UNIVERSITY OF TECHNOLOGY, SYDNEY

Surface-type Classification in Structured Planar Environments under Various Illumination and Imaging Conditions

by

Andrew Wing Keung To

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Engineering and IT
Electrical, Mechanical and Mechatronic Systems Group
Centre for Autonomous Systems

July 2015

Declaration of Authorship

I, Andrew Wing Keung To , declare that this thesis titled, 'Surface-type Classification in Structured Planar Environments under Various Illumination and Imaging Conditions' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

UNIVERSITY OF TECHNOLOGY, SYDNEY

Abstract

Faculty of Engineering and IT

Electrical, Mechanical and Mechatronic Systems Group

Doctor of Philosophy

by Andrew Wing Keung To

The recent advancement in sensing, computing and artificial intelligence, has led to the application of robots outside of the manufacturing factory and into field environments. In order for a field robot to operate intelligently and autonomously, the robot needs to build an environmental awareness, such as by classifying the different surface-types on a steel bridge structure. However, it is challenging to classify surface-types from images that are captured in a structurally complex environment under various illumination and imaging conditions. This is because colour and texture features extracted from these images can be inconsistent.

This thesis presents a surface-type classification approach to classify surface-types in a structurally complex three-dimensional (3D) environment under various illumination and imaging conditions. The approach proposes RGB-D sensing to provide each pixel in an image with additional depth information that is used by two developed algorithms. The first algorithm uses the RGB-D information along with a modified reflectance model to extract colour features for colour-based classification of surface-types. The second algorithm uses the depth information to calculate a probability map for the pixels being a specific surface-type. The probability map can identify the image regions that have a high probability of being accurately classified by a texture-based classifier.

A 3D grid-based map is generated to combine the results produced by colour-based classification and texture-based classification. It is suggested that a robot manipulator is used to position an RGB-D sensor package in the complex environments to capture the RGB-D images. In this way, the 3D position of each pixel is precisely known in a common global frame (robot base coordinate frame) and can be combined using a grid-based map to build up a rich awareness of the surrounding complex environment.

A case study is conducted in a laboratory environment using a six degree-of-freedom robot manipulator equipped with a RGB-D sensor package mounted to the end effector. The results show that the proposed surface-type classification approach provides an improved solution for vision-based classification of surface-types in a complex structural environment with various illumination and imaging conditions.

Acknowledgements

I would like to thank my supervisors Prof. Dikai Liu, and Dr Gavin Paul for their continual support and assistance throughout the course of my research. Your guidance and countless hours spent towards improving my research work has led to a more complete, quality thesis.

Thanks to the rest of the team at the Centre of Autonomous Systems, and Prof. Gamini Dissanayake, the head of the research centre, for providing an excellent environment that has facilitated great research interactions and exchange of ideas. Fellow research student Gibson Hu for providing encouragement and support throughout the course of the candidature.

I would finally like to thank my immediate family members Nelson To, Anita Luk, Anson To, grandparents and Ayesha Tang for believing in me to strive to do my very best.

This work is supported in part by the ARC Linkage Project: A robotic system for steel bridge maintenance, the Centre for Autonomous Systems (CAS), the NSW Roads and Maritime Services (RMS) and the University of Technology, Sydney (UTS).

Contents

List of Figures

List of Tables

Abbreviations

DOF	Depth of Field
FOV	Field of View
GLCM	Grey-Level Co-occurrence Matrices
IR	Infrared
PCA	Principal Component Analysis
RGB-D	Red, Green and Blue colour-space image with corresponding Depth image
SVM	Support Vector Machines
UTS	University of Technology, Sydney

Nomenclature

	General Formatting Style
$f(\cdots)$	A scalar valued function
$\mathbf{f}(\cdots)$	A vector valued function
$[\cdots]^T$	Transpose
$\lvert\cdot\rvert$	Absolute value
$\lVert\cdot\rVert$	Vector length and normalised vector
C	Covariance matrix
d	distance between two points
D	A diagonalised matrix
i	Index value in a list
n	Variable signifying the last index of a set or to refer to a count
P	Probability
(u, v)	Index values in a 2D array or an image
$\vec{v}$	A vector
Ω	2D image
τ	Threshold
θ	Angle between two directional vectors
	Specific Symbol Usage
${}^oT_e(\vec{Q})$	Homogenous transformation between the robot base coordinate frame and the end-effector at pose $\vec{Q}$
${}^oT_s$	Homogenous transformation matrix between the robot base coordinate frame and the sensor coordinate frame

${}^{e}T_{s}$	Homogenous transformation matrix between the robot end-effector coordinate frame and the sensor coordinate frame
${}^{s}T_{c}$	Homogenous transformation matrix between the sensor coordinate frame and the camera coordinate frame
C	Principle point of a pinhole camera model
d_c	Surface point-to-RGB camera coordinate origin distance
d_l	Light source-to-surface point distance
d_p	Plane of focus to camera distance
d_t	The viewing distance used to capture the training image dataset
F	Focal length of a camera
I_l	Light source intensity value
I_r	Reflected light source intensity value
K_d	Set notation for the diffused reflectance values
K_s	Set notation for the specular reflectance values
$\vec{l}$	The position vector of the light source relative to the RGB camera coordinate frame
n_t	Number of surface-types
P	A vector (or set) of 3D points or vertices
q	Robot manipulator's joint angle
$\vec{Q}$	Robot manipulator's joint angle vector, $[q_1, q_2, ...q_n]^T$
α	Bisector angle between $\vec{v_c}$ and $\vec{v_l}$
β_g	Gaussian blur coefficient
β_k	Skewing coefficient
β_s	Scaling coefficient
δ	Lens f-number
μ	Length of the voxel cube in the surface-type map
ω_v	Weighting factor applied to a voxel containing texture-based classification results
Ω_s	Depth image from the IR camera
Ω_c	Greyscale calibration image used to calculated the light source position

Ω_{cs}	An image of the greyscale calibration image Ω_c containing the specular reflectance region
Ω_{cd}	An image of the greyscale calibration image Ω_c containing the diffused reflectance region
Ω_t	A simulated texture pattern image
φ	Circle of confusion
σ	Surface roughness albedo
θ_c	Angle of incidence between the normal of a 3D surface point and the straight line between the surface point and the RGB camera coordinate origin
θ_l	Angle of incidence between the normal of a 3D surface point and the straight line between the surface point and the light source
θ_t	The viewing angle used to capture the training image dataset
τ_s	Pixel intensity threshold for identifying the specular reflectance region in an image
τ_d	Pixel intensity threshold for identifying the diffused reflectance region in an image
$\vec{v_\eta}$	Normal vector of a 3D surface point
$\vec{v_c}$	Direction vector between the surface point and the RGB camera coordinate origin
$\vec{v_l}$	Direction vector between the light source point and the RGB camera coordinate origin
$P(M_k)$	Discrete probability distribution of the surface-types for $k \in \{1, \dots n_t\}$, given n_t number of surface-types.
$P(M_k\|E)$	Probability of surface-type state given the evidence E
$P(E\|M_k)$	Probability of an evidence given the surface-type
$P(E)$	Probability of evidence
P_{d_c}	Probability value of a pixel being a surface-type based on viewing distance
P_{θ_c}	Probability value of a pixel being a surface-type based on viewing angle

P_{d_c,θ_c}	Probability value of a pixel being a surface-type based on viewing distance and viewing angle

Combinations of Variables

(a_2, a_1, a_0)	Polynomial coefficients for camera radiometric response in the reflectance model
$\{D_{n_1}, D_{f_1}\}$	Depth of field threshold range
$\{D_{n_2}, D_{f_2}\}$	Spatial resolution threshold range
$(K_{d,R}, K_{d,G}, K_{d,B})$	Diffused reflectance value for each RGB colour channel
$(K_{s,R}, K_{s,G}, K_{s,B})$	Specular reflectance value for each RGB colour channel
(x_c, y_c, z_c)	Axes of RGB camera's 3D Cartesian coordinate frame
(x_o, y_o, z_o)	Axes of Robot base's 3D Cartesian coordinate frame
(x_e, y_e, z_e)	Axes of End-effector's 3D Cartesian coordinate frame
(x_s, y_s, z_s)	Axes of Depth sensor's 3D Cartesian coordinate frame
$(\tau_n, \tau_f, \tau_\theta)$	Threshold parameters to calculate an image pixel's probability of being a surface-type
(ω_1, ω_2)	Weighting coefficients to calculate an image pixel's probability of being a surface-type

Glossary of Terms

Complex environment	A 3D workspace that has multiple planar surfaces arranged in various positions and orientations.
Confusion matrix	A specific table that allows the visualisation of classification results. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class.
Environmental awareness	In the context for a robot this can include but is not limited to the knowledge of, a geometric map of the environment that describes the location of surfaces and obstacles, and a semantic map that provides a label for objects, surface-types and locations within the environment.
Grid	A type of representation based on occupancy grids used to divide a space into discrete grid cells. For surface-type map in 3D this becomes voxels.
Grit-blasting	The abrasive removal of surface rust and/or paint using a high pressure grit stream.
Surface-type map	Model of the geometry and surface-type of surfaces in the environment.
RGB-D	The combination of a colour image represented in the RGB colour-space (red, green, blue) with the addition of depth data that corresponds with each colour image pixel.

