the shapes of these distributions can change significantly. Manually segmenting image pairs and applying the method to corresponding image regions is suggested by the authors in these cases. The colour transfer method of Pitie [54] repeatedly projects the RGB histogram onto a randomly oriented two-dimensional plane passing through the centre (grey-point) of the histogram. One dimensional histogram matching is performed on the marginal distributions of the 2D projected histogram. This process is iterated for a set-time to determine an overall mapping function, the mapping is then applied to re-colour images. The problem with this method is that it transforms a source distribution to be equivalent to a target distribution, when run to convergence this destroys true features of the source distribution. Moreover, features of the original histogram are likely to be destroyed when stopping the algorithm early as suggested by the authors. This procedure has been used to transform images for visual effect but the stopping criteria for the algorithm are ad-hoc and are of questionable value in system that requires statistical correctness.

The methods of Reinhard [51], Xiao and Ma [53] and Pitie [54] are all evaluated on a small number of images. The visual results of transformed images are presented. No end-user studies or quantitative evaluations are performed. The main advantage of the global transformation methods presented is their simplicity, which allows them to be applied to align colour histograms with little concern for the nature of the data. The main limitation of these global methods is that they do not make use of informative local features of the data distributions which can in turn lead to poor performance.

– Labelled/Partially labelled data

The methods at this level of the taxonomy attach labels to the data from the different alignment sets, then they transform the colour data to align the corresponding labels. Methods to label the data are often highly specialized and cannot be applied to other problems; these methods can be grouped according to whether they assume a particular structural form to the data such as the presence of particular objects in the scene or that objects will appear in a predefined order in a video sequence.

* Structured data

Objects with known reflectances can be introduced into a scene to reduce the complexity of finding corresponding points in colour space. A MacBeth chart is a standard chart used for colour management that has 24 patches of known reflectance (shown in Figure 2.9). Typically, 24 colour points in RGB space are computed by finding the mean colours of regions obtained from each patch. The process is repeated with different cameras or lighting conditions and another 24 RGB points are found; the correspondences between points are known, so point alignment transformations can be applied to align the colour histograms. A chart-based approach has been used in a multi camera food

inspection environment by Tao [55]. In diagnostic imaging, Colposcopy is a method to identify cervical cancer by ranking lesions in order of severity. Colour inconsistencies affect the ability of physicians to make meaningful comparative diagnostics; the method of *Li et al.* [56] calibrates cameras for colposcopy using a grey chart and a standard MacBeth colour chart. The method first removes vignetting variations by inspecting the grey chart. A general camera transform is modelled as a homogeneous 4×3 transform followed by a third order polynomial transform in each channel. The transforms are computed to align the corresponding source and target patches extracted from MacBeth charts.

Illie and Welsch [57] improve the colour consistency between multiple colour cameras for use in a photometric stereo system. Their method places a Macbeth chart in the shared field of view of the cameras. The chart and squares are automatically detected. The calibration process has two main steps: 1) hardware parameters of the cameras are adjusted to minimize the variance of the same patches obtained from different camera views. 2) Different alignment transforms are computed that align the mean colours of the patches obtained from different cameras. The transforms explored are the 3×3 RGB transform and a hierarchy of polynomial transformations. The polynomial transforms perform best according to the introduced criterion. The quantitative evaluations are performed with the chart, no indication of the effect on more general imagery is given. Macbeth charts are a powerful tool that facilitate a number of powerful colour inconsistency removal methods. However, it is often impractical or impossible to insert a Macbeth chart into a scene each time the lighting changes to perform a re-calibration. In many scenarios it may be impossible to place a Macbeth chart into a scene at all.

An alternative to using colour charts is to deliberately construct situations that limit the complexity of the scenes. Robocup is an annual robotics competition that requires autonomous robots to compete in a game of soccer. Colour calibration has a significant effect on the performance of these robots and the methods used take advantage of the fact that the main object colour classes are known in advance. For example, it is known that the ball is always orange, the terrain is green and the pitch lines are white. Jungel [58] developed a calibration system for Aibo robots that utilises this prior information from the spatial and colour domain. These approaches take advantage of scene knowledge to develop robust approaches but are highly specific to the task at hand.

Image overlap is a constraint utilized in the field of panoramic image stitching where multiple photographs of a scene are stitched together to produce a larger panoramic image. Different images obtained as an input to a panoramic stitching process exhibit differences in colour for the matching pixels due to local variations such as vignetting and global variations between cameras such as exposure time, white balance, gain and so on. Failure to compensate for these radiometric differences results in visible seam

stitches between the images. Numerous methods [59][27][26][60] find partial label correspondence in the matched overlapping regions and use this information as part of their colour histogram alignment procedure. Tian and Gledhill [60] apply diagonal, 3×3 and homogenous 4×4 (*affine*) transforms that align the histograms of the overlapping regions. Jia and Tang [26] find a vignetting function and a separate global monotonic correction function per image. The functions are found by a tensor voting approach that seeks local smoothness; in this approach, no explicit model of vignetting or camera effects is specified. The approaches of Litvinov and Schechner [59] and Goldman and Chen [27] develop explicit models of colour inconsistency effects and use the matched regions as part of the fitting procedure.

* Unstructured data

The term *unstructured data* is used here to refer to data that is not tuned for the task of colour labeling.

- **Colour data only:** labels are attached to colour data from colour histograms alone. Jeong and Jaynes [61] propose a colour transfer methodology to improve object tracking performance between multiple cameras with non-overlapping fields of view. The first step of their process performs background and foreground modeling that assumes the foreground is moving. All moving pixels are assembled into an appearance model in the U-V space for each camera view. An affine transformation is found between U-V histograms for each camera by fitting a Gaussian mixture model to each histogram and then aligning the corresponding components of different models. This method is interesting as the processing after the motion-detection step is performed entirely in colour space. However, the method does not perform consistently better than the diagonal transform on RGB histograms. The results show that the method performs worse than the diagonal model when tracking low numbers of objects. Performance relative to the diagonal model improves for higher numbers of objects. The use of the U-V colour space means that dependency on illumination is reduced. However, bringing the colour response of multiple cameras into the same reference space requires precise characterisation of each camera's response using imaging charts. This step is not performed and so the U-V colour space correspondence can be approximate at best. This approach shows the potential of histogram alignment for improving the performance of a tracking application but raises questions about the suitability of the GMM feature mapping approach as the best way to do this.
- **Incorporate other features:** Fredembach *et al.* [62] propose a region based image labeling approach to improve automatic colour correction methods. The stated aim of this approach is to label image regions according to labels such as *skin* and *vegetation*. The idea is that once labeled, the colours of corresponding regions can

be adjusted. The method performs a segmentation in DC-Lab space by performing K-means clustering with $K=8$ and then merging clusters that are below a manually set distance threshold. Colour features that measure the blue content of the image are also introduced and the method is tested on a number of images containing different objects on backgrounds with a high blue content. The number of scenes evaluated is limited and there is no quantitative evaluation of colour inconsistency removal performance.

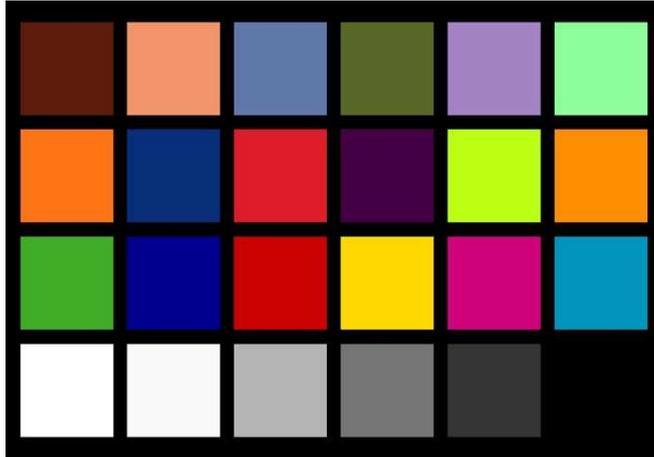


Figure 2.9: A Macbeth chart, commonly used for colour calibration tasks.

2.5 Transformation Methods

This section catalogs colour inconsistency transformations and methods for computing the transformations. Transforms are related to the different methods described in the taxonomy.

2.5.1 Transformations

Between set methods in the taxonomy use transformations that fall into three categories. These are 1) Independent polynomials in each channel, 2) Correlated Polynomials in each channel and 3) General monotonic transforms in each channel. Each transformation moves an n -dimensional colour value \mathbf{s} to a new position in colour space \mathbf{q} . The scalar values s_i and q_i are the i th elements of the respective $n \times 1$ column vectors, \mathbf{s} and \mathbf{q} . Expanding these categories further:

1. **Independent Polynomial** An order d polynomial is applied to each dimension separately. For the i th dimension,

$$q_i = \alpha_{i0} + \sum_{k=1}^d \alpha_{ik} s_i^k. \quad (2.2)$$

The transform is called independent because the i th channel is not related to other channels. Equation 2.2 is the general form of the following transforms.

- **Additive.** $d = 1$ and the coefficient α_{ik} is set to 1 reducing the equation to a simple offset.

For a colour data-point,

$$\mathbf{q} = \mathbf{s} + \mathbf{a}, \quad (2.3)$$

where \mathbf{a} is a $n \times 1$ vector of scalar offsets. It is not common the use the additive transform for inconsistency correction, although many transforms contain an additive element. The additive transform is worth considering for its simplicity.

- **Multiplicative.** $d = 1$ and the additive coefficient, α_{i0} , is set to 0. The multiplicative transform is the dominant model in colour constancy. It is also called the Von Kries transform, diagonal model or *gain* transformation. For a colour data point,

$$\mathbf{q} = \mathbf{D}\mathbf{s}. \quad (2.4)$$

\mathbf{D} is an $n \times n$ diagonal matrix where diagonal entries are the multiplicative scaling factors in each channel.

- **Linear.** $d = 1$. The linear transform of a colour data point can be represented using $(n+1) \times 1$ homogeneous representations of \mathbf{q} and \mathbf{s} , \mathbf{q}_h and \mathbf{s}_h , such that,

$$\mathbf{q}_h = \mathbf{T}\mathbf{s}_h. \quad (2.5)$$

\mathbf{T} is an $(n+1) \times (n+1)$ homogeneous matrix. The multiplicative elements are in the i th row and j columns where $i = j$ and $i = 1..n$. The additive elements are indexed by i , $(n+1)$ where $i = 2..(n+1)$. All other elements are zero. \mathbf{T} represents a scaling, rotation and shift of the colour data-point \mathbf{s} .

- **General.** $d \geq 2$. Polynomials of order 2 or greater are represented as a separate matrix multiplication in each channel.

$$q_i = \begin{bmatrix} 1 & s_i & s_i^2 & \cdots & s_i^d \end{bmatrix} \begin{bmatrix} \alpha_{i0} \\ \vdots \\ \alpha_{id} \end{bmatrix} \quad (2.6)$$

2. **Correlated Polynomial** An order d correlated polynomial relates the i th dimension to all other dimensions by the relationship

$$q_i = \alpha_{i0} + \sum_{k=1}^d \sum_{j=1}^n \alpha_{j(d(k-1)+j)} s_j^k. \quad (2.7)$$

- **N by N similarity transform:** $d = 1$ and the additive coefficient, α_{i0} , is set to 0. An $n \times n$ matrix pre-multiplies \mathbf{s} according to:

$$\mathbf{q} = \mathbf{M}\mathbf{s}. \quad (2.8)$$

\mathbf{M} represents a scaling and rotation of the colour data-point \mathbf{s} .

- **General:** Equation 2.7 can be represented by multiplying a $1 \times (1 + nd)$ vector by a $(1 + nd) \times 1$ vector. Writing this for the case $d = 2, n = 2$ as an example.

$$q_i = \begin{bmatrix} 1 & s_1 & s_1^2 & s_2 & s_2^2 \end{bmatrix} \begin{bmatrix} \alpha_{i0} \\ \alpha_{i1} \\ \alpha_{i2} \\ \alpha_{i3} \\ \alpha_{i4} \end{bmatrix} \quad (2.9)$$

This is repeated for each colour channel.

2.5.2 Methods For Computing The Transformations

This section describes different methods to compute the transformations introduced in the previous section.

1. **Aligning moments:** A moment generating function represents a distribution in terms of its moments. The n th moment is

$$m_n = \int_{x \in D} x^n f(x) dx \quad (2.10)$$

In this work, the alignment of the first and second moments is considered. The first moment is also known as the mean or expected value, $E(X)$. The second moment is closely approximated by the variance $Var(X)$ which is the average squared deviation from the mean. The square root of the variance is the standard deviation, $SDev(X)$. The mean is a common summary measure of a colour distribution, the grey world algorithm assumes the mean colour in a scene is grey and aligns the corresponding mean colours in different images using a multiplicative transform. The colour transfer method of Reinhard *et al.* [51] aligns the mean and variance of a source and target colour distributions in each channel.

The distribution of a random variable S can be aligned with the distribution of a random variable Q using the following methods:

- **Mean alignment using additive transform** The scalar elements of equation 2.3 are

$$q_i = a_i + s_i. \quad (2.11)$$

The offset in the i th channel, a_i , is computed as

$$a_i = E(Q) - E(S). \quad (2.12)$$

The same offset is used for all data-points.

- **Mean alignment using multiplicative transform** The multiplier r_i in the i th channel is the entry in the i th row and column of equation 2.4. It is computed as:

$$r_i = \frac{E(Q_i)}{E(S_i)} \quad (2.13)$$

- **Mean and Variance alignment using linear transform** Setting $d = 1$ in equation 2.2 gives the linear equation,

$$q_i = \alpha_{i0} + \alpha_{i1} s_i. \quad (2.14)$$

The mean and variance are aligned between distributions S and Q using equation 2.14. The multiplicative coefficient α_{i1} is:

$$\alpha_{i,1} = \frac{SDev(Q_i)}{SDev(S_i)}, \quad (2.15)$$

and the additive coefficient α_{i0} is:

$$\alpha_{i0} = E(Q) - \alpha_{i1}E(S). \quad (2.16)$$

2. **Point Alignment Transforms:** Local features of colour histograms can be identified by points in the colour space. The Macbeth chart alignment method of Illie *et al.* [57] identifies corresponding points in the RGB histograms of images by finding the mean colours in each of the coloured squares on the chart. A point alignment transform moves a set of source points so the residual distance between the transformed points and the target points is minimized. This transform is then used to transform all source data-points. Illie *et al.* [57] apply this method to compute 3×3 RGB and second order correlated polynomials to calibrate multiple colour cameras, they find the correlated polynomials perform best in their application.

This segment describes methods for computing different transforms of the l source points to the l target points. The source and target points are represented by two $l \times n$ matrices, \mathbf{S} and \mathbf{Q} respectively. s_{ji} and q_{ji} are the scalar values of the j th points in the i th channels of \mathbf{S} and \mathbf{Q} . The transformation parameters that minimize the distance between the transformed source points and the target points can be solved using the following methods:

- **Align points with additive transform.** The additive shift, a_i , for the i th channel is computed as:

$$a_i = \frac{\sum_{j=1}^l q_{ji}}{l} - \frac{\sum_{j=1}^l s_{ji}}{l}. \quad (2.17)$$

All entries of the i th row of \mathbf{a} in equation 2.3 are set to a_i .

- **Align points with multiplicative transform.** The multiplier, r_i , is the i th diagonal element in the matrix \mathbf{D} in equation 2.4. The i th row of \mathbf{S} is \mathbf{x}_i and the i th row of \mathbf{Q} is \mathbf{y}_i , so,

$$r_i = \mathbf{y}_i^T \mathbf{x}_i. \quad (2.18)$$

This is repeated for all channels.

- **Align points with independent polynomials** A separate polynomial transformation is computed for each channel. An order d polynomial is computed in the i th channel that aligns the scalar source and target values. Writing each source and target value as linear constraints on

the coefficient values as:

$$\begin{bmatrix} 1 & s_{1,i} & s_{1,i}^2 & \dots & s_{1,i}^d \\ 1 & \vdots & \vdots & \vdots & \vdots \\ 1 & s_{l,i} & s_{l,i}^2 & \dots & s_{l,i}^d \end{bmatrix} \begin{bmatrix} \alpha_{0,i} \\ \vdots \\ \alpha_{d,i} \end{bmatrix} = \begin{bmatrix} q_{0,i} \\ \vdots \\ q_{d,i} \end{bmatrix}. \quad (2.19)$$

Writing this as $\mathbf{A}_i \mathbf{C}_i = \mathbf{D}_i$ and solving for \mathbf{C}_i gives $\mathbf{C}_i = \mathbf{A}_i^\dagger \mathbf{D}_i$ where \mathbf{A}_i^\dagger is the pseudo-inverse of \mathbf{A}_i . Solutions are found for each colour channel.

- **Align points with correlated polynomials** The relationship between each source and target point can be described by a correlated polynomial in each colour channel. Writing this in matrix form:

$$\begin{bmatrix} 1 & s_{1,1} & \dots & s_{1,n} & \dots & s_{1,1}^d & \dots & s_{1,n}^d \\ 1 & \vdots & \dots & s_{1,n} & \vdots & \vdots & \dots & \vdots \\ 1 & s_{l,1} & \dots & s_{1,n} & \dots & s_{l,1}^d & \dots & s_{l,n}^d \end{bmatrix} \begin{bmatrix} \alpha_{0,1} & \dots & \alpha_{0,n} \\ \vdots & \dots & \vdots \\ \alpha_{nd+1,1} & \dots & \alpha_{nd+1,n} \end{bmatrix} = \begin{bmatrix} q_{1,1} & \dots & q_{1,n} \\ \vdots & \dots & \vdots \\ q_{l,1} & \dots & q_{l,n} \end{bmatrix}. \quad (2.20)$$

Writing this as $\mathbf{AC} = \mathbf{T}$ and solving for \mathbf{C} gives $\mathbf{C} = \mathbf{A}^\dagger \mathbf{T}$.

- **Align points with N by N transform.** Rearranging $\mathbf{S} = \mathbf{MQ}^\mathbf{T}$, gives,

$$\mathbf{M} = \mathbf{Q}^\mathbf{T} (\mathbf{S}^\mathbf{T})^\dagger. \quad (2.21)$$

3. **Histogram matching and equalization:** The standard histogram equalization operation finds a monotonic transformation of a 1D histogram so that the intensity distribution across its bins are uniform. Histogram equalization finds a monotonic transform of the original intensity values so that the cumulative distribution of the transformed values is linear. Finlayson *et al.* [48] apply histogram equalisation to improve image retrieval rates from an uncalibrated image database. They transform all images with a histogram equalisation in each individual channel prior to the retrieval step.

Histogram matching finds a monotonic transform of the source histogram intensity distribution that matches the distribution of a target histogram. Pitie *et al.* [54] use repeated histogram matching as part of their colour transfer method.

The problem with histogram equalisation for colour inconsistency removal between images is that the available colour information in both images is not directly related. The histogram equalisation transform only depends on the form of the one dimensional input histograms, this means that corresponding features in the histograms are ignored. The problem with the histogram matching method is that any scale variations between corresponding clusters will be removed; this is erroneous when seeking a transform that removes lighting and camera effects only. Pitie *et al.* [54] compound this problem with their iterative algorithm to apply histogram matching along randomly projected axes.

4. **SVD based principal axis alignment:** The SVD colour transfer method of Xiao and Ma [53] computes a homogeneous rotation, scaling and translating that aligns the principal axes and means

of a source and target data-set. The method separately decomposes the covariance matrix of the source and target image data using an SVD decomposition, the distribution means are aligned and a rotation and scaling is computed that aligns the nearest axes. The method computes \mathbf{T} in equation 2.5, it assumes that the entire RGB colour distribution of the entire image is well modeled by an spheroid. This assumption can break down due to the multi-modal nature of the distributions, if individual modes move and deform independently the enclosing spheroids of the source and target distributions may not correspond correctly.

2.6 Motivation

This chapter has described the problem of colour inconsistency and techniques that are used to manage and correct these inconsistencies. The taxonomy organises colour inconsistency removal techniques according to how they attach labels to colour data. Different branches of the taxonomy incorporate different levels of prior knowledge of the problem domain into the method. Methods that incorporate high levels of prior knowledge typically perform well in the domain that they are designed for but are inapplicable or generalize poorly to other domains. Methods that label colour data reliably and accurately can apply more powerful alignment transforms than methods that label the data approximately. This thesis proposes that generic solutions to the colour inconsistency correction problem can be developed by solving the between set histogram alignment problem. The aim is to detect local features of histograms and apply point alignment transforms to perform the alignment, this rationale is explored because point alignment transforms have proven highly successful when using structured data methods with objects of known reflectance (such as MacBeth charts) [57][63]. To date, these transforms have not been applied from the colour histograms alone; the object tracking calibration method of Jeong [61] comes the closest to achieving this, but incorporates a motion segmentation step as the first part of the processing. In addition, it has only been tested in U-V space and depends on the Gaussian mixture model which is often a poor model for the real shapes of distribution. Nonetheless, Jeong's work is promising as it suggests that this approach is relevant to the field of object tracking.

Despite the proliferation of different methods and transforms, we find no comprehensive studies that explain which methods are best for minimizing colour inconsistency. It is common to study a colour inconsistency correction technique within an application framework such as colour image retrieval [46] or object recognition [64], these studies show that colour inconsistency methods improve performance within these frameworks but they do little or nothing to explain the details of the relative performance and behaviour of colour inconsistency methods. In addition, we find no data-bases that are constructed for the study of colour inconsistency correction that allow the alignment of all relevant local modes in the colour histograms to be easily tested.

The key driver for this project is the colour inconsistency problem encountered by the industrial partner Buhler Sortex. The research in this thesis has been conducted to add value to the proprietary methods described in Chapter 5, but also to relate these proprietary methods to other techniques used in the wider vision community. Understanding the Buhler Sortex methods and how generic colour inconsistency methods can be applied to both Buhler Sortex data and more general imagery helps understand the

problems faced in each area. It is informative to see how pragmatic solutions can be built on industrial technology with an eye on the wider developments and trends in the vision community.

In summary, this review motivates the need for generic feature based histogram alignment methods and a study of their performance on different colour inconsistent data-sets.

Chapter 3

Feature based histogram alignment

This chapter introduces a feature based histogram alignment (FBHA) method to align a source RGB histogram with a target RGB histogram. Aligning the colour histograms of images computes a colour transformation that aligns the colours of a source image with those in a target image. This chapter considers the case where two colour inconsistent images contain the same set of N single-coloured objects. Each histogram contains a number of dense regions that correspond to objects of interest. FBHA seeks a transform that aligns clusters that correspond to the same objects. FBHA is designed to handle multiple clusters of potentially different size, no explicit assumption is made about the shape of the distributions or the number of clusters present. FBHA assumes that the source and target images are of the same set of objects.

3.1 Feature based histogram alignment algorithm

The section outlines the FBHA algorithm. The steps to compute a colour transformation from a source image to a target image using FBHA are:

1. Compute histograms for the source and target images.
2. Compute the scale space of each histogram and extract salient features.
3. Reject obvious outlying features.
4. Match the remaining features.
5. Compute the coefficients of a point alignment transform to align matching features.
6. Transform the source image.
7. Test for failure by comparing the transformed and target histograms.
8. If FBHA fails, revert to a moment based transformation.

This algorithm was introduced by Senanayake and Alexander [65] to process the individual R,G and B channels of images. The treatment in this chapter is more general and explores the approach in more detail. The following subsections detail the steps of the algorithm and explain the rationale for taking this approach.

3.1.1 Feature Detection

This section introduces a feature detection step to find significant maxima in colour histograms. The approach uses methods developed in the realm of scale space analysis. First, background scale space theory is introduced and then the feature detection methods are described.

Background to Scale Space Methods

Meaningful structure exists at different scales in the world and so meaningful structure in images can exist at different scales. Scale space methods are techniques for extracting information from signals when the relevant scales are not known in advance. Koenderink [66] introduced a scale space theory for processing visual imagery using differential geometric descriptors. The scale space approach has motivated the popular SIFT feature [67] detection method that computes scale invariant local features from images. Lindeberg [68] provides a detailed review of scale space theory. The scale space representation of a N dimensional signal, f , can be computed by convolution with Gaussians, $g(\sigma)$ of varying widths, σ as

$$L(\sigma) = g(\sigma) * f, \quad (3.1)$$

where the Gaussian kernel is

$$g(\sigma) = (2\pi)^{-N/2} |\Sigma| \exp(-(\mathbf{x} - \mathbf{u})^T \Sigma^{-1} (\mathbf{x} - \mathbf{u})/2). \quad (3.2)$$

The covariance, Σ , is a diagonal matrix with diagonal entries set to σ^2 . This defines an isotropic Gaussian with standard deviation σ along each dimension. Convolution of a signal with a Gaussian at increasing widths blurs the detail of the signal until eventually all detail is smoothed away and a single mode remains. Scale space methods process the scale space of signals to extract meaningful information. The term deep structure has been used to refer to linked structures that can be extracted through the different levels of the scale space [66].

Deep structure feature detection

The local maxima of a N dimensional histogram, \mathbf{H} , provide local structure information. This section introduces a new deep structure feature detection method that avoids spurious local maxima detection. First, it removes maxima below a noise threshold, γ . Second, it finds local maxima at each level of the scale space that are connected to form paths over at least T levels. The local maxima in the histogram, \mathbf{H} , that lead to these scale space path are retained as salient features, \mathbf{F} . \mathbf{F} is a $v \times N$ matrix of feature points, each row indicates the co-ordinates of a detected feature in the histogram; the number of detected features, v , is specific to a histogram and the parameters chosen. The pseudo-code function $\mathbf{F} = \mathbf{FindPersistentMaxima}(\mathbf{H}, T, \text{BlurScales}, \gamma)$ summarizes these steps in algorithm 1. The parameters of the algorithm are:

1. T : the path length.
2. *BlurScales*: the T scales used for the blurring. The i th blurring parameter is σ_i . $\sigma_i = e^{0.1(i-1)}$, where $i = 1..T$.

3. γ : the noise floor parameter.

The function *FindPersistentMaxima* generates and parses the scale space of \mathbf{H} . It maintains an internal structure *DeepStructurePaths* that contains all information about the detected paths. There are two elements to constructing paths: 1) local maxima are detected using the function *DetectLocalMaxima* and 2) maxima connected across scales are added to a path stored in *DeepStructurePaths* by the function *FollowPaths*. For histograms of different dimensionality there are different ways of computing local maxima and different possible connectivity rules across scale. In 1D, a maximum occurs in a bin that is greater than its two neighbouring bins, in 2D the four N-E-S-W or all eight neighbouring bins can be inspected. Two maxima are connected across scales if they have bin positions that are connected by a pre-defined shape. In the 1D case the three neighbouring bins at the next level of the scale space are tested for connectivity. In the 2D case, the 5-connected or 9-connected bins at the next scale are examples of alternative connectivity rules. The maxima detection and connectivity rules are specified when applying the algorithm to data.

An example feature detection process in 1D is now given. Figure 3.1 shows a histogram of the green channel data from an image of plastic skittles. Local maxima are detected where a bin value is greater than its two neighbours. There are a large number of irrelevant local maxima detected in this histogram, Figure 3.1(b) highlights that a single cluster can contain many local maxima due to local spikiness at the top of a cluster. Note that local maxima are detected in the noisy portions of the histogram, these noisy maxima are discarded during the call to *ThresholdBins* with noise floor threshold, γ . Figure 3.2(a) shows the scale space representation of the histogram signal and local maxima at each scale, notice how maxima that are detected in the noise that surrounds a dominant cluster centre are eliminated as the corresponding feature paths terminate early in the scale space. Figure 3.2(b) shows the histogram and features that lead to paths at least 20 long by setting $T = 20$.

Algorithm 1 $\mathbf{F} = \text{FindPersistentMaxima}(\mathbf{H}, T, \text{BlurScales}, \gamma)$

$LM \leftarrow \text{DetectLocalMaxima}(\mathbf{H})$

$PrevLevelLM \leftarrow \text{ThresholdBins}(LM, \gamma)$

$DeepStructurePaths \leftarrow \text{InitialisePathStructure}()$

for $i = 1$ to T **do**

$ThisLevelLM \leftarrow \text{DetectLocalMaxima}(\mathbf{H} * \mathbf{g}(\sigma_i))$

$\text{FollowPaths}(DeepStructurePaths, PrevLevelLM, ThisLevelLM)$

end for

$\mathbf{F} \leftarrow$ Features at the start of the paths with lengths of T .

3.1.2 Feature Matching

This section introduces methods to match source histogram features, \mathbf{W} , with target histogram features, \mathbf{Q} . There is no guarantee that the number of detected source features, a , is the same as the number of detected target features, b . The goal is to find the best set of one to one assignments between a and b , each set of assignments is called a *match*. A maximum cardinality match finds the maximum number

of one to one assignments as a solution, there are $\frac{a!}{(a-b)!}$ solutions when $a > b$, $\frac{b!}{(b-a)!}$ solutions when $b > a$ and $a!$ solutions when $a = b$. Choosing the best match requires a notion of cost between points. The total Euclidean distance between the k th set of matched points is computed as,

$$E_k = \sum_{i=1}^{\min(a,b)} L2(\mathbf{w}_i, \mathbf{q}_i), \quad (3.3)$$

where $L2$ is the L2 norm. The match with the minimum Euclidean distance, E_k , is chosen. This value can be found by brute force search by computing all of the possible matches and the corresponding scores, E_k , for all k and then finding the minimum value. The minimum Euclidean distance does not guarantee that matches preserve rank ordering in a channel, this means that it allows folding transformations. In a rank ordered match, both matched points in each channel must be either less than or greater than all other matched points in the channel. Two options for feature matching are evaluated in this work:

1. The maximum cardinality, one to one, minimum Euclidean distance computed with brute force. (Referred to with the code: **CEM**)
2. Use **CEM**, then remove all non-rank preserving matches. This prevents folding transformations. (Referred to with the code: **CEM-DC**)

3.1.3 Feature Alignment

A point alignment transform is selected and used to align the matched features. Transforms with greater degrees of freedom are more flexible but are likely to over-fit the data. The best transformation for an alignment is determined by testing a range of transformations on a data-set to see which transformation performs best. The matching process sets the source points \mathbf{S} and the target points \mathbf{Q} so that the corresponding rows of \mathbf{S} and \mathbf{Q} are the matched points. Then the following steps are performed:

1. The chosen point alignment transform is computed. The point alignment transforms for \mathbf{S} and \mathbf{Q} and the methods for solving them are described in section 2.5.2. Different options for the point alignment transformation are multiplicative (eqn: 2.18), additive (eqn: 2.17), independent polynomial (eqn: 2.19), correlated polynomial (eqn: 2.20) or an N by N transform (eqn: 2.21).
2. The source histogram, \mathbf{A} , is transformed to $P(\mathbf{A})$ using the point alignment transform, P .
3. The Bhattacharyya coefficient, B , is computed between the transformed histogram $P(\mathbf{A})$ and the target histogram, \mathbf{B} as $\sum_{x \in \mathbf{X}} \sqrt{s_x t_x}$, where s_x and t_x are the corresponding bins of $P(\mathbf{A})$ and \mathbf{B} respectively. If the coefficient is less than a threshold, D , the point alignment transform is discarded and a moment based transform is computed to align the histograms.

Steps 2 and 3 are optional, they improve the robustness of the algorithm overall and may not be required when the point alignment transformations are likely to work.

3.1.4 Discussion

This section elaborates on the design choices of the FBHA algorithm and discusses the advantages of the approach taken. The FBHA steps described are summarized by the pseudo-code function *RobustFeatureBasedAlignment* in algorithm 2.

Algorithm 2 *RobustFeatureBasedAlignment*(**A**, **B**)

W = **FindPersistentMaxima**(**A**)**Q** = **FindPersistentMaxima**(**B**)**S**, **Q** = *MatchFeatures*(**W**, **Q**)Compute Point Alignment Transform using **S** and **Q**.Transform source histogram, **A**, using point alignment transform.Compute the Bhattacharyya metric, *B*, between transformed histogram and target.**if** $B < D$ **then** Perform moment based transform of source histogram, **A**.**end if**

The problems encountered in employing standard feature detection techniques motivate the deep structure scale space feature detection technique described. The motivation is to produce a feature detector that:

- Doesn't require the number of clusters to be specified as a parameter.
- Detects features at different scales without parameter adjustment.
- Is not dependent on random initialisation and thus gives consistent results for a single data-set.

Common feature detection methods such as K-Means [69] and EM-GMM [70](pg. 435) approaches require the number of data clusters to be specified as a parameter to the algorithm; both approaches use an iterative procedure to update initial cluster estimates. In K-Means, a data point is updated. In EM-GMM, each cluster is modeled with a Gaussian distribution whose mean and covariance is updated during the procedure. For anything but the simplest distributions, these methods give different results based on the initialisation points, resolving the correct clusterings from these results often involves manual intervention and parameter tuning. GMMs fitted with EM suffer similar initialisation problems to the K-Means algorithm, also data clusters are frequently not Gaussian which leads to poor fits.

Matching low numbers of features with brute force is sufficient, however matching large numbers of points can become expensive using this technique. In this work, the brute force approach is used because low numbers of features are present and the focus of the experimental work is on the histogram alignment performance. If speed of execution of the matching step becomes an issue in future work the Hungarian method [71] can be used to match larger numbers of features efficiently. In this approach, the feature matching problem is represented as a maximum cardinality, maximum flow problem on a bi-partite graph. The graph is bi-partite because there are two types of nodes corresponding to source and target features in this case; a bi-partite match is a one to one correspondence between a source and target feature. A cost is computed for each of the bi-partite matches, and then a maximum flow technique such as Ford-Fulkerson [72] is used to find the $\min(a, b)$ matches. The Hungarian algorithm finds the set of bi-partite matches that maximise the total cost between matches. The total Euclidean distance minimization can be performed using Hungarian matching by altering the cost function to find a maximum value. This is done by subtracting the Euclidean cost from a suitably large constant.

In summary, the FBHA algorithm allows the use of point alignment transforms to be investigated using automatic processing. It is designed to produce stable local features and simple one to one feature matches. A degree of robustness is built into the approach by testing for catastrophic failure of the point alignment transform.

3.2 Qualitative Evaluation

This section qualitatively evaluates FBHA, the aim is to gain an intuition into the behavior of the algorithm and how different options affect its behavior. FBHA is evaluated on image pairs using 1D, 2D and 3D versions of the deep structure feature detector. The images are from the image database described in chapter 4, a full description of the image data is deferred to concentrate on the algorithm behavior. Chapter 4 develops a quantitative methodology for testing colour inconsistency removal and tests colour inconsistency methods comprehensively. The aim of this section is to visually demonstrate:

1. Feature detection using the deep structure method,
2. the matched features,
3. transformed histograms,
4. transformed images.

3.2.1 1D FBHA

1D FBHA operates on pairs of one dimensional histograms. To apply 1D FBHA to dimensions, $N \geq 2$, the procedure is applied in each dimension separately. This section demonstrates 1D FBHA on a colour inconsistent image pair.

Images

Figure 3.3(a) shows the source image and 3.3(b) shows the target image; both images are of a red and cyan piece of paper captured under different lighting conditions. The transformed source image is shown in 3.3(c), its colours appear more similar to those of the target image. The increased visual similarity of the colours gives a qualitative indication that the colour inconsistency has been reduced by the 1D FBHA procedure.

FBHA steps

The deep structure feature detection step described in section 3.1.1 is performed on the red, green and blue channel individually using parameters, $\gamma = 0.005$ and $T = 9$. The detected features and corresponding **CEM** matched features are shown for the red, green and blue channels in figures 3.4(a), 3.4(b) and 3.4(c) respectively. Two features are detected and matched in the red channel histograms. The green and blue channel histogram pairs show three detected features in the target histograms and two detected features in the source histograms; the final matches in the green and blue channels discard one of the features from the target histogram. The matched features are used to compute a linear point alignment transform in each channel, the source image data-points from the red, green and blue channels are corrected using the transforms. A histogram of the corrected values is computed for each channel. The

corrected histograms and their corresponding target histograms are shown for the red, green and blue channels in figures 3.5(a), 3.5(b) and 3.5(c). The corresponding peaks in the corrected histograms are aligned in all three cases.

Observations

The 1D FBHA procedure allows a linear correction to be computed in each channel that aligns the local structure of the histograms. The computed linear transforms in each channel modify the original source image so that it is more similar to the target image; these qualitative results provide an indication of what is possible using the algorithm. The algorithm is able to identify and match the histogram peaks robustly even though the local structure of the histogram peaks is variable. Noticeable artifacts occur in the histograms of the transformed data, the histogram of the transformed red channel data in figure 3.5(a) exhibits spikes and the histograms of the transformed green and blue channel data in figures 3.5(b) and 3.5(c) exhibit gaps. These effects occur because of the discrete precision of image data, the image pixels are represented by integer values in the 0 – 255 range. Transformation of pixel values by a multiplicative transform less than 1 can cause transformed values to bunch together in particular bins which results in the histogram spikes observed in 3.5(a). Gaps in the histograms of transformed values can be produced by a multiplicative transform greater than 1 that effectively stretches the transformed values and so leaves gaps in the histograms as seen in 3.5(b) and 3.5(c). Section 3.2.4 provides further discussion of the issues surrounding these effects and methods to mitigate them.

3.2.2 2D FBHA

The 2D FBHA procedure computes 2D source and target histograms from two colour channels. An RGB image is transformed by running 2D FBHA on the red and green channels and 1D FBHA is on the blue channel. Any two channels could be chosen, but the green and red channels are selected because the standard RGB camera samples the red and green wavelengths more than the blue band. Recall that the human eye samples the red and green bands more than the blue band, this sampling strategy helps the human visual system uniquely identify most objects in the natural world. Reasoning by analogy, one can suppose that the RG histogram is more likely to produce well separated clusters than other channel combinations.

Images

The colour inconsistent image pair in this example contains four different types of coloured object. Figure 3.6(a) shows the source image captured in a room with a florescent lighting and 3.6(b) shows the target image under the same ambient lighting conditions with a red bulb held over the objects. Figure 3.6(c) shows the transformed source image using 2D FBHA on the red and green channels and 1D FBHA on the blue channel.

FBHA steps

The deep structure feature detection on the RG histogram uses $\gamma = 0.0002$ and $T = 11$. The connectivity rule in the path following step connects a local maximum to a current path if the local maximum is in the nine neighbouring bins at the end of the path. The 1D FBHA in the blue channel uses $\gamma = 0.005$

and $T = 9$. Figure 3.7(a) shows the square root of the source RG histogram and Figure 3.7(b) shows the square root of the target RG histogram. The complex shapes present in the histograms provide a visual illustration of the potential features in the histograms, a large number of local maxima are present in the histograms so it is important to pick the significant features of the histogram. Figures 3.8(a) and 3.8(b) show intensity plots of the source and target RG histograms respectively. Matched features are shown as green crosses and unmatched features are shown as red crosses. Yellow lines on the target histogram are drawn between the position of the feature on the target histogram with the position of the matched feature on the source histogram. The matched features in each channel are used to compute three independent linear transformations, these transforms are applied to the red, green and blue channel of the source image in Figure 3.6(a) to transform it to 3.6(c).

Observations

Detecting and matching features in the RG histograms is a challenging problem. The advantage of the 2D FBHA step is its ability to detect and match features in the RG histogram that may be obscured when inspecting the red and green channels individually. The RG histograms in figures 3.8(a) and 3.8(b) exhibit complex shapes and it is not obvious how to identify and match features manually. Upon visual inspection it can be concluded that 2D FBHA performs a reasonable job of feature detection and matching. However, the potential variability of histogram shapes mean that detecting and matching features from the histograms alone is not sensible in many cases. 2D FBHA reveals details of the histograms that may be obscured in the 1D version; however, the increased detail available in the 2D histogram must be balanced against the increased difficulty of the feature detection and matching step. This example applies independent linear transforms in the RG histogram even though it is possible to apply the correlated linear transform described by equation 2.7 with $d = 1$ and $n = 2$, this means that any advantage gained by applying 2D FBHA over 1D FBHA in the red and green channels is purely due to improved feature detection and association. Applying the correlated transform removes colour inconsistencies that are correlated between channels.

3.2.3 3D FBHA

3D deep structure feature detection uses $\gamma = 0.00001$ and $T = 11$. Maxima are detected where histogram bins contain values greater than those in the three-dimensional 26 connected neighborhood. Maxima features are connected over scales if the 27 connected neighborhood at the next scale contains a maximum.

Observations

3D deep structure feature detection was found to yield many short broken tracks, all ending at approximately the same length in the scale space. This meant that the features detected were not suitable for use in the later stages of the FBHA algorithm. Larger connectivity windows and different noise floor values were interactively tested but none improved the feature detection results. 3D deep structure feature detection is not reliable on the histograms of RGB images, so 3D FBHA is not explored further.

3.2.4 Shape preserving histogram transformations

Section 3.2.1 describes 1D FBHA and shows that computing histograms of the transformed data in each channel leads to histogram shapes that have gaps or spikes when compared to histograms of the original data. The spikes and gaps appear in such a way that the shape of the original histogram is no longer preserved. This section details how to transform a source histogram so that its shape is preserved. The interpretation of a histogram under transformation that preserves shape is discussed and compared to computing the histogram of the transformed source data-points.

The idea of a shape preserving histogram transformation is now introduced with an example in 1D. A 1D histogram with bins of unit size counts the values, q , for the x th bin so that $x - 1 < q \leq x$ where x is an integer bin index in the range $1 \dots 255$. The lower bin boundary for the x th bin is defined by $l_x = x - 1$ and the upper bin boundary is defined by $u_x = x$. For a given transform, the bin boundaries, u_x and l_x , are transformed to give new bin positions, the bin counts of the transformed histogram are reassigned to the bins of a target histogram with unit size. If a transformed bin lies within a single bin in the target histogram its count is assigned to this bin; however, if the transformed bin spans more than one bin in the target histogram its count is assigned proportionally to the spanned bins. This simple procedure preserves the shape of 1D histograms for monotonic transforms, the same principle of transforming bin boundaries and re-assigning bin counts can be extended to 2D and 3D histograms if required.

The correction transform is a model for the sources of colour inconsistency between the source and target data. Computing the histogram of the transformed source data does not give the same histogram as the shape preserved histogram. It is important to articulate the difference between these approaches and relate them back to the colour inconsistency problem. The motivation for using shape preserved histograms is that the gaps and spikes in a 1D transformed histogram do not appear and disappear in a predictable manner. Spikes and gaps in histograms of transformed data are likely to appear at different bins in the histograms being compared, this means that histogram metrics that compare corresponding bins will be perturbed by these effects. The spikes and gaps are genuinely present in the histogram of the transformed data. Given the source colour values, the transformed histogram represents an inference about the distribution of colour values that would be obtained under a different set of experimental conditions. The assumption that the transformed histogram must retain the same shape as the source histogram is a strong assumption; it says that if further observations of the same set of objects are available under the target lighting and camera conditions, one could expect the histogram of these values to follow the shape of the preserved histogram. As a greater number of colour values of the same objects are observed, the gaps in the target histogram would be filled and spikes would be smoothed out. One problem with the shape preserving procedure is that it can introduce a non-monotonic relationship between the source and target histogram that is not specified by the original transformation model. For example, if a linear correction is computed where the multiplicative component is greater than 1 the shape preserving procedure introduces a one to many mapping from a single bin in the source histogram to multiple bins in the target histogram. This adjustment preserves the histogram shape but a 1 to many relationship between source data-point values and target values is now introduced, there is no way to

know how to transform the individual data-points without further information.