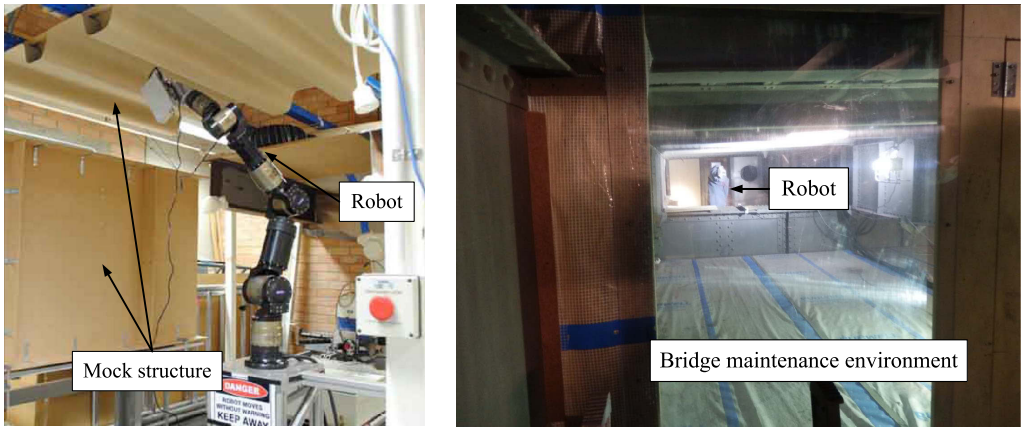
Robot manipulator	In this thesis, this is a six-degree of freedom Denso industrial robotic manipulator, with a RGB-D sensor tool mounted on the end-effector.
Sensor package	Generally refers to an IR-based depth sensing camera, a colour camera and a light source.
Surface	The face of an object/structure in the environment.
Surface normal	A 3D vector perpendicular to a surface.
Surface-type	The appearance of a surface described by the colour and texture.
Textural appearance	The visual appearance of a surface that can be changed by the image capture conditions.
Viewpoint	A position in space and an orientation of a sensor that results from a manipulator pose $\vec{Q}$. This can also be expressed in terms of the homogeneous transformation matrix, ${}^{0}T_{s}(\vec{Q})$
Voxel	Volumetric Pixel which represents a 3D cube-like volume in Euclidean space.

Chapter 1

Introduction

The application of autonomous robotic systems beyond the structured factory settings and into field environments is beginning to become a reality. Field robotic systems that can operate in complex structural environments can aid humans in performing tasks, often undertaking them with improved efficiency and with reduced risks. Unlike a human, a robotic system can operate without fatigue, while maintaining a more consistent level of accuracy and repeatability for prolonged periods.

The traditional industrial factory setting has provided a structured and predictable operating environment suitable for configuring robots to perform repetitive tasks. In contrast, a field environment can dynamically change over time; the environment around the robot can also be changed due to movements made by the robot. As a result, pre-configured robot routines are generally unusable for a field environment. Therefore, a field robot must have the ability to explore and sense the environment to generate and maintain an up-to-date environmental awareness. One such application that is considered in this research is autonomous grit-blasting robots for complex steel bridge structure maintenance (Figure 1.2*b*). In this application, a mobile industrial robot manipulator needs to autonomously conduct grit-blasting to remove old paint from steel surfaces in a steel bridge structural environment. To conduct this operation, besides geometric information of the environment, the robot needs to be aware of the surface-types (e.g. painted metal surface, blasted metal surface, rust, timber scaffolding,



(a) (b)

FIGURE 1.1: *a*) Mock robotic inspection setup in a laboratory; *b*) Actual bridge maintenance environment

etc). This thesis focuses on approaches and algorithms needed to accomplish surface-type classification in a complex structural three-dimensional (3D) environment under various illumination conditions. Varying illumination and imaging conditions can be a result of the light source being repositioned in the environment, or due to surfaces being at different distances and angles. This thesis presents: (a) a surface-type classification approach; (b) a colour feature extraction algorithm using Red, Green, Blue-Depth (RGB-D) images and a modified reflectance model; and (c) an algorithm to calculate a probability map of surface-types in an image.

Practical experiments are conducted to verify the algorithms using a sensor package consisting of an RGB image camera and a depth camera to capture the RGB-D images. The RGB-D sensor package is mounted to the end-effector of a robot manipulator, where the sensor-to-robot base position can be calculated using a hand-eye transformation. The outcomes of this thesis, including the surface-type classification approach and algorithms, are evaluated on a mock inspection setup in a laboratory (Figure 1.1*a*).

This chapter provides an introduction to the research work and the background to the practical application (Figure 1.1*b*), from which the challenges and problems are identified. This chapter also presents the motivation for the research, the scope of the research work, the main contributions, the relevant publications, and an outline of the subsequent chapters in this thesis.

1.1 Background

Steel bridges are an important component of the modern transportation system, with over 30,000 road and rail bridges in use across Australia and more than 400,000 steel bridges in the United States and Europe. Bridge maintenance and replacement form a large part of the expenditure associated with traffic infrastructure management. Statistically, Australia spends $200 million each year on bridge maintenance, with $19 million spent on maintaining and improving the iconic Sydney Harbour Bridge [16].

The most common cause of steel bridge failure is deterioration due to corrosion [17], which can significantly reduce the expected lifespan of the structure. To prolong the lifespan of a steel bridge, rust removal maintenance via grit-blasting and repainting is performed regularly. The process of grit-blasting involves accelerating abrasive media (grit) through a blasting nozzle by means of compressed air in order to generate an abrasive grit stream that can be used to treat surfaces such as to remove rust, prepare the surface for painting, remove burrs or rough edges, etc.

Grit-blasting can produce large reaction forces over 100 N [18], which causes operators to fatigue quickly. Operator fatigue not merely results in reduced productivity, but increases the risk of blast stream related injury or fatality. Long-term grit-blasting exposes the operators to hazardous contaminated particles as paint used on steel bridges can contain lead, mercury and asbestos particles. To this due to these particles become airborne and can be aspirated or ingested, leading to diseases including asbestosis, silicosis, lead poisoning, and mercury poisoning. Although respiratory and blast protective precautions are enforced in Australian workplaces, the most effective way of mitigating the risks is to remove the operator(s) from the hazardous environment. Therefore, research into a bridge maintenance robotic system is being conducted to address this mitigation strategy. The primary goal of the robotic system is to remove the operators from the hazardous task of grit-blasting and into a safer role of high-level control and decision-making. Secondary outcomes will be to increase productivity.

A sealed containment area is typically established to provide a stable platform for maintenance work. Figure 1.2*a* shows a containment area established for the

maintenance of the under deck structure of a steel bridge. As part of the research conducted in [6][19], a mobile robotic system capable of autonomously grit-blasting the steel surfaces on a bridge structure is developed. Figure 1.2*b* shows the deployment of the mobile robotic system in a containment area. Currently, the robotic system is capable of building a 3D geometric map of the surrounding surfaces, which has enabled automated path-planning and grit-blasting of the identified surfaces. However, a human inspector is still required to visually inspect the finish quality of a blasted surface to determine whether it is of acceptable quality or if re-blasting is necessary.