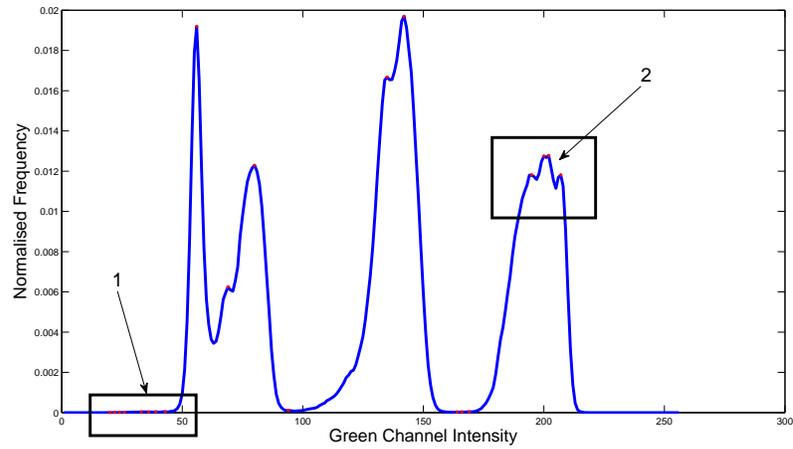
In conclusion, the shape preserved histogram minimizes effects that are likely to cause perturbations in common histogram alignment metrics. However, maintaining the shape constraint introduces a deviation away from the computed colour inconsistency correction transform. This can be seen because the histogram of the transformed data-points is not equivalent to the shape preserved histogram. The shape preserved histogram is of use when the source and target histogram shapes exhibit similar structure and deformations are moderate. In this case, the advantages of avoiding problems with histogram comparison metrics outweigh the introduced deviation from the colour inconsistency correction. When the source and target histograms contain complex shapes, the shape preservation procedure may not be appropriate as it is likely to model significant deviations from the computed colour inconsistency transform.

3.3 Summary Conclusions

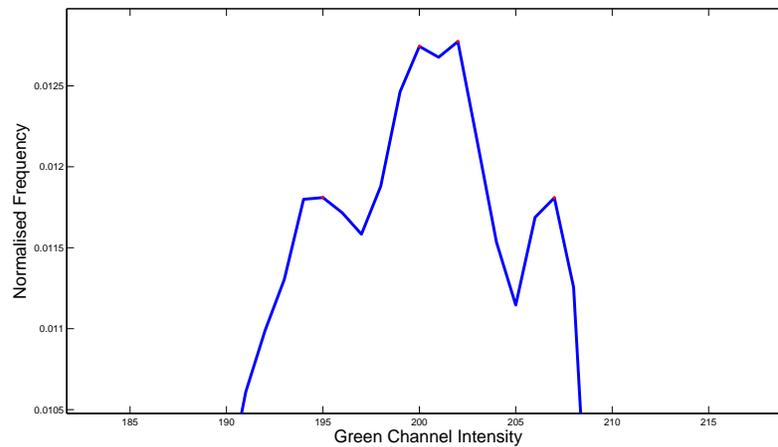
This chapter has introduced a method for removing colour inconsistencies called feature based histogram registration. The contributions of the method are:

- the introduction of an automatic feature detection and alignment approach. The feature based approach makes it possible to align the local structure of histograms using point alignment transforms. Point alignment transforms include correlated polynomials and can account for a wider range of variations than is possible when aligning the moments of the distribution using multiplicative or linear transforms.
- the introduction of a novel feature detector that exhibits stable performance over multiple executions. The detector does not require the number of clusters to be specified and makes no parametric assumptions about the form of the data.
- the introduction of two automatic feature matching strategies, **CEM** and **CEM-DC**.

FBHA has been evaluated in a purely qualitative manner to give the reader an understanding and feel for the steps of the algorithm, the 1D and 2D deep structure feature detectors are shown to work well. The 3D version of the feature detector does not work well enough to facilitate automatic feature detection and matching. Nevertheless, automatic feature based alignment of 1D and 2D histograms is useful because 1D and 2D data is common in machine vision applications. 1D FBHA has been shown to work on the red, green and blue channels of an image. 2D FBHA has shown to work on the red-green histograms, an RGB image can be manipulated by applying 2D FBHA on the red-green channels and 1D FBHA on the blue channel. The next chapter develops a quantitative assessment of colour inconsistency removal and evaluates the automatic FBHA approaches that have been introduced.

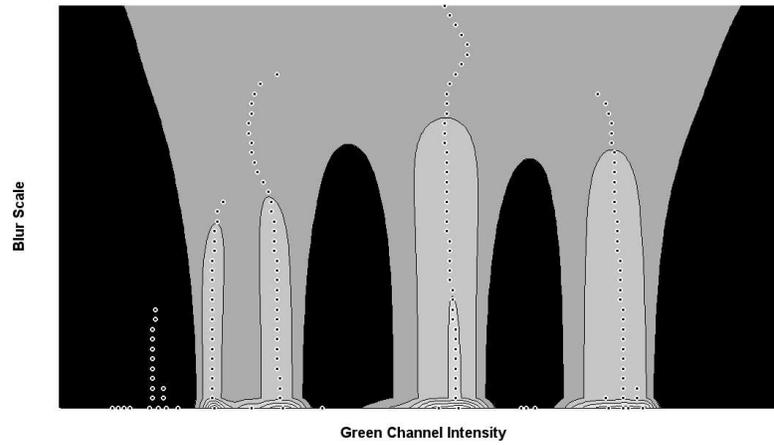


(a)

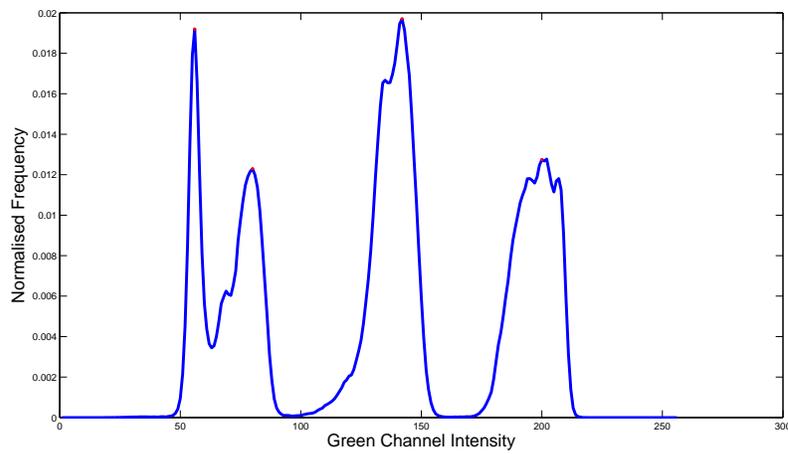


(b)

Figure 3.1: Example of the local maxima in a one dimensional histogram. 3.1(a) shows a histogram obtained from the green channel of an image of skittles in the image database. All local maxima are shown as red dots, maxima in low level noise is highlighted as 1) and multiple local maxima on a cluster are highlighted as 2). 3.1(b) shows a zoomed view of the local maxima in the cluster labelled 2).

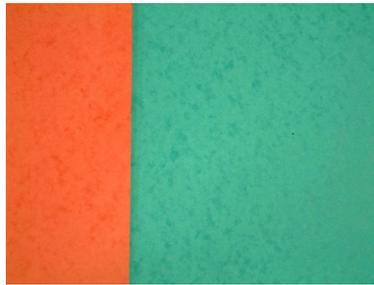


(a)

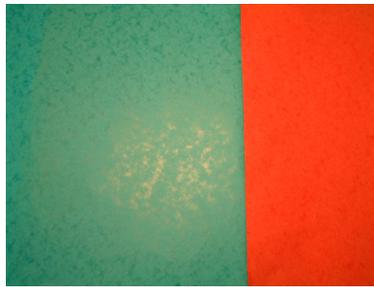


(b)

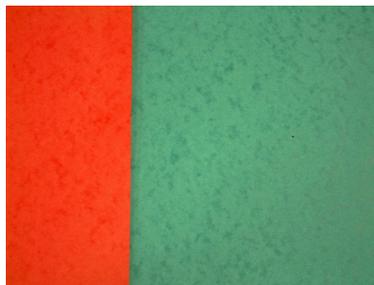
Figure 3.2: 3.2(a) shows a representation of the scale space of the histogram in Figure 3.1(a); horizontal slices indicate histograms blurred at increasing scale moving from bottom to top, denser regions of the scale space are rendered closer to white and local maxima at each scale are drawn as circles. The maxima form paths across scales, with paths from less significant peaks ending earlier in the scale space. 3.2(b) shows the persistent maxima using $T = 20$.



(a)

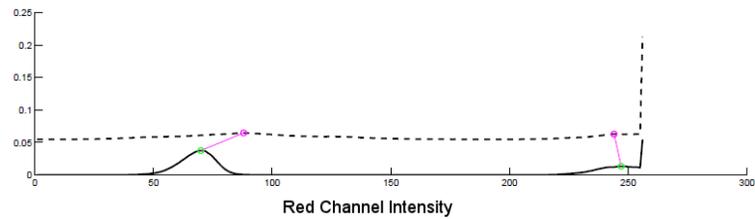
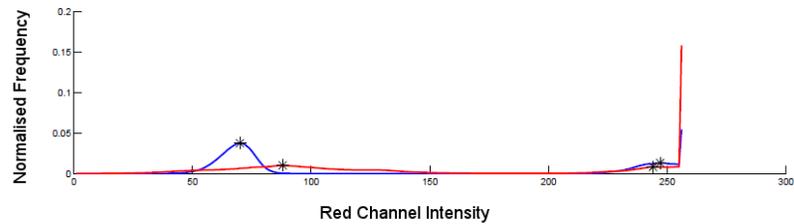


(b)

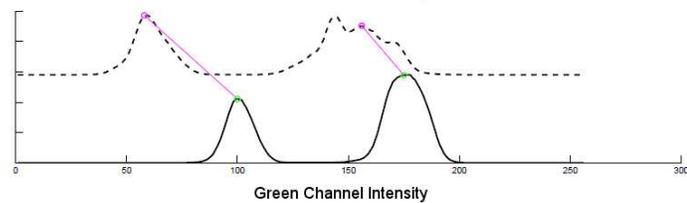
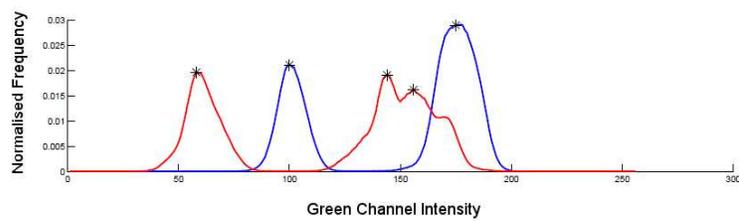


(c)

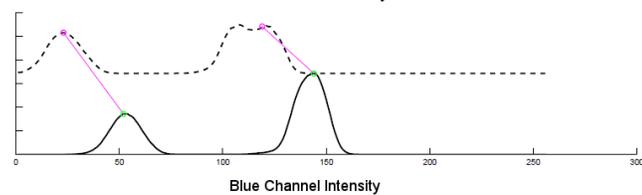
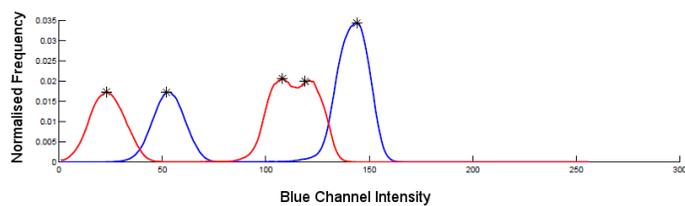
Figure 3.3: The source image in 3.3(a) and target image in 3.3(b) exhibit colour inconsistency. 3.3(c) shows the colours of the source image transformed using 1D FBHA with a linear transform in each channel.



(a)

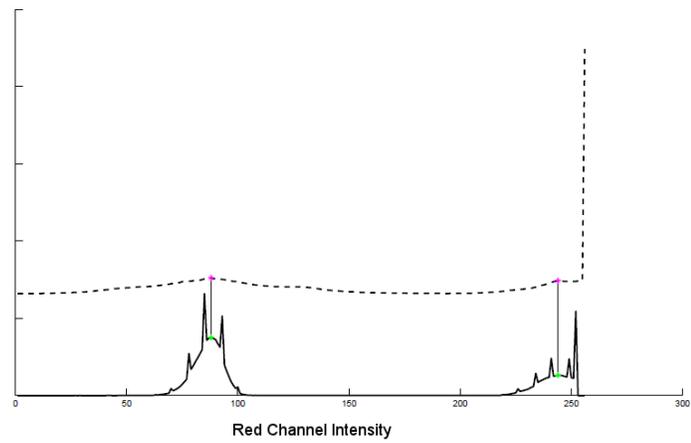


(b)

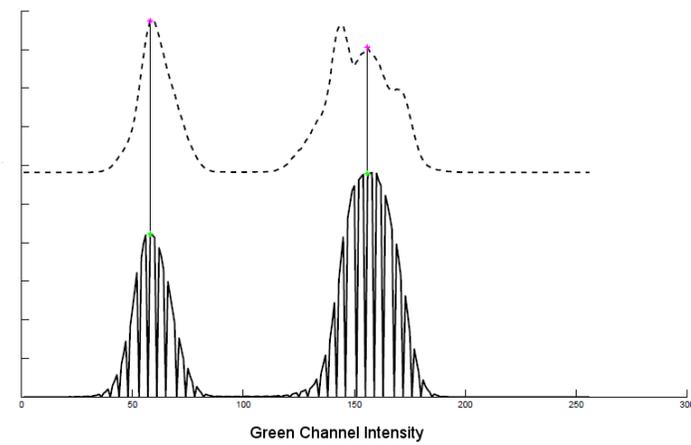


(c)

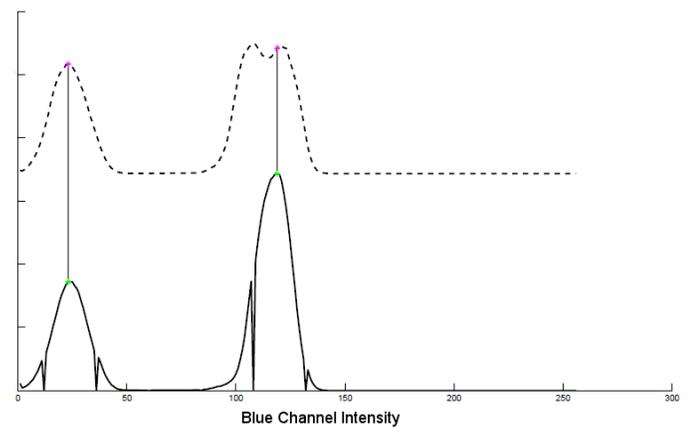
Figure 3.4: Source and target histograms are shown as overlaid line plots in the top portion of each sub-figure. The red plots show the target histogram in the red channel 3.4(a), green channel 3.4(b) and blue channel 3.4(c). The corresponding blue plots show the source histograms. Detected features are marked on the source and target histograms with a star. The bottom portions of each sub-plot show an exploded view of the source and target histograms and the matched features.



(a)



(b)



(c)

Figure 3.5: Exploded view of the histograms of the corrected data plotted with the solid line and the target histogram plotted with the dotted line. The aligned features are shown using a line to connect the aligned feature and target feature. Subfigures 3.5(a), 3.5(b) and 3.5(c) show the red, green and blue channels respectively.



(a)

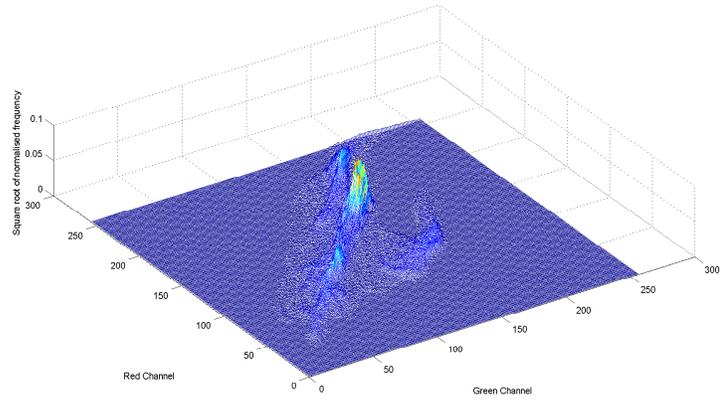


(b)

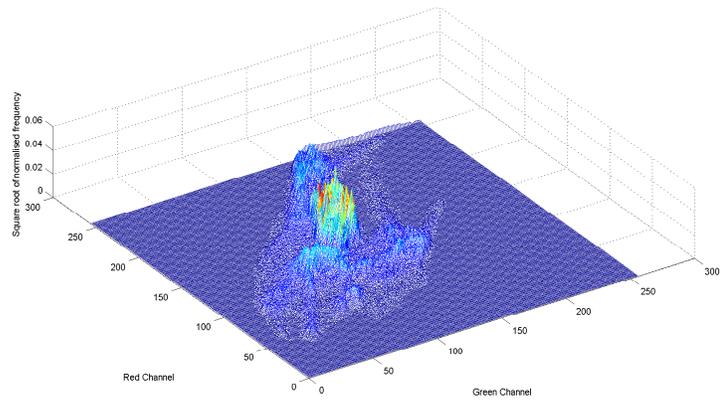


(c)

Figure 3.6: Image of plastic skittles. Figure 3.6(a) shows the source image and figure 3.6(b) shows the target image where a red light modifies the appearance of the skittles. Figure 3.6(c) shows the modified source image where 2D FBHA is used in the RG channels and 1D FBHA is used in the blue channel.

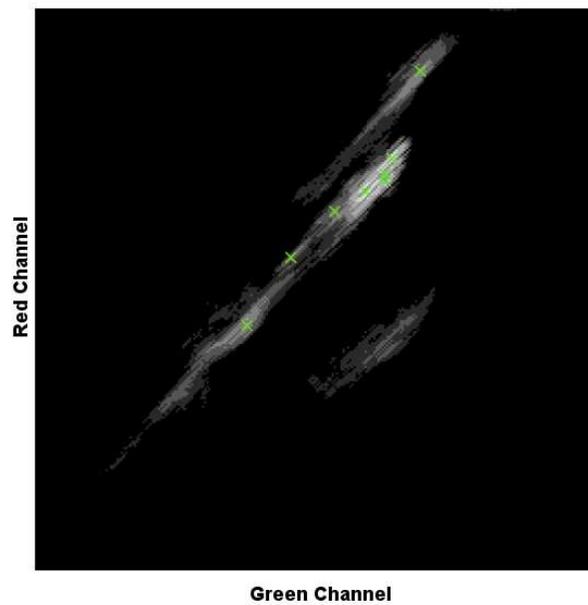


(a)

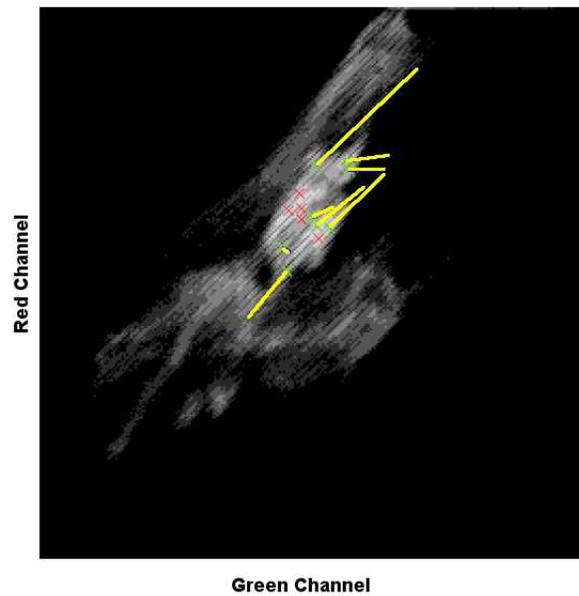


(b)

Figure 3.7: Sub-figure 3.7(a) shows the square root of RG histogram for the source image in Figure 3.6(a) and sub-figure 3.7(b) shows the square root of the RG histogram for the target image in Figure 3.6(b). Taking the square root of the histograms allows the shapes of local features at different scales to be observed more easily on a single plot.



(a)



(b)

Figure 3.8: Feature detection and matching for the source and target RG histograms shown in Figures 3.7(a) and 3.7(b). The source RG histogram in 3.7(a) is shown as an intensity plot in sub-figure 3.8(a), dense regions of the histogram are shown closer to white and less dense regions are shown closer to black. The target RG histogram in 3.7(b) is shown as an intensity plot in 3.8(b). Matched detected features are shown with a green cross, detected features that remain unmatched are shown with a red cross. A blue line is drawn on the target RG histogram from the matched feature on the target histogram to the position of the matched feature on the source histogram.

Chapter 4

An image database for testing RGB colour alignment

This chapter introduces an image database for evaluating colour inconsistency removal methods. The database is structured to investigate different sources of colour inconsistency. A quantitative methodology for evaluating and ranking different colour alignment methods is introduced. The methodology is used to compare FBHA and alternative methods.

4.1 Database design

This section motivates and describes the image database. Consider the task of removing the colour inconsistency between the two images in Figure 4.1. The plastic toys in 4.1(a) are illuminated with ambient lighting, 4.1(b) shows the same objects illuminated by an additional red light. A colour inconsistency removal method should find transformations of the images so that the individual colours of the yellow skittles, blue skittles, green balls and grey backgrounds each become more coherent in colour space. Trivial solutions such as setting colours to the same value should be ignored. Recall that colour inconsistencies are non-unique mappings from the material properties of an object to observed colours. A fair comparison of methods should evaluate when the colours from homogeneous materials become more



(a)

(b)

Figure 4.1: Two images of plastic toys on a grey cardboard background. In 4.1(a) the scene is lit using clear bulbs, in 4.1(b) a red bulb is placed above the scene.

self similar while remaining distinct from the colours of other materials. The database contains images that:

1. are composed of a low number of simple objects that contain large regions of homogenous colour.
2. are captured under different experiment conditions leading to different colour inconsistencies.
3. vary the relative scale of the different materials present in the image.
4. are labelled so that colours corresponding to different materials can be identified.

The subsequent subsections describe more details about the motivation for these choices.

4.1.1 Objects

Four different sets of objects are chosen to create images that contain 2 to 4 different materials. Scenes with low numbers of distinct materials are chosen because this allows the behavior of the colour clusters that correspond to different scene materials to be studied with clarity. Notice that it is the number of distinct materials in the scene that is important and not the number of objects.

The objects are:

1. Red and cyan paper strips shown in Figure 4.2(a). Two different materials are present.
2. Red, green and blue paper strips shown in Figure 4.2(b). Three different materials are present.
3. Purple, yellow and green plastic skittles and balls on an uncluttered grey background shown in Figure 4.2(c). Four different materials are present.
4. Brown, yellow and red stuffed animals arranged on an uncluttered grey background shown in Figure 4.2(d). Four different materials dominate the images, a number of different materials occupy a small fraction of the image. These are the black eyes of the red bear, the white and black labels and the flag on the chest of the red bear.

The images of paper provide examples of a planar object, the plastic objects provide examples of specular reflections and the teddy bears are diffuse reflectors. Figure 4.3 shows the database hierarchy, this structure groups similar variation types together.

4.1.2 Capture conditions

The database contains images of four different object classes that are captured by varying four different experimental conditions. The experimental conditions varied are:

1. The camera used.
2. The local illuminating light.
3. The ambient illuminating light.
4. The scale of the objects in the image.

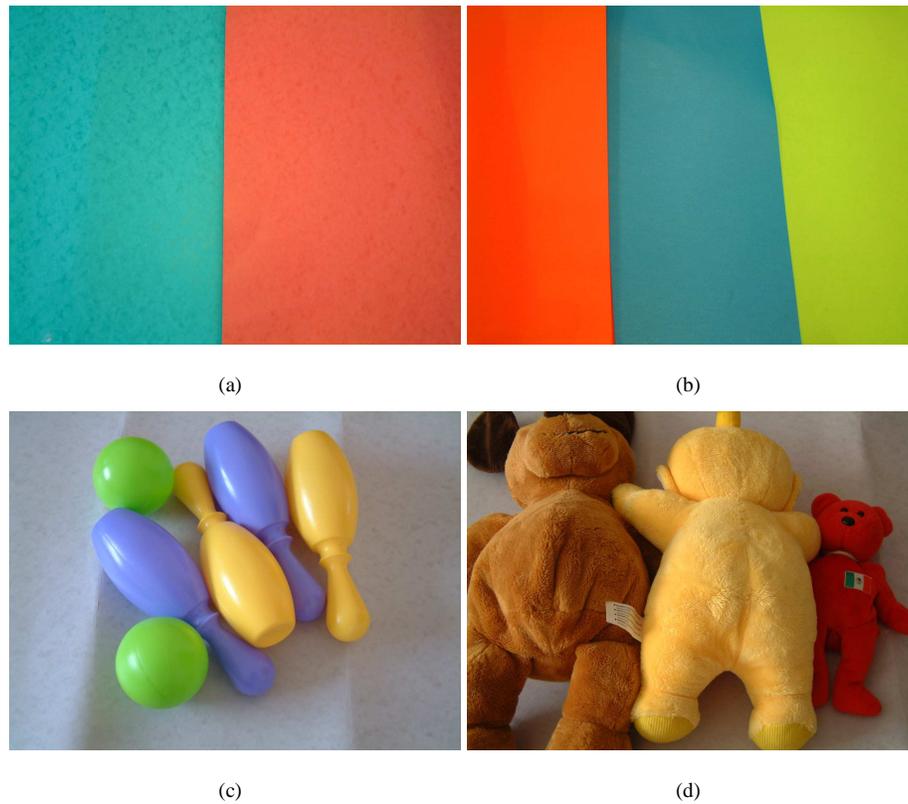


Figure 4.2: Typical images from the four different object categories used in the colour alignment database; for each object category, objects are imaged under different scale, lighting and camera conditions. Image 4.2(a) shows a representative image from the set Red and Cyan paper set. Image 4.2(b) shows an image from the set of red, green and blue paper strips. Image 4.2(c) shows and image of the plastic skittles and ball on a grey background. Image 4.2(d) shows an image from the set of stuffed animals.

The first three experimental conditions lead to observer and illuminant colour inconsistencies. Object scale variation is a common confounding condition that makes object matching problems harder. Therefore, it is important that colour inconsistency removal methods can handle data with scale variations. Colour inconsistency removal methods can be tested on two or more images of the same object type from the database. The difference in experimental conditions between any pair of images is known, so it is possible to test whether specific experimental conditions confound specific methods.

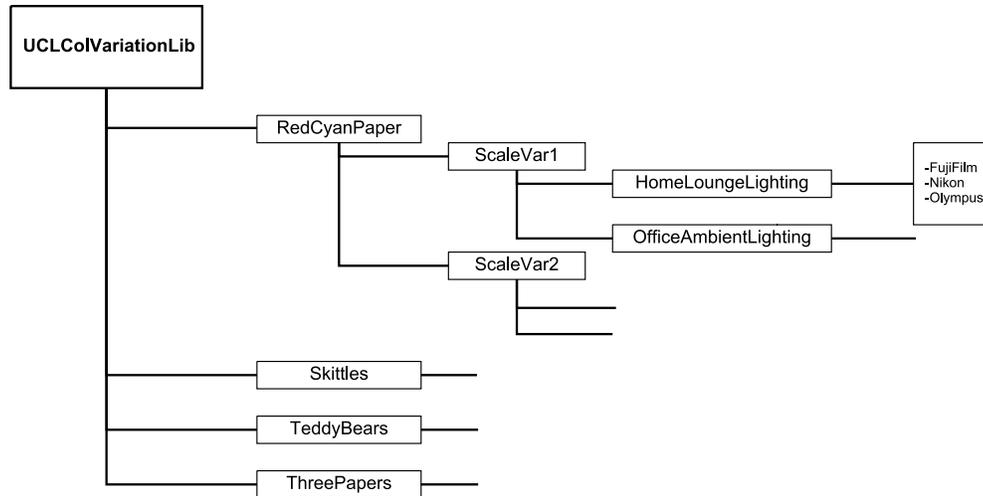


Figure 4.3: Organisation of the UCL colour variation database. The directory structure under each of the four object type folders is identical. The unique parts of the hierarchy are shown. At the lowest level of the hierarchy there are three folders corresponding to the different cameras; each camera directory contains five images corresponding to five different local lighting conditions.

Each of the sets of objects are imaged using:

- 3 different cameras.
- 2 different object scales.
- 5 different local lighting conditions.
- 2 different ambient lighting conditions.

These combinations lead to 60 images per object set and 240 for the entire database. The three cameras used to capture the pictures are shown in figure 4.6. The flash was switched off for each camera and automatic settings were used, a tripod was used to minimize hand shake and the camera was allowed to focus first by pressing the shutter down half way before taking the picture.

The object scale in the image was varied by moving the camera, adjusting the zoom and readjusting the positions and numbers of objects where appropriate. The objects were imaged under two different sets of scale conditions; human judgement was used to keep the relative composition of objects and background approximately constant across different camera and lighting conditions. Figure 4.4 shows example images from the four different object classes imaged under 2 different scale conditions.

Images are captured using five different local lighting condition conditions and two ambient lighting conditions. The two ambient conditions were created by capturing all images in an office with a high degree of florescent lighting shown in Figure 4.5(b) and a living room with a large window shown in Figure 4.5(a). The office environment lighting provides constant ambient lighting, whereas the living room lighting varied significantly over the capture period due to larger changes in sunlight and clouds. Readings from a light meter shown in figure 4.5(d) were used to verify this. For each camera, scale and ambient lighting condition, a scene was imaged under five different local lighting conditions. Figure 4.5(c) shows the four different coloured bulbs used: 1) a clear red 60W bulb, 2) a frosted green 60W bulb, 3) a frosted yellow 60W bulb, 4) a clear 40W bulb. The fifth lighting condition was due to the ambient lighting only. A dimmer switch set up was used to adjust the bulb brightness to avoid high degrees of over-saturation in the image; this was particularly important when using a clear bulb and highly reflective paper. Objects were placed on the floor and the coloured bulbs were attached at a fixed distance above the floor. All images are captured in Jpeg format, users of the database should be aware that Jpeg is a lossy format. The Jpeg format is less reliable than formats such as Tiff in areas of spatial detail. The use of Jpeg does not affect the ability to test histogram alignment algorithms.

4.1.3 Object labeling

Each image in the data-base has an associated layered mask, each layer in the mask is a binary image that defines a polygon or polygons that mark regions of the image with homogenous material properties. The polygons were marked up manually for each image region in all images. The colours of each material can be accurately compared across imaging conditions by gathering and comparing pixels from each mask layer. Figure 4.7 shows examples of the image regions extracted by using the associated masks, the masks do not approach the object boundary too closely to avoid ambiguous pixels.

4.1.4 Image variation sets

The term *image variation sets* is introduced here to refer to the set of all image pairs with a particular combination of imaging conditions changing between the two images in the pair. There are 60 images associated with each set of objects so there are ${}^{60}C_2 = 1770$ image pairs in total. When transforming a single image in the pair to match the other image in the pair there are 3540 possible source to target image pair combinations. Table 4.1 lists the different image variation sets along with a short hand code and the number of image pairs in each set.

4.2 Existing Colour Datasets

There are a number of freely available computer vision datasets. This section identifies these data-sets, their design rationale and what they are used for. The reasons why these data-sets are not appropriate for the investigations performed in this thesis are identified, this motivates the need for the new image data-base presented in this chapter.

The existing image data-bases are:

1. The University of East Anglia(UEA) colour constancy database [73]. It contains images of various kinds of wallpapers captured under different lighting conditions and from different cameras.

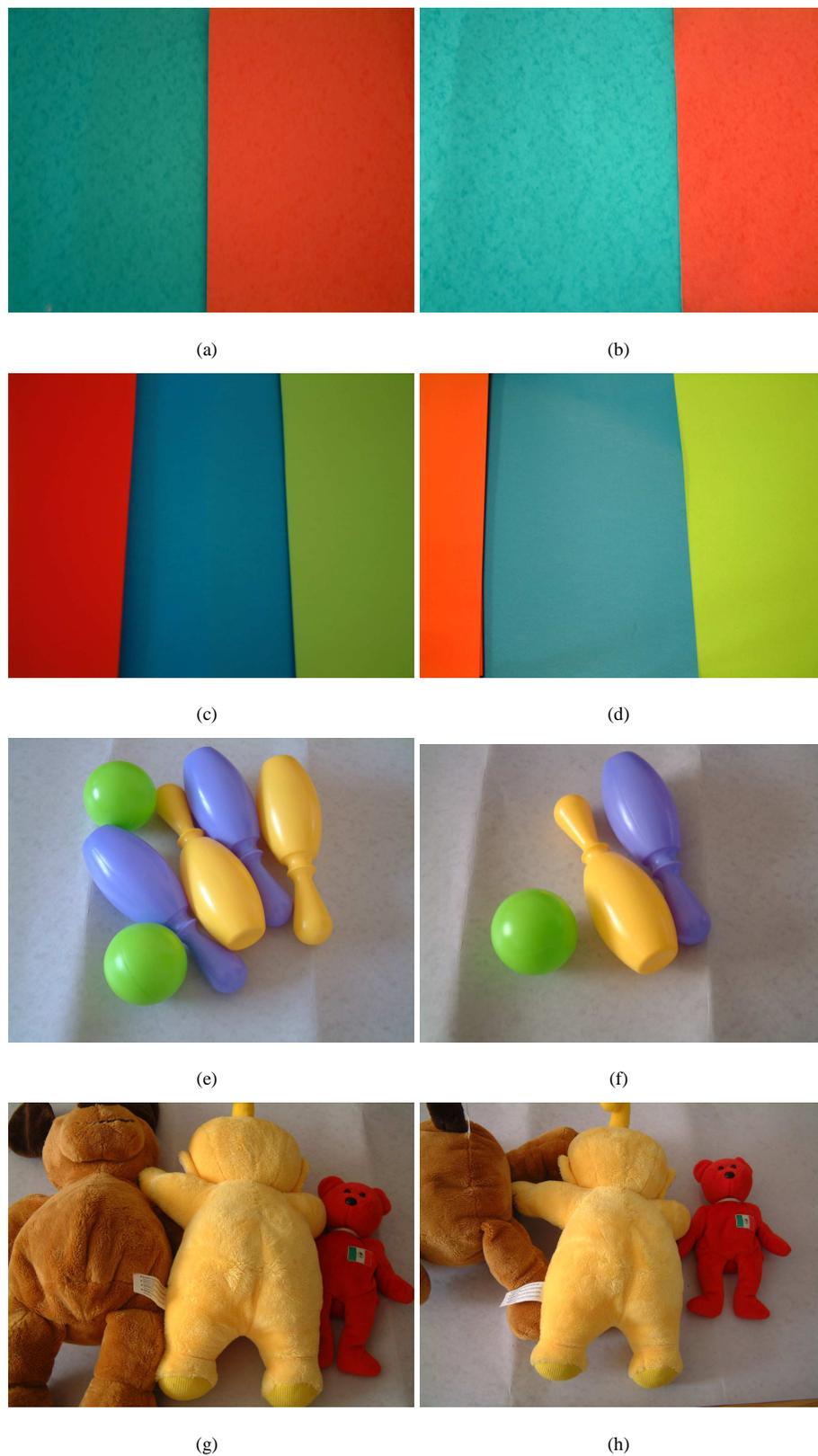


Figure 4.4: Images from the four different image sets illustrating the different scale variations captured and categorised in the database. The red and cyan pieces of paper occupy roughly equal portions of the image in 4.4(a), the cyan paper occupies a larger portion of the image in 4.4(b). The red, green and blue strips are arranged to occupy approximately a third of the image each in 4.4(c); 4.4(d) shows a scale adjustment of the red and blue paper. Images 4.4(e) and 4.4(f) illustrate scale variation in the skittles set. Images 4.4(g) and 4.4(h) illustrate scale variation in the Teddy Bears set.



Figure 4.5: Locations and equipment used to create different lighting conditions. 4.5(a) shows the naturally lit lounge and 4.5(b) shows the office lit by florescent bulbs. 4.5(c) shows the bulbs and dimmer switch used create local lighting variation, 4.5(d) shows the light meter used to approximately monitor the ambient lighting conditions.

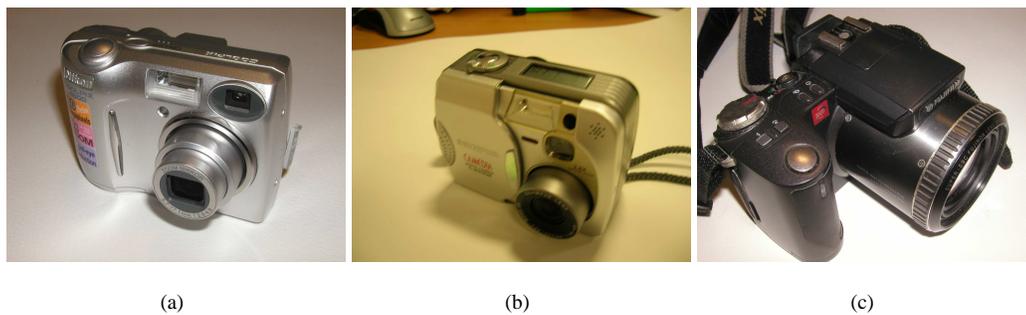
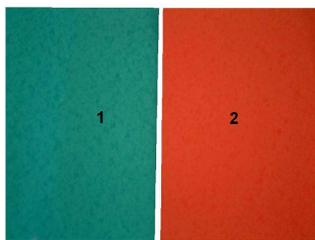


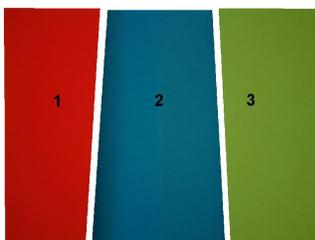
Figure 4.6: The three different cameras used to acquire the colour variation database. These are 4.6(a): a Nikon Coolpix 4600, 4.6(b): an Olympus Camedia C40 Zoom and 4.6(c): a FujiFilm FinePix 6900 Zoom.



(a)



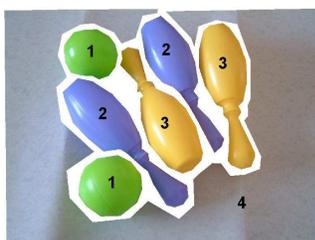
(b)



(c)



(d)



(e)



(f)



(g)



(h)

Figure 4.7: Example masked regions from sample images from the four different object types. Images 4.7(a), 4.7(c), 4.7(e) and 4.7(g) show numbered distinct masked regions for the corresponding images in 4.7(b), 4.7(d), 4.7(f) and 4.7(h).



Figure 4.8: Two images from the UEA uncalibrated colour database. Both images are of wall paper under the same lighting and camera conditions. The data-base contains the same images taken under different combinations of lighting and camera changes.



Figure 4.9: Different objects in the SOIL database.

Table 4.1: The image variation sets in the UCLColvariation database. Each variation set refers to a subset of the image pairs for an object type in the database, the image pairs in the subset differ in experimental capture conditions as described. The short hand codes are used to refer to these image variation sets.

| Varied Conditions | Code | Num image pairs |
|---|--------------------|-----------------|
| Scale | 000(S) | 60 |
| Ambient Lighting | 00(L-AL)0 | 60 |
| Local Lighting | 0(L-LI)00 | 240 |
| Camera | (C)000 | 120 |
| Ambient lighting and Scale | 00(L-AL)(S) | 60 |
| Local lighting and ambient lighting | 0(L-LI)(L-AL)0 | 240 |
| Camera and local lighting | (C)(L-LI)00 | 480 |
| Camera and ambient lighting | (C)0(L-AL)0 | 120 |
| Camera and scale | (C)00(S) | 120 |
| Local lighting and scale | 0(L-LI)0(S) | 240 |
| Camera, Local lighting and ambient lighting | (C)(L-LI)(L-AL)0 | 480 |
| Local lighting, ambient lighting and scale | 0(L-LI)(L-AL)(S) | 240 |
| Camera, ambient lighting and scale | (C)0(L-AL)(S) | 120 |
| Camera, local lighting and scale | (C)(L-LI)0(S) | 480 |
| Camera, local + ambient lighting and scale | (C)(L-LI)(L-AL)(S) | 480 |

Figure 4.8 gives an example of two images of different wallpaper patterns taken under the same lighting and camera conditions. Images of these and other wallpaper patterns are captured under three different lighting conditions and by four different cameras. The UEA data-base has been used to test image retrieval performance in the presence of colour inconsistencies [46]. The design of the data capture for UCLColVariationLib follows the UEA design by varying lighting and camera conditions systematically. The UEA data-base does not contain examples with low numbers of material properties, also the relative amounts of the different materials are not varied so that the effect of scale changes in the corresponding clusters of a histogram can be examined. UCLColVariationLib introduces these examples so that the behavior of colour inconsistency removal algorithms can be studied on simple examples.

2. The SOIL database from Surrey university [74] contains commonly obtained supermarket items imaged under lighting and pose changes. Figure 4.9 illustrates the objects present in the database. The Soil database varies the 3D viewpoint, the illumination intensity, occlusion and scene distracters and structural appearance variation and has been used to test object recognition algorithms [75]. The scene objects are captured against a black background and occupy a small fraction of the image. The objects typically have multi-coloured intricate patterns and logos that are common on product packaging. The high number of different coloured regions mean that these objects

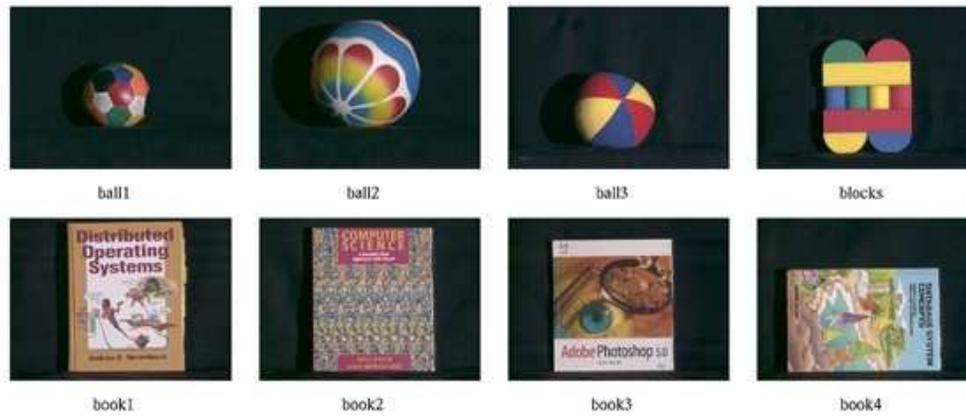


Figure 4.10: Sample images from the SFU database.

present a more difficult colour inconsistency removal task than the objects in the UCLColvariationLib data-base. Again, the strength of UCLColvariationLib is that it allows the simple cases to be studied first.

3. The Simon Fraser University(SFU) database [76] is similar to the SOIL database in design. It divides images into a training set of objects with fixed pose and changing illuminant and a test set with random pose under the same illuminants. Example images are shown in Figure 4.10. The SFU data-base has been used to evaluate colour constancy algorithms [77]. The objects are multi-coloured with small detailed regions, like the SOIL database these coloured regions lead to scattered small regions in the colour histograms. The SFU database does not introduce scale variation in a systematic way. These factors mean that the UCLColVariationLib is a better starting point for investigating colour histogram alignment methods.

This section has presented existing data-bases that have been successfully used in colour inconsistency research, because of the reasons mentioned these data-bases do not match the requirements of simplicity and systematic object scale variation that the study in this thesis requires. To summarize, the advantage of UCLColVariationLib is that it introduces colour inconsistencies for very simple objects so that the differences between the colour inconsistent histograms are more easily interpreted than those from these existing data-bases.

4.3 Histogram alignment metrics

One way to obtain quantitative measures of colour transfer performance is to compare histograms of the transformed images. This section lists a variety of metrics for histogram comparison and studies their properties on simple synthetic histograms. The histogram metrics fall into three categories: 1) Bin-Bin metrics, 2) Cross-Bin Metrics and 3) Manually defined metrics.

4.3.1 Bin to Bin Measures

Bin to Bin metrics are based on comparisons between the corresponding bins in histograms. Common examples are the Bhattacharyya distance [78], Mutual Information [79] or the Kullback-Leibler (K-L)

divergence [79]. Bin to Bin measures are not discriminative for histograms containing non overlapping or sparse data; also, comparisons between multi-modal clusters are biased by the largest overlapping clusters. This section introduces two popular bin to bin measures for comparing a histogram \mathbf{s} with another histogram \mathbf{t} ; \mathbf{s}_x is bin x in \mathbf{s} and \mathbf{t}_x is the corresponding bin in \mathbf{t} . The histograms contain the same number of bins over the domain \mathbf{X} . The Bhattacharyya coefficient is

$$B(\mathbf{s}, \mathbf{t}) = \sum_{x \in \mathbf{X}} \sqrt{\mathbf{s}_x \mathbf{t}_x}, \quad (4.1)$$

and the K-L divergence is

$$KL(\mathbf{s}|\mathbf{t}) = \sum_{x \in \mathbf{X}} \mathbf{s}_x \log \frac{\mathbf{s}_x}{\mathbf{t}_x}. \quad (4.2)$$

The Bhattacharyya coefficient is used extensively due to its simplicity and numerical stability when dealing with zero bins [78]. The K-L divergence is an asymmetric measure known as the relative entropy in information theory, its numerical computation requires translating all histogram bin values away from zero to avoid division by zero.

4.3.2 Cross-Bin Measures

Cross bin measures [80][81][82] compute a metric based on corresponding and non-corresponding histogram bins, they have been used successfully in vision-based database lookup applications to alleviate biases in bin-bin comparisons when comparing partially or non-overlapping clusters of different size. Colour histograms are commonly sparsely populated, so cross bin measures are a robust choice for comparing them. The Earth Mover's Distance is perhaps the first example of a cross-bin measure that has found applications in Computer Vision; it is computed as the minimum work required to transform one distribution to another when posed as the transportation problem in linear programming. The EMD has $O(n^3)$ complexity where n is the number of histogram bins; because of this computational expense the EMD is typically used to compare simple histograms. Subsequent work has sought to replicate the benefits of the EMD within a computationally efficient framework; examples are the pyramid match kernel [81] and the diffusion distance [82]. The diffusion distance is of particular interest because of its simplicity and speed of computation. The diffusion distance considers the difference between two distributions as

$$\mathbf{d} = \mathbf{t} - \mathbf{s} \quad (4.3)$$

where the corresponding bins of \mathbf{s} are subtracted from \mathbf{t} . \mathbf{d} is blurred by convolution with a Gaussian kernel $G(\cdot; \sigma)$; the L1 norm is computed between the blurred distance and a matrix of zero entries. The is repeated over a range of kernel widths s to t and the results are summed as

$$\sum_{\sigma=s}^t L1(\mathbf{d} * G(\cdot; \sigma)). \quad (4.4)$$

4.3.3 Manually defined metrics

The term *manually defined metrics* is introduced to describe metrics that are computed from labeled data that is obtained from the marked up masks associated with each image in the database. There are m different labels attached to the data, j indexes the labels. Two different manually defined metrics are

described here. The first metric is the total Euclidean distance between labeled clusters, the mean of the colours labeled j is \mathbf{q}_j for the source distribution and \mathbf{w}_j in the target distribution. The L2 norm is computed between corresponding means and the results are summed over all labels, as,

$$E = \sum_{j=1}^m L2(\mathbf{q}_j, \mathbf{w}_j). \quad (4.5)$$

The Euclidean metric is not discriminative for changes in cluster orientation at fixed distances between cluster centres; the second metric reduces this problem, it computes an average Mahalanobis distance over the clusters. The Mahalanobis distance in both directions for the j th label is,

$$\phi_j = \sqrt{(\mathbf{q}_j - \mathbf{w}_j)^{-1} \Sigma_{\mathbf{q}_j} (\mathbf{q}_j - \mathbf{w}_j)} \quad (4.6)$$

and

$$\beta_j = \sqrt{(\mathbf{w}_j - \mathbf{q}_j)^{-1} \Sigma_{\mathbf{w}_j} (\mathbf{w}_j - \mathbf{q}_j)}, \quad (4.7)$$

where $\Sigma_{\mathbf{q}_j}$ and $\Sigma_{\mathbf{w}_j}$ are the covariances of j th source and target components respectively. For each pair of corresponding components the average of these distances is computed as,

$$J_j = \frac{\phi_j + \beta_j}{2}. \quad (4.8)$$

The average Mahalanobis metric is computed as the sum of the averages of ϕ_n and β_n ,

$$\sum_{n=1}^L J_n. \quad (4.9)$$

4.3.4 Empirical comparison of metrics

This section develops an intuitive notion of how different histogram alignment metrics vary. Synthetic source and target histograms are generated from a parametric model, the parameters of the model are specified so that they correspond to intuitive geometric transformations. Then, metrics are computed for a sequence of histogram pairs. The pairs are generated by adjusting the model parameters to investigate different forms of alignment. One motivation is to understand the effects of overlapping and non-overlapping clusters on the behavior of the metrics.

Model

A two dimensional Gaussian Mixture Model (GMM) is specified here in terms of a set of intuitive parameters. The GMM is a weighted sum of Gaussian distributions, each Gaussian is usually specified in terms of its mean, a scaling parameter and a covariance matrix. One problem with specifying the shape of each mixture component with a covariance matrix is that it does not relate clearly to an intuition about manipulating the model. This formulation defines each Gaussian component with its mean at the origin, the standard deviations along the x and y axes are specify the shape of the component. A rotation of each component in the X-Y plane is then specified. This subtle change in parametrization allows one to think in terms of rotating, stretching and translating each component.

Each histogram is generated by a mixture model as a linear combination of m component densities, $p(x|j)$. A mixture model has the form:

$$p(x) = \sum_{j=1}^m p(x|j)P(j), \quad (4.10)$$

where the coefficients $P(j)$ are called the mixing parameters. The mixing parameters satisfy

$$\sum_{j=1}^m P(j) = 1, \quad (4.11)$$

and

$$0 \leq P(j) \leq 1. \quad (4.12)$$

The component densities $p(x|j)$ are normalised such that

$$\int p(x|j)dx = 1. \quad (4.13)$$

The component densities can take any parametric form. In this work, each component is an N dimensional anisotropic Gaussian distribution,

$$p(x|j) = (2\pi)^{-N/2} |\Sigma| \exp(-(\mathbf{x} - \mathbf{u})^T \Sigma^{-1} (\mathbf{x} - \mathbf{u})/2), \quad (4.14)$$

where $N = 2$. The covariance matrix, Σ is expressed in terms of θ , σ_x and σ_y using a singular value decomposition,

$$\Sigma = \mathbf{U} \mathbf{S} \mathbf{V}^T. \quad (4.15)$$

\mathbf{S} is a diagonal scaling matrix. The diagonal entries specify the variance along the x and y axes of a zero mean Gaussian. In terms of standard deviations, σ_x and σ_y this is

$$\mathbf{S} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}. \quad (4.16)$$

The Gaussian is rotated counter-clockwise θ degrees in the X - Y plane. This is specified by setting \mathbf{U} to a standard rotation matrix so,

$$\mathbf{U} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}, \quad (4.17)$$

and $\mathbf{U} = \mathbf{V}$.

Method

A sequence of source and target histograms are generated using the model, the sequence of parameters is chosen to investigate a particular experimental hypothesis. A histogram in the sequence is generated by specifying each of its m clusters. The j th cluster in the l th source histogram is specified by five parameters as, $S_{lj}(P(j), \mathbf{u}, \theta, \sigma_x, \sigma_y)$. The corresponding cluster in the target histogram is identified as $T_{lj}(P(j), \mathbf{u}, \theta, \sigma_x, \sigma_y)$.

Five metrics are computed from each histogram pair in the sequence, these are:

1. The Bhattacharyya coefficient (eqn:4.1).
2. The Kullback-Leibler distance (eqn:4.2).
3. The Diffusion distance (eqns:4.3.2, 4.4).
4. The total Euclidean distance (eqn: 4.5).

5. The average Mahalanobis distance (eqn: 4.9).

The results for the sequence are plotted for each metric. All metrics are normalized to the range 0..1. The Bhattacharyya coefficient is normally in the range 0..1 where a score of 1 indicates the highest level of similarity. Bhattacharyya coefficient scores are reflected about the axis $y = 0.5$ so that 0 indicates the highest level of similarity. After these steps, all scores for a sequence are shown in the range 0..1 and a lower score indicates a greater degree of similarity between histograms.

Experiments

This section describes the empirical results obtained from three different sequences of histograms. The sequences are designed to show the effects of non-overlapping clusters and the bias of larger overlapping clusters on the metrics.

Sequence 1: Comparison of single mode variations Bin to bin metrics only consider the relationship between the corresponding bins in the histogram. This leads to the presupposition that bin to bin metrics will not discriminate between changes in orientation when the degree of cluster overlap is low.

Hypothesis: Bin-Bin metrics are not discriminative at large distances.

Sequence: The source and target histogram contain a single cluster each. The sequence is generated by iterative rotation and translation of the target cluster. The rotations are $\theta_1 = 0$, $\theta_2 = 20$, $\theta_3 = 45$, $\theta_4 = 90$, and the means of the target histogram cluster are, $\mathbf{u}_1 = [0, 0]$, $\mathbf{u}_2 = [5, 0]$, $\mathbf{u}_3 = [10, 0]$, $\mathbf{u}_4 = [15, 0]$, $\mathbf{u}_5 = [20, 0]$, $\mathbf{u}_6 = [25, 0]$, $\mathbf{u}_7 = [30, 0]$, $\mathbf{u}_8 = [35, 0]$. The source histogram does not change throughout the sequence, it is $S_{l1}(1, \mathbf{u}_1, \theta_1, \sigma_x, \sigma_y)$. $\sigma_x = 10$ and $\sigma_y = 30$ for all histograms. The target histogram sequence is described by the pseudo-code in algorithm 3.

Algorithm 3 *GenerateHistoSequence1*

```

count ← 1
for o = 1 to 8 do
  for p = 1 to 4 do
     $T_{count1}(1, \mathbf{u}_o, \theta_p, \sigma_x, \sigma_y)$ 
    count ← count + 1
  end for
end for

```

Results Figure 4.11 shows the values of the metrics for the sequence. Representative transformations from the sequence are illustrated in Figure 4.12. The most noticeable observation is that none of the metrics varies smoothly across the sequence. The total Euclidean distance is the only metric that exhibits a uniform repeating pattern. The total Euclidean distance exhibits a step like response as it does not register the changes in target orientation but registers the change in mean. The average Mahalanobis distance changes smoothly in response to orientation changes when the clusters are close and with more variation when the clusters are far apart. The Bhattacharyya coefficient, Kullback-Leibler and Diffusion distance all vary in a highly non-uniform manner across the sequence. The Kullback-Leibler stops discriminating between changes in target orientation at position 15 in the sequence and the Bhattacharyya coefficient

stops at position 20. The Diffusion distance continues to discriminate between different orientations when the distance between clusters is large.

Conclusions The results show that the KL-Divergence and Bhattacharyya bin to bin metrics are not discriminative when the degree of cluster overlap is low. This is important because according to these metrics two overlapping clusters that differ significantly in orientation score higher than two non-overlapping clusters that share the same orientation. Furthermore, these results highlight the value of using manually defined metrics such as the total Euclidean distance and the Average Mahalanobis distance; the advantage of these metrics is noted even though the distributions compared only contain a single cluster.

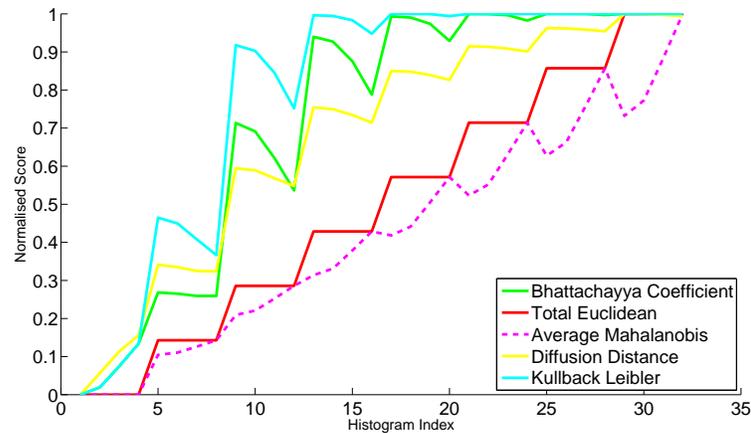


Figure 4.11: Distance metric comparison for the histogram comparisons described in Experiment 1.

Sequence 2: Overlapping cluster bias for increasing total cluster distance The metrics are commonly used to compare multi-modal histograms. The bias towards overlapping clusters demonstrated in sequence 1 motivates an exploration of what happens in the multi-modal case. Since the metrics summarize the alignment of multiple clusters using a single number, the effect of relative improvements in individual clusters is explored.

Hypothesis: When comparing multi-modal histograms, changes in highly overlapping clusters dominate changes in clusters with lower overlap.

Sequence: Both source and target histograms contain 2 clusters each. All clusters are the same size and shape. Both the source and target histograms have the parameters, $P(1) = 0.5$, $P(2) = 0.5$, $\sigma_x = 10$ and $\sigma_y = 30$. The orientation θ is set to 0 for all clusters. The source histogram has cluster centres (0,0) and (50,0). The sequence of cluster transformations for the target histogram is shown in table 4.2. The sequence of four source and target histogram pairs are shown in 4.16(a), 4.16(b), 4.16(c), and 4.16(d). The sequence is designed so that the total Euclidean distance between the corresponding clusters increases in the sequence 0, 10, 15, 20. The 3rd pair in the sequence increases the distance between the 1st component clusters and keeps the second cluster in the same position. The 4th pair in the sequence decreases the distance between the 1st component clusters and increases the distance between the second component clusters.

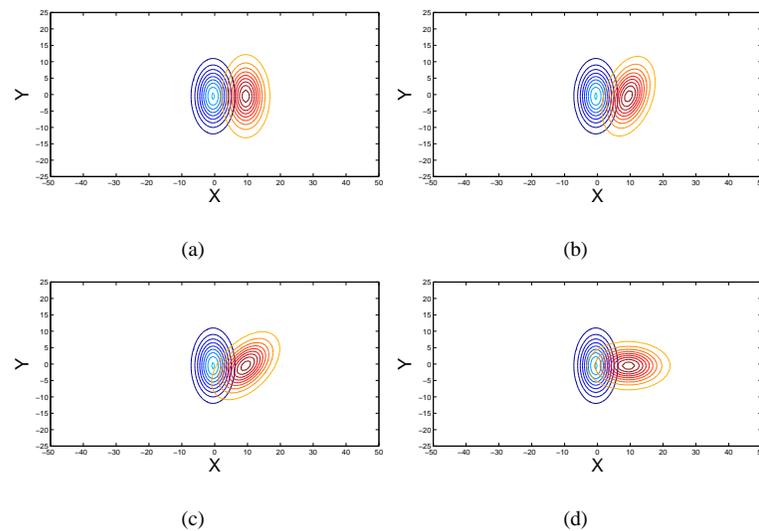


Figure 4.12: Contour plots of source (blue) and target (red) Gaussian distributions for a target Gaussian translation of 5 on the x-axis. Target values of θ illustrated are 0 4.12(a), 20 4.12(b), 45 4.12(c) and 90 4.12(d). The complete sequence repeats these target cluster rotations at different translated positions.

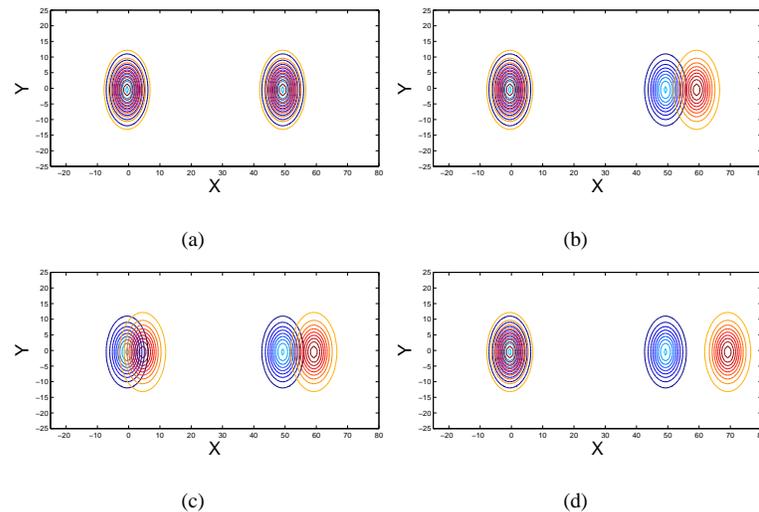


Figure 4.13: Contour plots of source (blue) and target (red) bimodal distributions. In 4.13(a) the distributions are identical, 4.13(b) moves the 2nd target cluster by 10 along the x axis. 4.13(c) moves the 1st target cluster by 5 along the x-axis, keeping the second cluster displacement at 10. 4.13(d) aligns the first cluster components and displaces the second cluster by a distance of 20. The total distance between corresponding clusters is increasing across the sequence which allows the bias of metrics towards movement in the overlapping clusters to be investigated.

Table 4.2: Cluster Positions for the target histogram.

| Component 1 mean | Cluster 2 mean |
|------------------|----------------|
| (0,0) | (0,50) |
| (0,0) | (0,60) |
| (0,5) | (0,60) |
| (0,0) | (0,70) |

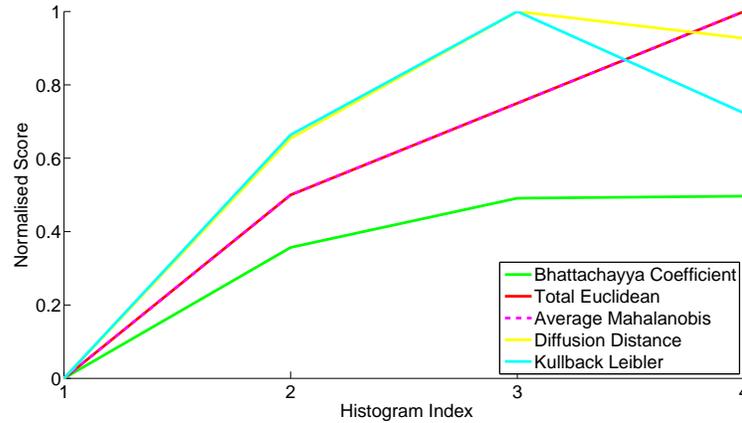


Figure 4.14: Distance metric comparison for the histogram comparisons described in Experiment 2.

Results Figure 4.14 shows plots of the metrics. Representative transformations from the sequence are illustrated in Figure 4.13. The second cluster becomes non-overlapping at position 4 in the sequence. The Diffusion distance and the Kullback Leibler distance decrease from the 3rd to 4th position. All other metrics increase between these positions. The Bhattacharyya coefficient only increases a small amount, the total Euclidean and Average Mahalanobis give the same values across the the sequence because there is no difference in the orientation of the clusters.

Conclusions The Kullback-Leibler and Diffusion distance metrics show heavy bias from the non-overlapping cluster, the decrease in the value of these metrics contradicts the increase shown by the other metrics. The bin-bin and cross-bin measures can be used when it is acceptable to heavily penalise non-overlapping clusters. Manually defined metrics have clear benefits when considering clusters that do not always overlap because they discriminate between alignment improvements of non-overlapping clusters.

Sequence 3: Large cluster bias variation under equivalent translation Multi-modal histograms commonly contain clusters that are different sizes. Depending on the application, the smaller clusters in a histogram may represent very important information. When using a metric to summarize the alignment of multiple modes it is important to understand how the metric changes with movement of the larger clusters.

Hypothesis: Transformation of the larger clusters in a multi-modal histogram comparison has the great-

est effect on the metric.

Sequence: Both source and target histograms contain 2 clusters each. The first cluster in both histograms has the mixing parameter, $P(1) = 0.7$. The second cluster in both histograms has a mixing parameter, $P(2) = 0.3$. All clusters use $\sigma_x=10, \sigma_y=30$. The mean of the 1st cluster in the source histogram is (0,0), the mean of the second cluster is (0,50). The sequence of cluster means for the target histogram is shown in table 4.3. Figure 4.16 shows contour plots of the overlaid histograms for the sequence.

Results Figure 4.15 shows plots of the metrics. At position 2 in the sequence the larger cluster is offset by 5 units, at position 3 the smaller cluster is offset by 5 units. The Kullback-Leibler distance and the Diffusion distance exhibit a large variation due to the movement of the different sized clusters. The variation in the Bhattacharyya coefficient is smaller, but present. The manually defined metrics show invariance to the movement of different sized clusters.

Conclusions When comparing multi-modal histograms movement of the largest clusters dominates the scores computed by the bin-bin and cross bin metrics. Manually defined metrics alleviate this problem and are a good choice when manual labeling is possible and it is important to consider the alignment of a number of metrics irrespective of their individual sizes.

Table 4.3: Cluster Positions for the target histogram.

| Component 1 mean | Cluster 2 mean |
|------------------|----------------|
| (0,0) | (0,50) |
| (0,5) | (0,50) |
| (0,0) | (0,55) |
| (0,5) | (0,55) |

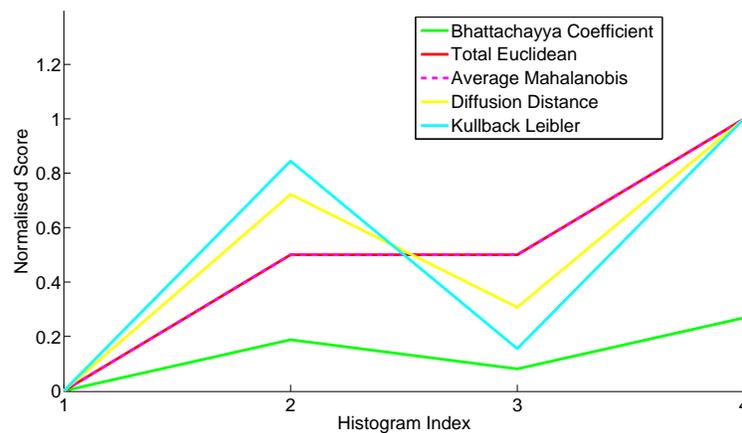


Figure 4.15: Distance metric comparison Experiment 3.

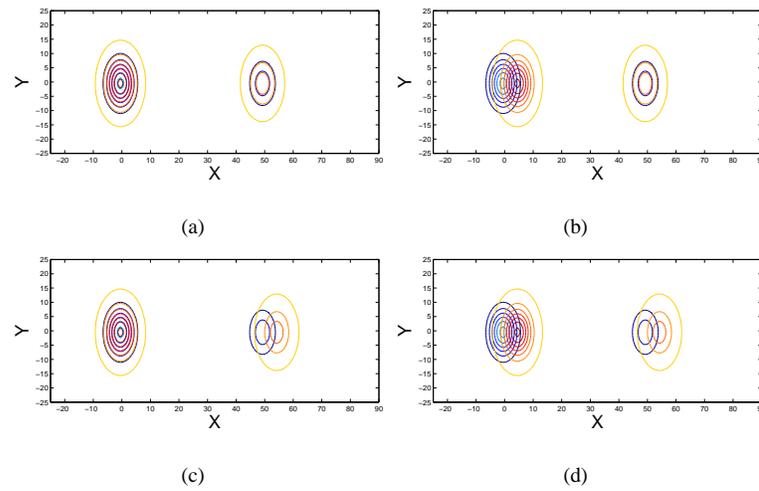


Figure 4.16: Contour plots of source (blue) and target (red) bimodal distributions. For both histograms the first cluster has a weight of 0.7 and the second cluster has a weight of 0.3. 4.16(a) shows the histograms perfectly overlapping. In order to investigate the bias on metrics of the larger overlapping clusters the large and small clusters are individually translated (in 4.16(b) and 4.16(c)) before moving both clusters together 4.16(d). 4.16(b) moves the larger cluster, 4.16(c) moves the smaller cluster and 4.16(d) moves both clusters together.

4.3.5 Discussion

The empirical evaluation highlights a number of advantages of the manually defined metrics over the more commonly used bin-bin and cross-bin metrics. The manually defined metrics discriminate between alignments when the corresponding clusters are far apart, they also evaluate the movement of overlapping clusters and larger clusters more fairly than bin-bin and cross-bin metrics. The average Mahalanobis distance is the best metric for evaluating histogram alignment in a colour inconsistency removal application. This is because clusters may not overlap and can be of different sizes, the average Mahalanobis distance can rank alignments that produce incremental improvements fairly. The average Mahalanobis distance is chosen over the total Euclidean distance as it considers the orientations of the corresponding clusters.

4.4 Quantitative evaluation of RGB colour alignment

This section uses the UCLColVariation database and the developed methodology to improve the current understanding of colour inconsistency removal methods. Colour inconsistency removal transformations are comprehensively evaluated and FBHA is compared to competing approaches. Additionally, two assumptions of the FBHA approach are investigated. The first assumption is that point alignment transforms give better performance than non-point alignment transforms, it is important to investigate this assumption because a key benefit of FBHA is the ability to use point alignment transforms. However, the extra work to perform automatic feature detection and matching is only justified if the point alignment transforms show superior performance. Second, the closest total Euclidean distance matching strategy

used by FBHA is investigated; the mean colours of manually labeled regions are used as ground truth features to check that matched results are sensible. Finally, the performance of FBHA is evaluated and its behavior is explored.

4.4.1 Experiment 1: Feature Based Alignment Hypothesis

Aims:

This experiment compares different colour inconsistency removal transforms on the UCLColVariation data-base. The comprehensive ranking of these transforms provides important information about the best ways to remove colour inconsistencies. Additionally, the ranked transforms allow point alignment transformations to be compared to alternatives. If point alignment transforms perform best then this motivates the FBHA approach.

Hypothesis:

Transformations that align local histogram features give better alignment scores than transforms that use global properties of the histograms when applied to image pairs in the alignment database.

Method:

This experiment extracts all 1770 image pairs for each object set. For each image pair, one image is the source image and the other is the target image. A list of n transforms are used to produce n transformed images for each source image. Each transformed image is compared to the target image using the average Mahalanobis metric to produce n results scores per image pair. Results are grouped by transform and image variation set for comparison, the distribution of results for each transform are compared to produce a ranking of the transformations under the experimental conditions of the image variation set.

The list of transforms is presented here. For detailed mathematical descriptions refer back to section 2.5. Parameters are specified along with a short-hand code for subsequent identification. The non feature-point transforms used are:

1. **Identity Transform.** Code: Untouched.
2. **Additive alignment of the 1st moment** in each channel using equations 2.11 and 2.12. Code: Moment1-ShiftEachChan.
3. **Multiplicative alignment of the 1st moment** in each colour channel using equation 2.13. Code: Moment1-MultEachChan.
4. **Alignment of 1st and 2nd moments** in each colour channel using linear transforms computed using equations 2.14, 2.15 and 2.16. Code: Moment1-2-MultiShiftEachChan.
5. **Histogram equalization** performs a standard histogram equalisation in each channel. Code: HistEqData, described in section 2.5.2.
6. **Histogram matching** finds the monotonic transform in each channel that matches the source and target histograms in each channel. Code HistMatchData, described in section 2.5.2.

7. **SVD based principal axis alignment.** The method of Xiao and Ma [53] computes a homogeneous rotation, scaling and translation that aligns the principal axes and means of a source and target data-set. Code: SVDSimilarityTrans, described in section 2.5.2.

The feature point alignment methods use the mean RGB colours extracted from the training region of the marked up polygons associated with each image. These features are considered to be the best ground truth available because they are obtained from regions of the image manually annotated by a human, the feature correspondences are also known from the mark up data. Features are manually provided in this experiment to test the experimental hypothesis. The feature point transforms are:

1. **Multiplicative feature point alignment** using equation 2.18. Code: AlignPtsGain.
2. **Additive feature point alignment** using equation 2.17. Code: AlignPtsShift.
3. **N by N feature point alignment** using equation 2.21. Code: AlignPtsNbyN.
4. **Independent linear feature point alignment** using equation 2.19 with $d = 1$. Code: NDIndepPolyOrder1.
5. **Independent quadratic feature point alignment** using equation 2.19 with $d = 2$. Code: NDIndepPolyOrder2.
6. **Independent Cubic feature point alignment** using equation 2.19 with $d = 3$. Code: NDIndepPolyOrder3.
7. **Independent quartic feature point alignment** using equation 2.19 with $d = 4$. Code: NDIndepPolyOrder4.
8. **Correlated linear feature point alignment** using equation 2.20 with $d = 1$. Code: NDCorrPolyOrder1.
9. **Correlated quadratic feature point alignment** using equation 2.20 with $d = 2$. Code: NDCorrPolyOrder2.
10. **Correlated cubic feature point alignment** using equation 2.20 with $d = 3$. Code: NDCorrPolyOrder3.
11. **Correlated quartic feature point alignment** using 2.20, $d = 4$. Code: NDCorrPolyOrder4.

Investigation of the distribution of average Mahalanobis distances for each transform reveals that the results histograms for a transform are highly skewed and non-Gaussian. Ranking the transforms requires a meaningful ordering of these results distributions. For Gaussian distributions a paired t-test is commonly used; however, the non-Gaussian form of these distributions mean that the t-test is inappropriate.

Non-parametric boot-strap statistics are a computational method of performing statistical inference that are based on random re-sampling with replacement of the original data. The boot-strap procedure allows confidence intervals to be constructed when a parametric formula is not available to describe

the data; a key advantage of the approach is that it is simple to implement. Efron [83] provides a detailed coverage of boot-strap methods. In this work the boot strap procedure compares the distribution means; to achieve this, results distributions are compared by computing a confidence interval around the sampled mean of each distribution. Histogram alignment increases as the Mahalanobis score decreases, so if the mean of a results distribution for a transform A is significantly lower the mean of the results distribution for transform B then transform A performs better with a degree of confidence. The pseudo-code in algorithm 4 outlines the steps to determine whether a transform A or B scores better. Going through the steps in the pseudo-code, \mathbf{r}_A is a vector of all scores for transform A and \mathbf{r}_B is a vector of all results of transform B . The next loop computes estimates of the distribution mean. The function *Random.SampWithReplacement* takes the vector of scores and produces a new sample with the same number of elements, the new sample is produced by repeatedly sampling a randomly selected value from the vector until a new sample of the same size is collected. The original vector remains untouched. The expectation operation $E()$ computes the mean value of the re-sampled set of values at each iteration and assigns mean values to the i th element of the arrays *BootStrapMeanA* and *BootStrapMeanB*. After the loop, the next line computes a vector of differences, \mathbf{d} , that contains the difference between the corresponding elements of *BootStrapMeanA* and *BootStrapMeanB*. $\text{cdf}(\mathbf{d})$ computes the cumulative histogram of the difference values in \mathbf{d} , then the function *ConfidenceIntervals* extracts the confidence interval limits l_b and u_b at the C confidence level. If zero falls between these limits then there is no significant difference between the distributions. If zero does not fall between these limits then the transform with the lower average score is the best. The overall ranking process is described by the pseudo-code in algorithm 5, the number of times that each transform scores better than all other transforms is counted. A higher count indicates a superior ranking between transforms, all transforms are ranked and ties between transformations are allowed. Position 1 is used to indicate the best transform and thus the highest count.

Results: The total processing time to compute all transformed images and results scores for all image pairs from each of the four object data-sets was approximately 1 month on a Dell Inspiron 1525 laptop with a 2Ghz dual core processor and 2GB of RAM; a single processor core was used for the computations. A comprehensive list of transformation rankings grouped by variation set and object group is shown in Appendix 9.2. Each ranking is displayed using a colour coded format where a different colour is used for each transform, Figure 4.17 shows the colours used for each transform. Figure 4.18 shows the coloured coded ranking for transformations from the (C)(L-LI)(L-AL)(S) image variation set using the red-cyan paper data-set. Each colour coded box contains a number that indicates the ranking of the transform where 1 is the best and lower positions are worse, the first position is always shown as the bottom box and the last position is shown as the top box. Note that positions 2,9,10,11 and 12 are occupied by multiple transforms in this example, this means that no significant performance difference was detected by the procedure at these respective positions in the ranking. Subsequent positions down the ranking are interpreted as being significantly worse at confidence level C according to the comparison procedure. This section presents ranked transforms for five different image variation sets. These sets are chosen as illustrative examples of the main points, the Appendix 9.2 should be consulted as required.

Algorithm 4 $ScoresBetter(A, B)$: Test if method A scores better than method B

$\mathbf{r}_A \leftarrow$ Average Mahalanobis scores for method A

$\mathbf{r}_B \leftarrow$ Average Mahalanobis scores for method B

for $it = 0$ to $NumBootstapIts$ **do**

$BootstrapMeanA[it] = E(RandomSampWithReplacement(\mathbf{r}_A))$

$BootstrapMeanB[it] = E(RandomSampWithReplacement(\mathbf{r}_B))$

end for

$\mathbf{d} = BootstrapMeanA - BootstrapMeanB$

$[l_b, u_b] = ConfidenceIntervals(cdf(\mathbf{d}), C)$

if $l_b \leq 0 \leq u_b$ **then**

Bootstrap distribution means are not significantly different

else if $E(\mathbf{r}_A) < E(\mathbf{r}_B)$ **then**

Method A results significantly better than method B results with confidence C

else

Method A results significantly worse than method B results with confidence C

end if

Algorithm 5 Rank all alignment methods

for $T1 = 1$ to $NumTransforms$ **do**

for $T2 = 1$ to $NumTransforms$ **do**

if $T1 \neq T2$ **then**

$Results(T1, T2) = ScoresBetter(AllScores(T1), AllScores(T2))$

end if

end for

end for

Group $Results$ by transform type and sort from lowest(best) to highest(worst).

Count the number of transforms outperformed for each transform to give the final ranking. Ties are allowed.

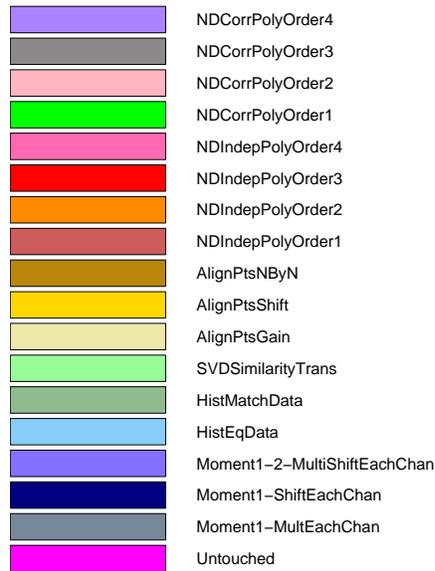


Figure 4.17: Colour coding scheme to represent the different alignment transforms.

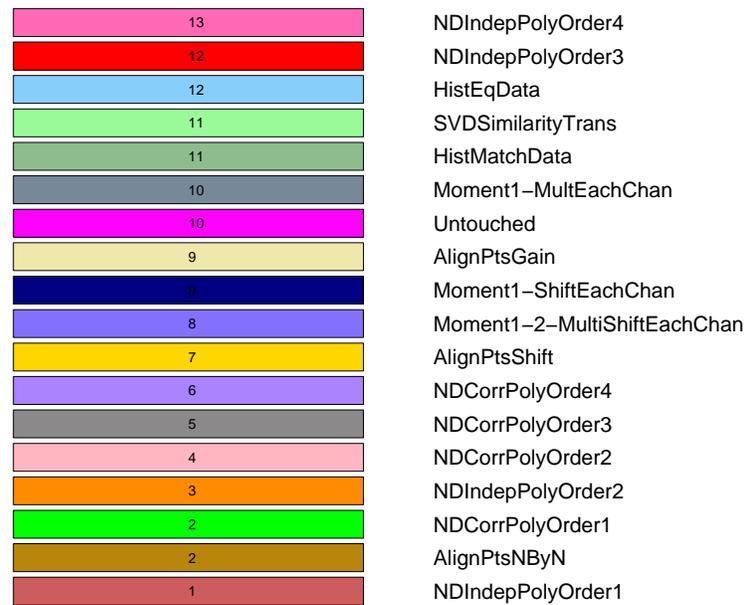
The highlighted results are:

1. All image pairs (all variations) in Figure 4.19,
2. (C)(L-LI)(L-AL)(S) in Figure 4.20,
3. 0(L-LI)(L-AL)0 in Figure 4.21 ,
4. 0(L-LI)(L-AL)(S) in Figure 4.22,
5. 00(L-AL)0 in Figure 4.23.

A further break down of the structure of the transform ranking variation on an image by image basis for the Red-Cyan data-set is shown in Figure 4.24(a). Figures 4.24(a) to 4.24(f) show the transformations that ranked in the 1st to 6th positions respectively for all 1770 image pairs. A coloured square is used to represent the transform and show how transform performance varies on an image by image basis.

Conclusions: Two key findings from this experiment are: 1) Transformation performance varies significantly across capture conditions and data-sets, 2) Feature point transforms robustly align colour histograms with the highest degree of alignment. Elaborating on these findings:

1. **Transformation performance variation** The transformation rankings computed using the bootstrap statistic procedure show that feature point transforms perform well. Also, a transformation that performs well under one set of experimental condition can perform badly under another. For example, the NDCorrPolyOrder1 transform performs well on the skittles data-set under 0(L-LI)(L-AL)0, 0(L-LI)(L-AL)(S) and 00(L-AL)0 variations shown in Figures 4.21(b), 4.22(b) and 4.23(b)



(a)

Figure 4.18: Ranked transformation methods with (C)(L-LI)(L-AL)(S) variation : 1) Red-cyan paper

respectively. However, Figure 4.20(b) shows the same transform performing very badly under (C)(L-LI)(L-AL)(S) variation. This performance variation tells us that the best transformations must be chosen on a per data-set and experimental variation basis to give the most significant levels of performance improvement. Transformation rankings differ across data-sets and experimental conditions, a change in scene objects leads to the biggest variation in the performance of the transformations. Observe that selecting an independent linear point alignment transform (NDIndepPolyOrder1) gives robust performance improvements across the data-sets and experimental conditions.

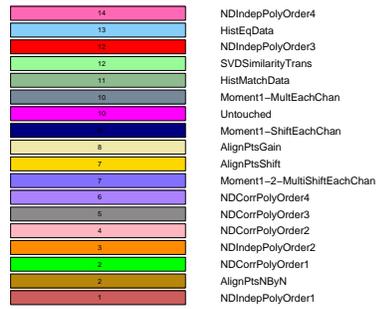
Recall that the boot-strap ranking procedure is necessary because of the high variability of the results for each transform, the ranking shows that there is a performance hierarchy among the transforms. However, transform performance can vary significantly on an image by image basis. Figure 4.24(a) provides an intuition for how variable the results are, it shows that no single method performs best across the different image pairs. By contrast, note the existence of structure in the first and second positions in Figures 4.24(a) and 4.24(b), disorder increases from the first to sixth position in 4.24(f) where no transform ranks consistently in sixth position. This tells us that the best histogram alignment transform varies between image pairs, even when the images are of similar objects. This is an important result for computer vision designers seeking colour inconsistency removal transforms. It means that it is possible to select a transform that performs

reasonably well over a range of conditions, but the best transform for an image pair must be found on a case by case basis. Also, the experiment has shown that outlier results are typical across the different transforms; a transform that performs well on one image pair can perform badly on another with seemingly innocuous differences in experimental capture condition.

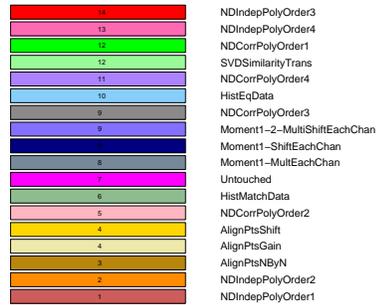
2. **Feature point transforms** The results demonstrate that a point feature based alignment transform always performs better than the next best non-point feature based method. However, not all point feature transforms outperform non-point feature based methods; in particular, third and fourth order polynomial transforms are susceptible to performing badly due to over-fitting the data. This supports the idea that a well chosen feature point alignment transform can robustly align histograms; the original hypothesis that all point-feature transforms perform better than non-point feature methods cannot be supported as we find some that perform badly. The correlated polynomials give some of the best alignment scores but are susceptible to failing poorly under some conditions such as the skittles data-set in 4.19(b). The linear correlated polynomial is robust across different conditions where the camera is held constant - this hints that correlated transforms could be of greater use when calibrating between colour data obtained from the same camera (this argument could extend to cameras of the same make and model).