Therefore, it is desirable to implement a machine vision system to automate the surface inspection process such that a human inspector is not required. In order to efficiently inspect all the surfaces in the surroundings, a machine vision system can be mounted to the end-effector of a robot manipulator such that it is possible to position the imaging device(s) at different viewpoints to capture images of the different surfaces. But when a machine vision system is mounted to the end-effector of a robot to perform surface-type (classified based on the colour and/or the texture) inspection of a bridge, the following challenges need to be addressed: (a) the inspection environment can be structurally complex, containing multiple surface planes where one observation viewpoint will not be able to inspect all the surfaces; (b) movement limitations of a robot manipulator may make it difficult to position sensors to capture an image of a single surface plane in the environment; (c) varying illumination conditions in the environment may produce images of a surface with non-uniform illumination; and (d) the surface inspection process will need to be performed in an online manner such that the results can be used by the robotic system to make decisions during the grit-blasting process. Hence, a surface-type classification approach needs to be devised that can address these challenges so that a mobile robotic system can be used to inspect surfaces on a steel bridge.

(a) (b)

FIGURE 1.2: *a*) A sealed containment area established for bridge maintenance; *b*) A mobile robotic system deployed for steel bridge maintenance

1.2 Motivation

In order for a robotic system to conduct grit-blasting in a complex structural bridge environment, the robotic system must be capable of building an up-to-date environmental awareness, including not only a geometric map but also surface-types (e.g. to identify areas that should not be blasted). To date, research into geometric map building of a steel bridge using a six degree-of-freedom robot manipulator with a range sensor mounted on the end-effector has been successfully conducted [20]. The geometric map has enabled the robotic system to autonomously grit-blast surfaces in a safe and collision-free manner, albeit indiscriminately.

From the safety and correct grit-blasting operation standpoint, surface-type awareness is necessary. An awareness of the surfaces-types will enable the robotic system to grit-blast only the required surface areas, evaluate the grit-blast finish quality, and perform additional touch-up grit-blasting work upon evaluation. However, an outstanding challenge is to provide the robotic system with the capability of non-intrusively collecting surface data, and classifying the different surface-types in the environment.

Recent research in vision-based classification approaches have demonstrated improved robustness towards image inconsistencies caused by illumination, sensor noise, sensor

movement and position. However, machine vision and classification approaches for surface-types have not been demonstrated for applications in complex structural environments where illumination conditions can vary, such as on a steel bridge structure. Firstly, data acquisition in a complex structural environment can be constrained and limited because the movement of the robot manipulator can be restricted by the environment. Secondly, the field environment generates additional data inconsistency due to vibration, dust, and complex 3D structures. A review of the literature (see Chapter 2) reveals that the current algorithms and classification approaches have not been demonstrated for such field applications. It is the purpose of this thesis to perform research into the approaches and algorithms that can enable a robotic system to autonomously acquire, classify and map the surface-types in a field environment.

1.3 Scope

This research aims to develop a surface classification approach that facilitates the non-intrusive sensing and classification of surface-types in complex steel bridge structural environments by a robotic system. Research work will be conducted to develop an algorithm to improve colour feature extraction from images with non-uniform illuminations and an algorithm to assess the classification results of images produced using a texture-based classifier. Ihe research also covers the implementation of a multi-modal RGB-D sensor package (RGB camera and depth camera) to a robot manipulator to acquire data for classification. Overall, the scope of this work shall enable a robotic system to classify surface-types in a complex structural environment under various illumination conditions.

The following assumptions are made for this research: (a) it is assumed that the environment remains static for the duration of data acquisition and all light source in the environment are known and controllable (not outdoors daylight conditions); (b) the multi-modal sensor package can be rigidly mounted and calibration can be performed to establish the transform between the sensors, and the hand-eye transform between the sensor package and a robot manipulator; and (c) the target field environment consists of planar and Lambertian surfaces.

In the case study, a robotic bridge maintenance scenario is used. Because the grit-blasting diameter is assumed to be approximately 200 mm, the spatial resolution of the surface-type classification approach needs to meet the criteria of being equal to or finer than the aforementioned blast diameter. The detailed configuration and setup of the robot platform and additional hardware used in the case study are provided in Chapter 6.

1.4 Contributions

The theoretical contributions of this thesis are:

- A surface-type classification approach. The approach is composed of two algorithms to address the change in illumination and textural appearance of surface-types.
- An algorithm for the extraction of colour features using a modified reflectance model and RGB-D images. This algorithm can be applied to images of surfaces that are illuminated by a light source positioned at various viewing distances and angles.
- An algorithm to assess the classification results produced by a classifier trained with texture features. A probability map is generated for each pixel to show the probability of being a specific surface-type. This algorithm can be applied to images captured under various image capture conditions (viewing distance, viewing angle, camera optical settings) where the textural appearance of a surface-type can change from image to image.

The practical contribution of this research work is:

- A software package written in Matlab for surface-type classification and result visualisation is developed as part of an industry funded project to grit-blast a steel bridge using robots. The package is intended to be used onsite during robotic bridge maintenance work.

1.5 Publications

1.5.1 Journal Papers

1. A. W. K. To, G. Paul, and D. K. Liu. Surface-type Classification Using RGB-D. IEEE Transactions on Automation Science and Engineering, Vol.11, pages 359-366, 2013.

2. A. W. K. To., G. Paul., D. K. Liu., An Approach to Identifying Classifiable Regions of an Image Captured by Autonomous Robots in Structural Environments. Robotics and Computer Integrated Manufacturing, (Accepted, 2015)

3. A. W. K. To, G. Paul, N. M. Kwok, and D. K. Liu. An effective trajectory planning approach for autonomous robots in complex bridge environments. International Journal of Computer Aided Engineering and Technology, Vol.1, No. 2, pages 185-208, 2009.

1.5.2 Conference Papers

4. A. W. K. To, G. Paul, D. Ruston-Smith, D. K. Liu, and G. Dissanayake. Automated and Frequent Calibration of a Robot Manipulator-mounted IR Range Camera for Steel Bridge Maintenance, International Conference on Field and Service Robotics (FSR'12), July 16-19, Matsushima, Japan, pages 1-14, 2012.

5. A. W. K. To, G. Paul, and D. K. Liu. Image Segmentation for Surface Material-type Classification using 3D Geometry Information. International Conference on Information and Automation (ICIA'10), June 20-23, Harbin, China, pages 1717-1722, 2010.

6. A. W. K. To, G. Paul, N. M. Kwok, and D. K. Liu. An integrated approach to planning for autonomous grit-blasting robot complex bridge environments. Intelligent Production Machines and Systems (I*PROMS), UK, 2008

7. D. Rushton-Smith, A. W. K. To, G. Paul, and D. K. Liu. An Accurate and Reliable Approach to Calibration of a Robot Manipulator-Mounted IR Range Camera For Field Applications. International Symposium on Robotics and Mechatronics, Singapore, 2013.

1.6 Thesis Outline

Chapter 2 presents a review of research work that is relevant to the development of the proposed approach and algorithms. The topics reviewed include the different types of environmental awareness, the technologies and approaches that can be used to inspect a surface, and the vision-based approaches for the classification of surface-types using colour and texture features.

Chapter 3 presents the proposed surface-type classification approach for complex structural environments under various illumination and imaging conditions. This chapter details the processes used in the approach to precisely position an RGB-D sensor package with a robot manipulator, to calculate the viewing distance and viewing angle for each pixel in the captured RGB-D image, and to build a surface-type map in 3D from the classified images.

Chapter 4 presents an algorithm for the extraction of colour feature using RGB-D images and a modified reflectance model. A reflectance model using a single light source is presented and the process to extract three reflectance-based features from each colour channel (RGB) of a pixel is provided. A classifier is trained using the extracted reflectance-based colour features to classify surface-types in images with non-uniform illumination. Experiments are conducted to compare the classification performance of the reflectance-based colour features against other colour-space features including RGB and the CIELab colour-space component a*b*.

Chapter 5 presents an algorithm to generate a probability map that can be used to assess the classification results produced by a texture-based classifier. An analysis is performed to show the effect of image capture condition (viewing distance, viewing angle and camera optical settings) on texture features extracted from an image in terms of image qualities (focus quality, spatial resolution, perspective distortion). From the analysis, capture condition thresholds are defined and are used in a function to calculate a pixel's probability of being a surface-type. The process of using the function to generate a probability map for the pixels in an image is detailed. Experiments are conducted to verify the effect of image capture conditions on (focus quality, spatial

resolution and perspective distortion) and consequently on the accuracy of surface-type classification results. An experiment is also conducted using a probability map to identify regions in an image that have a high probability of being accurately classified.

Chapter 6 presents a case study that combines the colour-based and texture-based classification results to produce a surface-type map in 3D. The case study is performed using an RGB-D sensor package mounted to the end-effector of a six degree-of-freedom robot manipulator. The specifications of the RGB-D sensor package and robot manipulator used in the case study are provided, and the process to calibrate the RGB-D sensors and the transformations between the different coordinate frames are also detailed. Furthermore, the process to evaluate the accuracy of the classification results using a benchmark surface-type map is presented. The experiments are conducted using classifiers trained with colour-based and texture-based features extracted from a training RGB-D dataset. Surface-type classification is performed on images collected by the RGB-D sensor package positioned by the robot manipulator using different movement strategies.

Chapter 7 presents the conclusion to the thesis, discussion on limitations of the proposed surface-type classification approach, and future work.

Chapter 2

Review of Related Work

This chapter reviews the research works that are relevant to the development of concepts and methods required to address the research challenges associated with surface-type classification by a robotic system in a complex structural environment. The reviewed work includes the different types of environmental awareness, the technologies and approaches that are used to inspect a surface, and the vision-based approach for classifying surface-types based on colour and texture features.

2.1 Environmental Awareness

Gaining an environmental awareness in a context that is useful towards the task at hand is important. Environmental awareness for an autonomous robot can include, but is not limited to, a geometric map of the environment that describes the location of surfaces and obstacles, and a semantic map that provides a label for objects, surface-types and locations within the environment. Various approaches and technologies have been developed to build geometric and/or semantic awareness of an environment in the form of maps.

A top-down 2D geometric map of the environment can be used to provide the information necessary for robot path/motion planning and navigation within a 2D workspace. The geometric map consists of grid cells forming a discrete topological map; each cell can in the state of unknown, occupied and unoccupied. In an ideal case, 2D geometric mapping

can be achieved using a depth sensor and an odometer mounted onto a mobile platform. A dead reckoning approach [21] using odometry data can track the movements of the mobile platform from which the collected depth data at different positions are combined together to build a 2D geometric map. However, noise-free odometry is generally impossible for a real-world scenario and a dead reckoning approach will result in a 2D geometric map with accumulating offset errors [21]. As such, various approaches have been devised to generate an accurate 2D map for a real-world scenario. This work in [22] demonstrated 2D geometric mapping for an enclosed area such as a factory or storehouse using a 2D laser range finder mounted on a conveyance car. The position of the car was estimated by detecting the line-segments of the walls to calculate the orientation and distance of the car from the walls. For an environment that has curved surfaces, two algorithms were proposed in [23] to register consecutively collected depth data together to compute the relative position change of the depth sensor. The first algorithm computed the relative position change between two depth scans by matching tangent lines using distance minimisation, and the second algorithm solved a point-to-point least squares problem to compute the relative position change. In summary, it has been shown that there are approaches available to generate 2D geometric maps that are accurate for real-world applications with the ability to correct offset errors.

In order to perform tasks in 3D space, a 3D geometry map is necessary. Three dimensional geometry mapping has been demonstrated from a static platform using a 2D laser range finder mounted onto the end-effector of a robot manipulator [7]. The robot manipulator provides a way to accurately combine the depth data into a common reference frame using kinematics and a hand-eye transform between the end-effector and the laser range finder. A 3D geometry map of a bridge structural environment has been generated by sweeping a laser range finder to scan the environment using a robot manipulator. For 3D geometry mapping on a mobile platform, [24] has demonstrated the use of a vertically sweeping laser range finder mounted onto a Segway robot mobility platform (RMP). The depth data from different scans were combined together using a probabilistic registration approach that incorporated a generic surface probability prior, to guide the optimisation process towards maps that closely resemble the real environment. A 3D geometry map of an office environment had been produced using this approach. However, mapping using

a 2D laser range finder has been shown to be time-consuming and sensitive to vibration; therefore, alternative depth sensors have also been investigated to build a 3D geometry map.

Stereo vision and depth camera are alternative depth sensor technologies that can be used to build a 3D geometric map. In stereo vision, depth is calculated by identifying the disparity between matching features in two simultaneously captured images from different viewing perspectives. Stereo vision has been demonstrated to detect the distance of objects in the environment for localisation [25], identify an object for grasping [26], and generate a 3D colour map as shown in Figure 2.1 for visualisation [1]. However, a limitation of stereo vision is the need for texture disparity in captured images to calculate depth. Therefore, plain environments with low texture disparity, i.e. low illumination and plain surfaces, do not perform well with stereo vision. To use stereo vision in a plain environment, pattern projection approaches[1] [27] have been suggested to generate artificial features in the environment. Alternatively, a single camera setup with an onboard projected light-source can also be used to build a 3D geometric map in an environment that has low texture disparity. The Time of Flight (ToF) based depth camera [28] which measures depth by projecting a flash of LIDAR into the scene and calculates the return time of the light has been in use by the robot community since the 90s. However, commercially available ToF camera tend to be low resolution, and costly. The recent availability of an RGB-D camera that combines an RGB and IR camera has provided a cost-effective and lightweight solution to performing 3D geometric mapping with additional colour information (3D colour map). The use of an RGB-D camera to generate a 3D colour map in real-time has been shown to be possible in [29]. In this approach, on-the-fly reconstruction of a static indoor environment was achieved by implementing an iterative closest point (ICP) algorithm using the graphics processing unit (GPU). Robustness in combining the depth data was improved by using both geometric and photometric information when performing the nearest neighbour search. In terms of precision differences between stereo vision and the IR depth camera, [30] has conducted a comparison between a Microsoft Kinect (IR depth camera) and the Bumblebee2 camera (stereo vision). The results showed that due to the higher resolution of the Bumblebee2, it has better localisation performances over the Kinect. However, studies are still required to test the sensors in an industrial setting

under unideal capture conditions to further identify the limitations of each technology. Nevertheless, the availability of these depth sensors that are capable of capturing colour such that a 3D colour map can be generated has improved semantic labelling in 3D.

FIGURE 2.1: 3D geometric map with additional colour information [1]

Semantic labelling of surfaces and objects within a geometric map can improve decision-making for tasks. For example, semantic labelling of locations (e.g. bedroom, kitchen) can improve human-centric tasks beyond what is capable with only map coordinates. Various approaches have been devised that use image data, depth data and a combination of both to perform semantic labelling of a geometric map. In [31], an approach has demonstrated the use of image data to semantically label different areas of a 2D geometric map. The image data was integrated into a Conditional Random Field (CRF) that also considered higher level semantics such as adjacency and place boundaries to provide location labelling. This approach was shown to be able to achieve a high percentage of correct labelling of locations even when using images taken from the Internet. Furthermore, [32] has demonstrated the use of image data for unsupervised semantic labelling of indoor scenes. In this approach, Scale-Invariant Feature Transform (SIFT) [33] and Generalised Search Tree (GiST) features [34] were extracted from images to generate Visual Words (VW) [35]. Context-based features were extracted from Bags of VW using two-dimensional histograms. Experimental results showed that the context-based features can be successfully used to label different locations such as office, corridors, etc. In the case of semantically labelling objects within an environment, [32]

has demonstrated the use of SIFT features extracted from depth data. Stereo vision was used to provide the depth data from which the objects and their orientation in space can be labelled. The approach has been practically demonstrated on an industrial robot to perform object grasping and alignment operations. In addition, with the recent advancements in RGB-D sensor technology that can achieve high-resolution image and depth data capture at high frame rates, improved approache to semantically label objects using RGB-D information have been devised. RGB-D information can be used to provide both the geometric shape and colour information of an object for semantic labelling. The work in [2] demonstrated an RGB-D based approach that combined multiple observations of a scene to semantically label multiple object types. In this approach, image-based object detection was applied to the image data, where each pixel was assigned class probabilities of being a certain object class-type. The class probabilities of each pixel were projected into a voxel-based 3D geometric map where a Markov Random Field that combines the class probabilities with 3D shape cues was used to infer and label objects. This approach was able to accurately label multiple objects including a bowl, a cap, a cereal box, etc. Figure 2.2 shows examples of the 3D colour maps with labelled objects of different class-types from this approach. The geometric shape of an object is a good descriptor for semantically labelling objects into different class-types, but is ineffective for labelling surface-types.