Other observations are:

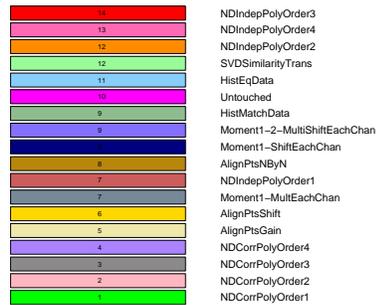
- The SVD alignment method of Xiao and Ma [53] (Code:SVDSimilarityTrans) performs badly across all examples. This shows that aligning the two multi-modal colour distributions using rotation, scaling and translation based on the principal axes of the distributions is not a good idea if alignment of the individual modes is the desired goal. The results presented in the original paper offer no quantitative validation and it is thought that this method may be of value aligning uni-modal or near uni-modal distributions.
- Histogram equalization (Code:HistEqData) and matching (Code:HistMatchData) perform uniformly badly. In particular, histogram equalization has no knowledge about the target distribution. Both methods give poor alignment of the distribution modes and need not be considered further.



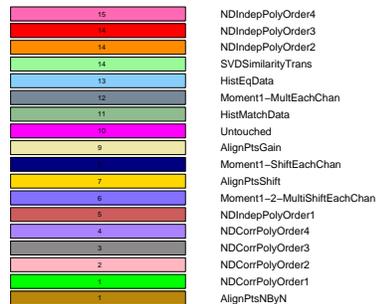
(a)



(b)

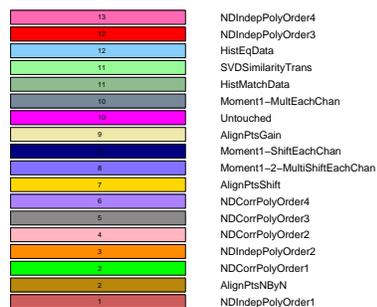


(c)

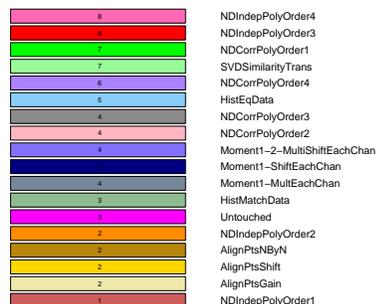


(d)

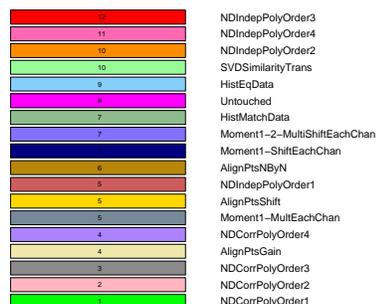
Figure 4.19: Ranked transformation methods for all 1770 image pairs from : 1) Red-cyan paper 4.19(a), 2) Skittles 4.19(b), Teddy bears 4.19(c) and three paper strips 4.19(d).



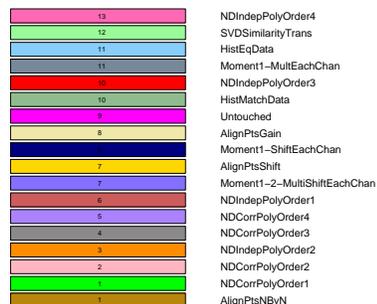
(a)



(b)

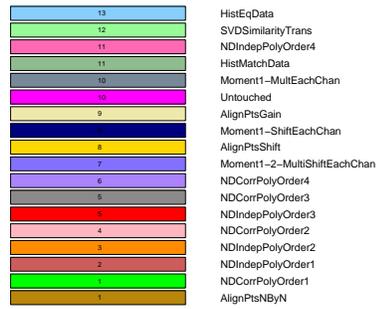


(c)

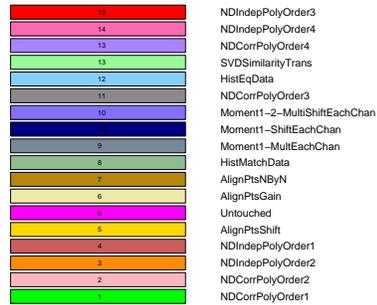


(d)

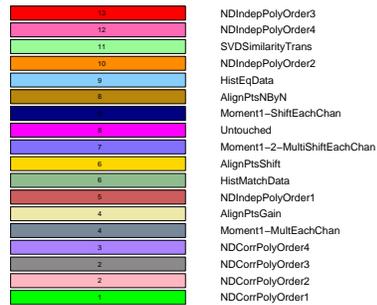
Figure 4.20: Ranked transformation methods with (C)(L-LI)(L-AL)(S) variation : 1) Red-cyan paper 4.20(a), 2) Skittles 4.20(b), Teddy bears 4.20(c) and three paper strips 4.20(d).



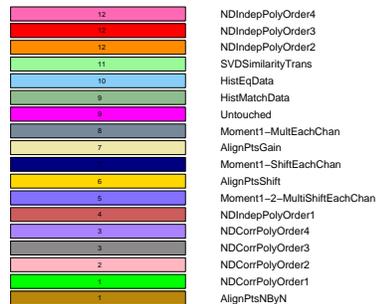
(a)



(b)

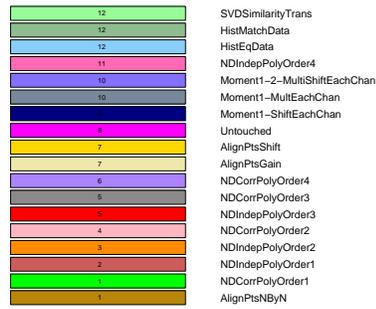


(c)

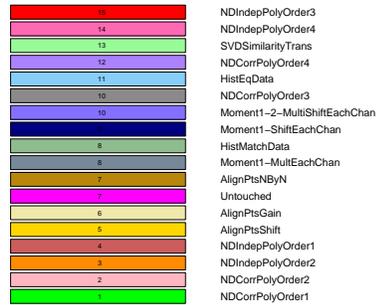


(d)

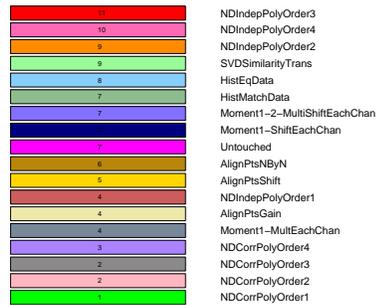
Figure 4.21: Ranked transformation methods for image pairs with 0(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 4.21(a), 2) Skittles 4.21(b), Teddy bears 4.21(c) and three paper strips 4.21(d).



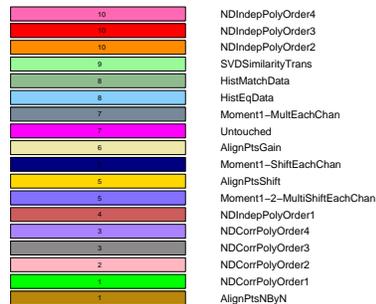
(a)



(b)

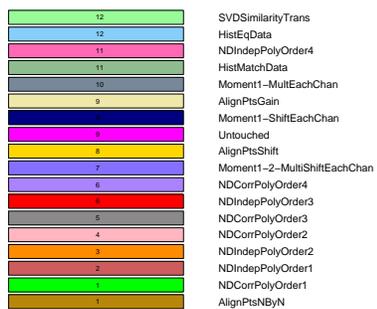


(c)

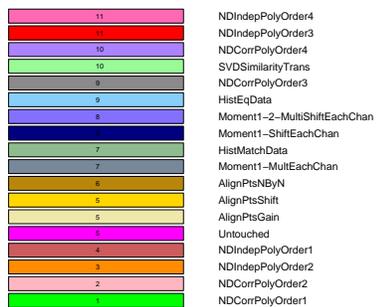


(d)

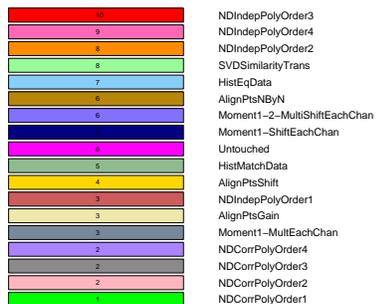
Figure 4.22: Ranked transformation methods for image pairs with 0(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.22(a), 2) Skittles 4.22(b), Teddy bears 4.22(c) and three paper strips 4.22(d).



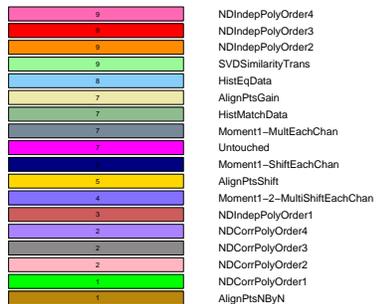
(a)



(b)



(c)



(d)

Figure 4.23: Ranked transformation methods for image pairs with 00(L-AL)0 variation for: 1) Red-cyan paper 4.23(a), 2) Skittles 4.23(b), Teddy bears 4.23(c) and three paper strips 4.23(d).

The ranked transformations for each image pair in the set are represented using a $z \times z$ colour coded matrix. For the n th ranked position, the matrix tells us which transforms ranked at the n th position for the different source and target alignments. Rows index the source images and columns index the target images. The upper triangular part of the matrix is populated according to the colour scheme described in figure 4.17. Black entries in the matrix indicate that no transformation and evaluation was performed for the indexed source and target image combination. Performing the transformations indicated by the non-diagonal black entries would reveal whether the structure of the results is symmetric, it is suspected that such an investigation would reveal a non-symmetric structure.

4.4.2 Experiment 2: Closest Euclidean Feature Match Hypothesis

Aims: This experiment tests whether the minimum total Euclidean distance matches features correctly. This is tested because FBHA uses this during the matching step. The features used in the evaluation are the mean RGB colours of the hand marked up regions for each image (*the ground truth features*).

Hypothesis: The correct match between ground truth features can be found by choosing the match with the minimum total Euclidean distance between points.

Method The CEM feature matching method described in section 3.1.2 is tested. For each of the 1770 image pairs in all four object sets the mean RGB colours of the masked regions for both images are computed. The first image in the pair has a mean RGB colours, \mathbf{W} , and the second image has b mean RGB colours, \mathbf{Q} . The masks always contain the same number of marked up regions, so, $a = b$ for each image pair. All possible matches are enumerated and the total Euclidean distance between matched points is computed for each match. The match with the minimum total Euclidean distance is compared to the correct match known from the mask mark-up.

Results All 1770 matches for all four object data-sets matched correctly.

Conclusions The correct match is picked correctly in all cases by the minimum total Euclidean distance. The strength of this constraint is surprising, especially given the different types of variation in the database. The success of this test indicates that if the histogram features can be found robustly and accurately then the minimum total Euclidean distance between features is a good constraint to match with.

4.4.3 Experiment 3: FBHA comparison

Aims This experiment compares FBHA to alternative transforms. Experiment 1 demonstrates that feature point alignment transforms perform well when using features computed from labeled masks. First, FBHA is compared to the entire list of candidate transforms used in Experiment 1, this contrasts the impact of manually defined features with features that are automatically detected and matched. Second, FBHA is compared to transforms that can be used without manual intervention or other forms of image based feature processing. This shows how FBHA compares to its direct competitors.

Hypothesis Automatic FBHA methods perform better than manually specified alternatives.

Method Three FBHA configurations are run on three different image variation sets for all four object data-sets. The bootstrap transformation comparison procedure is run to compare all transforms listed in Experiment 1 with the three FBHA configurations.

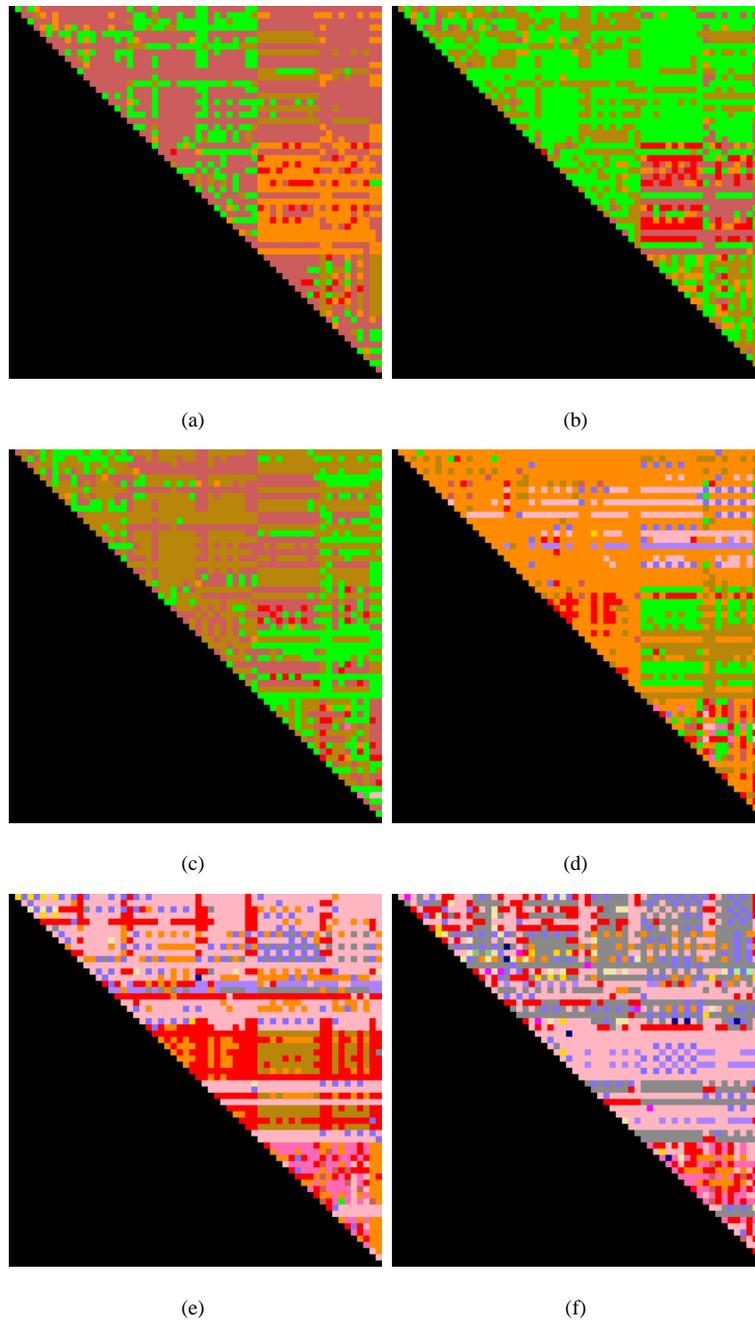


Figure 4.24: Colour coded matrices indicating the transformations that ranked 1st 4.24(a), 2nd 4.24(b), 3rd 4.24(c), 4th 4.24(d), 5th 4.24(d) and 6th 4.24(f). There are 60 images in this set, a coloured entry in the i th row and j th column indicates the transform that mapped the i th image to the j th image in the set and gave an average mahalanobis score that ranked at the position represented by the matrix. The colour coding scheme is shown in Figure 4.17.

The three FBHA configurations were used in section 3.2, the short-hand codes for these are:

1. (**Code:** [1D-Maxima]-[1DSS-[1]]-CEM). 1D deep structure feature detection in each of the red, green and blue source and target histograms. Feature detection parameters are $\gamma = 0.005$, $T = 9$ and a path is followed in the scale space if connected by 1 bin. CEM feature matching is performed and a linear feature point transform aligns the source and target points in each channel.
2. (**Code:** [1D-Maxima]-[1DSS-[1]]-CEMDC). This configuration is the same as ([1D-Maxima]-[1DSS-[1]]-CEM) except CEMDC feature matching is used. This matching strategy eliminates matches that do not preserve rank ordering.
3. (**Code:** [RG2D-B1DMaxima]-CEM2D-CEMDC). The deep structure feature detection on the RG histogram uses $\gamma = 0.0002$ and $T = 11$. The connectivity rule for the path following step connects a local maxima to a current path if the local maxima is in the nine neighbouring bins at the end of the path. The 1D FBHA in the blue channel uses $\gamma = 0.005$ and $T = 9$. **CEM** matching is used in the RG channels and **CEMDC** matching is used in the blue channel. Detected and matched features are used to compute a linear feature point transform that aligns source and target points in each channel.

The image variation sets used are 0(L-LI)(L-AL)0 , 0(L-LI)(L-AL)(S), (C)(L-LI)(L-AL)(S). The variation set 0(L-LI)(L-AL)0 is chosen to examine the effects of lighting variation and 0(L-LI)(L-AL)(S) is chosen to see whether object scale effects the results under the same colour inconsistency conditions. (C)(L-LI)(L-AL)(S) is used to compare the transforms when all experimental conditions are varying.

The second part of the experiment compares FBHA against methods that do not use the labeled data from the image mask, these methods are focussed upon because they compete directly with FBHA. The methods from Experiment 1 that require no manual intervention are:

1. Multiplicative alignment of the 1st moment. Code: Moment1-MultEachChan.
2. Additive alignment of the 1st moment. Code: Moment1-ShiftEachChan.
3. Alignment of 1st and 2nd moments. Code: Moment1-2-MultiShiftEachChan.
4. Histogram equalization. Code: HistEqData.
5. Histogram matching. Code: HistMatch.
6. SVD based principal axis alignment. Code: SVDSimilarityTrans.

The bootstrap procedure is used to compare this list of transforms with the FBHA methods, the number of times that a transform performs best is counted. This is done because initial tests showed that FBHA does not perform well when using the full ranking procedure. Instead a simple count is used to show the number of times that a FBHA method performs best. The cases when FBHA performs worse than all other methods in the list are classified as failure cases, these cases are inspected manually and categorized. The failure case categorization is valuable as it highlights assumptions of the FBHA method that

are not applicable to the data.

Results

Figures 4.30, 4.31 and 4.32 show the ranked results for the 0(L-LI)(L-AL)0, 0(L-LI)(L-AL)(S) and (C)(L-LI)(L-AL)(S) variation sets respectively. The rankings do not show a consistent performance advantage of FBHA over the other automatic methods and so the initial hypothesis is rejected. The performance of FBHA is found to be highly variable across the different image variation sets and object sets. For example, 1D FBHA outperforms a multiplicative alignment of the means for the skittles data-set under 0(L-LI)(L-AL)0 and 0(L-LI)(L-AL)(S) variation sets shown in Figures 4.30(b) and 4.31(b) respectively; however, 1D performs poorly under (C)(L-LI)(L-AL)(S) variation show in figure 4.32(b).

Other observations are:

1. The performance of the FBHA approach is not comparable to feature point alignment that use features computed from the manually labeled masks.
2. 1D FBHA methods perform better than the hybrid ([RG2D-B1DMaxima]-CEM2D-CEMDC) method.
3. FBHA transforms perform better than some transforms that do not require manual intervention, but the ordering of the rankings varies considerably between different conditions and data-sets.

Figures 4.33 and 4.34 show bar charts of the number of times that an automatic transform performs best for the 0(L-LI)(L-AL)0 variation set for all object data sets. The count is normalised between the 0-1 range. Figures 4.35 and 4.36 show the counts for the 0(L-LI)(L-AL)(S) variation set for all objects. Figures 4.37 and 4.38 show the (C)(L-LI)(L-AL)(S) variation set results. Transform performance varies between variation set conditions and different object sets, the transforms that perform best most frequently are Moment1-2-MultShiftEachChan and Moment1-MultEachChan. Interestingly, the identity transform (Untouched) performs best for the (0)(L-LI)(L-AL)0 and (C)(L-LI)(L-AL)(S) variation sets with the red, green and blue paper strips data set; this means that the histograms of the images in this variation set are in better initial alignment than the histograms transformed by either the FBHA method or moment based transforms. The FBHA method [1D-Maxima]-[1DSS-[1]]-CEM performs best under 0(L-LI)(L-AL)(S) variation for the red-cyan data set. Although a single FBHA does not perform better than the alternatives in a consistent way, the different FBHA methods perform better than the alternatives approximately 30 percent of the time on average and between a range of approximately 10 to 55 percent.

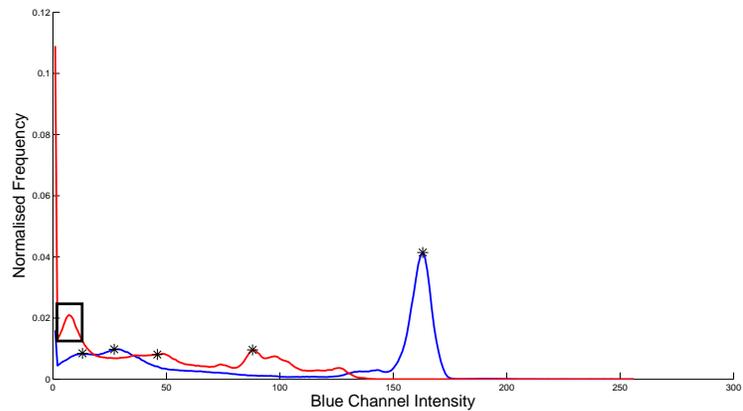
The list of FBHA failure cases compiled by hand illustrates when FBHA fails and why this is so, they also illustrate when the structure of the histograms is mismatched according to the FBHA assumptions. These failure cases are an important contribution as they identify specific problems that must be solved by future work to improve the FBHA framework. The failure case categories are identified from image pairs with poor alignment scores when using the FBHA methods, the cases are categorized according to:

1. Correctness of the feature detection step.

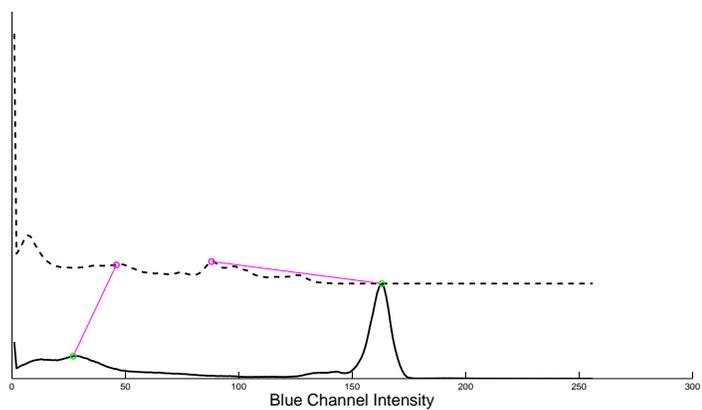
2. Correctness of matching step.
3. Match between structural features of the histograms.

The cases highlighted using 1D FBHA with CEM matching, these are:

1. **False negative feature detection and incorrect matching.** Figure 4.25 shows an example of a false negative feature detection that typically results when the deep structure path threshold, T , is set too high. Recall that a single set of threshold values are used across all experimental conditions and data-sets. Although picking a single threshold has proven reasonably robust, feature detection failures can occur. Figure 4.26(a) highlights the position of the missing feature and figure 4.26(b) shows the resulting matched features. The result is that a significant peak in the target histogram plays no part in the alignment.
2. **False positive feature detection and incorrect matching.** An irrelevant feature can be detected at erroneous feature points, this can happen at regions that contain spikes in the histograms. Figure 4.26(b) shows an example of the false positive feature in figure 4.26(a) that leads to a false match in figure 4.26(b). Another example of the potentially catastrophic effects on the matching of misplaced features is shown in figure 4.27, figure 4.29(a) shows the detected features and highlights two features that have been detected at almost the same position. Figure 4.27(b) shows the resulting matches from these features, they do not preserve ranking ordering and result in poor alignment. The CEM-DC drops these matches but then can not align these parts of the histogram as a result. False positive features can be mitigated by increasing the deep structure threshold, a balance exists between removing these features and maintaining robust detection of the true features.
3. **Correct feature detection, incorrect matching, structural mismatch between 1 pair of corresponding clusters** Figure 4.28 shows how a mismatch in the structure of corresponding clusters can confound FBHA. The centre cluster in the source histogram has two peaks and the centre cluster of the target histogram has one. The CEM matching scheme associates each peak in the centre cluster of the source histogram to two different clusters in the target histogram.
4. **Correct feature detection, incorrect matching, structural mismatch between multiple pairs of corresponding clusters.** Figure 4.29 shows how mismatches in the structure of multiple corresponding clusters can confound FBHA. The source and target histograms show a matching cluster on the left side of the plot, the source histogram cluster has two peaks and the corresponding target histogram cluster has one. The centre target cluster has two peaks and its corresponding cluster in the source histogram has one. The corresponding clusters at the right hand side of the source and target plots also have a different number of peaks. The effect of these structural mismatches in the histograms is that matched features lead to mismatches between the clusters as shown in figure 4.27(b).

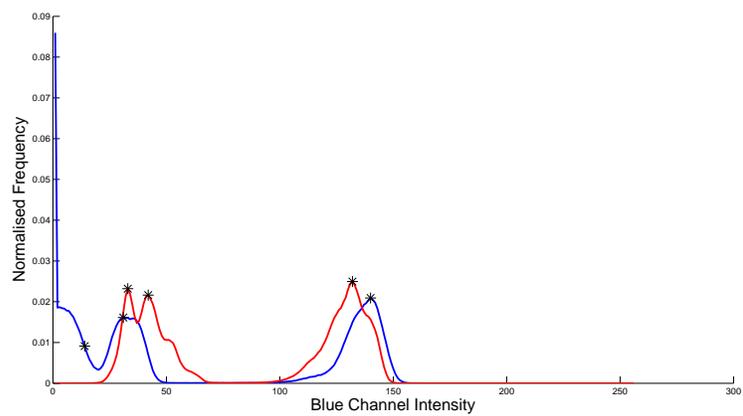


(a)

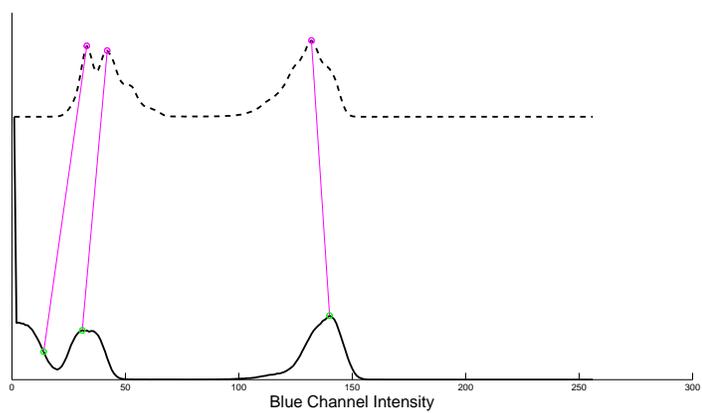


(b)

Figure 4.25: Example of false negative feature detection and incorrect matching. 4.26(a) shows source (blue plot) and target histograms (red plot) in the blue channel for the teddy bears data set. The location of the missing feature in the target histogram is highlighted by the black box. 4.26(b) shows the final matches produced by CEM.

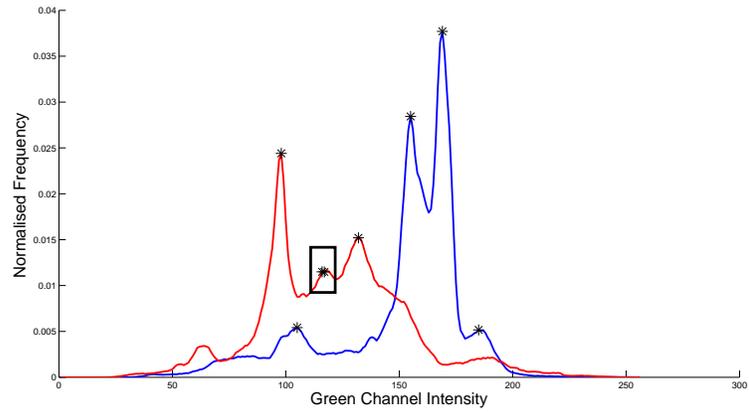


(a)

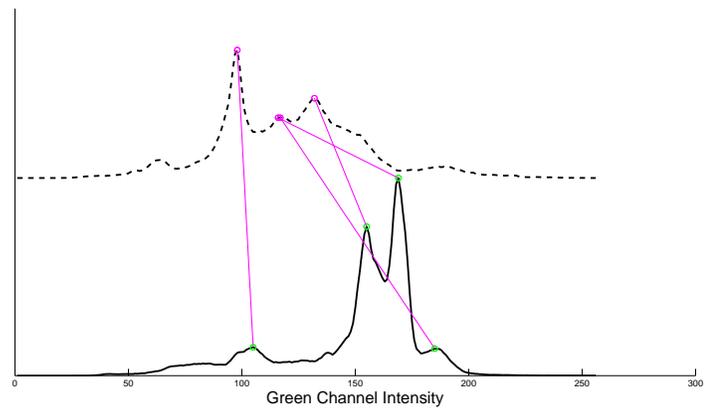


(b)

Figure 4.26: Example of false positive feature detection and incorrect matching. The left most match is deemed to be incorrect. 4.26(a) shows the detected features and 4.26(b) shows the matches.

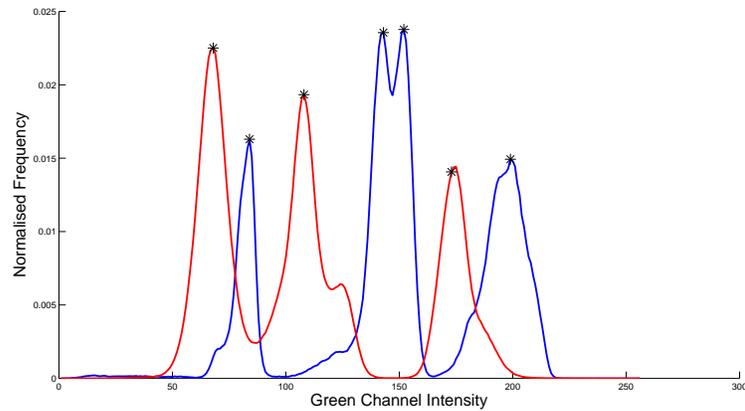


(a)

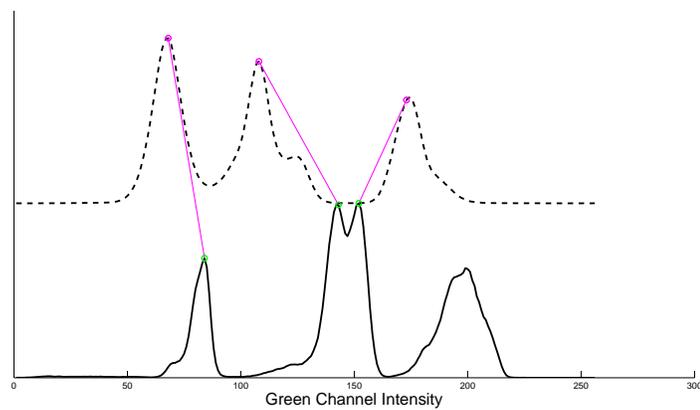


(b)

Figure 4.27: Example of false positive feature detection resulting in catastrophic matching failure. 4.27(a) shows the detected features and 4.27(b) shows the matches.

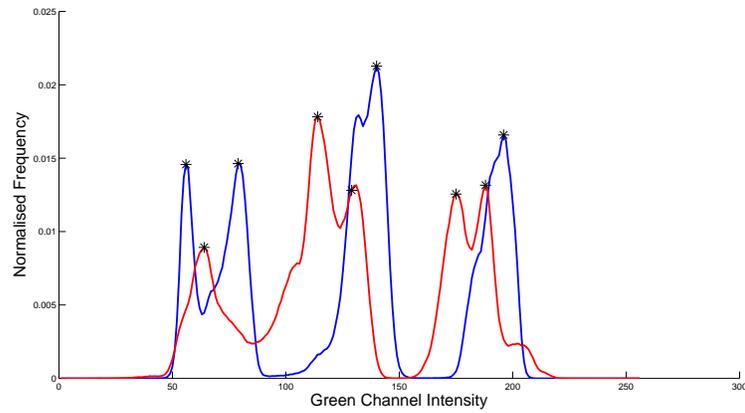


(a)

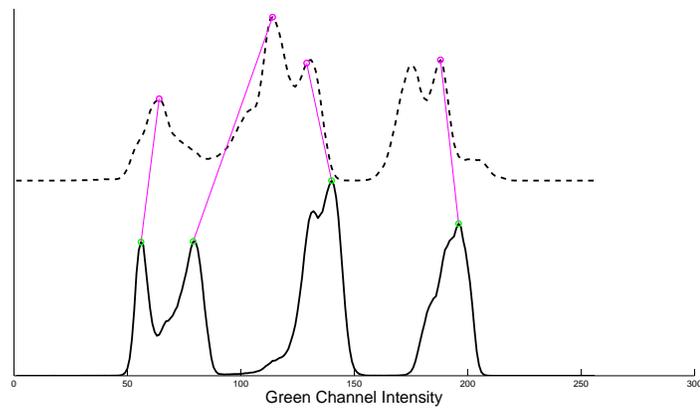


(b)

Figure 4.28: Example of correct feature detection, incorrect matching and a structural mismatch in 1 cluster. Source and target histograms from the green channels of images of the red, green and blue paper strips are shown in 4.28(a), the red plot is the target histogram and the blue plot is the source histogram. Detected features are shown as crosses. 4.28(b) shows an exploded view of the source and target histograms and the final correspondences generated by the CEM matching step.



(a)



(b)

Figure 4.29: Example of correct feature detection, incorrect matching and a structural mismatch in 2 clusters. Source and target histograms from the green channels of images of the red, green and blue paper strips are shown in 4.28(a), the red plot is the target histogram and the blue plot is the source histogram. Detected features are shown as crosses. 4.28(b) shows an exploded view of the source and target histograms and the final correspondences generated by the CEM matching step.

Conclusions The initial hypothesis that FBHA performs better than comparable alternatives is rejected because FBHA does not perform robustly across the range of colour inconsistency conditions and data-sets tested. This means that FBHA cannot be substituted for simpler but more robust transforms such as moment alignment transforms under the conditions tested. However, closer inspection of the results shows that FBHA methods give the best performance in 30 percent of cases on average and up to 50 percent of cases under some conditions. The overall ranking of FBHA is low when the failure examples are considered because catastrophic alignment failure frequently results when one of the FBHA failure cases occurs.

The investigation into why FBHA can fail has led to important an important discovery about colour inconsistent data. This is that for colour inconsistent images of simple object sets, the clusters that correspond to each scene colour can vary in unpredictable ways. In particular, it is not sufficient to assume that a single peaked cluster will appear in the colour histogram for each material type present in the imaged scene. A single peaked colour cluster in a histogram from one set of conditions can map to a cluster with multiple peaks across apparently simple changes in colour inconsistency. The most significant conclusion of this is that a peak matching strategy is not sufficient to ensure correct associations between corresponding clusters. It has been possible to discover this because the feature detection step robustly detects features across a wide range of conditions given the same parameters. Although false positive and false negative feature detections occur in this experiment, the structural mismatches between clusters and the inability of FBHA to resolve these are the dominant effect that negatively effects the robustness of FBHA in the experiment.

In summary, FBHA can robustly detect and match histogram peaks using the same set of parameters across a wide range of colour inconsistent data. For cases where the corresponding histogram clusters have a single significant peak, the FBHA approach produces good results and provides a distinct advantage over other methods. However, structural mismatches in the corresponding clusters occur frequently in colour inconsistent data so more robust performance will only be achievable if extra steps are taken to reason about what constitutes a single cluster. Suggestions for future work that may lead to improved FBHA robustness are discussed in Chapter 6.

4.5 Conclusions

This chapter makes four key contributions:

1. **A freely available data-base for evaluating colour inconsistency correction methods is introduced by the author.** The data-base is unique because it contains examples of colour inconsistency for simple scenes containing a low number of easily identified material properties; this data-base structure allows the colour histograms to be studied with reasonable expectations about the number of clusters present. The data-base introduces different physical sources of colour inconsistency so that different physical situations can be studied. Because the data-base contains ground truth labels for each image, it could be useful for evaluating the performance of clustering algorithms.

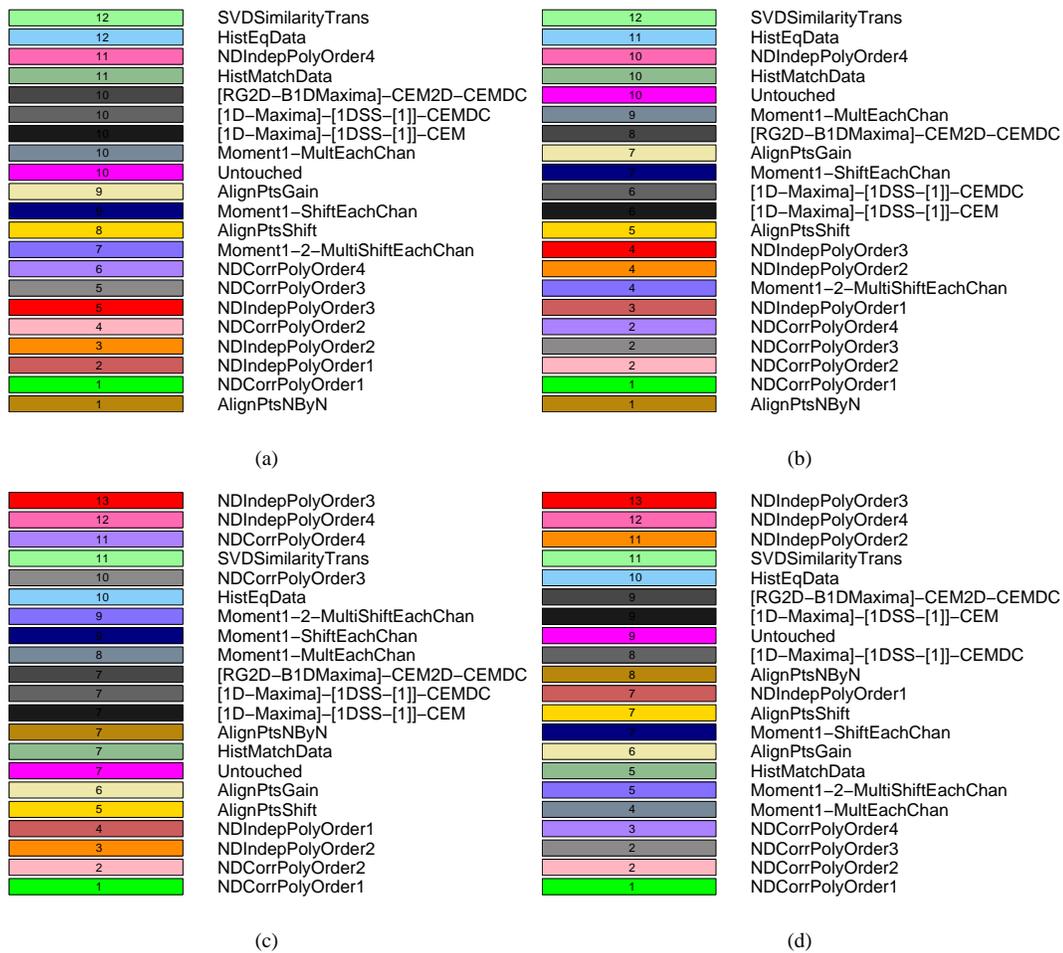


Figure 4.30: Rankings of FBHA and the competing methods evaluated in experiment 1 in section 4.4.1. Shows 0(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 4.30(a), 2) Skittles 4.30(b), 3) Teddy bears 4.30(c) and 4) three paper strips 4.30(d).

2. Existing histogram metrics are critiqued and a new metric for labeled data is introduced.

Quantitatively ranking the alignment performance of different algorithms requires a metric to score results. Different classes of metrics have been evaluated and the pros and cons of each metric have been explored. A new histogram comparison metric for labeled data is introduced, the average Mahalanobis distance. This metric discriminates between alignment improvements of overlapping and non-overlapping clusters in multi-modal histograms.

3. Colour inconsistency removal transforms are quantitatively ranked.

This work compares a large number of transforms that have been used in different colour inconsistency removal applications. The evaluation performed is independent of a particular application and so it informs the behavior of these transforms in a wide range of situations. Point alignment transforms of labeled ground truth data are shown to align histograms better than non-point alignment transforms, this validates the need for automated methods that can apply point alignment transforms to align histograms. One surprising finding that emerged from the results is the variability of transform

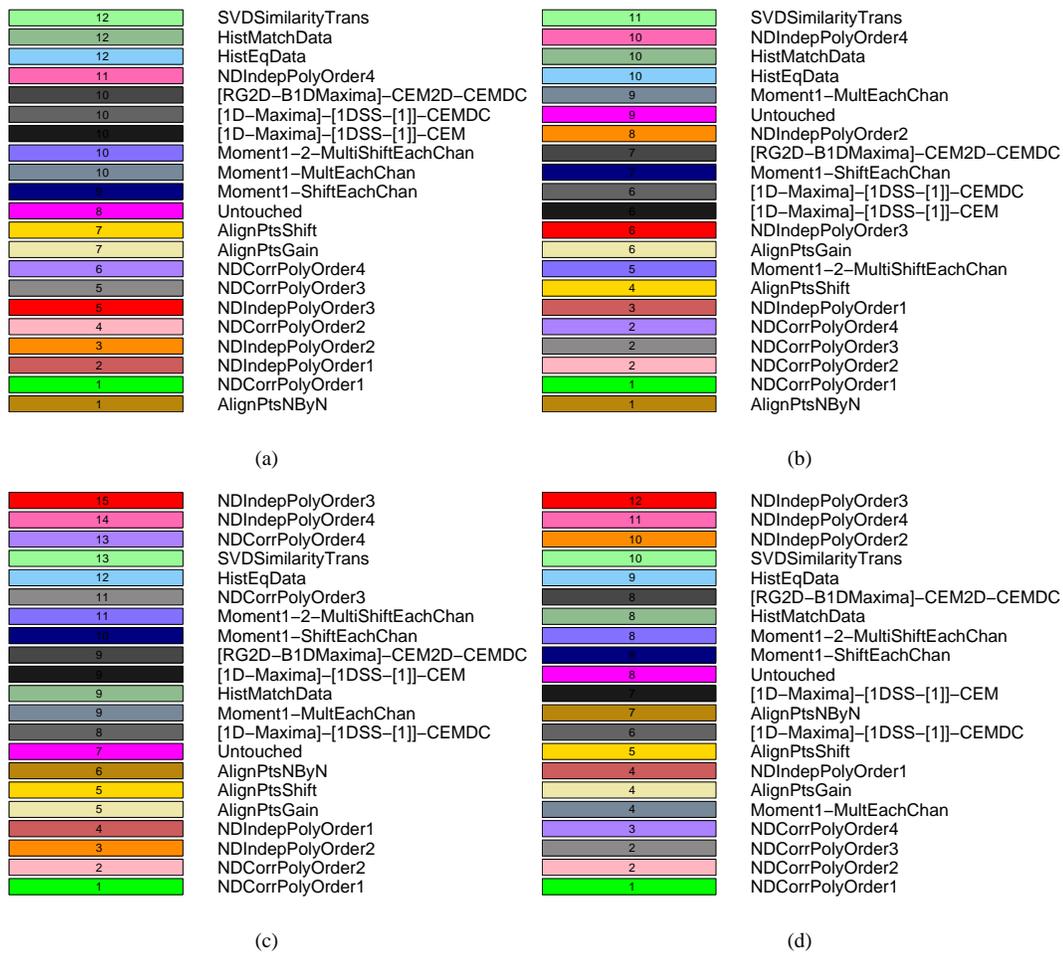


Figure 4.31: Rankings of FBHA and the competing methods evaluated in experiment 1 in section 4.4.1. Shows 0(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.31(a), 2) Skittles 4.31(b), 3) Teddy bears 4.31(c) and 4) three paper strips 4.31(d).

performance; this means that the best transformation to remove colour inconsistencies varies on a case by case basis even for similar colour inconsistent data-sets. Nevertheless, the bootstrap confidence tests show that a dominant ordering of the transforms emerges for the majority of cases.