Overall, the reviewed approaches and technologies have provided a generalised concept of environmental awareness in terms of geometric mapping and semantic labelling. In order to semantically label surface-types and surface conditions in high-resolution, a further review of the surface sensing approaches and sensor technologies is needed.

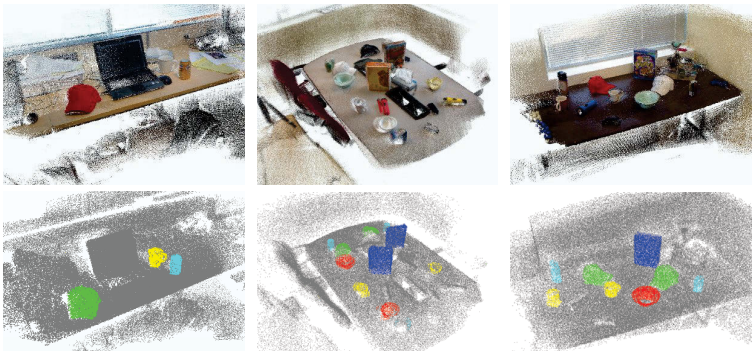


FIGURE 2.2: 3D scene labelling results for three complex scenes, where: bowl is red, cap is green, cereal box is blue, coffee mug is yellow and soda can is cyan [2]

2.2 Sensor Technologies and Sensing Approaches Used for Surface Inspection

Surface inspection is performed to assess for surface conditions such as defects [4] [36] [37], deterioration [13] and finish quality [38]. Given the differences among surface inspection tasks, a range of sensors and sensing approaches are available [39]. The following is a review of the sensors and sensing approaches that have been demonstrated to provide non-destructive surface inspection in different industrial applications.

In applications that require a high degree of precision and accuracy (i.e. sub-millimetre), commercial contact-based sensors are available for use [3]. The following is a summary of the different sensors and sensing approaches that are shown Figure 2.3. A chemical kit can be used for the detection of surface contaminants including: chlorides, sulphates, nitrates, ferrous ion, etc. To detect for contaminants, testing samples are extracted from the surface and measured using an analogue sensor such as pH paper, or digitally using a digital salt detector that measures conductivity. For surface profile measurements, various inspection approaches have been developed, such as surface profile comparators with a magnifier for sub-millimetre surface profile measurement. Additionally, surface profile can also be measured using replica profile tape (Appendix D illustrates the use of profile tape to extract surface profile samples). Once a surface sample is extracted onto a replica profile tape, a texture gauge can be used to measure the coarseness of the surface profile with accuracy in the order of microns. For the measurement of surface coating thickness that is

not directly observable from the exterior, ferrous and non-ferrous thickness measurement gauges can be used. In both types of gauges, a sensing probe is placed in contact with the surface and a thickness reading is provided with accuracy of up to ± 2.5 micron. Overall, from the reviewed contact-based sensors and approaches, it is shown that it is possible to perform surface inspection with precise measurements in the order of microns. However, these sensors and sensing approaches can only measure a small surface area at a time, thus impractical for the full inspection of all surfaces in an area such as a bridge structure. It will be time-consuming to set up and costly to perform over a large inspection area. Therefore, alternative sensors and sensing approaches have been investigated for surface inspection over a large area.

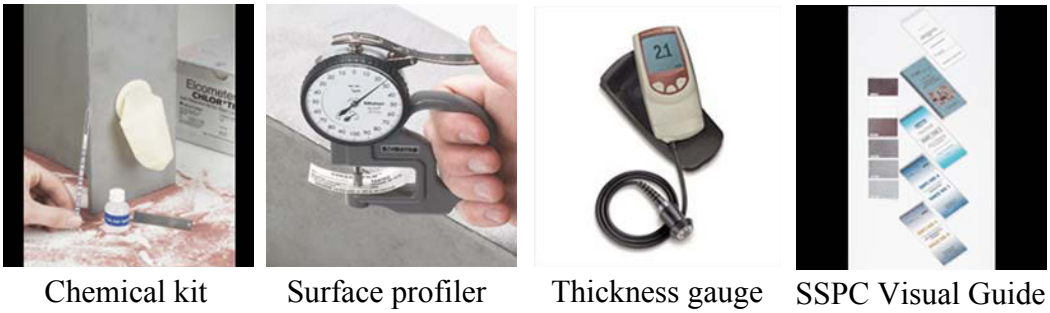


FIGURE 2.3: Commercial surface inspection instruments [3], please refer to Appendix D for SSPC visual guide

In order to perform large area and non-contact surface inspection, laser-based sensing approaches have been investigated. In [40], a laser-based position-sensitive detector (PSD) has been presented to inspect the inner surface of a pipe for defects and diameter changes. The laser-based device was able to scan the entire inner wall of the pipe and could transverse the length of the pipe by being mounted on a micropipe robot. Experiments performed on a pipe with a diameter between 9.5–10.5 mm have shown that the approach was able to detect defects within the pipe with ± 0.1 mm accuracy. Another laser-based approach has been demonstrated for the surface-type inspection of a steel bridge structure [7][41]. In this approach, a laser range finder was used to collect laser range data and intensity data from surfaces in the environment. Surface-type classification was performed using the collected data and has yielded accurate results for different surface-types including wood, rust, and steel. Although it was possible to

perform non-contact surface inspection with high-precision and for a large area using a laser, it was also time-consuming to collect the high-resolution data. Thus, currently it can be impractical to collect high-resolution laser data over a large area required for the classification of finer texture surfaces such as rust and blast grading as described by the visual inspection standards provided by The Society of Protective Coatings (SSPC, formally the Steel Structures Painting Council): SSPC VIS2 and SSPC VIS5 (Appendix D).

For real-time applications, vision-based approaches currently provide the best option for large area inspection of finer texture surfaces. In marble slab grading, a vision system has been demonstrated to categorise marble slabs into different aesthetic groups based on texture attributes including: smooth gradients of colour, and the existence of veins and spots [4]. Figure 2.4 shows the inspection system consisting of a conveyor delivery system with a vision system providing diffused and controlled lighting for the acquisition of marble slab images. Texture features were extracted from the collected images using principal component analysis, and classified with a multi-layer perceptron neural network classifier. The classification results showed a high success rate of 90% for a set of 30 marble textures classified into three classes. Another industry that uses a vision system to perform surface defect detection is metal sheets/strips manufacturing processes [42][43][44]. In [45], real-time surface inspection of hot rolled strips has been demonstrated using an algorithm that combines image processing techniques including: image segmentation (real-time), four types of defect detection (just-in-time), and defect classification (idle-time). This vision system was able to identify defects on metal strips with a maximum width of 1650 mm and a moving speed of 18 m/s. Furthermore, [46] presented an analysis of the illumination on steel surface-type to design the optimum imaging modes of surface defects on steel plates. The approach was able to obtain high quality images that show surface defects in high contrast, while satisfying the production speed requirement of the steel sheet manufacturing process.

Overall, this review shows that a vision system provides the best approach for surface inspection in real-time, for fine texture defects, and over a wide surface area. However, in the situation where a vision system is to be used in dimly lit and complex structural environments, illumination and texture inconsistencies can occur in images. Figure 2.5

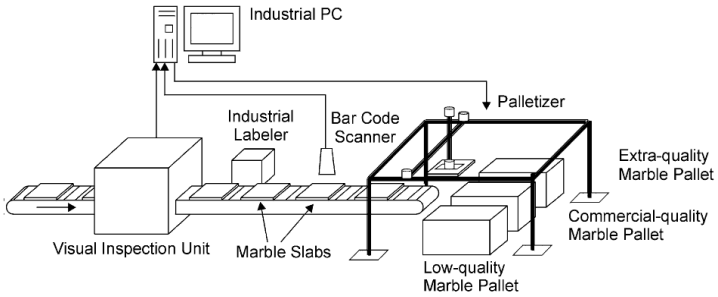


FIGURE 2.4: Automation of the marble quality classification process: from image acquisition to the pallets [4]

shows an example of a bridge structure environment where surface inspection is to be performed using a camera mounted onto the end-effector of a robot manipulator with a light source. Due to the impracticality of repositioning the light source for each new observation, the robot manipulator may need to move around the light source in a way that results in changes to the camera distance and angle to surfaces. As such, the captured images can appear non-uniformly illuminated and have texture inconsistencies such as focus blur, spatial scaling, and perspective distortion. In order to accurately classify the surface-types from these images, algorithms that are robust to the inconsistencies are needed. A review of vision-based classification approaches is provided to identify the outstanding problems that need to be addressed when implementing a vision system in a complex structural environment under various illumination conditions.

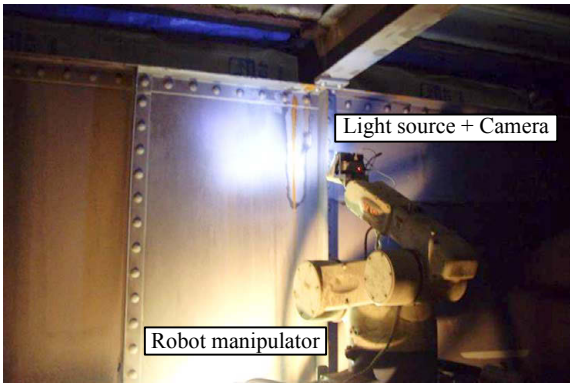


FIGURE 2.5: Directional light source and camera mounted to the end-effector of a bridge maintenance robot to inspect for rust and grit-blasting quality [5] [6] [7]

2.3 Vision-based Classification of Surface(s) with Non-uniform Illumination

In research where the lighting in an environment is assumed to uniformly illuminate a surface; texture features can be robustly extracted using Grey-Level Co-occurrence matrix (GLCM, refer to Appendix B.1 for usage in the thesis) to classify various surface-types. Figure 2.6 shows examples of uniformly lit images that can be classified using GLCM features, including an insect wing image [8] to classify insect types; steel surface images to classify for different rust and sandblast (cleanliness) grades [9]; and an aerial image [10] to classify for vegetation types. However, when surfaces are non-uniformly illuminated, their visual appearance described by the pixel intensity values will vary. This can adversely affect surface-type classification results when using GLCM features which may misinterpret non-uniform illumination as texture. In order to improve the classification accuracy for surface-types in an environment where illumination consistency cannot be guaranteed, further research has been performed into generating illumination-tolerant texture features. Local Binary Patterns [47] (LBP, refer to Appendix B.2 for usage in the thesis) is a very popular feature type that has demonstrated high tolerance against monotonic illumination. Various extensions have also been made to the original LBP to improve illumination tolerance. One such work is shown in [48] which aims to improve the robustness against reflective illumination with a proposed gradient local binary pattern (GLBP). The GLBP is shown to improved texture classification accuracy for images with reflective illumination; demonstrated using the Outex and Brodatz database with simulated reflective illumination added to each test sample using a linear transform function. In addition to investigating LBP, there have been other studies towards improving vision-based classification of surface texture under non-uniform illumination. A general view of this area of research includes, the investigation of using SVM to provide robust texture classification with variation in pattern scale [49], a proposed bidirectional feature histogram approach to represent texture surfaces that address changes in lighting and viewing conditions [50], a proposed 3D textons approach to describe the spatial variation in texture due to reflectance and surface normal [51], a parametric approach for estimating the likelihood of

homogeneously textured images.[52], an approach to remove illumination variance (shadows) from images using intrinsic images [53]. The next paragraph will look more specifically into the classification approaches that have demonstrated illumination tolerance, divided into key areas of statistical approach, filter-based approach, and model-based approach. For a statistical approach, [54] demonstrated material classification using image patch exemplars extracted from images obtained from an unknown viewpoint and illumination. In this approach, training models of texture were built using statistical descriptions of filter responses. The approach was tested on the CURet, San Francisco and Microsoft Textile databases, and showed that small image patches (3×3 pixels patch) contained sufficient information to discriminate between different textures. A filter-based approach was demonstrated in [55] to provide illumination-invariant texture classification when given a single training image. A set of filter banks that comprised of Gabor filters at four orientations and a Gaussian filter, were applied to extract an illumination-invariant feature vector. Experiments were conducted to classify texture images from the PhoTex database and the results showed accuracy above 95% for test images with only illumination tilt changes, and 60–80% for test images with non-equal illumination slants. Furthermore, a texture recognition approach based on surface rendering has been demonstrated in [56]. The approach used a model-based system to classify image texture seen from different distances and under different illumination directions by rendering 2D image texture as it would appear under different imaging capture conditions. The approach was tested by classifying images with different imaging conditions than the training image dataset. Overall, there exists various approaches to successfully classify texture surfaces captured under non-uniform illumination.

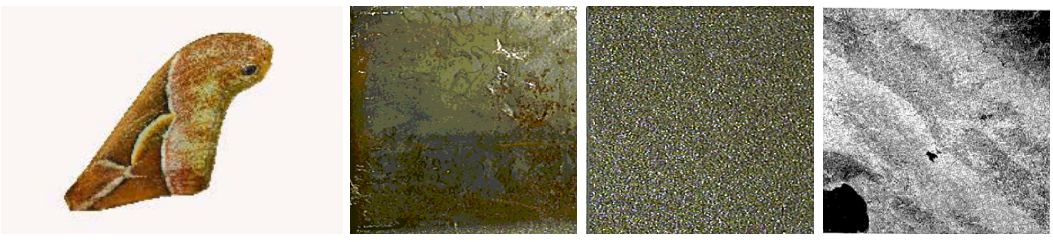


FIGURE 2.6: Images captured with uniform illumination [8][9][10]

A colour-based approach to perform rust detection on non-uniformly illuminated images has been demonstrated in [57]. Fourteen colour-spaces (RGB, rgb, I1I2I3, HSV, HIS, YUV, YIQ, YCbCr, YCgCr, XYZ, W*U*V*, L*u*v*, L*a*b*, and L*C*h*) were evaluated to find the best colour-space components for the discrimination of rust from a background surface under non-uniform illumination. The evaluation identified a*b* as the best colour-space components for rust detection, and they were selected as the features to demonstrate an adaptive ellipse approach to segment the rust regions in the images. The experimental results showed that these features provided high rust detection accuracy. Further investigation of rust detection in [11][58] has proposed the use of a Fourier Transform to initially segment background and rust regions for training data. Fourier transform was performed in the S colour-space component of hue, saturation and intensity (HSI) to detect the background region in the image, and in the b* colour-space component from L*a*b* to detect the rust regions in the image. The identified pixels for rust and background regions were used to train a Support Vector Machine (SVM), and were subsequently used to classify the test images. Figure 2.7 shows the classification results for an original image sample and a simulated non-uniform illumination image for a single surface plane. Classification results were presented that uses the proposed SVM approach (SVMRA: Support Vector Machine Rust Assessment), the previously proposed adaptive ellipse approach (BE-ANFIS: Box-and-Ellipse-Based Adaptive Network Based Fuzzy Inference System), and a benchmark K-Means approach (SKMA: simplified K-means algorithm). The SVM-based approach showed the best performance in terms of accuracy and processing time. Currently the approach can be applied for images with non-uniform illumination on a single surface plane, with a consistent background type.

Image intensity adjustment is another approach for improving classification. Intensity adjustment of an image with non-uniform illumination has been demonstrated in [59]. In this work, Bidimensional Empirical Mode Decomposition (BEMD) morphology has been applied to reduce the levels of shade and highlights and produce an image with improved intensity consistency. The intensity adjustment approach was evaluated using a K-Means approach to detect rust in images. Improved accuracy over the original image was demonstrated. The detailed intensity adjustment approach can be used for images that contain a single surface plane to bring the intensity values to a consistency level