4. **FBHA is quantitatively compared substitutable alternatives** FBHA is evaluated on the database. The experiments tell us that FBHA performs well when aligning histograms that contain corresponding clusters that have 1 significant peak; in this case FBHA uses linear point alignment transforms to align histograms so performance is comparable to point alignment transforms that use features from manually marked up regions.

This work has identified that colour inconsistencies can cause unpredictable variations in the local peak structure of clusters, in particular a change in the structure of a cluster across different conditions confounds the FBHA algorithm presented. This knowledge informs future work, and shows that it is not sufficient to align point based features to remove colour inconsistency. It is thought that future work should attempt to map detected features to clusters before matching the clusters

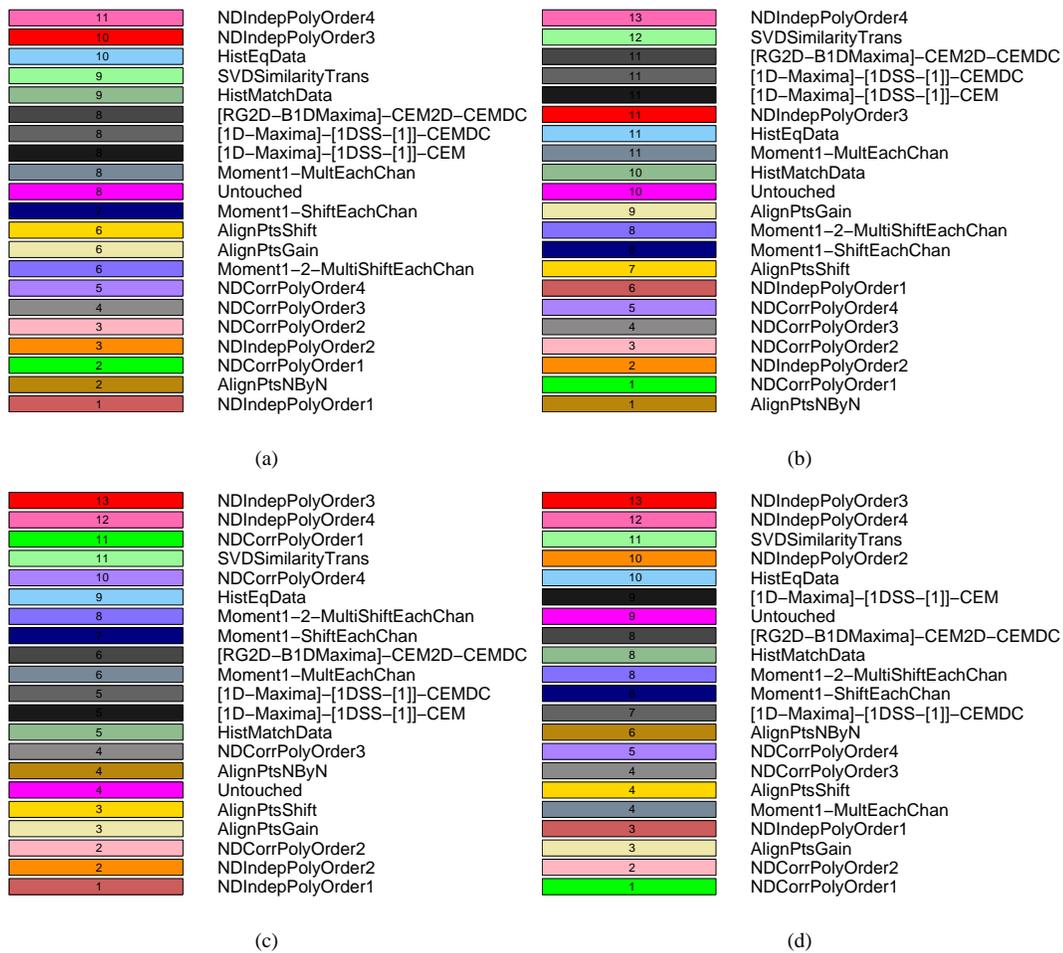
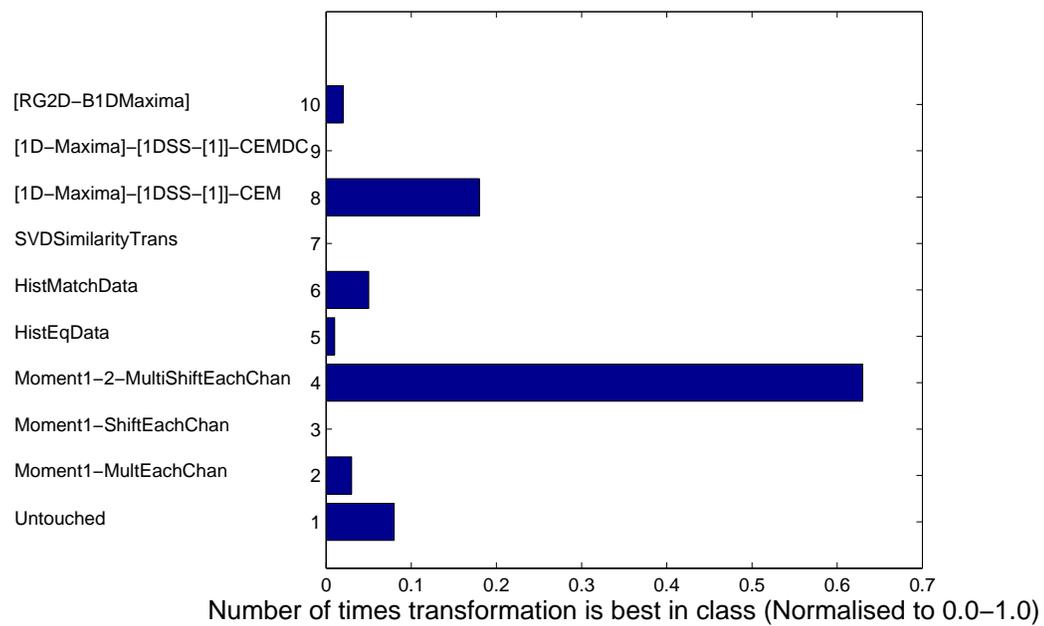
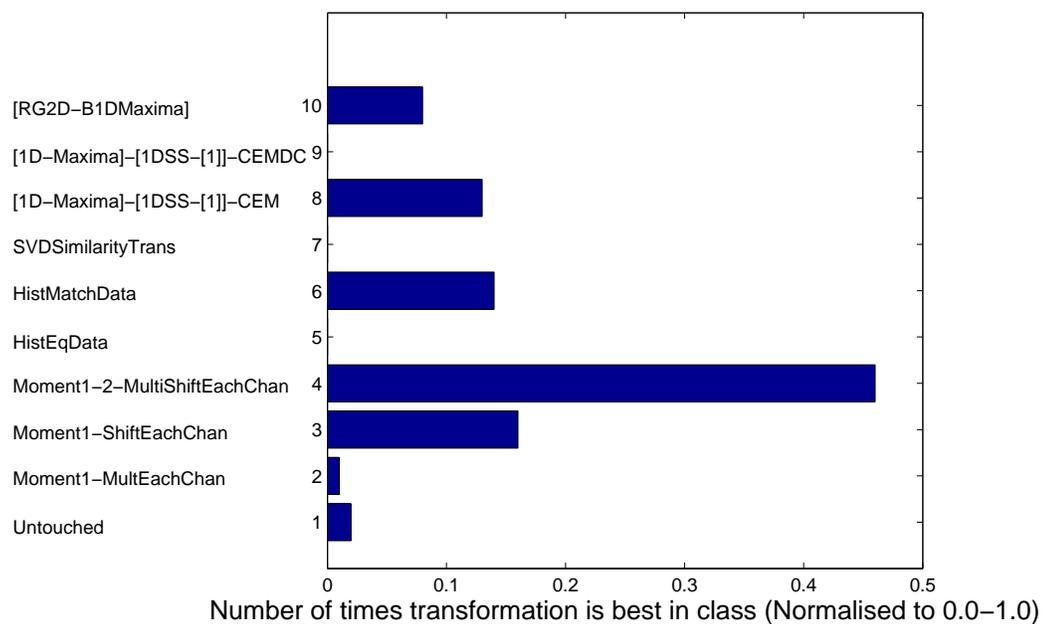


Figure 4.32: Rankings of FBHA and the competing methods evaluated in experiment 1 in section 4.4.1. Shows (C)(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.32(a), 2) Skittles 4.32(b), 3) Teddy bears 4.32(c) and 4) three paper strips 4.32(d).

between histograms, it is likely that topological reasoning about the histograms is necessary if further progress is to be made. Chapter 6 discusses some ideas for possible future exploration.

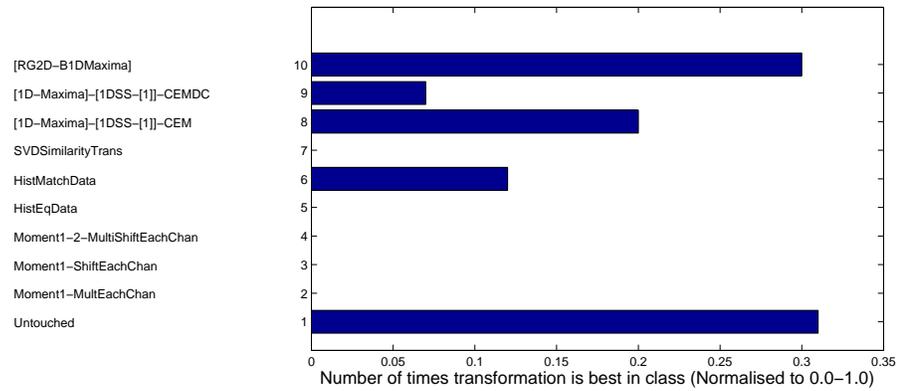


(a)

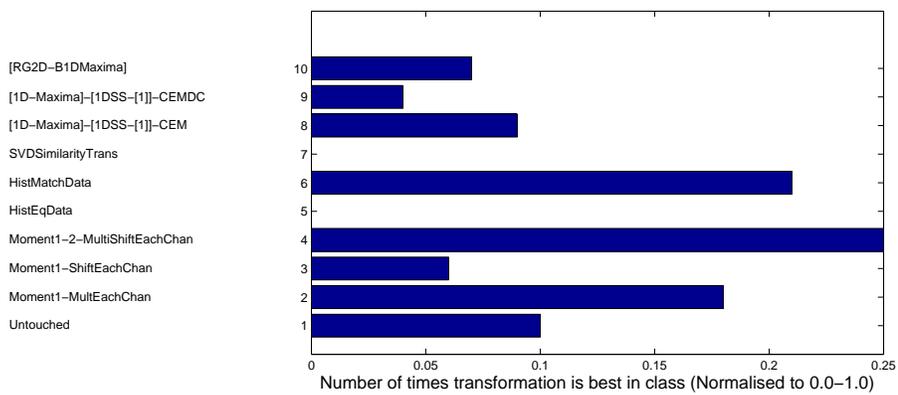


(b)

Figure 4.33: Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 4.33(a) and 2) Skittles 4.33(b).

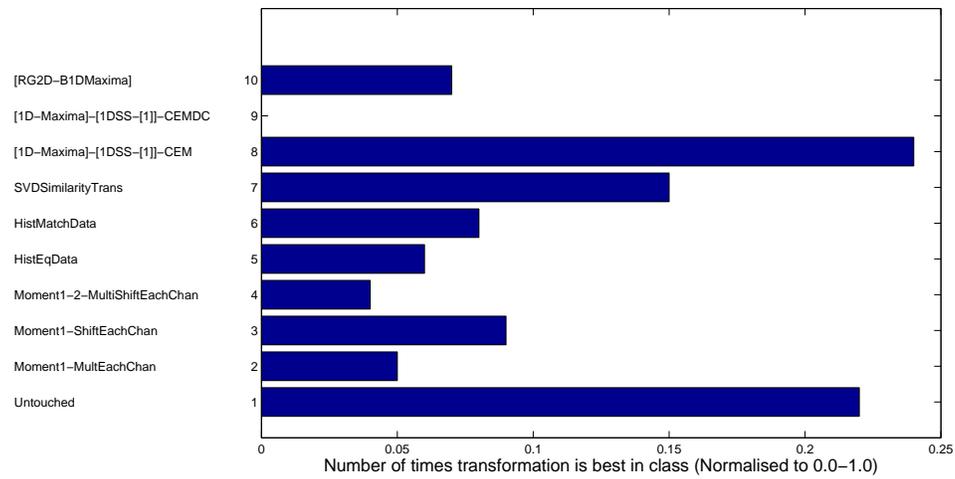


(a)

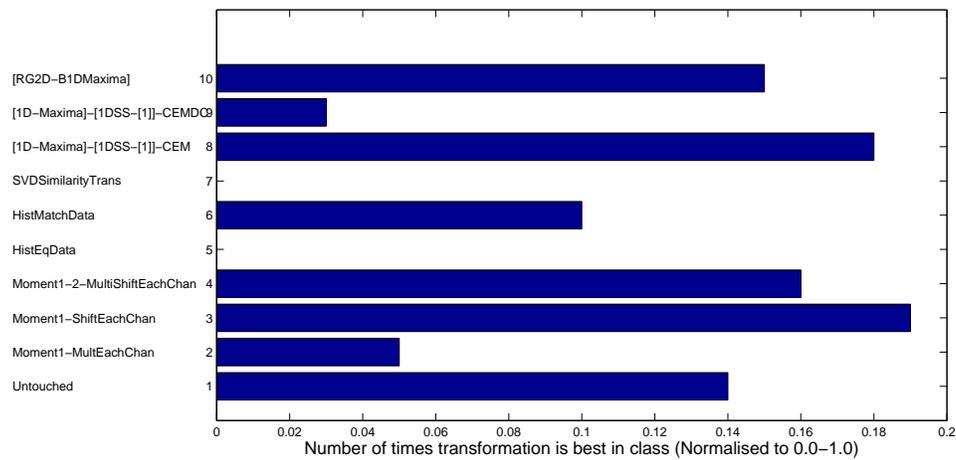


(b)

Figure 4.34: Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)0 variation for: 1) Teddy bears 4.34(a) and 2) three paper strips 4.34(b).

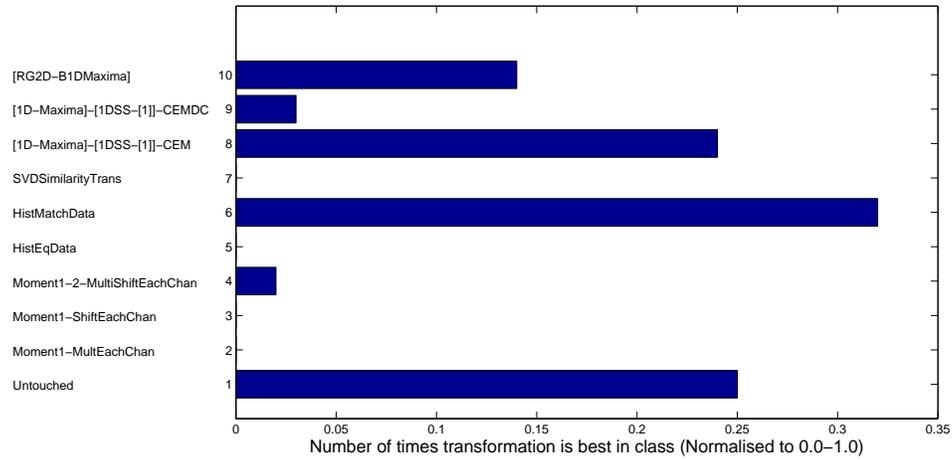


(a)

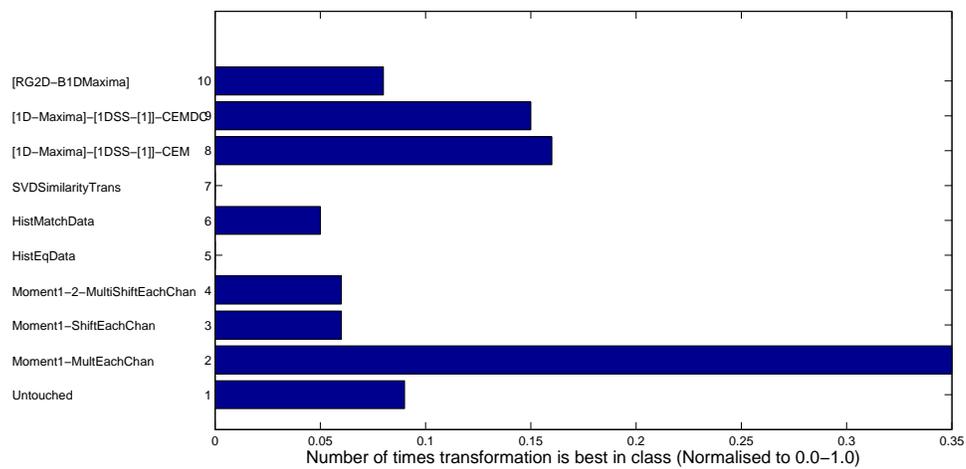


(b)

Figure 4.35: Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.35(a) and 2) Skittles 4.35(b).

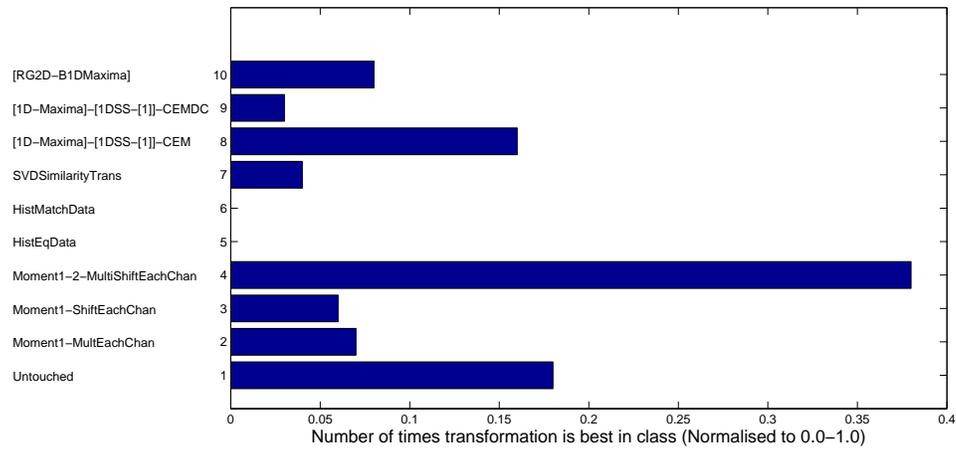


(a)

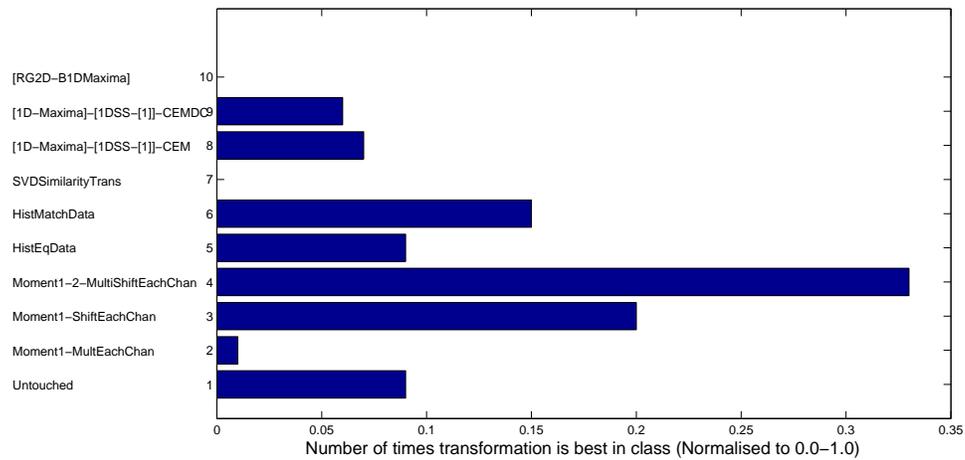


(b)

Figure 4.36: Normalised counts showing the number of times each transformation method performs best against the others with 0(L-LI)(L-AL)(S) variation for: 1) Teddy bears 4.36(a) and 2) three paper strips 4.36(b).

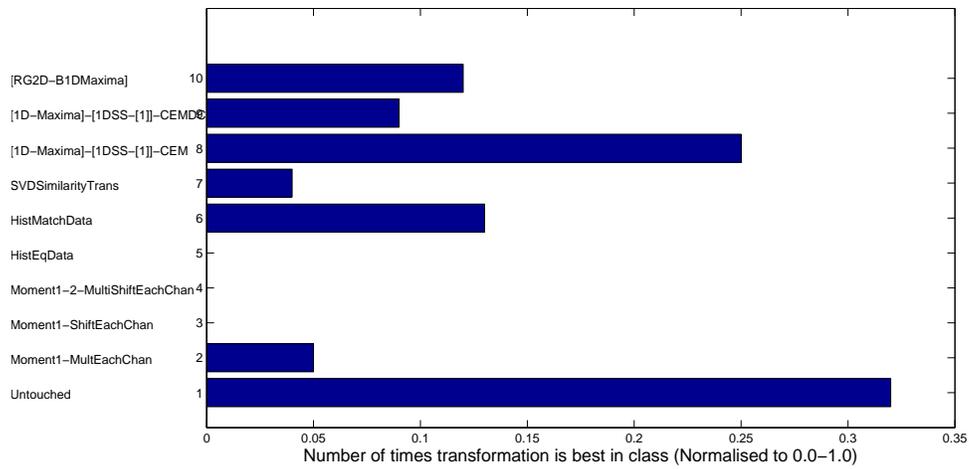


(a)

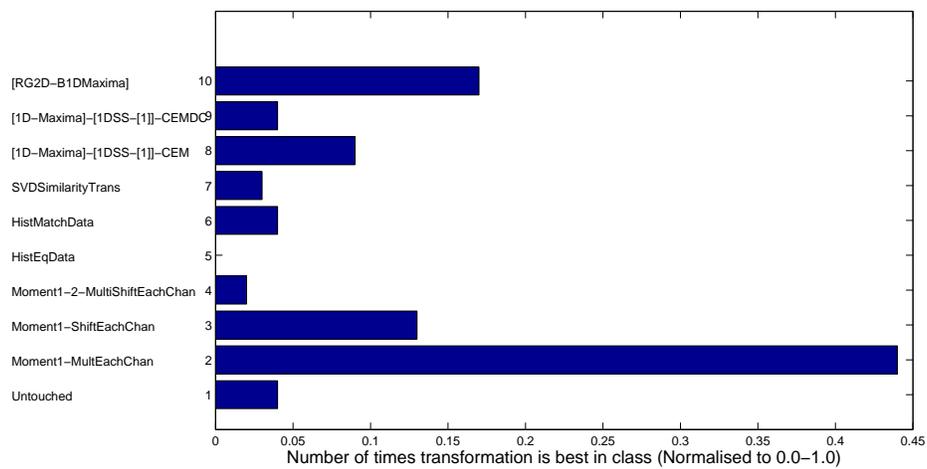


(b)

Figure 4.37: Normalised counts showing the number of times each transformation method performs best against the others with (C)(L-LI)(L-AL)(S) variation for: 1) Red-cyan paper 4.37(a) and 2) Skittles 4.37(b).



(a)



(b)

Figure 4.38: Normalised counts showing the number of times each transformation method performs best against the others with (C)(L-LI)(L-AL)(S) variation for: 1) Teddy bears 4.38(a) and 2) three paper strips 4.38(b).

Chapter 5

Application of feature based histogram alignment to Buhler Sortex machines

The previous chapter examined approaches to aligning pairs of colour histograms, this chapter investigates approaches to aligning sets of grey-level histograms obtained from video streams of food produce passing through a Buhler Sortex machine. The Z1 machine is introduced and the existence of colour inconsistency experienced by the machine is described as a histogram alignment problem. Two classes of approach to solving the histogram alignment are introduced and contrasted. The first approach involves segmenting the histograms and then applying piecewise transforms to the portions of the histogram. The second approach involves transforming the global properties of the histogram. The feature based histogram alignment method is introduced as a non-segmentation based method and is applied to Buhler Sortex data. All histogram alignment methods are quantitatively compared and the relative merits of these two approaches are discussed.

5.1 The Buhler Sortex Z-series

This section describes the operation of the Buhler Sortex Z-series machine, the grey-level histogram alignment problem and the current method for aligning the histograms. Aligning the histograms corrects unwanted appearance variation in the products observed by the machine. The Buhler Sortex machine constrains the imaging environment to label image data as background or product and defect. Once the background has been discarded the appearance of the product and defect is aligned across the camera view. Correcting the appearance in this way allows a single threshold to be set that separates the acceptable produce from the defect. The Z series machines are monochromatic optical sorting machines that come in different sizes. Figure 5.1 shows the single chute machine and figure 5.2 shows the Z+ three chute version. All machines operate by filling up the input hoppers with food produce to be inspected, a vibrator system then feeds the food product so that it falls down the chute in a uniform manner. The product falls past front and rear line scan cameras and a corresponding array of air ejectors. A computer vision system identifies defective product and fires the appropriate air ejector in order to channel the defective product to a reject receptacle. Figure 5.3 shows a schematic slice view diagram for a single chute in a Z-series sorting machine to illustrate the key operational points.



Figure 5.1: The single chute monochromatic Buhler Sortex Z1 sorting machine. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission.

5.1.1 Histogram alignment problem

Each camera used in the Buhler Sortex Z-Series is a monochromatic line scan camera that produces a one dimensional 1024 pixel wide image; the 1024 intensity values are then processed by calibration and sorting algorithms. Figure 5.4 shows a grey-scale image that represents 1024 continuous captures of rice falling past the 1D 1024 pixel CCD array. All such images in this section are portions of a video stream where the capture rate has been set to sample the object as accurately as possible as it falls past the CCD array aperture. Figure 5.5 shows a zoomed portion of the image in 5.4 that illustrates the recorded grey levels when imaging a few rice grains over a short period of time.

The appearance of the product varies with spatial position across the view; evidence of this variation can be observed by capturing approximately 20 seconds worth of data from a single camera and computing a histogram of grey level intensities for each pixel. Figure 5.6 shows the histograms $\mathbf{h}_1 \dots \mathbf{h}_{1024}$ of the intensities observed in the pixels $p_1 \dots p_{1024}$, the histograms show clear variation in intensity. Finding the correction transformations that align the histograms across the view allows the appearance of the product to be corrected across the view, this is called the *histogram alignment problem*. There is a scale variation in the relative amounts of product and background observed by pixels near the centre of the chute and pixels near the edges. Figure 5.9 shows a plot of the histograms from pixel 500 in green and from pixel 10 in blue. Notice the difference in size of the corresponding peaks, in addition to the displacement between the two histograms.



Figure 5.2: The three chute monochromatic Buhler Sortex Z+ sorting machine. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission.

For any given pixel, there is a significant difference between the amount of product, background and defect captured; this large difference means that the defect portions of the histogram are not visible in figure 5.6. Figure 5.7 highlights the defect portions of the histograms in red and plots the log histograms to show the variation among the different classes. Figure 5.8 provides a further sense of the variation across the within view histograms by plotting the log histograms as a three dimensional height field. These plots illustrate that the variation in intensity is reasonably small between pixels that are close together and more significant when comparing pixels a larger distance apart.

5.1.2 Current Approach and Commercial Confidence

The Buhler Sortex machine corrects appearance variation across the view. When produce falls down the chute it is inspected by a front and rear camera. The aim is to reject produce that has visible defects when inspected from either the front or rear. Separate thresholds are applied in each pixel in the front and rear of the chute, independent spatial processing in the front and rear is used to identify defect regions above a specified size and fire the air ejectors. Figure 5.10 summarizes these ideas.

The acceptable product is a significant feature of the histogram in each pixel, the defect product occurs much less frequently and can be difficult to discern from the histogram. Because of the significant scale difference between acceptable and defect product, the accept product is treated as the dominant feature of the histograms. Transforms are found that align the acceptable product portions of the histogram in each pixel, the resultant transforms are then used to map a single threshold value to an appropriate po-

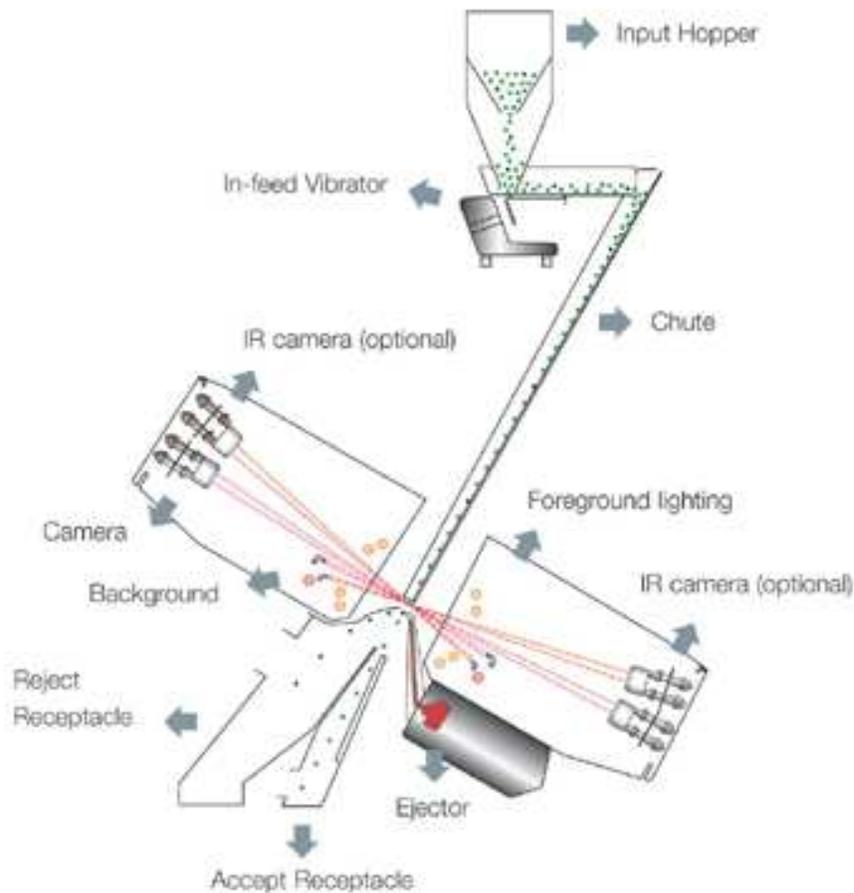


Figure 5.3: Side view slice of Z series machine chute. Camera systems inspect both the front and rear of the rice stream falling down the chute. The information from the front and rear views is use to reject food produce from the stream using air ejectors. Picture ©copyright Buhler Sortex Ltd, 2008. Reprinted with permission.

sition in each pixel. If different within-view alignment transforms are compared, a transform that gives lower acceptable product appearance variability across the view gives mapped thresholds that depend less on the product appearance variation.

Commercial confidence issues forbid direct disclosure and identification of the exact calibration methods used by the Buhler Sortex machines. The work in this chapter is based on a detailed investigation of existing product specifications, conversations with Buhler Sortex engineers and interactive investigation of machine behavior. Candidate methods for performing within-view calibration are introduced and quantitatively evaluated.

5.1.3 Motivation for improved calibration routines

It is believed that improved calibration procedures will lead to improved sorting performance. Buhler Sortex state, “There is a noticeable variation in performance across the width of the chute. The potential benefit from improving the calibration across the view has not yet been quantified. An improvement in

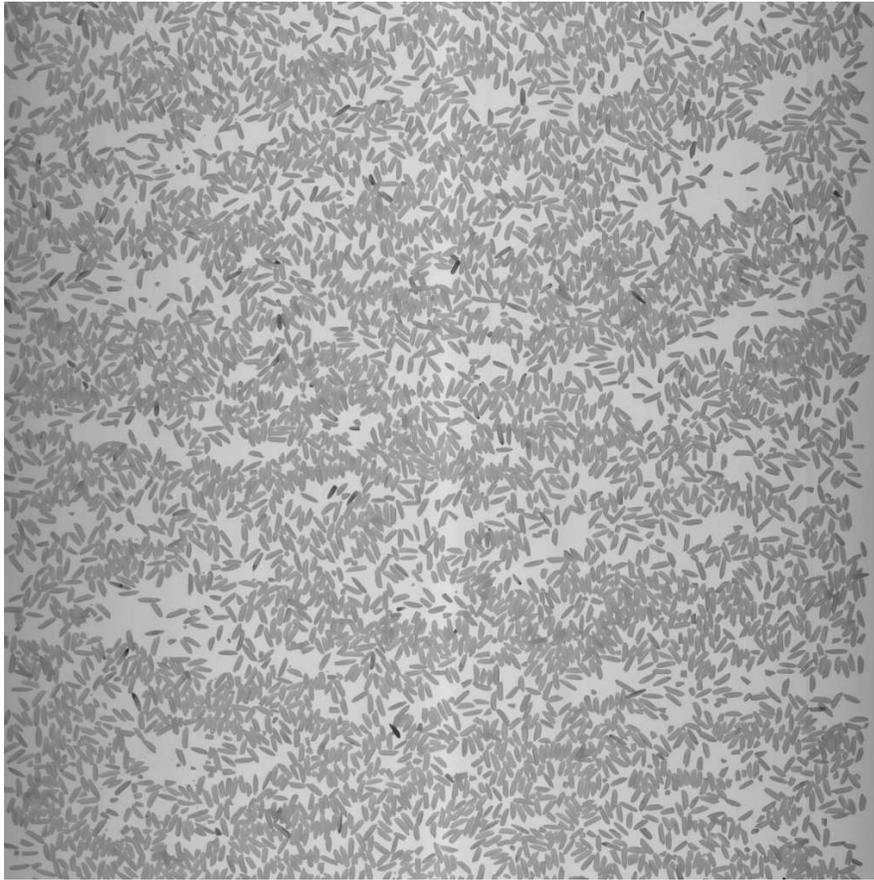


Figure 5.4: A 1024 by 1024 captured image of rice with approximately 3 percent defect. The i th column in the image represents 1024 sequential grey-level captures from the pixel in the CCD array at the i th column position. Picture ©copyright Bühler Sortex Ltd, 2008.

sorting performance of 0.5% or 1% would yield both economic and environmental benefits for world production of stable crops such as rice and wheat.” [84]

5.2 Product colour inconsistency reduction

This section introduces methods for colour inconsistency reduction of the product within a single camera view of the Z1 machine. When inspecting rice falling down the chute the camera sees three object classes, these are:

1. A white plate in the background,
2. the acceptable rice,
3. the defective rice and other contaminants.

During a calibration cycle the feed is stopped and the intensity of the white background plate is recorded. The angle of the plate is adjusted so that it is brighter than the intensity of the acceptable product. Rice defects are assumed to be darker than the acceptable rice. After calibration the three object classes are

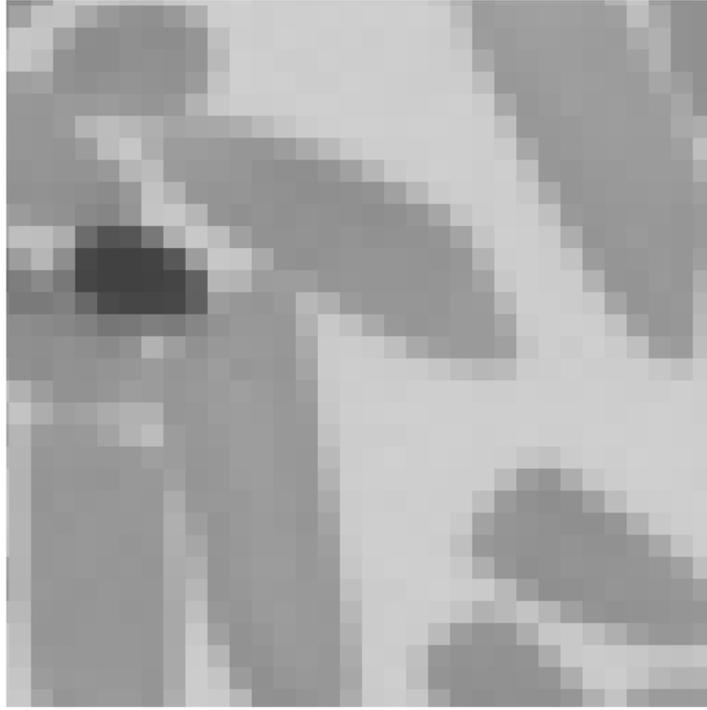


Figure 5.5: A portion of the capture stream that clearly shows the individual grey-levels that are recorded over a small spatial region and short time frame. Rows in the image are grey level values captured over time, columns indicate spatial position across the chute. Picture ©copyright Bühler Sortex Ltd, 2008.

ordered from dark to light on the intensity scale as : 1) defects+contaminants, 2) acceptable rice and 3) the white plate. When the feed is turned on, the histogram \mathbf{h}_i in the i th pixel of observed intensities is an additive combination of the histograms of the background \mathbf{b}_i , acceptable product \mathbf{p}_i and defect \mathbf{d}_i ; so,

$$\mathbf{h}_i = \mathbf{b}_i + \mathbf{p}_i + \mathbf{d}_i. \quad (5.1)$$

With the feed turned off, $\mathbf{h}_i = \mathbf{b}_i$. Two contrasting approaches to removing colour inconsistencies across the camera view are introduced in the next two sections. First, methods that align global properties of the histograms are introduced in section 5.2.1; these are termed *non-segmentation alignment methods*. Second, methods that align local properties of the acceptable product \mathbf{p}_i and defect portions of the histogram \mathbf{d}_i are introduced in section 5.2.2; these are termed *segmentation based alignment methods*. The next two sections 5.2.1 and 5.2.2 develop two approaches to histogram alignment within the view, the introduced methods have a number of potential sub-components. These options are introduced, along with short hand codes to refer to them.

5.2.1 Non-segmentation alignment methods

This section introduces two non-segmentation alignment methods. Methods for applying FBHA across the view are introduced along with methods to correct the global moments of the histograms. These

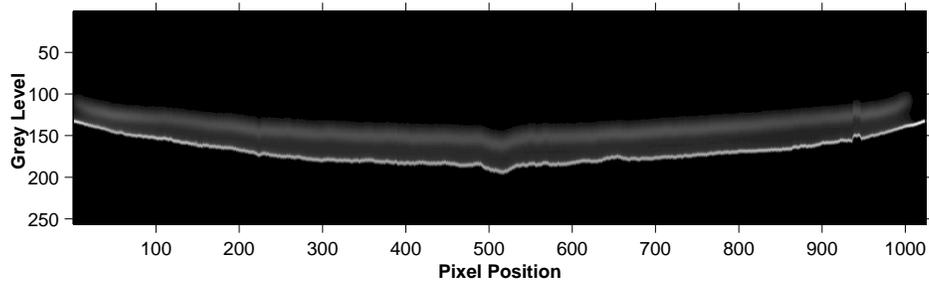


Figure 5.6: Histograms of product, defect and background for the 1024 pixels across the view obtained by computing histograms for the grey level values observed in each pixel. The vertical axis indexes the 256 different grey levels and the horizontal axis indexes pixel position. The frequency count is displayed as a grey-value, where high frequencies are rendered close to white and lower frequencies are rendered closer to black.

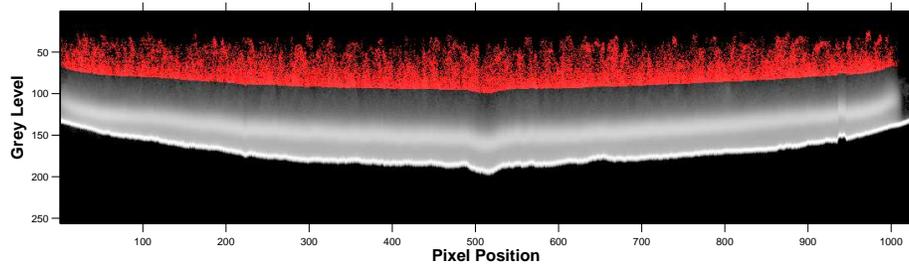


Figure 5.7: Log of the histograms in figure 5.6 with the defect portions of the histograms highlighted in red. This shows the distribution of the defect across the view despite the large scale variation between product, defect and background.

methods do not use a separate background intensity estimate.

FBHA within a view

Chapter 4 applied FBHA to align paired histograms. In the Buhler Sortex histogram alignment problem there is a histogram for each of the 1024 pixels in the camera view. This section describes a procedure to align histograms from a camera view, the main steps are 1) feature detection for each histogram \mathbf{h}_i , 2) association of all corresponding features and 3) aligning the features. The procedure ***WithinViewFBHA*** in algorithm 6 describes these steps as pseudo-code. ***WithinViewFBHA*** accepts a 255×1024 matrix H where the i th column contains the i th histogram, \mathbf{h}_i , from the within view data. Alternatives for the feature detection, feature association and alignment steps are described here in more detail.

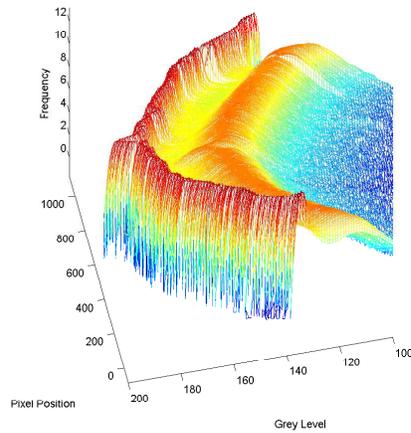


Figure 5.8: Three dimensional coloured height plot of the log histograms across the view displaying grey levels 100 to 200.

Algorithm 6 WithinViewFBHA(**H**)

// Detect and store the features in each pixel

for $i = 1$ to $NumPixelsInView$ **do**

$F(i).Features = FindPersistentMaxima(\mathbf{h}_i)$

end for

Matches = MatchFeaturesWithinView(F , MatchStrategy)

Compute alignment transform for each pixel that performs a feature based alignment to either a) centre pixel features (pixel 512) or b) the average matched feature values.