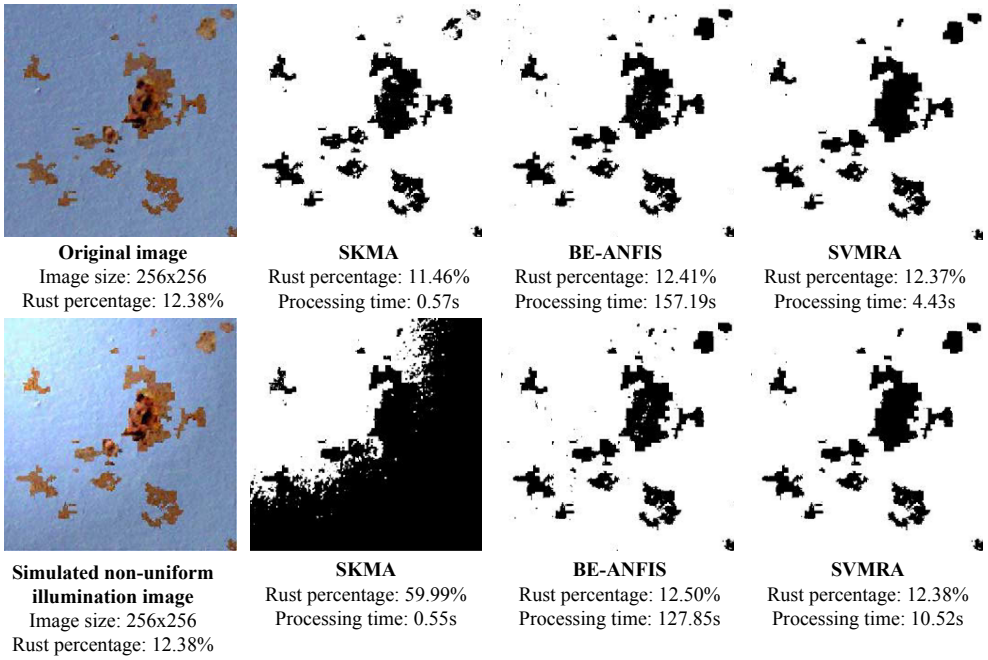


FIGURE 2.7: Rust classification results for an original image, and an image with simulated non-uniform illumination; Rust percentage = percentage of pixels in an image identified as rust [11]

dependent on the overall intensity range of the image. For images that contain a complex scene with multiple surfaces, the BEMD approach will be ineffective and as such, image adjustment approaches based on a reflectance model have been investigated to reduce illumination inconsistency. A light attenuation adjustment model has been applied to underwater terrain images [60]. In this work, a reflectance model and a 3D geometric map of the terrain were used to adjust the intensity of a pixel to correct for illumination attenuation, vignetting and colour patterns caused by underwater lighting. The adjusted images provided improved colour consistency for a 3D colour map. The work in [61] also demonstrated image adjustment to match the colour and illumination of objects captured underwater with the same object captured in air. An attenuation coefficient was estimated using multiple underwater images of objects captured at known-distance locations. The estimated coefficient was applied to correct subsequent images for improving recognition and classification operations.

Given knowledge of the light source position, viewing distance, and viewing angle of

surfaces in an image, the use of reflectance properties to directly classify surface-types is possible and has been demonstrated in [12]. In this approach, a subset of the Bidirectional Reflectance Distribution Function (BRDF) was estimated by using images captured under different lighting directions. The results of this work are shown in Figure 2.8, where two surface-types with similar colour appearance but with different reflectance properties can be distinguished. This approach is currently limited to a small surface patch placed inside a specially designed and calibrated lighting array to capture images with different lighting directions at known distances and angles [62].

The use of image and depth data captured by a portable stereo vision camera to estimate BRDF parameters was demonstrated in [63]. In this work, the BRDF parameters of different surface-types were estimated using a Population Monte Carlo algorithm, given the depth data of surface points and the direction vectors of the light source. The estimated posterior distributions of the BRDF parameters were used as the features to classify the surface-type in an image. Classification experiments were conducted on different surfaces where each test image showed only a single-type; classification was performed on an image level. Per-pixel multi-class classification of raw materials using BRDF was demonstrated in [64]. In this work, different wavelengths of light were used to illuminate surface-types such that discriminating BRDF spectral responses could be identified. The experimental results showed that images comprised of different surface-types including metal, plastic, fabric, ceramic and wood can be classified. It is noted that in order to accurately estimate the BRDF spectral response as features for surface-type classification, a complicated LED light array was used to provide different wavelengths of light at multiple directions to the surface of interest. Thus, further research into identifying the reflectance parameters for an image pixel, captured using a single light source by varying distances is necessary. Current research has indicated potential in this direction, such as light-field estimation for augmented reality using a single hand-held camera [65] and reflectance propriety estimation using a phone and onboard LED lighting [66]. With further research, methods of extracting reflectance parameters can be applied in practice to extract features for per-pixel surface-type classification in a complex field environment under various illumination.

FIGURE 2.8: Captured sample image (left), Classification based on single image (middle), Classification based on Hemispherical Harmonic coefficients (right) [12]

2.4 Vision-based Classification of Surface(s) with Texture Inconsistency

Texture in an image describes the inter-pixel relationship in terms of intensity value. There are many theoretical approaches to analyse an image to extract texture features, including statistical, geometrical, model-based and signal processing methods [67][68][69]. A case study of pattern recognition methods to classify a steel surface has demonstrated many of the theoretical approaches [70].

In summary, statistical analysis identifies the spatial distribution of pixel values (greyscale images, individual channels in a colour image, and depth images). Features extracted using statistical analysis include intensity co-occurrence [10], grey level difference [71] and local binary pattern (LBP) [47]. Geometrical analysis identifies the texture primitives in an image. Features extracted using geometrical analysis include spatial organisation of edges primitives [72]. Model-based analysis synthesises an observed intensity distribution into a model. Model-based features include parametric mosaic models [73]. Signal analysis examines the frequency components of an image. Features extracted by signal processing include Wavelet transform [74], Gabor transform [75], difference of Gaussian [76], and pseudo-Wigner distribution [77].

Many of the available texture analysis approaches have been shown to be able to extract features for the classification of real-world textures with appearance change. In [78], LBP was used to provide rotation invariant features for classifying remote sensing images. The features were compared against spectral only data and GLCM features for image

classification, and improved overall accuracy has been demonstrated. In [79], image classification was performed using features extracted at different viewing distances to improve tolerance to texture inconsistencies caused by spatial scaling. GLCM features were extracted at a number of distances and modelled into a continuous function of scale. The shape of this function was used to improve the discriminatory power of classes. This approach was demonstrated for Pap smear cell nuclei images and showed improved classification accuracy over the traditional approach that selects a number of the discrete distances to train a class. In the majority these works, texture-based classification has been demonstrated using image samples taken from texture databases with various levels of texture consistency, including MeasTex, VisTex, Outex, Brodatz and CuRet. For application in field environments, additional texture inconsistency can be expected due to unsupervised image capture conditions, including lens focusing, camera positioning, camera exposure adjustments and image cropping.

In the case where a camera is set up to observe surfaces in a complex structural environment, it is difficult to capture images of a single surface plane at consistent orientation and distance. Figure 2.9 shows an example image of a steel structure captured from an outdoors environment with irrelevant surfaces in the background. Rust grade classification was performed on the image using signal processing features and fractal dimension features trained with an SVM classifier [13]. The experimental result shows a 99% accuracy for training data, and 66% accuracy for test data, suggesting the difficulty of achieving accurate classification when images contain significant texture appearance change caused by inconsistent image capture conditions.

Inconsistent image capture conditions (specifically referring to the focus quality, camera positioning relative to the surfaces, and position of light source) can result in texture appearance change that is beyond the capability of a texture analysis approach to extract features while maintaining high class discrimination. Therefore, image processing has been applied to improve texture-based classification. An image quality adjustment approach [80][81] has been proposed that uses a reference image database for automated quality adjustment of test images for improving classification accuracy. Experimental results showed that the approach was able to improve the detection of contaminant particles in images captured in a dynamic environment involving large changes in image

FIGURE 2.9: Rusty signpost image [13]

quality. However, in the case where the scene in an image can change, it is difficult to use a reference image database to perform image quality adjustment. Alternatively, image quality assessment can be performed to detect and reject images with texture appearance change to maintain the overall accuracy of the classification process. Image quality assessment approaches based on geometric structured distortion and image definition (motion blur, noise), such as in [82][83], have been used to detect for texture appearance change. A limitation of these image quality assessment approaches is the assumption of texture inconsistency across the whole image. For an image that contains different levels of texture inconsistencies (i.e. depth of field image blur, multiple surfaces at different distances), these image assessment approaches are ineffective. Therefore, a possible area of further research is assessing an image pixel for texture inconsistencies from the training dataset. By being able to assess the texture inconsistency on a pixel level, it is possible to identify the accurately classified pixels/regions from an image that is classified using a training dataset that does not completely encapsulate the texture feature variance.