Feature detection

There are two dominant peaks in the within-view histogram: background and acceptable product. The amount of defect in a typical histogram is not large enough to find a discernable peak. The persistent scale space maxima are detected in each pixel. Figure 5.11 shows a grey-level histogram obtained from a pixel within the view and smoothed histograms using medium and large scales. Figure 5.12 shows the scale space of the same histogram. Figure 5.12 shows the local maxima detected at each scale, we see that two paths persist over the scale space - these correspond to the background and acceptable produce peaks. Notice that a large number of irrelevant local maxima can be eliminated by thresholding the scale space paths.

Feature matching

Three options for associating the detected features across the view are proposed. The pseudo-code function call *MatchFeaturesWithinView(F,MatchStrategy)* in algorithm 6 accepts the detected features in the parameter F and the parameter *MatchStrategy* selects one of the following methods for matching

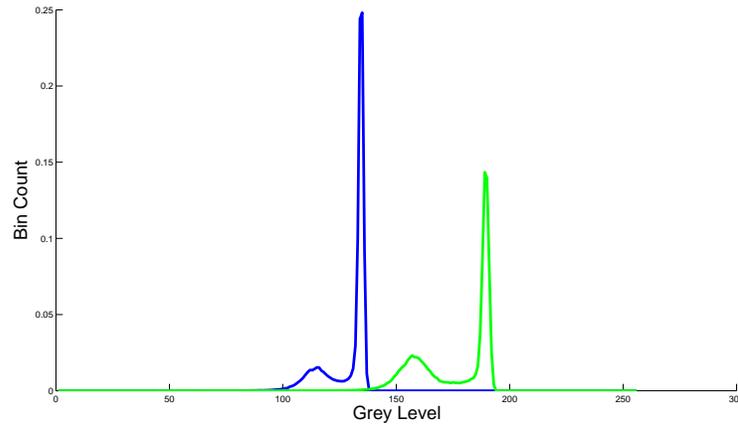


Figure 5.9: Plots of histograms from pixel 500 near the centre of the chute in green and from pixel 10 near the left edge of the chute in blue.

the detected features:

1. *AssociateToTargetPixel(Targetpixel)* maps features from all pixels to the features from a target pixel, *Targetpixel*. Figure 5.14 illustrates this method by showing a sample of associated features across the view; there are two types of feature, associated features of the same type are shown with a circle or a square. The features for the target pixel are drawn in red and the features for other pixels are drawn in blue. The features for the i th column are matched to the target features by finding the match that has the minimum total Euclidean cost between the features from the i th column and the *Targetpixel* column.
2. *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* is described in algorithm 7. The procedure associates features in the centre region of the view together by working outwards from a centre pixel, *Targetpixel* to two target pixels, *EdgeT* away from pixels 1 and 1024 on either side of the view. The remaining unmatched features on both sides are matched to the target edge pixels. Figure 5.15 highlights the stages of the algorithm graphically by colour coding the matched features across the view according to the stage of the algorithm when the features are matched. The procedure avoids directly matching pixels that are far apart and places less confidence on the features obtained near the edges of the chute.
3. The procedure *ThreeStageAssocAndFixup(F,Targetpixel,EdgeT)* associates features with *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* then scans across the view for missing features. Gaps are filled with linear interpolation between the detected features. Extrapolation is used at the edges if no product features are detected. Figure 5.16 shows an example of the linear interpolation step when features are missing from a central portion of the view.

Section 5.3.2 shows results that highlight the performance of these different approaches.

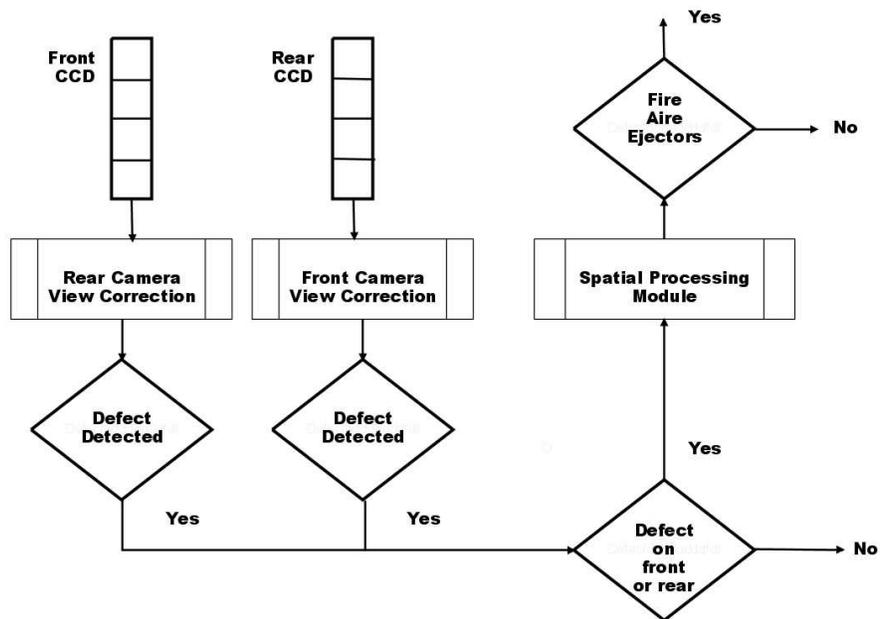


Figure 5.10: The logical flow of information from the front and rear cameras within a chute. Defect thresholding is performed on the front and rear views independently. The defect information is aggregated by a spatial processing module, the decision to fire the air ejector is base on the size of the detected defect and the machine settings.

Feature alignment

Point alignment transforms are used to align the associated features; for a recap, see section 2.5.2. A point alignment transformation is found for all pixels in the view, the transform in a pixel moves the associated features to new target positions. The associated source features from a pixel are represented by a 2×1 vector, s , where each entry indexes the position of the feature in the corresponding histogram. There are three choices for selecting the 2×1 target vector t ,

1. the associated features from pixel 512 are used. (This option is referred to as [**Target-512**])
2. the average background and product values. All associated features are represented as a 2×1024 matrix, M , where the 1st row represents the associated product intensities and the 2nd row represents the associated background intensities. Given this, the rows of t are computed as the average of the corresponding rows of M . (This option is referred to as [**Target-MeanCluster**])
3. the maximum background and product values. The rows of t are computed as the maximum value of the corresponding rows of M .

For each pixel, a point alignment transform is found. The transforms evaluated are:

1. Additive using equation 2.17.
2. Multiplicative using equation 2.18.

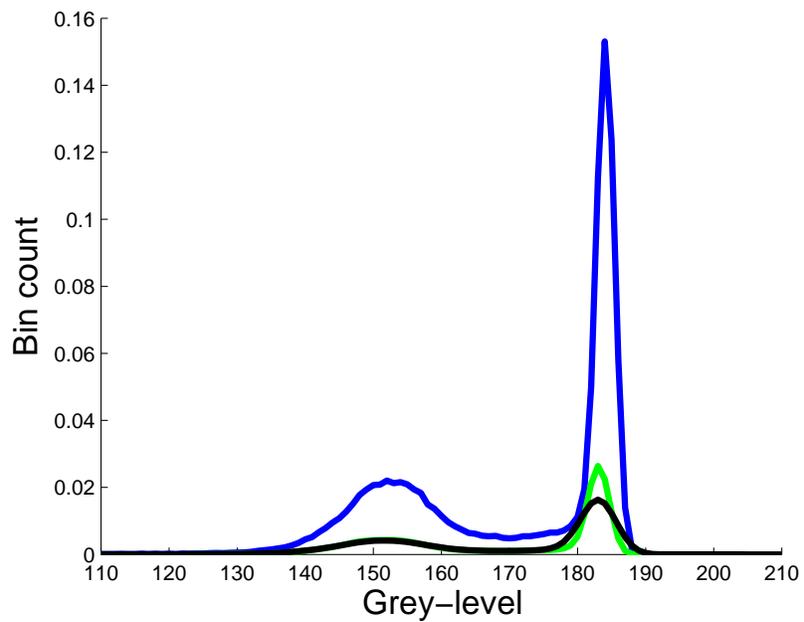


Figure 5.11: Histogram of grey-level intensities from a single pixel in blue and its representation at medium and high levels of blurring (plotted in green and black respectively).

3. Linear using equation 2.19, $d = 1$.
4. Quadratic using equation 2.19, $d = 2$.
5. Cubic using equation 2.19, $d = 3$.

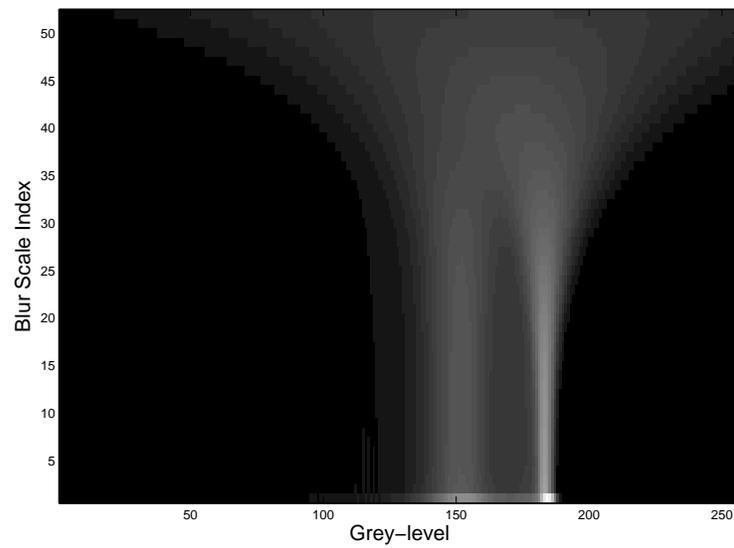


Figure 5.12: Grey level representation of the scale space of a grey-level histogram. Each row in the image represents the blurred values of the histogram at different scales, higher values are rendered closer to white and lower values are rendered closer to black. The blur scale index indexes the lowest scale at the bottom row to the highest scale at the top row.

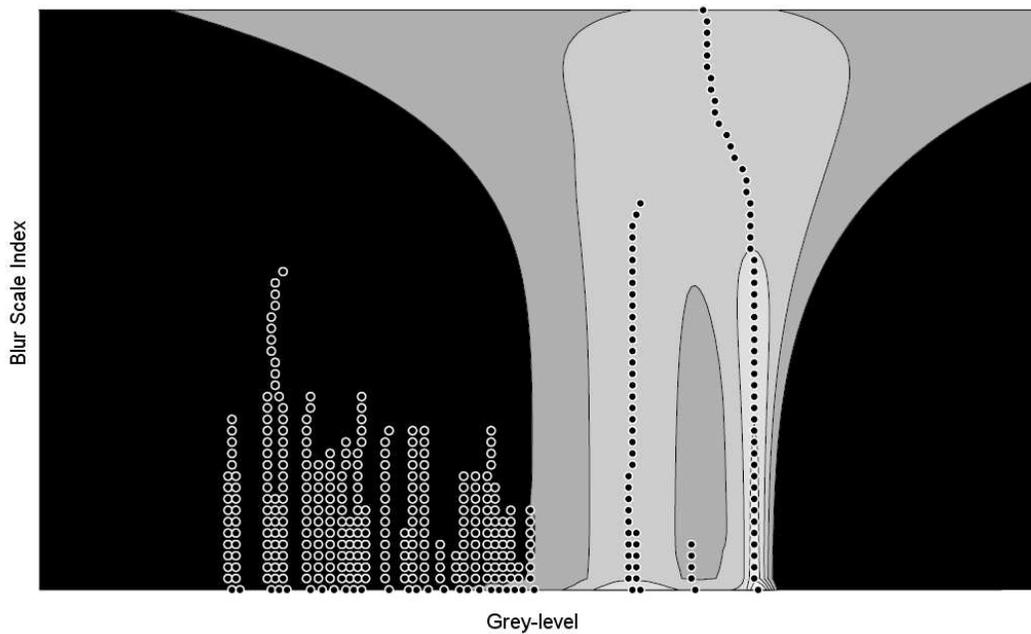


Figure 5.13: Contour plot of the scale space shown in figure 5.12. Local maxima at each scale are displayed using a circle.

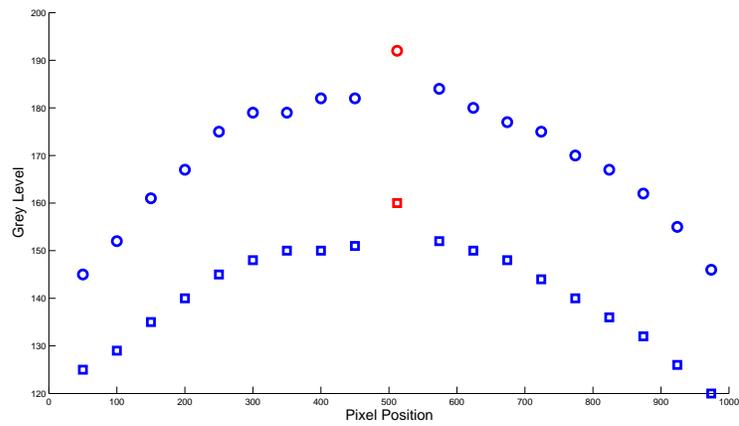


Figure 5.14: Associated features across the view using the *AssociateToTargetPixel(Targetpixel)* method. Features indicated with a circle are associated together and features indicated with a square are associated together. Features from the target pixel are drawn in red.

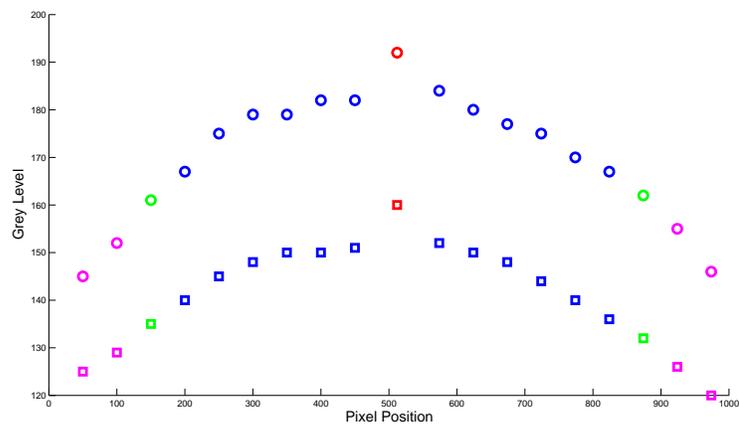


Figure 5.15: Associated features across the view using the *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* method described in algorithm 7. Features indicated with a circle are associated together and features indicated with a square are associated together. The initial seed features are drawn in red, the algorithm associates the features drawn in blue to the target features in two passes. First the features on the left are associated by matching the features in a column to the matched features in the adjacent column, this process is repeated for the features on the right of the initial seed features. The features drawn in green are *EdgeT* pixels away from the side of the chute; the green features on the left side of the chute are target features for the features on the left side of the chute marked in pink, the green features on the right side of the chute perform the same purpose for the matched features on the right side of the chute.

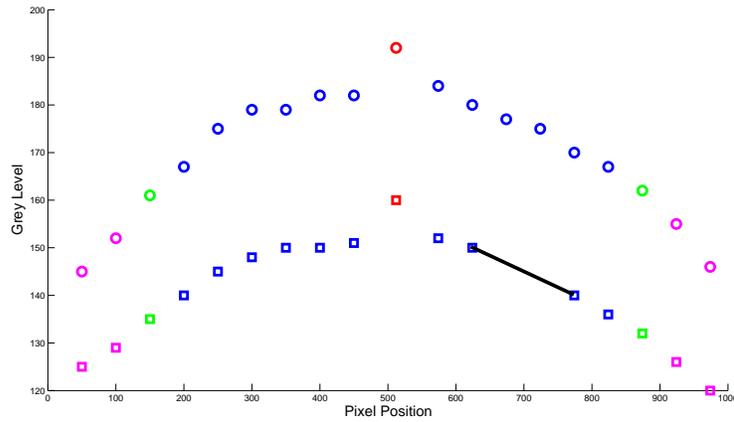


Figure 5.16: Associated features across the view using *ThreeStageAssocAnd-Fixup(F,Targetpixel,EdgeT)*. The method uses *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* shown in Figure 5.15 as a first step, then the algorithm scans for gaps in the features across the view. The black line shown indicates the detection of a gap and the interpolated line between the detected features.

Alignment of global moments

This section enumerates possible transforms for aligning the moments of the histograms in each pixel. No background removal segmentation is performed, i.e. $\mathbf{h}_i = \mathbf{b}_i + \mathbf{p}_i + \mathbf{d}_i$. The different transforms are:

Global additive transform A global additive(shift) transformation in each pixel maps all grey-level intensities q to $q + \omega_i$. A shift in each pixel is computed to align the mean value in each pixel to a target. The mean in each pixel is computed as,

$$f_i = E(\mathbf{b}_i + \mathbf{p}_i + \mathbf{d}_i). \quad (5.2)$$

The additive transform is computed as,

$$\omega_i = \mu_t - f_i \quad (5.3)$$

Three choices for the target μ_t are investigated,

- **Code: GlobalShiftToMean.** μ_t is set to the mean of all histogram means.
- **Code: GlobalShiftToMax.** μ_t is set to the maximum of all histogram means.
- **Code: GlobalShiftToTarget.** μ_t is set to the mean of the t th pixel. The pixel is chosen manually.

Global multiplicative transform A global multiplicative transform correction maps all grey-level intensities q to gq . A separate multiplier g_i is computed for each pixel p_i to align the distribution means. The mean in each pixel is

$$f_i = E(\mathbf{b}_i + \mathbf{p}_i + \mathbf{d}_i). \quad (5.4)$$

Algorithm 7 ThreeStageAssociateFeatures(**F**, **Targetpixel**, **EdgeT**)

// Set targets for the left and right sides.

$T_l \leftarrow EdgeT$

$T_r \leftarrow 1024 - EdgeT$

$AllMatches(Targetpixel).Features \leftarrow F(Targetpixel).Features$

$LastTargetMatch \leftarrow AllMatches(Targetpixel).Features$

// Stage 1: Sweep outward from Target pixel, associating the features in each pixel to its neighbouring pixel.

$i = Targetpixel - 1$

while $i \geq T_l$ **do**

$AllMatches(i).Features \leftarrow MatchFeatures(F(i).Features, LastTargetMatch).$

$LastTargetMatch \leftarrow AllMatches(i).Features$

$i \leftarrow i - 1$

end while

$i = TargetPixel + 1$

$LastTargetMatch \leftarrow AllMatches(Targetpixel).Features$

while $i \leq T_r$ **do**

$AllMatches(i).Features \leftarrow MatchFeatures(F(i).Features, LastTargetMatch).$

$LastTargetMatch \leftarrow AllMatches(i).Features$

$i \leftarrow i + 1$

end while

$TargetMaxima = AllMatches(T_l).Features$

for $i = 1$ to $T_l - 1$ **do**

$AllMatches(i).Features \leftarrow MatchFeatures(F(i).Features, LastTargetMatch).$

end for

for $i = T_r + 1$ to 1024 **do**

$AllMatches(i).Features \leftarrow MatchFeatures(F(i).Features, LastTargetMatch).$

end for

This is used to compute the multiplicative transform

$$g_i = \frac{\mu_t}{f_i} \quad (5.5)$$

The different ways of computing a multiplicative correction in each pixel are:

- **GlobalGainToMean:** μ_t is set to the mean of all histogram means.
- **GlobalGainToMax:** μ_t is set to the maximum of all histogram means.
- **GlobalGainToTarget:** μ_t is set to the mean of the t th pixel. The pixel is chosen manually.

Global linear transform A linear transformation in each pixel maps all grey-level intensities q to $\lambda_i q + \omega_i$. The multiplicative component aligns the standard deviation of the i th pixel with the target standard deviation, we write this as

$$\lambda_i = \frac{\sigma_t}{\sigma_i}. \quad (5.6)$$

The additive component is

$$\omega_i = \mu_t - \lambda_i f_i. \quad (5.7)$$

The standard deviation in a pixel is computed using all product, background and defect intensities obtained and $f_i = E(\mathbf{b}_i + \mathbf{p}_i + \mathbf{d}_i)$.

The three transform-target combinations are:

- **Code: MeanVarToMean.** The target mean, μ_t , is defined as $\mu_t = \frac{\sum_{i=1}^{1024} f_i}{1024}$. The target standard deviation, σ_t , is defined as $\sigma_t = \frac{\sum_{i=1}^{1024} \sigma_i}{1024}$.
- **Code: MeanVarToMax.** The maximum mean and standard deviation are used as target values μ_t and σ_t .
- **Code: MeanVarToTarget:** The mean and standard deviation of the i th pixel are the target values, μ_t and σ_t . The pixel is chosen manually.

5.2.2 Segmentation based alignment methods

This section introduces methods to compute a piecewise alignment of the product portion of the histograms. There are three elements to the segmentation based alignment approach. First, a background threshold is used to remove the background portions of the histograms. Second, an erosion step is applied to discard the intensities corresponding to edge pixels. Third, the remaining portions of the histograms are aligned. Background segmentation methods and transformation methods are described; different options are identified for each step so that the performance of different combinations of these options can be evaluated.

Methods for Background removal

Two types of background removal method are introduced. First, segmentation methods based on average intensity statistics are introduced and second, thresholds based on associated persistent maxima are introduced.

Average intensity thresholds

With the feed turned off, it is possible to observe the intensity of the background plate on its own and compute the average intensity in each pixel. The mean grey-level value in the i th pixel b_i of the background histogram \mathbf{b}_i is

$$b_i = E(\mathbf{b}_i). \quad (5.8)$$

When the feed is turned on, the average intensity of the background, acceptable produce and defect is computed in each pixel as,

$$f_i = E(\mathbf{b}_i + \mathbf{p}_i + \mathbf{d}_i). \quad (5.9)$$

A simple method of thresholding the background is to compute a fraction of f_i in each pixel. This threshold is computed as,

$$r_i = P f_i, \quad (5.10)$$

where P is a fraction between 0 and 1. The value of P is set manually. This method is referred to as **PercMean**.

A second method is to compute a threshold t_i in each pixel that is an offset from the background mean by a fixed proportion d of the distance between the background mean and r_i . This is,

$$t_i = b_i + d(r_i - b_i). \quad (5.11)$$

This method is referred to as **DiffOffset**. Pixels with grey-levels less than or equal to r_i in the case of **PercMean** thresholding and t_i in the case of **DiffOffset** thresholding are classified as product or defect.

A feature of the within view data is that less product is observed by the edge pixels because produce bounces off the sides of the chute. The implication of this is that average intensity thresholds can misclassify background as product near the edge of the chute. Figure 5.17 shows how the **DiffOffset** threshold, t_i , approaches the background mean level, b_i , on the right hand side of the chute. In this example the effect is less pronounced on the left hand side of the chute. A further processing step termed **ExtrapEdges** seeks to replace average intensity thresholds at the edges of the chute by using a simple linear model to perform extrapolation. The procedure *ExtrapEdgeThresholds* described in algorithm 8 accepts the existing thresholds, t_i or r_i depending on the method used. Outlier thresholds are discarded from both edges of the chute, separate lines are fit to thresholds on either side of the chute using a fixed window size. The fitted lines are then extrapolated on each side to generate the replacement thresholds. Figure 5.18 shows the modified thresholds for the *ExtrapEdgeThresholds* procedure.

Persistent maxima offset thresholds The persistent feature detection and matching step allows a background segmentation threshold to be computed without a separate background estimate. This is performed by

Algorithm 8 *ExtrapEdgeThresholds(T,EdgeT,Fitsize)*

 $LeftLimit \leftarrow EdgeT$
 $RightLimit \leftarrow 1024 - EdgeT$
 $LeftLine \leftarrow \text{Fit line to pixels } (LeftLimit + 1)..(LeftLimit + Fitsize)$
 $RightLine \leftarrow \text{Fit line to pixels } (RightLimit - Fitsize)..(RightLimit - 1)$

 Extrapolate $LeftLine$ to pixel 1, replacing all extrapolated pixels.

 Extrapolate $RightLine$ to pixel 1024 replacing all extrapolated pixels.

1. Finding the persistent deep structure features in each pixel.
2. Associating features using $ThreeStageAssocAndFixup(\mathbf{F}, \text{Targetpixel}, \text{EdgeT})$ with gap filing
3. In a pixel, we compute a background segmentation threshold t_i , as:

$$t_i = b_i - P(b_i - p_i), \quad (5.12)$$

where b_i is the detected background feature, p_i is the product feature and P is a fraction that can be set from 0 to 1. In this work, $P = 0.5$.

This method is referred to as **DStructMidPoint**. Figure 5.19 shows plots of the background features, the product features and the background segmentation thresholds.

Erosion step

Pixels at the edge of the rice grain give inaccurate grey level values for the product due to pixels partially sampling the product and background. These outlying grey-level values are removed from the product brightness distribution using an erosion image processing operation. The edge pixels are rejected by first producing a binary thresholded image of acceptable and defective product, the thresholds are computed using an average intensity thresholding method; one of **PercMean**, **DiffOffset** or **ExtrapEdges** is chosen. Next, an erosion image processing operator is run to identify edge pixels. These edge pixels are discarded as they do not represent the intensity of the product well.

Local Transforms

This section introduces transforms to perform alignment of the product portion of the histograms. The product histograms are the result of applying a background removal segmentation to the within view histograms and then applying an optional erosion step. For each transform the different target alignment values are enumerated.

Multiplicative correction transform

The appearance of the product and defect distributions are corrected across the view by aligning the means of the combined product and defect distributions. A multiplicative transform correction maps product grey-level intensities q to gq . A separate multiplier g_i is computed for each pixel p_i to align the means of the segmented product and defect distributions. The mean in each pixel is

$$\mu_i = E(\mathbf{p}_i + \mathbf{d}_i). \quad (5.13)$$

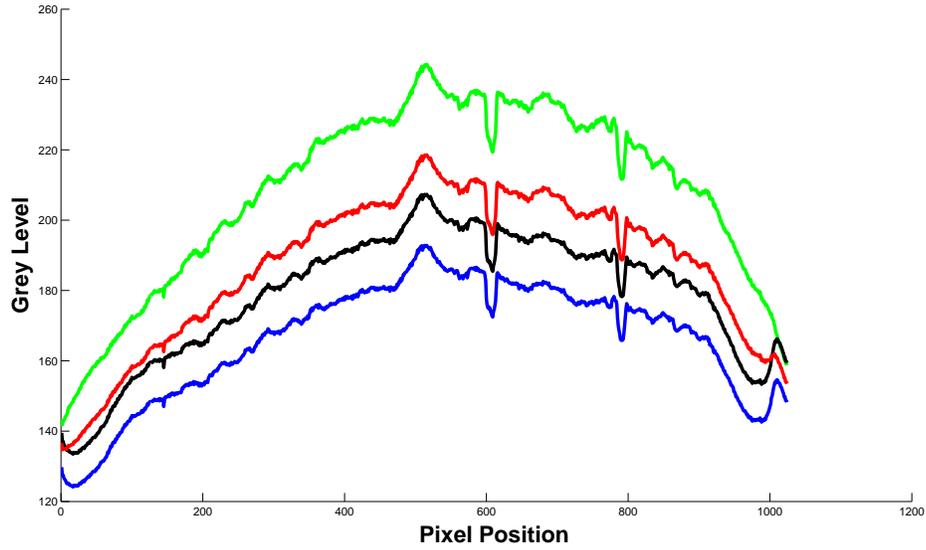


Figure 5.17: Thresholds computed within a single chute using average intensity statistics method. The green plot shows the mean background values, b_i (eqn: 5.8), computed by turning the feed off to inspect the background. The black plot is the mean value f_i in each pixel (eqn: 5.9) with the feed turned on. The blue plot is the threshold r_i , in each pixel computed with **PercMean**. The red plot is the threshold t_i in each pixel (eqn: 5.11), computed with **DiffOffset**.

This is used to compute the multiplicative transform

$$g_i = \frac{\mu_t}{\mu_i}. \quad (5.14)$$

The different ways of computing a multiplicative correction in each pixel are:

- **Code:GainToMean.** The target, μ_t , is the mean of all means. i.e. $\mu_t = \frac{\sum_{i=1}^{1024} \mu_i}{1024}$.
- **Code:GainToMax.** The maximum mean computed in each pixel is used as the target. i.e. $\mu_t = \max(\mu_i), \forall i$
- **Code:GainToTarget.** The mean of the i th pixel is the target value, μ_t . The pixel is chosen manually.

Additive correction transform

An additive(shift) transformation in each pixel maps product grey-level intensities q to $q + \omega_i$. A shift in each pixel is computed to align the mean value in each pixel to a target. The mean in each pixel is

$$\mu_i = E(\mathbf{p}_i + \mathbf{d}_i). \quad (5.15)$$

The additive transform is

$$\omega_i = \mu_t - \mu_i. \quad (5.16)$$

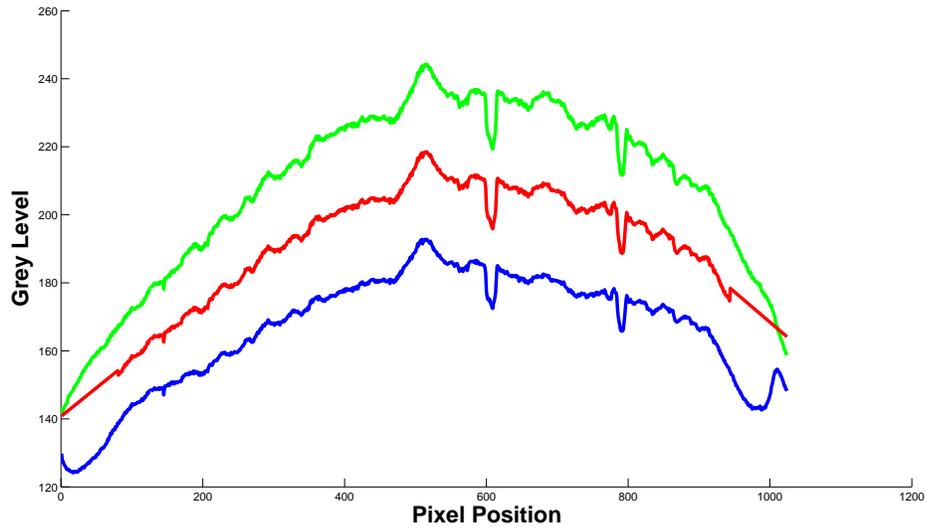


Figure 5.18: The red plot shows modified thresholds in each pixel using the `ExtrapEdgeThresholds` procedure described in Algorithm 8. The green plot shows the mean background values, b_i (eqn: 5.8), computed by turning the feed off to inspect the background. The blue plot is the threshold r_i , in each pixel computed with **PercMean**. Note that the extrapolated red `DiffOffset` lines cross the green plot on the right hand side. This is undesirable behaviour.

The target μ_t is set to the mean of all mean values (**Code: ShiftToMean**), the maximum of all mean values (**Code: ShiftToMean**) or the mean value from a manually chosen target pixel (**Code: ShiftToTarget**).

Linear correction transform

A linear transformation in each pixel maps product grey-level intensities q to $\lambda_i q + \omega_i$. The multiplicative component aligns the standard deviation of the i th pixel with the target standard deviation, we write this as

$$\lambda_i = \frac{\sigma_t}{\sigma_i}. \quad (5.17)$$

The additive component is

$$\omega_i = \mu_t - \lambda_i \mu_i. \quad (5.18)$$

The three transform-target combinations are:

- **Code: MeanVarToMean.** The target mean, μ_t , is defined as $\mu_t = \frac{\sum_{i=1}^{1024} \mu_i}{1024}$. The target standard deviation, σ_t , is defined as $\sigma_t = \frac{\sum_{i=1}^{1024} \sigma_i}{1024}$.
- **Code: MeanVarToMax.** The maximum mean and standard deviation are used as target values μ_t and σ_t . $\mu_t = \max(\mu_i), \forall i$ and $\sigma_t = \max(\sigma_i), \forall i$.
- **Code: MeanVarToTarget:** The mean and standard deviation of the i th pixel are the target values, μ_t and σ_t . The pixel is chosen manually.

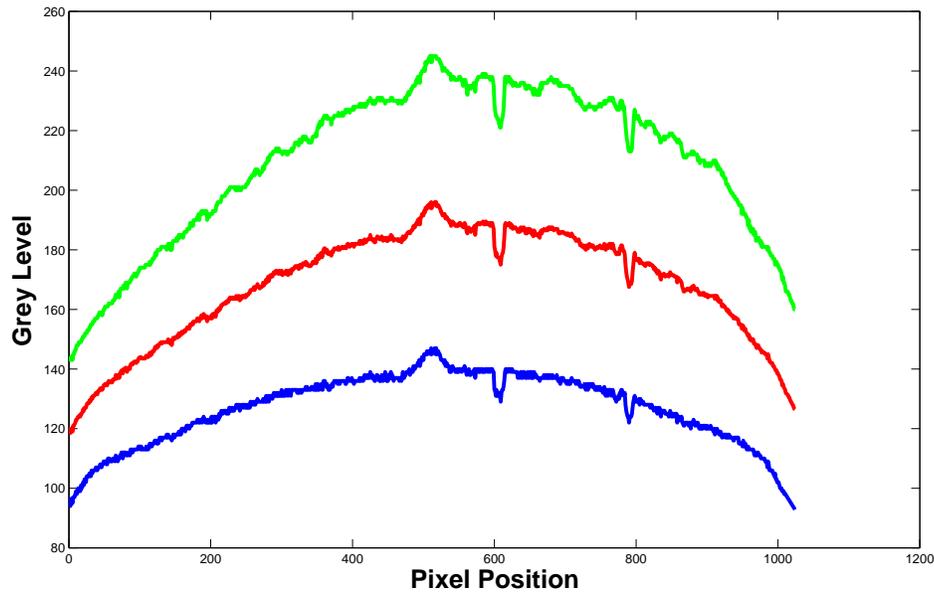


Figure 5.19: Persistent deep structure features and background segmentation thresholds computed using the DStructMidPoint method. Background features are plotted in green, the blue plot shows the product features. The red plot shows the background segmentation thresholds computed in equation 5.12

5.3 Experimental Evaluation

This section experimentally compares the alternative histogram alignment methods. First, we qualitatively investigate the behavior of the persistent maxima detection and association procedures on Buhler Sortex data. Second, we quantitatively compare the introduced colour inconsistency corrections. The next section introduces the data used in the experiments.

5.3.1 Data

This section describes the procedures and system developed during the EngD to capture data from the Buhler Sortex machine in order to investigate the histogram alignment problem.

A new capture system

Figure 5.20 shows the Z1 machine and the real time data capture setup that has been developed by the author specifically for this project. The architecture details have been classified confidential by Buhler Sortex. The capture setup allows data to be captured from a single monochromatic camera view in real time. This setup provides significant advantages over previous capture setups at Buhler Sortex that had a 25 second delay between captured frames. The non real-time nature of the previous capture solution meant that a recirculation rig was needed to recycle to product during a data capture. Figure 5.21 shows a recirculation rig that pumps the product back up to the input hopper via a mechanical system. With the previous setup, the rice was physically polished as it recycled through the rig thus changing its brightness over time. The developed system avoids these problems.



Figure 5.20: Buhler Sortex Z1 sorting machine and PC based capture system. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission.



Figure 5.21: Buhler Sortex Z1 sorting machine and recirculation rig. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission.



Figure 5.22: The author operating the touch screen interface on the Buhler Sortex Z1. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission.



Figure 5.23: Camera and sorting electronics. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission.

All data referenced in this chapter was captured using the new capture system. The development of this system was a significant undertaking that occupied a significant portion of the 1st year of the project. All parts were ordered, assembled and custom software was written and debugged as part of the project. The developed system allows real time streaming data to be captured for the first time from Z1 machines, this ability to capture this data will prove beneficial in a variety of other projects.

Data capture procedure

The Z1 series machine is first calibrated using the in-built calibration routines. Figure 5.22 shows the author operating the Buhler Sortex Z1 machine. Once fully calibrated, the camera is unplugged from the machine's internal electronics shown in Figure 5.23 and plugged into the custom capture setup. This does not affect the internal state of the machine.

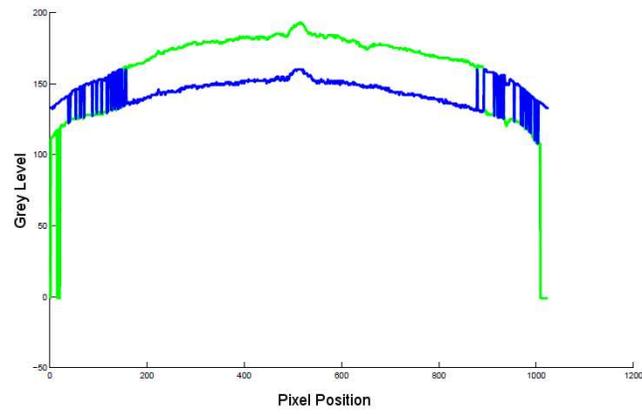
A continuous flow of rice was created by filling the top input hopper with rice three times, this was sufficient for calibration and data capture. Notice the ladder positioned next to the Z-Series during a capture session in Figure 5.20, the top input hopper shown in Figure 5.24 is filled with rice. Rice is



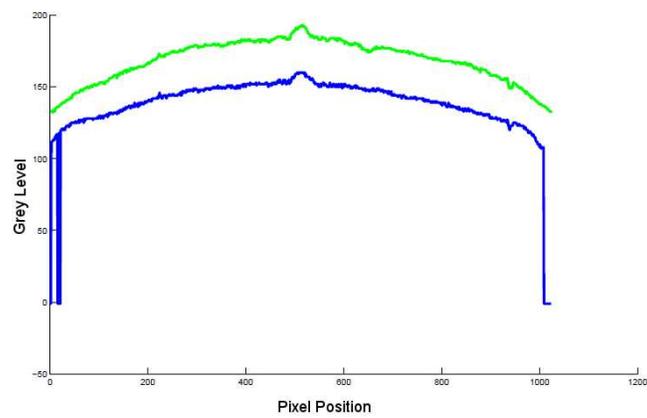
Figure 5.24: Top chute to be filled with rice on the Buhler Sortex Z1. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission.



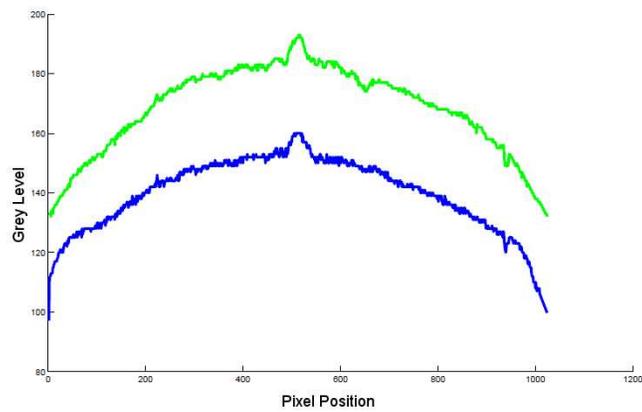
Figure 5.25: Bottom of chute on the Buhler Sortex Z1. Picture ©copyright Buhler Sortex Ltd, 2008. Printed with permission.



(a)

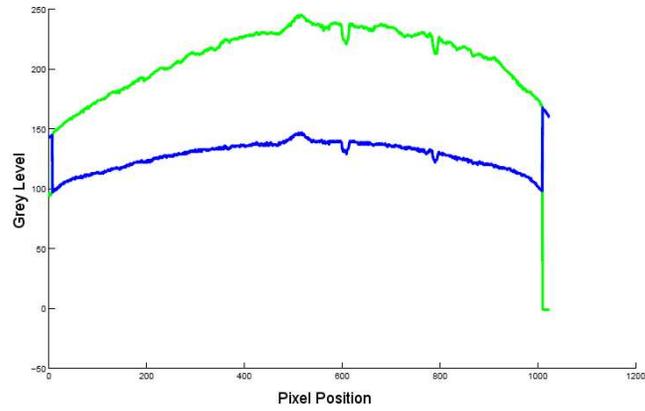


(b)

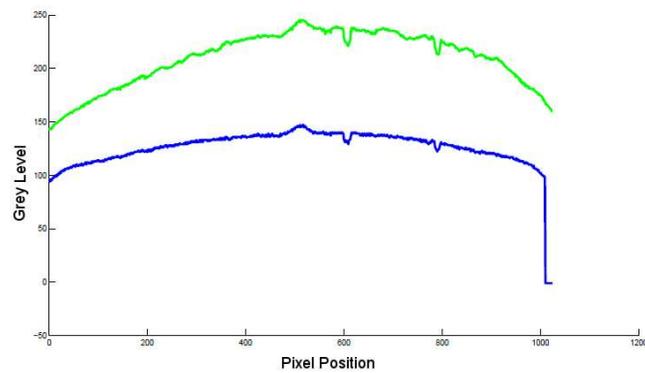


(c)

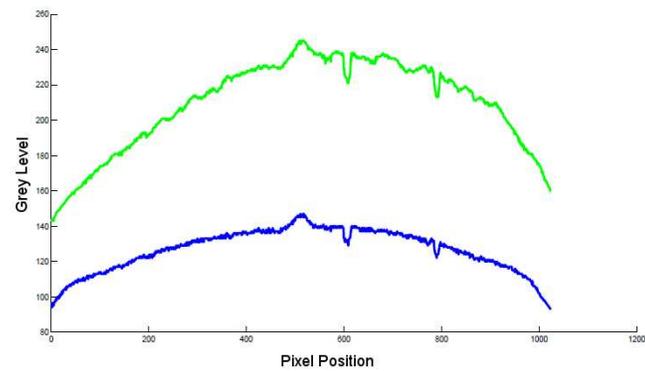
Figure 5.26: Associated persistent features from the front view using an offset of 110 using *Associate-ToTargetPixel(Targetpixel)* in 5.26(a), *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* in 5.26(b) and *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* with gap filing in 5.26(c).



(a)

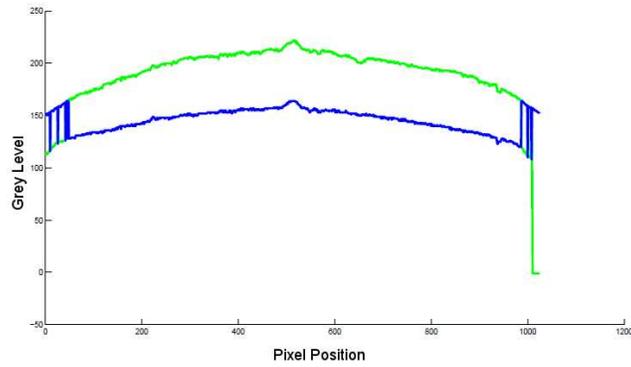


(b)

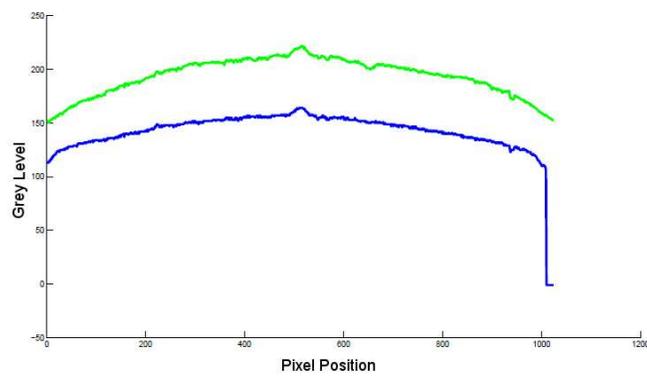


(c)

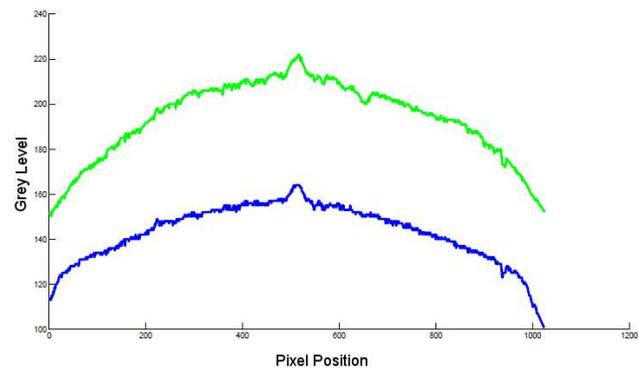
Figure 5.27: Associated persistent features from the rear view using an offset of 110 using *Associate-ToTargetPixel(Targetpixel)* in 5.27(a), *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* in 5.27(b) and *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* with gap filing in 5.27(c).



(a)

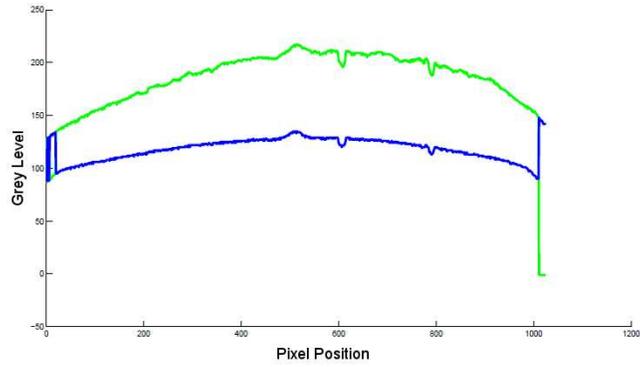


(b)

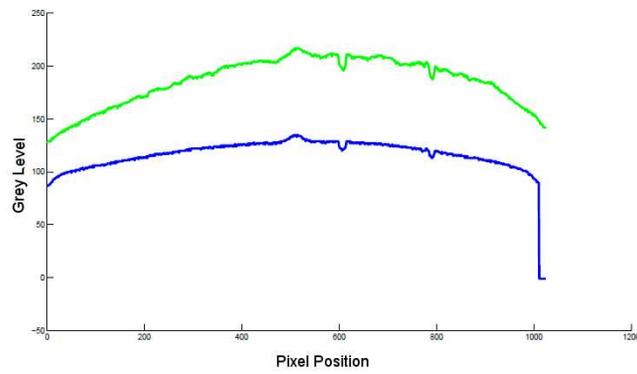


(c)

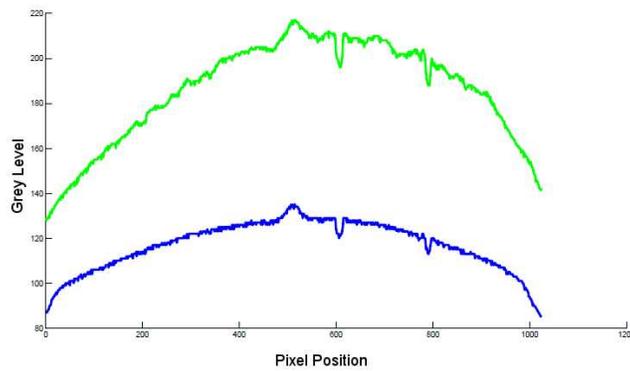
Figure 5.28: Associated persistent features from the front view using an offset of 120 using *Associate-ToTargetPixel(Targetpixel)* in 5.28(a), *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* in 5.28(b) and *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* with gap filing in 5.28(c).



(a)



(b)



(c)

Figure 5.29: Associated persistent features from the rear view using an offset of 120 using *Associate-ToTargetPixel(Targetpixel)* in 5.29(a), *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* in 5.29(b) and *ThreeStageAssociateFeatures(F,Targetpixel,EdgeT)* with gap filing in 5.29(c).

collected at the bottom of the machine as shown in Figure 5.25 and a bucket is used to refill the top input hopper, a separate operator controls the capture software during this process.

The Z-series has a *background offset* parameter to the calibration routines that can be set from the graphical user interface. A higher background offset value increases the distance between the product reference, r_i , and background mean, b_i , across the view by adjusting the angle of the white calibration plate during the calibration cycle. Therefore, a high background offset value increases contrast between the background and product grey-levels making the segmentation more robust. There is a trade off between setting a high background offset and using up the dynamic range of the camera to record background and acceptable product grey-levels, a higher offset means that there is a smaller range of grey-levels to discriminate between the acceptable product and the defect. The capture system can capture the data feed from only one camera at a time. American parboiled rice is used with 2-3 percent defect to compare the effects of:

1. Different background offset settings - (110 and 120), and
2. Front and rear views.

White lamps were used in the machine. The data-sets captured are:

1. the front view, calibrated with an offset of 110.
2. the rear view, calibrated with an offset of 110.
3. the front view, calibrated with an offset of 120.
4. the rear view, calibrated with an offset of 120.

The offset values of 110 and 120 give a low and high contrast between the background and rice respectively. The offset is commonly set to 110 in production sorting setups, data is captured with the 120 setting because the product and background peaks are further apart which should be an easier histogram alignment task.

5.3.2 Qualitative evaluation of feature detection and association

Aims

To assess the feature detection and association steps within a Buhler Sortex view.

Method

For each of the four data-sets, the following steps are performed:

- Compute histograms in each pixel for a portion of the video stream.
- Compute persistent maxima from the histogram in each pixel using $FindPersistentMaxima(H)$ in algorithm 1 using scales, $\sigma_i = e^{0.1(i-1)}$, where $i = 1..T$. The scale persistence threshold, T , is set to 17. The noise floor threshold, γ , is set to 0.001.
- Match the detected features using the three different routines introduced in section 5.2.1. These are:

- *AssociateToTargetPixel*(**Targetpixel**), with **Targetpixel** = 512,
- *ThreeStageAssociateFeatures*(**F,Targetpixel,EdgeT**). **Targetpixel** = 512 and *EdgeT* = 30,
- *ThreeStageAssociateFeatures* with an additional gap filling step.

Plot the associated background features using a green line the associated product features using a blue line.

Results and conclusions

Figures 5.26 and 5.27 show results for the front and rear, 110 offset datasets respectively. Figure 5.28 and 5.29 show results for the front and rear, 120 offset datasets. The most noticeable result from the plots is that the *ThreeStageAssociateFeatures* procedure followed by a gap filling step performs the best. The problems with the two association methods highlight issues when matching features across the view. Figure 5.26(a) shows how the minimum Euclidean distance can mismatch features between histograms that exhibit large deformations. The features in each pixel are matched to pixel 512 which causes mismatches between the background and foreground features at both ends of the chute, this is because the mismatched features are closest together. Figure 5.26(b) shows that finding associations between neighbouring pixels working out from the centre of the view resolves this problem. Two further problems with the associated features are evident in the matched product features in figure 5.26(b):

1. Absent features are represented as a zero grey levels and so the plotted feature line drops to zero. Further investigation of the features in 5.26(b) showed that the optimal feature persistence threshold is different for these few pixels. Despite this, we note that the same feature detection parameters are used in all pixels for all four data-sets; only these pixels exhibit this issue.
2. Product edge features are frequently not present at the edges of the chute due to low or no product passing these pixels. This can be seen in figures 5.26(b), 5.27(b), 5.28(b) and 5.29(b).

We have shown that the combination of *ThreeStageAssociateFeatures* with gap filling works well across the data-sets.

5.3.3 Quantitative evaluation of colour inconsistency corrections

Aims

The aim of all methods is to minimize the colour inconsistency of the acceptable food across different pixels. The product is the focal point of the investigation because the background is removed and the defect does not form significant peaks in the histogram.

Method

The introduced methods to perform appearance correction within a view are comprised of steps with a number of options. The different combinations are evaluated, and all methods are applied to each data-set. An alignment score is computed after applying each method. The following steps detail the experimental procedure used to evaluate alignment methods on each of the four within-view data-sets.

For each data-set:

1. Divide the video stream into a training and test portion.
2. Compute the histograms for each pixel across the view using the training set.
3. For each alignment method, compute the alignment transformations in each pixel across the view using the training set.
4. Apply the computed transformations in each pixel to the test set data histograms.
5. Use the ground truth labels (see below) to extract the aligned histogram data across the view for each class label.
6. Discard edge histograms computed from pixels that have observed insignificant amounts of product.
7. Normalize all remaining histograms.
8. Compute the variance of the aligned class histograms using the summary variance measure described in equation 5.21 (see below).

Ground truth

The histogram alignment methods evaluated may contain a segmentation and alignment transform step. The different methods are compared by evaluating their alignment performance on a data-set that has been labeled as the ground truth. To produce an acceptable ground truth, the existing calibration method is used to label portions of each data-set with the class labels 1) Background, 2) Accept product, 3) Edge Pixel and 4) Defect.

The steps used to compute the four labelled classes are:

1. The **DiffOffset** method described in equation 5.11 is used to generate thresholds to segment the background.
2. The background portion of the signal is removed and the remaining signal is aligned across the view using the multiplicative correction transform. This is computed using equation 5.14 and **GainToMax**.
3. A fixed defect threshold is applied across the corrected view signal to label the defect. The threshold is adjusted manually until a visually acceptable result is obtained.
4. The product signal is isolated and used to create binary images of the rice. An erosion filter is run to classify the edge pixels, the remaining non-eroded pixels are labelled as the product.
5. The labelled class information is used to create a four coloured mask that overlays the original grey level data. The product regions are inspected by eye and a manual correction is performed. Missing product pixels are added and false product classifications are removed.

This procedure produces a highly robust labelling of the acceptable product data. The defect portions of the histograms are not directly studied in this work.

Metrics

The variance of the transformed ground truth classes tells us how much residual colour inconsistency remains. When making comparisons, a better alignment method leads to lower variances of each individual class. First, the data is transformed and then the ground truth labels are used to compute class-labeled histograms in each pixel. Each of the C labels can be used to extract portions of this ground truth histogram. To reduce of the sensitivity of the metric to scale variations between different instances of the same class, the transformed histogram components for each class are normalized.

For the c th class, we compute the variance in each grey-level of the transformed ground truth histograms components labeled c . For a set of N labeled histograms from the CCD pixels $\{p\}$ we write the variance of the vector of histogram bin values for the i th grey-level of the c th class $\mathbf{g}_{i,c}$ as:

$$\text{var}(\mathbf{g}_{i,c}) = \frac{\sum_{p=1}^N (g_{i,p,c} - E(\mathbf{g}_{i,c}))^2}{N}, \quad (5.19)$$

where p indexes the CCD pixel, $g_{i,p,c}$ is the bin-count from the i th grey-level of the p th histogram for class c . The expected grey-level at intensity i for class c is

$$E(\mathbf{g}_{i,c}) = \frac{\sum_{p=1}^N g_{i,p,c}}{N}. \quad (5.20)$$

We summarize the grey level variance using a single number by summing over all grey-levels according to,

$$V_c = \sum_{i=1}^{256} \text{var}(\mathbf{g}_{i,c}). \quad (5.21)$$

The different set of pixels $\{p\}$ that we consider are:

1. From the correction of a single view.
2. From the alignment of front and rear views.

Care must be taken to reject edge pixels that have not observed rice falling past, this can occur at the edge of the chute due to rice bouncing off the sides; including such pixels in the metric can cause outlandish results and so we remove the histograms from these pixels from the alignment evaluation. Outlier pixels are identified by inspecting edge pixels histograms and flagging those with no product peaks, the outlier pixels are saved along with the ground truth information and used during each evaluation.

Experiment 1: segmentation driven within-view alignment

Hypothesis: Alignment of the first two moments of the eroded product distribution leads to the best alignment score when compared to other product alignment methods.

Method and Results Within view alignment methods that segment the product distribution and then align this portion of the distribution are used to align the test set data as described in the previous section. The methods evaluated each comprise a product segmentation method, a transformation and a choice of target values. For each data-set we rank the transformations according to the score on the aligned product portion of the distribution. The best 15 scores are displayed as a bar chart for each data-set in figure 5.30.

Conclusions Segmentation driven within-view alignment methods that use linear correction transforms outperform methods that use the multiplicative correction. This is seen by comparing linear and multiplicative correction results where all other conditions are held constant; for example, DiffOffsetEros0-93ExtrapMeanVarToMax performs better than DiffOffsetEros0-93ExtrapGainToMax in Figure 5.30(a). This pattern is repeated among other transforms. In addition, the best performing methods DiffOffsetEros0-93ExtrapMeanVarToMax in Figure 5.30(a) and PercMean0-93ErosMeanVarToMax in Figure 5.30(b) both use the linear correction. This tells us that the linear correction reduces the appearance variance of the product across the view; during dark-sort thresholding appearance variation in the product is significant as it affects the effectiveness of the thresholds used across the view. The results also show the importance of discarding edge pixels to gain an estimate of product brightness as almost all of the top 15 ranked methods on all data-sets utilize the erosion step.

Also of note, is the variation in alignment score according to the choice at target parameters. For both the multiplicative and linear transformation there is an advantage gained by aligning to the maximum values; this ensures that the multipliers are positive and the range of the bins occupied by product is increasing, this yields more similar histograms compared with the case where the range is being reduced. Finally, the alignment scores are sensitive to the method used to segment the product across the view, the best performing method can vary according to the data-set. In all cases the methods are dependent on product reference value computed using 5.10 with $P = 0.93$. Individually tuning P may yield improved results on specific data-sets. A common value was chosen across all data-sets for simplicity.

Further individual tuning of this parameter may lead to improved results in some cases, however this parameter is set by observing its effect on aggregate system performance by a Buhler Sortex engineer during setup.

Experimental 2: global histogram alignment

Hypothesis: FBHA transformations outperform shift, multiplicative or linear alignment of the distribution moments.

Method and Results Shift, Multiplicative and linear alignment of the distribution moments are compared against the FBHA procedure with linear, quadratic and cubic correction transforms. Alignment scores for the 110 calibration offset data-sets are shown in 5.32 and 5.33 shows the 120 calibration offset data-sets. FBHA3MatchMax performs best in all cases. All FBHA methods perform better than all moment based corrections on the front view 110 and 120 data-sets, this is seen in Figures 5.32(a) and 5.33(a) respectively. Most FBHA methods perform better than moment based corrections on the rear view data-sets shown in Figures 5.32(b) and 5.33(b). The GlobalMeanVarToMax and GlobalMeanVarToTarget methods outperform FBHA1MatchMean for the rear view 110 data-set and they outperform FBHA2MatchMean and FBHA1MatchMean on the rear 120 data-set.

Conclusions FBHA is shown to be robust and effective, it outperforms other global histogram alignment methods. The parameters of the algorithm are shown to be robust and features are extracted from the data in an unsupervised manner. There are no hard wired assumptions in the algorithm about the number of clusters in the data. The results show an improvement of the alignment score from linear through to

cubic transforms.

The global transformations do not align the individual product distribution components as well as the segmentation driven approach. However, the FBHA approach is more general and aligns the background components of the distribution as well.

5.4 Summary Conclusions and Discussion

This chapter makes four key contributions:

1. **FBHA robustly aligns the appearance of Buhler Sortex in-feed data.**

The procedure *ThreeStageAssociateFeatures* with gap filling detects and associates features within the view reliably so that FBHA can be performed. We learn that correct association of the features across the view can be achieved by using the knowledge that neighbouring pixels give rise to similar histograms and edge pixels frequently observe either low levels of product or no product at all. FBHA with linear, quadratic and cubic transformations equalizes the appearance of the product and background across the view. These combinations outperform shift, multiplicative and linear alignments of the global moments of the histograms. Figures 5.32 and 5.33 show that the best global feature based alignment method is FBHA3MatchMax on all four data-sets. Feature based histogram alignment methods that use a third order polynomial give product variation scores approximately a factor of two lower than the next best moment based transform GlobalMeanVarToMax. This validates using the feature based approach compared to moment based approaches.

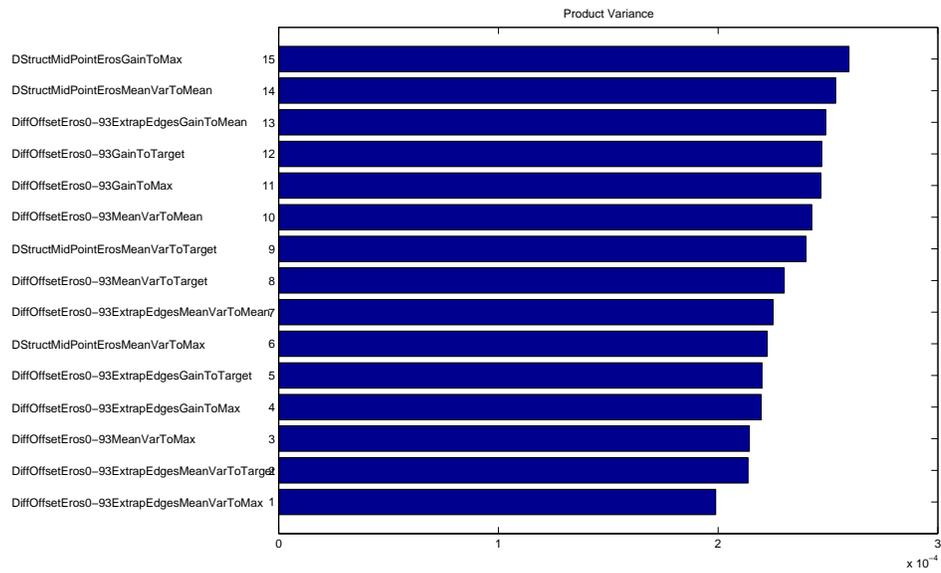
2. **Background removal thresholds are computed using in-feed data.** Stopping the feed to inspect the background plate is a costly procedure because lost sorting time reduces the productivity of a sorting machine. We have introduced an alternative to the average intensity statistics method that does not require separate background estimates with the feed turned off. The *DStructMidPointErosMeanVarToMax* method processes histograms obtained with the feed turned on and performs well on all data-sets; it gives close to the best result in 5.30(a), 5.30(b) and 5.31(a). It performs best in 5.31(b), the method works well because it robustly segments the product and background across the view before applying the linear correction step to the product. Global moment based transforms perform worse in most cases because they do not consider the multi-modality of the distributions to be aligned. *DStructMidPointErosMeanVarToMax* could give superior performance over long periods of operation compared to the Buhler Sortex algorithm, this is because the Buhler Sortex method's background estimates will become more inaccurate if the intensity of the background changes over time. Further tests and data capture could explore whether this scenario arises in production set-ups.

3. **Performance effects of component permutations of the Buhler Sortex algorithm are evaluated** We discover that the erosion step is critical to the performance of all segmentation based methods. The top fifteen segmentation based methods in Figures 5.30 and 5.31 all use the erosion step to discard the edge pixels. We also discover that no single background segmentation method performs best across all data-sets, the difference in product variance across the view for the top

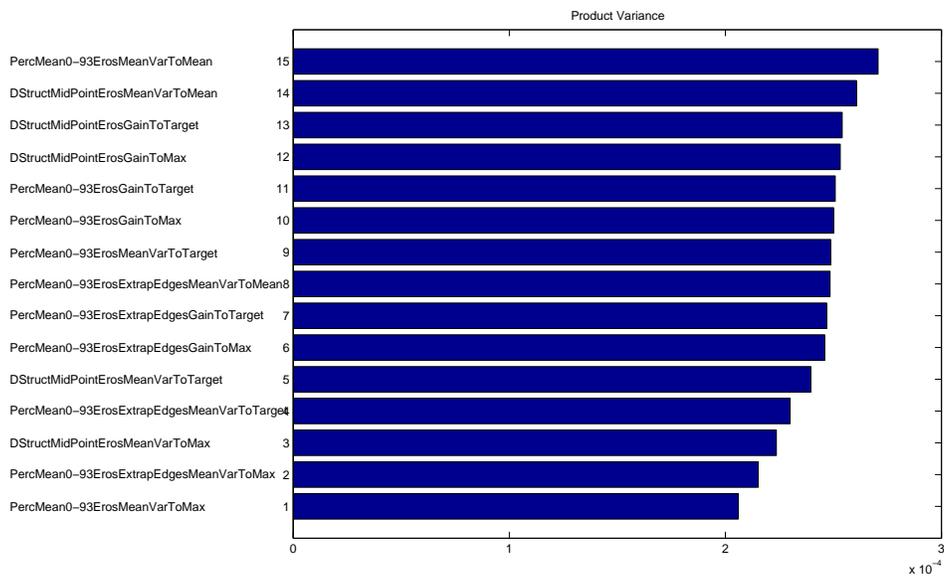
fifteen segmentation based methods is very small compared to the global histogram alignment results. This variation is approximately 0.5×10^{-4} between the best and worst results in 5.30(a), 5.30(b) and 5.31(b). The variation is approximately 1.0×10^{-4} in 5.31(a). The results contrast significantly with the global histogram alignment scores which are significantly higher for all methods. The differences between the best global histogram transformation and FBHA3MatchMax, the best performing global moment based transformation are approximately: 3×10^6 GlobalMeanVarToMax in 5.32(a), 1.75×10^7 in 5.32(b), 4×10^6 in 5.33(a) and 1.5×10^7 in 5.33(b). We note that all global histogram alignment methods are significantly worse than background segmentation driven methods.

4. **A linear correction of the product appearance is introduced.** The linear correction gives lower appearance variation across the view compared to the multiplicative correction. This is significant because the multiplicative correction has significant support in the literature [49]. This tells us that it is worth aligning the mean and variance of a histogram mode when the histogram can be reliably labelled.

Global and segmentation driven local correction transforms have been examined and contrasted. The controlled environment and constraints of the *dark sort* procedure mean that alignment of the product mode of the set of histograms is of utmost importance. We learn that in these cases, segmentation driven algorithms are favorable. We have shown how the new feature detection procedure can be used to perform the segmentation - it is important to realize that the FBHA procedures used have no parameters to indicate the number of clusters present in the data. This is a key design feature of this approach, specifying the number of colour clusters frequently leads to brittle assumptions; the bottom up feature extraction procedure deserves further examination on Buhler Sortex bi-chromatic machines. Future work may also seek to develop the idea of performing segmentation driven piece-wise alignments on sets of 2D or 3D colour histograms.

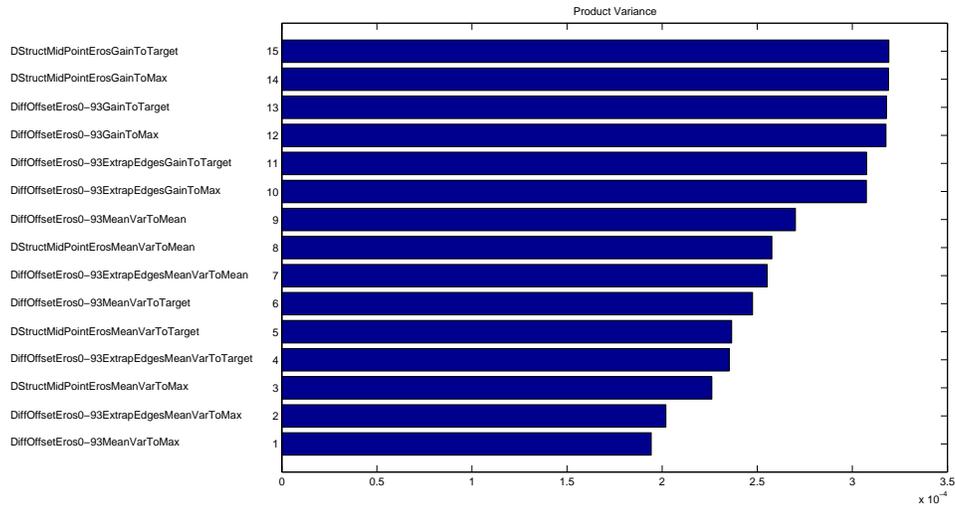


(a)

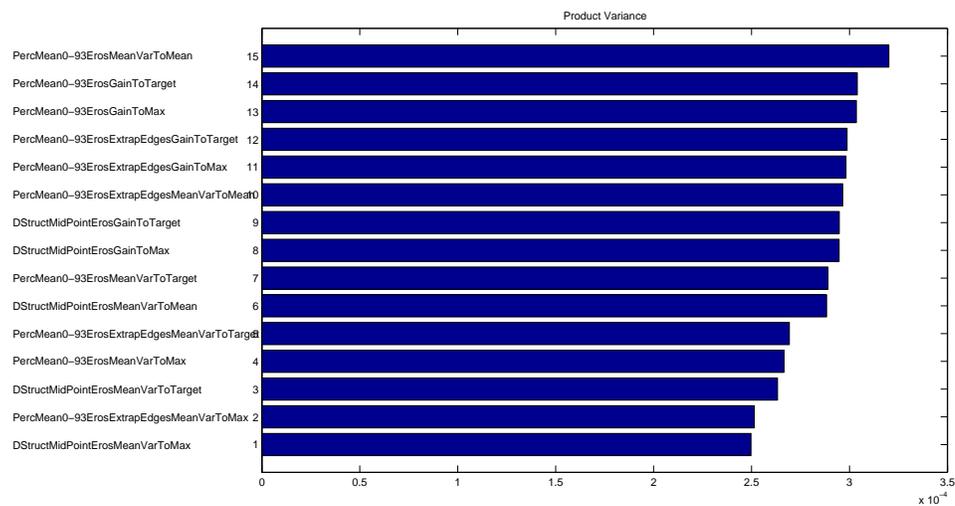


(b)

Figure 5.30: The fifteen best performing within view transformation methods applied to the front 5.30(a) and rear 5.30(b) view data-sets with a calibration offset of 110. The scores indicate the variance of the product components of the histograms across the view after correction. A lower variance indicates better alignment.

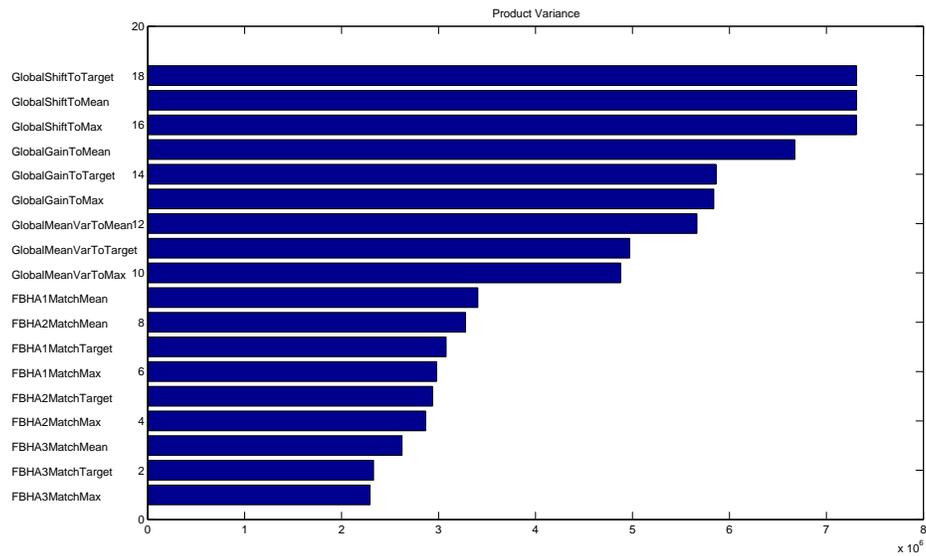


(a)

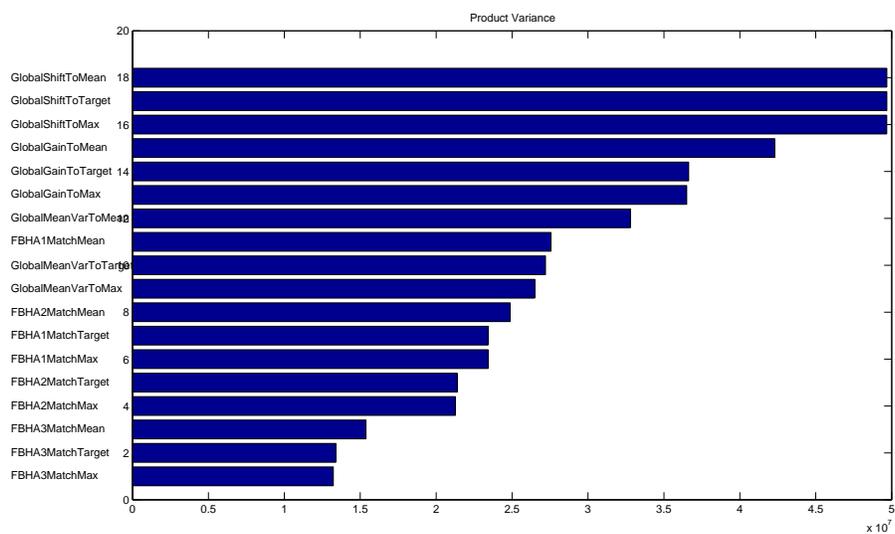


(b)

Figure 5.31: The fifteen best performing within view transformation methods applied to the front 5.31(a) and rear 5.31(b) view datasets with a calibration offset of 120.

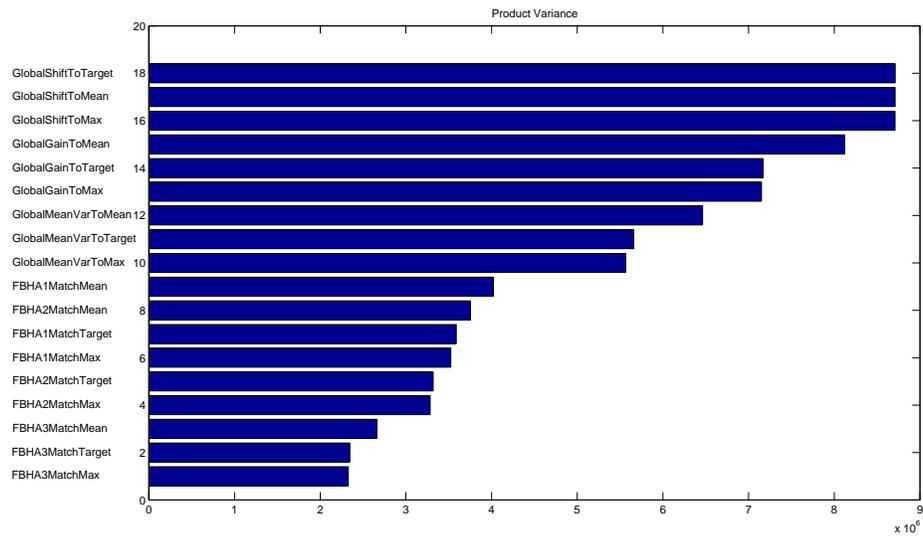


(a)

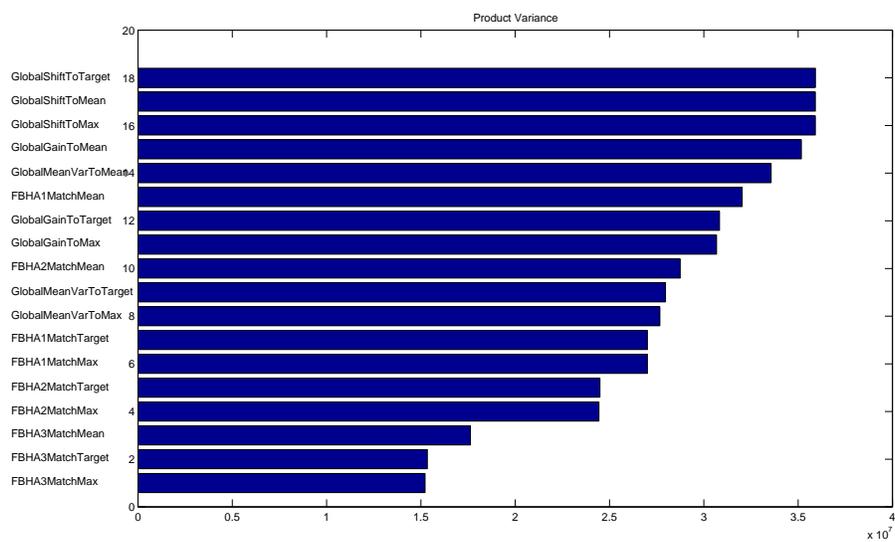


(b)

Figure 5.32: Variance of the product components of the histograms across the view after correction with moment based and feature based global correction transforms on data from the front 5.32(a) and rear 5.32(b) views with an offset setting of 110.



(a)



(b)

Figure 5.33: Variance of the product components of the histograms across the view after correction with moment based and feature based global correction transforms on data from the front 5.33(a) and rear 5.33(b) views with an offset setting of 120 .

Chapter 6

Conclusions and Further Work

This chapter highlights the commercial relevance of the work in this thesis then discusses the empirical findings presented in Chapters 4 and 5. It considers what the empirical findings tell us about removing colour inconsistencies using an automatic histogram alignment approach. The limitations of the work are identified and suggestions for future improvements and extensions are discussed.

6.1 Commercial relevance and contributions

This section highlights the commercially relevant achievements in this thesis

6.1.1 Direct applicability of findings

Chapter 5 provides a detailed investigation of methods for reducing colour inconsistencies in the product appearance across the view of a Buhler Sortex Z1 machine. Section 1.2 reasoned that a performance improvement that saves just 0.5 percent of the processed food volume will yield an extra 270 tonnes of product per machine per year. Buhler Sortex engineers know that colour inconsistencies are one significant factor that affect the Sorting performance of the machines; the introduced linear correction gives the best segmentation driven alignment and the FBHA alignments gives the best non segmentation alignment method. Deployment of the calibration algorithms into sorting setups will enable the effect of the improved calibration on the performance on the food sorting process to be tested.

6.1.2 Future applications

Future work may extend the FBHA approach to work with bi-chromatic colour data. The experiments performed on the image data-base in Chapter 4 provide insights into the behavior of FBHA in 2D. The background removal thresholds computed using the deep structure features proved extremely robust on the Buhler-Sortex data, the need to perform alignment of individual modes motivates the extension of FBHA to perform piece wise alignment of corresponding clusters. The need to segment individual modes in order to align them well suggests that it would be desirable to integrate calibration and segmentation into a fully automatic approach.

6.2 FBHA

The feature based histogram alignment approach has been applied to a database of colour inconsistent images in Chapter 4 and to grey level video streamed data in Chapter 5. The deployment of the same

basic approach on these two different applications was motivated by the need for generic colour inconsistency removal techniques; FBHA shows itself to be a generic approach to aligning histograms with corresponding clusters that have a single dominant peak. FBHA behaves in a robust manner on Buhler Sortex data and a significant number of image pairs in the image database.

The results of the histogram alignment experiments make it possible to draw conclusions about assumptions made by the FBHA approach. These are:

- **Point features.** Histogram peaks are good features to match when the corresponding clusters have the same number of peaks. Small irrelevant peaks are smoothed away by the deep structure feature detection procedure. However, it is unclear how to match different numbers of peaks for the corresponding cluster. Deep structure feature detection is shown to robustly find peaks in 1D and 2D histograms, the technique does not generate robust features from 3D histograms of the RGB data. It is thought that tracking the connected path of maxima in the scale space of 3D histograms leads to broken tracks as the number of degrees of freedom for the path following step is increased. The deep structure feature detector is useful because its results are not dependent on initial seeding points as is the case with algorithms such as K-Means and Expectation maximisation mixture model fitting; the detector does not require the number of clusters to be input in the algorithm and has proven robust to its input parameters.
- **The structure of colour inconsistent histograms.** The experiments on the RGB image database show that high variability between corresponding clusters in colour inconsistent histograms is common. This means that it is difficult to match point features correctly due to these unpredictable changes in the structure. The FBHA procedure failed most often on these cases, this shows that point feature detection and a CEM or CEM-DC matching strategy does not work successfully in many cases. Ambiguous matches between feature points can only be resolved if feature points are associated with a cluster. The lesson here is that matching feature points alone is not sufficient in many cases, correct matching can only be achieved in these cases by reasoning about the clusters in the histograms. Section 6.4 discusses potential approaches to extend the current FBHA approach to meet this challenge.

The Buhler Sortex problem did not exhibit the same problems of histogram structure variation as the RGB histogram alignment problem. Grey-level histograms obtained from the Sorting machine have clear and unambiguous clusters that enable the FBHA to work well. The stable nature of the histograms obtained from the Buhler Sortex machines is probably not surprising, as these machines are engineered for high performance imaging and use quality optical and electronic components. However, the commodity cameras used for the RGB image database capture are produced with different aims in mind, in particular they are designed to produce visually pleasing images using optics and electronics that must be produced at competitive high street prices.

Applying the same generic approach to colour data from different sources has proved informative. The results show that a generic approach has value, but caution must be exercised when applying a

method that works well under one set of acquisition conditions to data-sets that are acquired under a different set of conditions.

- **Alignment transforms.** FBHA allows point alignment transforms to be used to align colour histograms. Previously, this class of transform was restricted for use by applications that labeled the colour data using information from the spatial domain. For example, by extracting the corresponding coloured squares of a MacBeth chart in different images and using the mean colours of each square as the feature points. Point alignment transforms are powerful, they can align histograms more precisely than other commonly used colour transforms such as moment based alignments. Section 4.4.1 in chapter 4 investigates the feature based alignment hypothesis and finds that point alignment transforms perform better than all other classes of method tested.

Global and piecewise local transforms are investigated in this thesis. The experiments in chapter 4 transform all colour values using different global transforms. Chapter 5 evaluates both global and local transforms on Buhler Sortex data, section 5.3.3 presents experiments that quantitatively evaluate these transforms. These experiments show that local transforms align the appearance of the rice better than global transforms, they also show that point alignment transforms are the best choice among global transforms. It makes sense that piecewise transforms that align corresponding histogram clusters can produce better alignments than global alignments that aim to align multiple clusters, this is because piecewise transforms adjust each mode of the histograms individually. Applying piecewise local transforms is difficult because the corresponding clusters must be segmented prior to computing a local alignment transform. The Buhler Sortex machine utilises a robust imaging setup so that segmentation of the corresponding clusters is effective. However, it is more difficult to segment the corresponding clusters of more variable histograms like those of the RGB image database. The work in this thesis highlights that the alignment performance of global non feature based transforms, global feature based transforms and piecewise local transforms exist in a performance hierarchy. Global feature based transforms perform better than global non feature based transforms and piecewise local perform better than global feature based methods on the Buhler Sortex data. It is reasonable to infer that piecewise transforms would also improve the alignment of the RGB image data if suitable segmentations of the histograms can be found. A general observation is that the transforms that perform better are harder to apply automatically as incorrect feature detection or segmentation steps can cause them to fail catastrophically. The results in this thesis suggest that a promising line of future investigation could seek to unite ideas from segmentation and calibration into a single framework, section 6.4 discusses some possible approaches.

- **Multiple histogram alignments.** Chapter 5 demonstrates that multiple histograms can be aligned using the FBHA method. The qualitative assessment of feature detection and association in section 5.3.2 shows how features from Buhler Sortex within view data can be correctly associated using a three stage feature matching procedure. The procedure matches the centre, left and right edge regions separately. Then, the three regions are associated in a final step. Attempts to match

features from all pixels to the features from the centre pixel failed, this result shows that closest Euclidean feature matching (CEM) can fail when the incorrectly matching features are close together. When working with Buhler Sortex data, it is possible to use the knowledge that features from neighbouring pixels are likely to be more similar than features from pixels that are far apart; the three stage feature association uses this knowledge to perform correct matching. FBHA experiments on the RGB image database focus on matching image pairs, the performance of the FBHA method needs to be improved in future work to justify its application to aligning multiple RGB histograms. Automatic alignment of multiple RGB histograms is more difficult than the Buhler Sortex problem because there is no inherent ordering among the images in the data-set and correct cluster association is more difficult because of the structural variation that exists between corresponding clusters.

6.3 Existing methods

Experiment 1 in section 4.4 provides a unique evaluation of commonly used colour alignment transforms. The findings from this work inform machine vision system designers to make intelligent choices when selecting transforms in their work. Particular lessons of note are that transform performance can vary greatly for small changes in experimental conditions. The collated transform rankings can be used by practitioners needing to select a transform for an application, the rankings give a sense of alignment performance and robustness. The high variability of transform performance highlights an area for future work, an automatic transform selection method would be desirable in this case. For automatic model selection to work an improved non parametric model is needed, this option is discussed further in section 6.4.

6.4 Further work

The limitations of the work in this thesis suggests four areas for future research that could advance the state of the art of automatic histogram alignment algorithms. The ultimate aim is to produce a fully modular *black box* algorithm that can be used to remove colour inconsistencies in any computer vision scenario. This section states the research areas along with the problem they should address and some suggestions for potential lines of investigation. The areas for further research are:

1. **Extend FBHA to handle topological features.**

Problem: Point features are useful when the corresponding clusters contain the same number of significant peaks. However, feature points are not easy to match correctly when the structure of corresponding clusters is significantly different. The difficulty of matching the structurally varying clusters that occur in the RGB database motivates the development or usage of cluster detection algorithms that are robust to these changes in local topology.

Potential approaches: A cluster detection method is needed that finds significant peaks in the histograms and also the relationship between these peaks. The method needs to find features that occur at different and unspecified scales, also the algorithm should not require the number of

clusters or their shape to be specified. Klemela [85] has introduced a level set tree method for the visualisation of multivariate density estimates that appears to meet these criteria. The method builds a tree structure from the separated parts of level sets of a function, this is called a level set tree. Klemela proposes the method as a way of visualising general multivariate functions and shows promising results on synthetic histograms generated using Gaussian mixture models in three and four dimensions. Two relevant questions to investigate are: 1) Can level set trees be used to robustly detect features in the 1D, 2D and 3D histograms of images from the RGB image database?, and, 2) Can level set trees of these histogram be robustly matched? Work on comparing the topological structure of 3D shapes [86] may provide some insights into the best ways to perform the matching.

If this technique is found to work, it would represent a significant advance. Volume estimates are computable for each cluster in the level set tree, so it is possible to express the posterior probability of each data point belonging to the identified clusters in the level set tree. A Bayesian expression that describes the probability that an unseen data-point belongs to each of the clusters identified by the level set tree would be of great value. It would be a natural next step to develop this formulation if initial tests on the level set tree are positive. The Bayesian formulation would make it easy to combine labeling information from other algorithms that use prior information as label probabilities can be combined by use of the multiplication rule.

2. Unordered set automatic multiple histogram alignment

Problem: The FBHA approach is not currently tested on the unordered set, multiple histogram alignment problem. The alignment experiments on the Buhler Sortex data show how multiple alignment can be performed but the approach uses the knowledge that neighbouring pixels produce more similar histograms. The experiments on the RGB database only test alignments on pairs of histograms. Future work could explore the challenges of aligning multiple histograms when no ordering information between the histograms is known.

Potential approaches: A sensible starting point for research on this topic would be to devise algorithms to align Buhler Sortex data that do not utilise the inherent ordering between pixels. Potential lines of investigation could take inspiration from work into the alignment of multiple range views [87] and point sets [88][89]. Attempts could then be made to extend the multiple alignment approach to the RGB data, it is presumed that the ability to do this successfully is dependent on the ability to successfully manage the topological features described in the 1st item in this list.

3. Produce a unified cluster segmentation and alignment routines

Problem: The alignment of corresponding clusters using piece wise transforms requires a cluster segmentation step to be performed before the alignment step. Work on the Buhler Sortex problem shows that the segmentation method and alignment are coupled; applying piece wise transforms in more general settings (such as the RGB data-base) requires methods that integrate ideas from segmentation and alignment.

Potential approaches: The problem of dealing with data from multiple sources or multiple learned models has been tackled by the machine learning community. Cluster ensembles [90][91] are methods to combine multiple partitionings of a set of objects into a single consolidated clustering. An insight from the alignment experiments on Buhler Sortex data is that a single clustering is not sufficient to segment all the histograms correctly. Instead, it is important to find correct clusterings in each histogram and the transformations between these clusterings. It is thought that development in this area would benefit from development of the first two research items in this list.

4. **Extend segmentation framework to allow priors to be seamlessly integrated into the labeling process**

Problem: Correct histogram alignment is dependent on correct labeling of the histograms prior to alignment, either by feature detection or complete labeling. Fully automatic histogram alignment algorithms do not utilise many powerful sources of prior information in particular problem domains. A challenge for the development of a generic approach is to retain a modular framework that allows prior information from other labeling processes to be seamlessly integrated.

Potential approaches: The development of Bayesian labelings that do not impose unnatural shape constraints on the distributions would allow prior information from other sources to be integrated probabilistically in a natural manner. In striving for this goal, it is important not to impose distributions that do not fit the data just because they are easy to deploy; for example the Gaussian mixture model is frequently used to model highly non Gaussian distributions in many computer vision applications. Advances to research item 1 in this list would naturally lead to these extensions. Information from the spatial or temporal domain provide powerful cues and should not be ignored when aiming to build the best systems, it would be interesting to integrate generic spatial segmentation approaches such as graph partitioning [92] with the automatic histogram alignment approach. Additionally, striving for clear modularity will lead to wider deployment of algorithms and deeper insights in the future.

Chapter 7

Summary of Achievements

This chapter summarises the achievements made by this thesis. Section 1.3 introduced the automatic histogram alignment problem and section 1.4 outlined the goal to develop unsupervised alignment algorithms that can align the corresponding clusters in colour histograms. The achievements that meet these goals and improve understanding are:

- **Introduced taxonomy of colour inconsistency removal techniques.** Chapter 2 provides a new way of looking at colour inconsistency correction methods by organising methods into a taxonomy. The relationship between apparently disparate methods is made explicit, common transformations are identified and related to different methods in the literature. The chapter can be used on its own by anyone interested in an overview of basic colour theory and colour inconsistency removal.
- **Introduction of a new feature based histogram alignment approach.** Chapter 3 introduces a new feature based histogram alignment approach. The algorithm makes effective use of a scale space technique to robustly detect features in 1D or 2D colour histograms. The introduction of the scale space feature detector solves an important feature detection problem, it finds histogram peaks at different scales robustly and efficiently. FBHA can be successfully used to align histograms that have similar structures using feature point alignment transforms. Feature point transforms are shown to be a useful and powerful class of transform; previous colour inconsistency removal applications relied on manual labeling or domain specific prior information such as the presence of Macbeth charts to use these transforms.
- **Design and capture of colour inconsistent databases.** The experimental design and subsequent data-capture of the RGB image database described in Chapter 4 and the Buhler Sortex video data described in Chapter 5 were both significant undertakings. In the case of the Buhler Sortex data capture, a new capture system and associated software were designed and constructed by the author as part of this project.
- **Introduction of procedures and metrics to rank colour inconsistency removal methods.** Chapter 4 introduces a new metric for evaluating the alignment of labeled or partially labeled histograms, the average Mahalanobis distance. The metric is a fair way to rank multi-modal alignments, its choice is justified by empirical comparison with existing metrics. Chapter 4 also in-

troduces a ranking methodology based on bootstrap statistics, the method produces an ordered ranking of all methods tested on the RGB image database. The bootstrap methods handle the highly non-Gaussian nature of the results distributions being compared with an associated degree of confidence.

Chapter 5 introduces procedures to rank different colour inconsistency removal methods on the Buhler Sortex methods, these procedures are specific to the Buhler Sortex data and procedures and show the methods that give the lowest variation in product appearance across the chute.

- **Empirical investigation of methods on RGB data-base** Chapter 4 provides a comprehensive comparison of existing colour inconsistency removal methods. To the author's knowledge, this is the first such application independent ranking of its kind. The ranking can be used by other practitioners as an initial assessment before investing time implementing or using some of the methods in custom systems. The chapter also investigates the FBHA algorithm on RGB data using 1D and combinations of 1D and 2D histogram alignments; both the strengths and weaknesses of the automatic FBHA procedure are highlighted.
- **Empirical investigation of methods on Buhler Sortex data** Chapter 5 investigates both global and piece-wise transforms of the data. FBHA with a cubic transform is found to be the best global alignment transform. Also, the feature detection and association step can be used to segment the product portions of the distribution without the need to inspect the background separately. This functionality alone, could lead to stopping the product feed less so that efficiency is increased. A large number of permutations of system components were tested in this chapter, commercial confidence means that relative improvement to the system are not highlighted directly in this thesis. Relative improvements to the current system can be discussed with the thesis examiners at oral examination only (*Note: this is a contractual requirement*).
- **Clear positioning of the existing work for future research** Significant advances have been made towards the goal of fully automatic and general histogram alignment procedures. Where limitations have been found, they have been exposed, explained and suggestions for future research have been made. Hopefully, this approach will facilitate continued progress on this topic.

Chapter 8

Glossary

BRDF: Bidirectional Reflectance Distribution Function.

CAM: Colour Appearance Model.

CAT: Chromatic Adaptation Transform.

CCD: Charge Coupled Device.

CIECAM97s: A colour appearance model that predicts a number of human colour appearance phenomena such as chromatic adaptation.

CMOS: Complementary Metal Oxide Semiconductor.

CMYK: Cyan, Magenta, Yellow and Key(black) subtractive colour model used in colour printing.

CIE: International Commission on Illumination.

CIE 34 XYZ: CIE colour space based on positive matching functions determined using experiments that use two degrees of visual angle.

CIE 63 XYZ: CIE colour space based on positive matching functions determined using experiments that use ten degrees of visual angle.

FBHA: Feature Based Histogram Alignment.

EM: Expectation Maximisation algorithm.

GMM: Gaussian Mixture Model

HSV: Hue, Saturation and Value colour space.

HSL: Hue, Saturation and Lightness colour space.

RGB: Red, Green and Blue colour space.

YUV: Colour space defined in terms of luminance (Y) and chrominance (UV).

SIFT Scale Invariant Feature Transform.

SVD: Singular Value Decomposition.

U-V: The UV plane of the YUV colour space.

Chapter 9

Appendix

9.1 The Pseudoinverse

The inverse \mathbf{A}^{-1} of a matrix \mathbf{A} exists only if \mathbf{A} is square and has full rank. In this case $\mathbf{Ax} = \mathbf{b}$ has the solution $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. The pseudoinverse \mathbf{A}^\dagger is a generalization of the inverse, and exists for any $m \times n$ matrix. We assume $m > n$, if \mathbf{A} has full rank we define

$$\mathbf{A}^\dagger = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$$

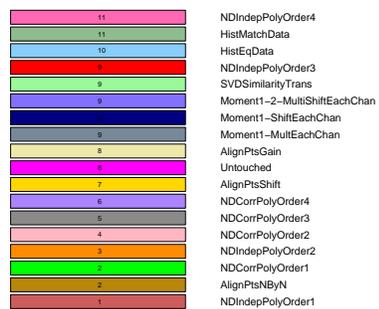
and the solution of $\mathbf{Ax} = \mathbf{b}$ is $\mathbf{x} = \mathbf{A}^\dagger \mathbf{b}$. The best way to compute \mathbf{A}^\dagger is using singular value decomposition. With $\mathbf{A} = \mathbf{USV}^T$, where \mathbf{U} and \mathbf{V} are $n \times n$ orthogonal matrices and \mathbf{S} is an $m \times n$ diagonal matrix with real, non negative singular values.

We find,

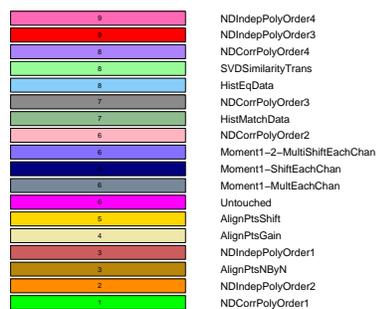
$$\mathbf{A}^\dagger = \mathbf{V}(\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T \mathbf{U}^T.$$

If the rank of \mathbf{A} is less than n , then $(\mathbf{S}^T \mathbf{S})$ does not exist, and one only uses the first r singular values; \mathbf{S} becomes an $r \times r$ matrix and \mathbf{U} and \mathbf{V} shrink accordingly.

9.2 Ordering Results

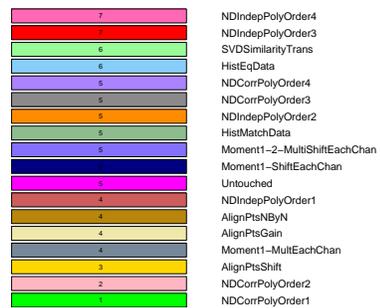


(a)

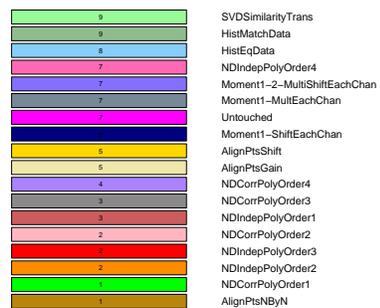


(b)

Figure 9.1: Ranked transformation methods for image pairs with 000(S) variation for: 1) Red-cyan paper 9.1(a) and 2) Skittles 9.1(b).

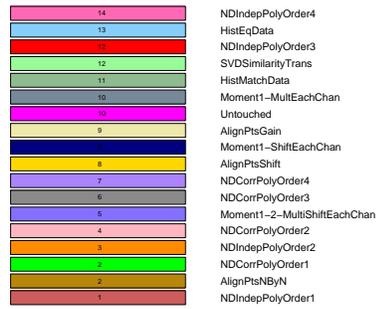


(a)

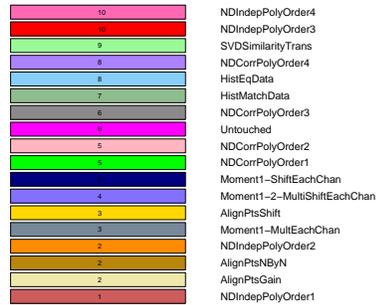


(b)

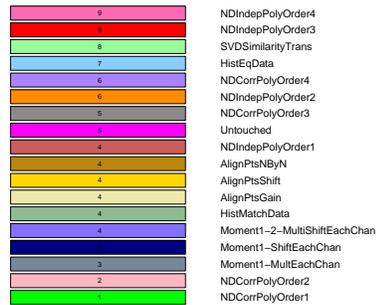
Figure 9.2: Ranked transformation methods for image pairs with 000(S) variation for: 1) Teddy bears 9.2(a) and 2) three paper strips 9.2(b).



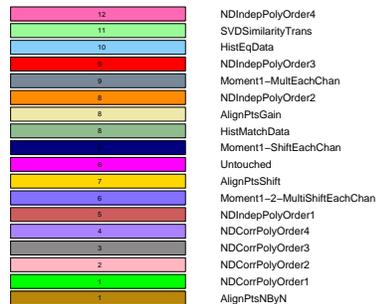
(a)



(b)

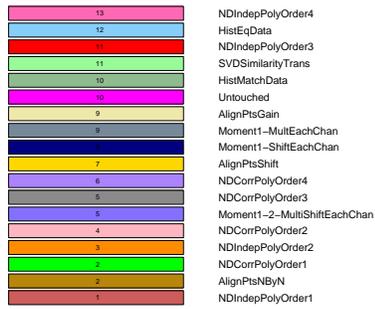


(c)

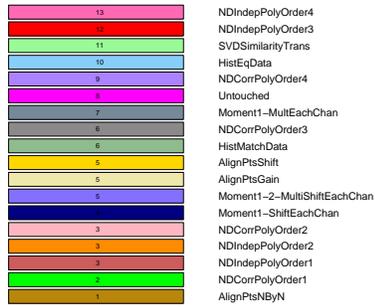


(d)

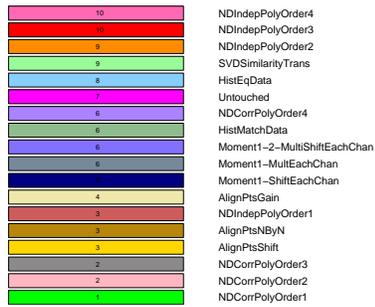
Figure 9.3: Ranked transformation methods for image pairs with 0(L-LI)00 variation for: 1) Red-cyan paper 9.3(a), 2) Skittles 9.3(b), Teddy bears 9.3(c) and three paper strips 9.3(d).



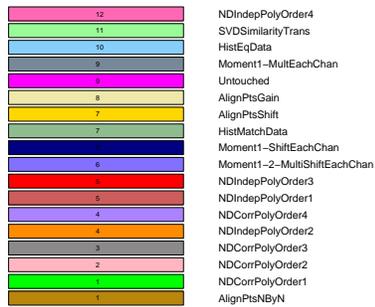
(a)



(b)

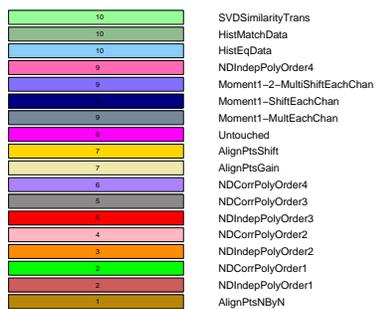


(c)

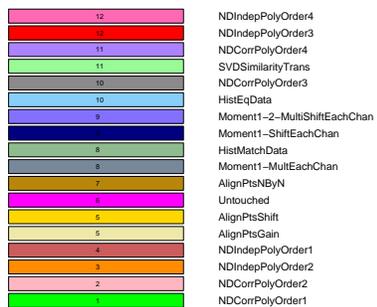


(d)

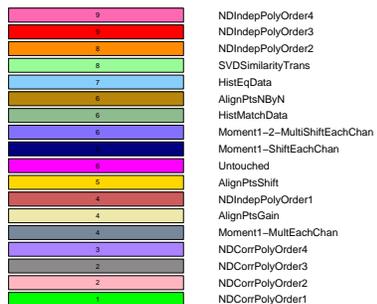
Figure 9.4: Ranked transformation methods for image pairs with (C)000 variation for: 1) Red-cyan paper 9.4(a), 2) Skittles 9.4(b), Teddy bears 9.4(c) and three paper strips 9.4(d).



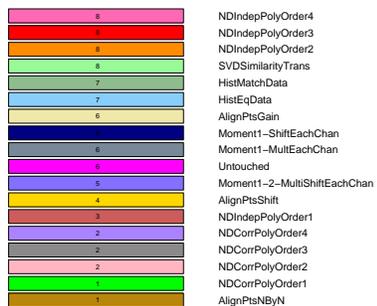
(a)



(b)

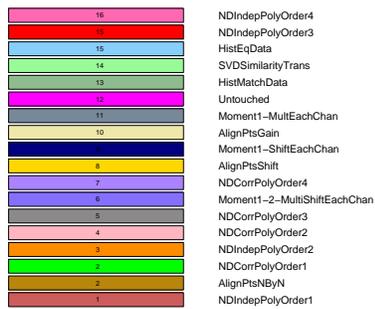


(c)

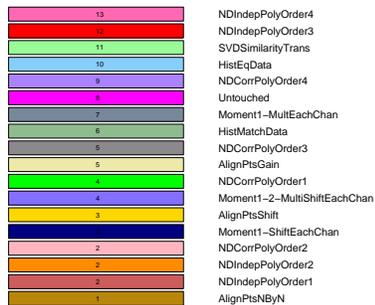


(d)

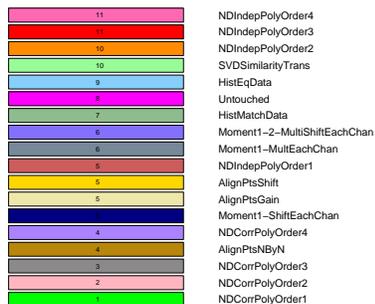
Figure 9.5: Ranked transformation methods for image pairs with 00(L-AL)(S) variation for: 1) Red-cyan paper 9.5(a), 2) Skittles 9.5(b), Teddy bears 9.5(c) and three paper strips 9.5(d).



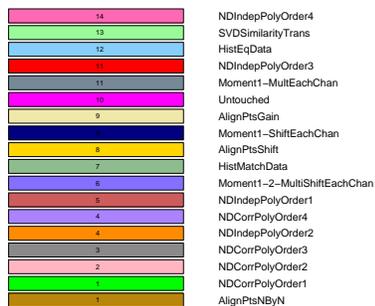
(a)



(b)

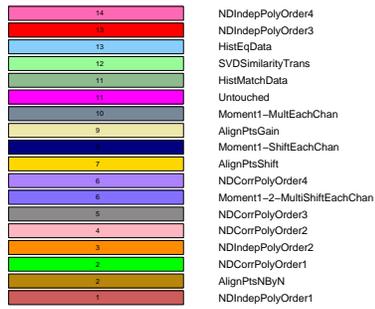


(c)

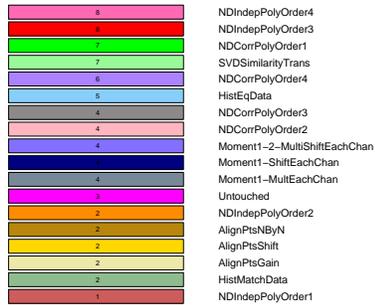


(d)

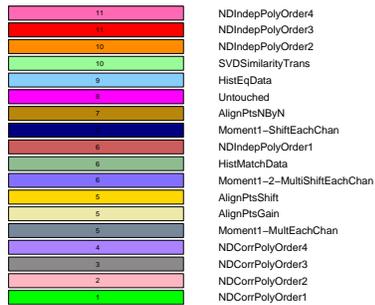
Figure 9.6: Ranked transformation methods for image pairs with (C)(L-LI)00 variation for: 1) Red-cyan paper 9.6(a), 2) Skittles 9.6(b), Teddy bears 9.6(c) and three paper strips 9.6(d).



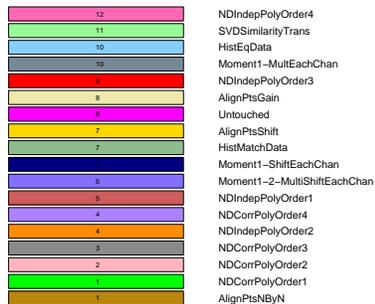
(a)



(b)

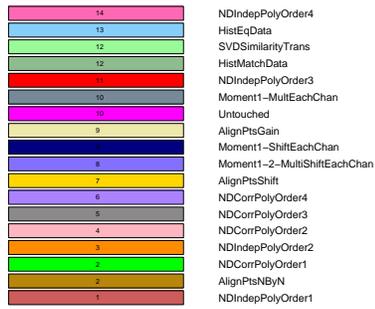


(c)

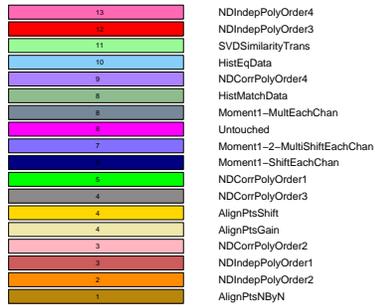


(d)

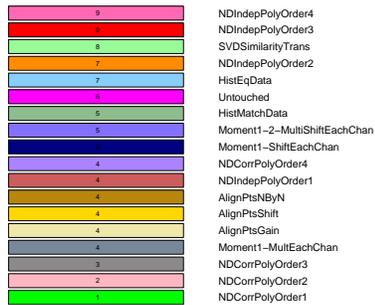
Figure 9.7: Ranked transformation methods for image pairs with (C)0(L-AL)0 variation for: 1) Red-cyan paper 9.7(a), 2) Skittles 9.7(b), Teddy bears 9.7(c) and three paper strips 9.7(d).



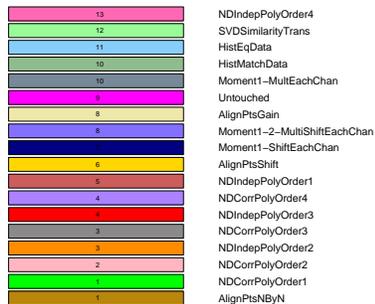
(a)



(b)

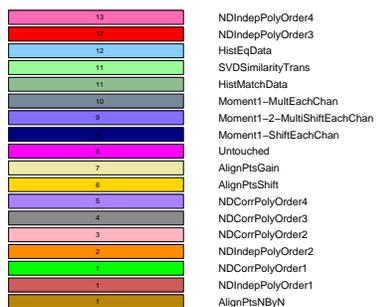


(c)

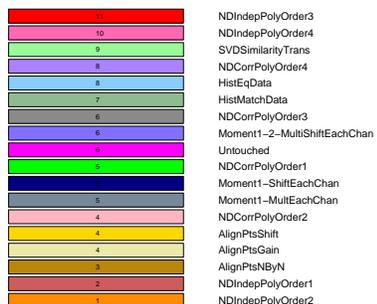


(d)

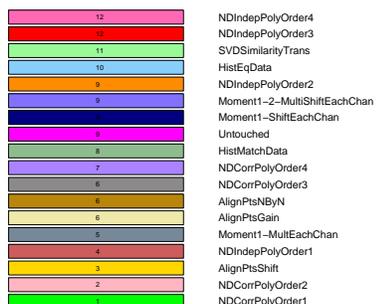
Figure 9.8: Ranked transformation methods for image pairs with (C)00(S) variation for: 1) Red-cyan paper 9.8(a), 2) Skittles 9.8(b), Teddy bears 9.8(c) and three paper strips 9.8(d).



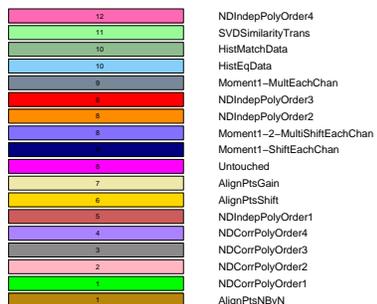
(a)



(b)

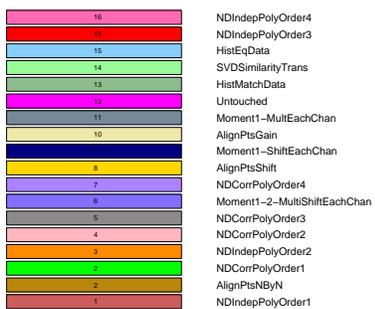


(c)

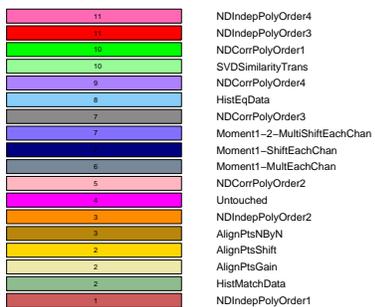


(d)

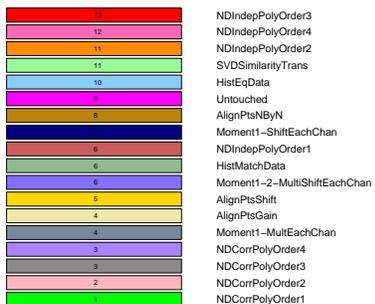
Figure 9.9: Ranked transformation methods for image pairs with 0(L-LI)0(S) variation for: 1) Red-cyan paper 9.9(a), 2) Skittles 9.9(b), Teddy bears 9.9(c) and three paper strips 9.9(d).



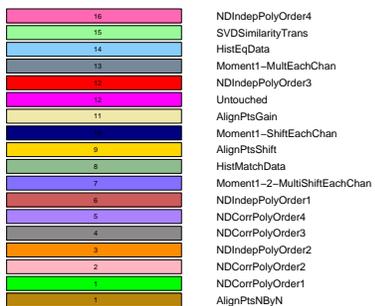
(a)



(b)

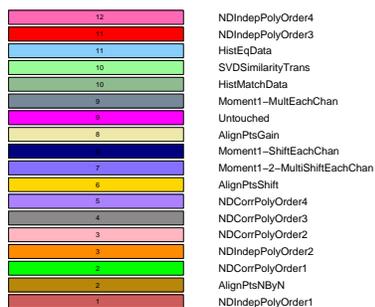


(c)

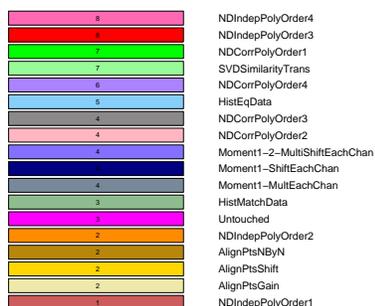


(d)

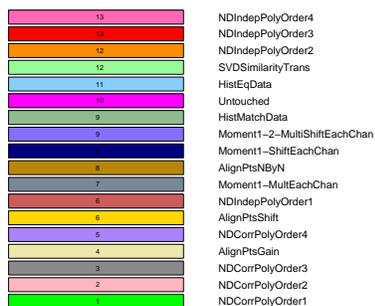
Figure 9.10: Ranked transformation methods for image pairs with (C)(L-LI)(L-AL)0 variation for: 1) Red-cyan paper 9.10(a), 2) Skittles 9.10(b), Teddy bears 9.10(c) and three paper strips 9.10(d).



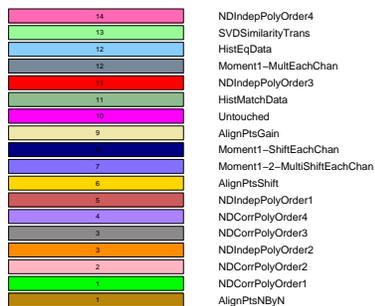
(a)



(b)

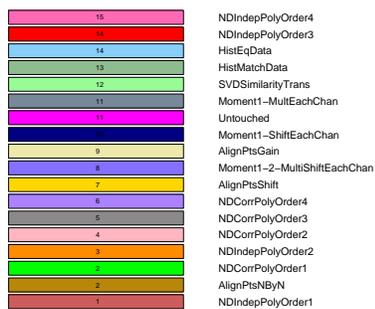


(c)

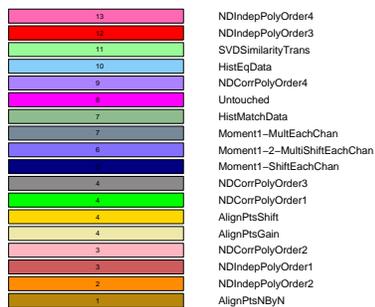


(d)

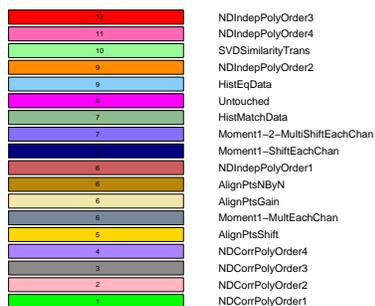
Figure 9.11: Ranked transformation methods for image pairs with (C)0(L-AL)(S) variation for: 1) Red-cyan paper 9.11(a), 2) Skittles 9.11(b), Teddy bears 9.11(c) and three paper strips 9.11(d).



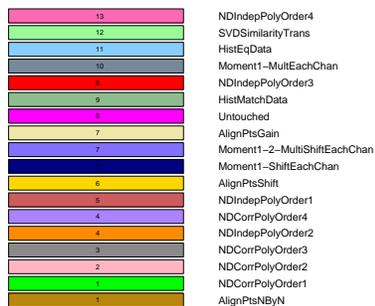
(a)



(b)



(c)



(d)

Figure 9.12: Ranked transformation methods for image pairs with (C)(L-LI)0(S) variation for: 1) Red-cyan paper 9.12(a), 2) Skittles 9.12(b), Teddy bears 9.12(c) and three paper strips 9.12(d).

Bibliography

- [1] T. Rofer. Region-based segmentation with ambiguous color classes and 2-d motion compensation. In *Lecture Notes in Artificial Intelligence. Robocup 2007*, pages 369–376, 2008.
- [2] GE Lighting. <http://www.gelighting.com/> (*last viewed, 2009*).
- [3] Wikipedia Creative Commons. http://en.wikipedia.org/wiki/file:schematic_diagram_of_the_human_eye_en.svg (*last viewed, 2009*).
- [4] Wikipedia Creative Commons. <http://commons.wikimedia.org/wiki/file:pinhole-camera.png> (*last viewed, 2009*).
- [5] Dalsa Imaging. Image sensor architectures for digital cinematography. www.dalsa.com/ (*last viewed, 2009*). In *Document 03-70-00218-01 Dalsa corporation*.
- [6] A.S. Glassner. *Principles of digital image synthesis. Vol. 2*. M. Kaufmann, 1995.
- [7] A. Gilbert and R. Bowden. Incremental Modelling of the Posterior Distribution of Objects for Inter and Intra Camera Tracking. In *In Proc. British Machine Vision Conference*, volume 1, pages 419–428, Oxford, UK, September 2005.
- [8] M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.
- [9] Buhler Sortex. <http://www.sortex.com> (*last viewed, 2009*).
- [10] E. Kress-Rogers and J.B. Brimelow. *Instrumentation and sensors for the food industry*. CRC Press/Woodhead Publishing Limited, 2001.
- [11] The RoboCup Federation. <http://www.robocup.org/> (*last viewed, 2009*).
- [12] M. Sridharan and P. Stone. Towards illumination invariance in the legged league. In *In The Eighth International RoboCup Symposium*, pages 196–208. Springer Verlag, 2004.
- [13] M. Sridharan and P. Stone. Towards Eliminating Manual Color Calibration at RoboCup. *Lecture notes in Computer Science*, 4020:673, 2006.
- [14] E.H. Adelson and J.R. Bergen. The plenoptic function and the elements of early vision. *Computational Models of Visual Processing*, 1, 1991.

- [15] I.S. Grant and W.R. Phillips. *The elements of physics*. Oxford University Press, 2001.
- [16] D.A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.
- [17] R.S Berns. *Billmeyer and Saltzman's Principles Of Color Technology*. Wiley-Interscience, 2000.
- [18] S.R. Marschner, S.H. Westin, E.P.F. Lafortune, K.E. Torrance, and D.P. Greenberg. Image-based BRDF measurement including human skin. In *Proceedings of the 10th Eurographics Workshop on Rendering*, pages 139–152, 1999.
- [19] A. Gibson, R.M. Yusof, H. Dehghani, J. Riley, N. Everdell, R. Richards, J.C. Hebden, M. Schweiger, S.R. Arridge, and D.T. Delpy. Optical tomography of a realistic neonatal head phantom. *Applied Optics*, 42(16):3109–3116.
- [20] J. Guild. The Colorimetric Properties of the Spectrum. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Math. or Phys. Character (1896-1934)*, 230(-1):149–187, 1932.
- [21] G. Wyszecki and W.S. Stiles. Color Science: Concepts and Methods, Quantitative Data and Formulae. *Color Science: Concepts and Methods, Quantitative Data and Formulae, 2nd Edition*, by Gunther Wyszecki, WS Stiles, page 968, 2000.
- [22] GERMAN TEAM national robocup entry. <http://www.robocup.de/germanteam> (last viewed, 2009).
- [23] B. Horn. *Robot Vision*. MIT Press, 1986.
- [24] R.G. Willson. *Modelling and Calibration of Automated Zoom Lenses*. PhD thesis, The Robotics Institute, Carnegie Mellon University, 1994.
- [25] D. Hasler and S. Susstrunk. Colour handling in panoramic photography. *Proc SPIE*, Jan, 2001.
- [26] J. Jia and C.K. Tang. Tensor voting for image correction by global and local intensity alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1):36–50, 2005.
- [27] D.B Goldman and J.H. Chen. Vignette and exposure calibration and compensation. In *The 10th IEEE International Conference on Computer Vision*, Oct. 2005.
- [28] D. Litwiller. CCD vs. CMOS. *Photonics Spectra*, 2001.
- [29] J. Janesick. Dueling Detectors. *OE Magazine*, 2(2), 2002.
- [30] G. Kamberova. Understanding the systematic and random errors in video sensor data. In *GRASP Lab., Department of Computer and Information Science, University of Pennsylvania, Tech. Rep., 1997. 4*, 1997.
- [31] B.E. Bayer. Color imaging array, July 20 1976. US Patent 3,971,065.

- [32] R. Kimmel. Demosaicing: image reconstruction from color CCD samples. *IEEE Transactions on Image Processing*, 8(9):1221–1228, 1999.
- [33] M.D. Fairchild, IS & T the Society for Imaging Science, and Technology. *Color appearance models*. Addison-Wesley Reading, Mass, 1998.
- [34] T.H. Goldsmith. What birds see. *Scientific American*, 2006.
- [35] P.W. Woods, C.J. Taylor, D.H. Cooper, and R.N. Dixon. The use of geometric and grey-level models for industrial inspection. *Pattern Recognition Letters*, 5(1):11–17, 1987.
- [36] Z. Yao, K. Sakai, X. Ye, T. Akita, Y. Iwabuchi, and Y. Hoshino. Airborne hyperspectral imaging for estimating acorn yield based on the PLS B-matrix calibration technique. *Ecological Informatics*, 2008.
- [37] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 232–237, 1998.
- [38] N. Sanders and C. Jaynes. Class-specific color camera calibration with application to object recognition. In *IEEE Workshop on the Applications of Computer Vision*, Breckenridge, CO, 2005.
- [39] W. Yu. Practical anti-vignetting methods for digital cameras. *IEEE Transactions on Consumer Electronics*, 50:975–983, 2004.
- [40] Y. Zheng, S. Lin, and S.B. Kang. Single-image vignetting correction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2006.
- [41] H. Beyer. *Geometric and radiometric analysis of a CCD-camera based photogrammetric close-range system*. PhD thesis, Institut für Geodäsie und Photogrammetrie, 1992.
- [42] G.E. Healey and R. Kondepudy. Radiometric ccd camera calibration and noise estimation. *PAMI*, 16(2):267–276, 1994.
- [43] G. Schaefer, S. Hordley, and G. Finlayson. A combined physical and statistical approach to colour constancy. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 1, 2005.
- [44] S.A. Shafer, Dept. of Computer Science, and University of Rochester. *Using Color to Separate Reflection Components*. Dept of Computer Science, University of Rochester, 1984.
- [45] E.H. Land and J.J. McCann. Lightness and Retinex Theory. *Journal of the Optical Society of America*, 61(1):1, 1971.
- [46] G. Ciocca, D. Marini, A. Rizzi, R. Schettini, and S. Zuffi. Retinex preprocessing of uncalibrated images for color-based image retrieval. *Journal of Electronic Imaging*, 12(1):161–172, 2003.

- [47] K. Barnard, V. Cardei, and B. Funt. A comparison of computational color constancy algorithms. I: Methodology and experiments with synthesized data. *Image Processing, IEEE Transactions on*, 11(9):972–984, 2002.
- [48] G.D. Finlayson, S.D Hordley, G. Schaefer, and G.Y Tian. Illuminant and device invariant colour using histogram equalisation. *Pattern Recognition*, 38(2):179–190, 2005.
- [49] J. V. Kries. *Chromatic Adaptation*, volume 1. Festschrift der Albrecht-Ludwigs-Universitat, 1902.
- [50] M. Schroder and S. Moser. Automatic color correction based on generic content-based image analysis. In *Ninth Color Imaging Conference*, pages 41–45, 2001.
- [51] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley. Color transfer between images. *Computer Graphics and Applications, IEEE*, 21(5):34–41, 2001.
- [52] D.L. Ruderman, T.W. Cronin, and C.C. Chiao. Statistics of cone responses to natural images: implications for visual coding. *Journal of the Optical Society of America A*, 15(8):2036–2045, 1998.
- [53] X. Xiao and L. Ma. Color transfer in correlated color space. In *VRCIA '06: Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications*, pages 305–309, New York, NY, USA, 2006. ACM Press.
- [54] F. Pitie, A.C. Kokaram, and R. Dahyot. N-Dimensional Probability Density Function Transfer and its Application to Colour Transfer. *ICCV'05 Volume 2*, pages 1434–1439, 2005.
- [55] Y. Tao. Closed-loop search method for on-line automatic calibration of multi-camera inspection systems. *Transaction of the American Society of Agricultural Engineers*, 41(5):1549–1555, 1998.
- [56] W. Li, M. Soto-Thompson, Y. Xiong, and H. Lange. A new image calibration technique for colposcopic images. *Proc. SPIE*, 6144:1973–1984.
- [57] A. Ilie and G. Welch. Ensuring color consistency across multiple cameras. In *The 10th IEEE Conference on Computer Vision (ICCV)*, Beijing, China, October 2005.
- [58] M. Jungel. Using layered color precision for a self-calibrating vision system. *The Eighth International RoboCup Symposium*, 2004.
- [59] A. Litvinov and Y. Schechner. Radiometric framework for image mosaicking. *Optical Society of America*, 22(5):839–848, 2005.
- [60] G.Y Tian, D. Gledhill, and D. Taylor. Colour correction for panoramic imaging. *Sixth International Conference on Information Visualisation*, pages 483–488, 2002.
- [61] K. Jeong and C. Jaynes. Object matching in disjoint cameras using a color transfer approach. *Special Issue of Machine Vision and Applications Journal*, 2006.

- [62] C. Fredembach, M. Schroder, S. Susstrunk, S. Lausanne, and S. Zurich. Region-Based Image Classification for Automatic Color Correction. *Proc. IS&T 11th Color Imaging Conference*, pages 59–65, 2003.
- [63] YV Haeghen, J. Naeyaert, I. Lemahieu, and W. Philips. An imaging system with calibrated color image acquisition for use in dermatology. *Medical Imaging, IEEE Transactions on*, 19(7):722–730, 2000.
- [64] Z. Lin, J. Wang, and K.K. Ma. Using eigencolor normalization for illumination-invariant color object recognition. *Pattern Recognition*, 35(11):2629–2642, 2002.
- [65] C.R. Senanayake and D.C. Alexander. Colour transfer by feature based histogram registration. In *British Machine Vision Conference*, pages 429–438, 2007.
- [66] J.J. Koenderink. *Solid shape*. MIT Press Cambridge, MA, USA, 1990.
- [67] D.G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, volume 2, pages 1150–1157. Kerkyra, Greece, 1999.
- [68] T. Lindeberg. Principles for automatic scale selection. *Handbook on Computer Vision and Applications*, 2:239–274, 1999.
- [69] JA Hartigan and MA Wong. A K-means clustering algorithm. *JR Stat. Soc. Ser. C-Appl. Stat.*, 28:100–108, 1979.
- [70] C.M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.
- [71] Wikipedia-Hungarian algorithm description. http://en.wikipedia.org/wiki/hungarian_algorithm (*last viewed, 2009*).
- [72] T.H. Gormen, C.E. Leiserson, R.L. Rivest, C. Stein, et al. Introduction to Algorithms. *MIT Press*, 44:651664, 1990.
- [73] UEA uncalibrated image database. <http://www2.cmp.uea.ac.uk/research/compvis/catsi.htm> (*last viewed, 2009*).
- [74] COIL database Surrey University. <http://www.ee.surrey.ac.uk/research/vssp/demos/colour/soil47/> (*last viewed, 2009*).
- [75] D. Koubaroulis, J. Matas, J. Kittler, and CTU CMP. Evaluating colour-based object recognition algorithms using the soil-47 database. In *Asian Conference on Computer Vision*, page 2, 2002.
- [76] SFU Database. <http://www.cs.sfu.ca/colour/data/> (*last viewed, 2009*).
- [77] B. Funt, K. Barnard, and L. Martin. Is machine colour constancy good enough? *Lecture notes in computer science*, pages 445–459, 1998.

- [78] F.J. Aherne, N. A. Thacker, and P. I. Rockett. The bhattacharyya metric as an absolute similarity measure for frequency code data. *Kybernetika*, 32(4):1–7, 1997.
- [79] D.J.C. MacKay. Information theory, inference, and learning algorithms. 2003.
- [80] S. Cohen and L. Guibas. The earth mover’s distance under transformation sets. *ICCV*, 1999.
- [81] K. Grauman and T. Darrell. The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2, 2005.
- [82] H. Ling and K. Okada. Diffusion Distance for Histogram Comparison. *CVPR*, 1:246–253, 2006.
- [83] B. Efron and R.J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1993.
- [84] G. Hamid. Personal communication from buhler sortex senior research engineer, 2005.
- [85] J. Klemelä. Visualization of Multivariate Density Estimates With Level Set Trees. *Journal of Computational & Graphical Statistics*, 13(3):599–620, 2004.
- [86] M. Hilaga, Y. Shinagawa, T. Kohmura, and T.L. Kunii. Topology matching for fully automatic similarity estimation of 3D shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 203–212. ACM New York, NY, USA, 2001.
- [87] D.W. Eggert, A.W. Fitzgibbon, and R.B. Fisher. Simultaneous registration of multiple range views for use in reverse engineering of CAD models. *Computer Vision Image Understanding*, 69(3):253–272, 1998.
- [88] S. Cunnington and AJ Stoddart. N-view point set registration: A comparison. *British Machine Vision Conference*, 1:234–244, 1999.
- [89] A. Fitzgibbon. Robust registration of 2d and 3d point sets. In *In Proc. British Machine Vision Conference, volume II, pages 411–420, Manchester, UK, September, 2001*.
- [90] A. Strehl and J. Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, 2003.
- [91] J. Ghosh, A. Strehl, and S. Merugu. A consensus framework for integrating distributed clusterings under limited knowledge sharing. In *Proc. NSF Workshop on Next Generation Data Mining, Baltimore*, pages 99–108, 2002.
- [92] P.F. Felzenszwalb and D.P. Huttenlocher. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.