2.5 Discussion

This chapter has presented a review of the research work relevant to the algorithms and approaches necessary to achieve surface-type classification in a complex structural

environment under various illumination conditions. Firstly, a review of the approaches that can be used to build different types of environmental awareness is presented. Currently, it is on-going research to accurately build a 3D geometric map with additional colour information (3D colour map) in real-time using an RGB-D sensor [90] The information that is provided by the 3D colour map is shown to be useful to semantically label objects and locations in order to improve decision-making beyond what map coordinates can provide. Secondly, a review of sensors and sensing approaches used to perform surface inspection is provided. The review has identified that a vision system is suitable for non-contact, large surface area, and real-time inspection of surfaces. Vision systems have been successfully implemented in factory settings using a fixed camera position and uniform lighting on the inspection surface to produce high detection/classification accuracy.

Finally, a review of algorithms and approaches is provided to examine the ways of implementing a vision system to classify surface-types from images with illumination and texture inconsistencies. It is identified that the current algorithms and approaches will not be adequate for a mobile vision system to be used in the field to inspect a complex structural environment under various illumination conditions. An image of a complex structural environment can contain multiple surfaces at different viewing distances and viewing angles relative to the camera and light source. As a result, the non-uniform illumination and texture appearance change can be more challenging in comparison to images of a single surface plane captured with a consistent camera position. These challenges include: (a) colour-based surface-type classification using colour-space conversion and/or intensity adjustment algorithms can produce low accuracy results due to the large illumination change in the images; and (b) texture-based surface-type classification, which has improved robustness against non-uniform illumination can also produce low accuracy results due to the inconsistency in texture appearance caused by changes in image capture conditions of the camera (viewing distance, viewing angle and camera optical settings).

This thesis shall present the approach and algorithms to enable surface-type classification in complex structural environments under various illumination conditions. Algorithms are presented to address the identified research problems of non-uniform illumination and

texture appearance change for images captured in a structurally complex environment. In the next chapter, an overview of the proposed surface-type classification approach, and the processes used to capture RGB-D images and represent classification results in a surface-type map are presented.

Chapter 3

Surface-type Classification Approach

From the review of related works in Chapter 2, it was identified that in order to implement a vision-based approach to perform surface-type classification in a structurally complex environment under various illumination and imaging conditions, the following is needed:

(a) A colour feature extraction algorithm that can improve classification accuracy of an image captured in a complex structural environment under various illumination conditions. The reviewed approaches and algorithms for extracting colour features may extract features that results in low classification accuracy due to the non-uniform illumination for a complex scene.

(b) An algorithm to assess the classification results of an image that has been classified using a texture-based classifier. When images of a complex environment are captured using different image capture conditions, the texture appearance of a surface-type may vary from surface plane to surface plane; resulting in the texture features being inconsistent. Therefore, different regions in an image can be misclassified, and a way to identify the regions in an image that are classified accurately is needed.

This chapter present an approach and two algorithms that addresses non-uniform illumination and texture appearance change to enable surface-type classification of a

complex structural environment. The process flow of the proposed surface-type classification approach is detailed. The details of the two developed algorithms are provided in the subsequent Chapters 4 and 5, while the remaining sections of this chapter present the processes of using a robot manipulator to position the RGB-D sensor package to capture RGB-D images, the calculation of depth information including viewing distance and viewing angle for each RGB image pixel, and the process of representing classification results in a surface-type map.

3.1 Surface-type Classification Approach

Figure 3.1 shows an overview of the proposed surface-type classification approach. In this approach, a robot manipulator is used to precisely position an RGB-D sensor package at different viewpoints in a complex structural environment (e.g. Figures 1.1, 1.2) to collect RGB-D images. The collected depth image is processed to produce depth information for individual RGB image pixels, including: 3D position of the pixel, viewing distance, viewing angle, direction vector, and surface normal. Two algorithms are used, including a colour feature extraction algorithm and a classification result assessment algorithm. In the colour feature extraction algorithm, the depth information is used with a modified reflectance model to extract reflectance values from each colour channel of an RGB pixel. The reflectance values are used as the features in a colour-based classifier to perform per-pixel surface-type classification. In the classification result assessment algorithm, the depth information and camera optical settings (image capture conditions) are used to generate a probability map of surface-types in an image. For each pixel in an image the map contains a probability of being a specific surface-type.

In brief, an RGB-D image is surface-type classified using two classifiers: one from a colour-based classifier using the proposed diffused reflectance values as features, and the other from a texture-based classifier using texture features. The classification results are combined into a 3D grid-based surface-type map with a probability map used to provide a weighting factor for the texture-based classification results. The choice of using a surface-type map in 3D to represent classification results can enable a global surface-type awareness of an environment to be incrementally built over multiple observations.

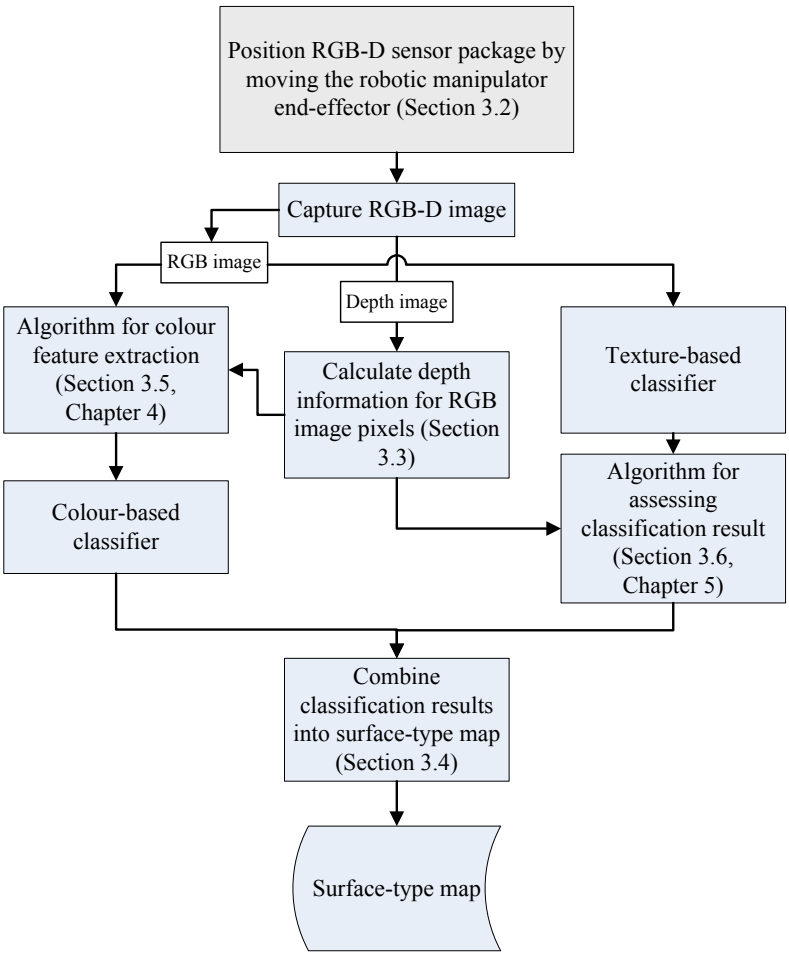


FIGURE 3.1: Overview of the proposed surface-type classification approach

3.2 Positioning of the RGB-D Sensor Package Using a Robot Manipulator

A robot manipulator is used to provide the means of precisely positioning an RGB-D sensor package at the different viewpoints to observe surfaces in a complex environment. The RGB-D sensor package consisting of a rigidly coupled red/green/blue (RGB) camera and a (D)epth camera that is mounted onto the end-effector of the robot manipulator, such that the position and orientation of the sensor package relative to the robot base frame at each viewpoint can be accurately calculated (limited by calibration process and/or robot manipulator odometry). By using a robot manipulator to position the RGB-D sensor

package, it is possible to register the RGB-D image collected at each viewpoint into a common reference frame.

The transform ${}^{o}T_{s}$, that describes the position and orientation of the Depth camera relative to the robot base frame is calculated using the end-effector-to-robot base transform ${}^{o}T_{e}(\vec{Q})$, and the hand-eye transform ${}^{e}T_{s}$. Provided with the Denavit and Hartenberg (D-H) parameters for a robot manipulator with n number of joints, the transform ${}^{o}T_{e}(\vec{Q})$, is expressed as:

$$ {}^{o}T_{e}(\vec{Q}) = \prod_{i=1}^{n} {}^{i-1}T_{i}(q_{i}) \tag{3.1} $$

where $\vec{Q} = [q_1, q_2, ...q_n]^T$ are the joint angles of a robot manipulator for achieving a certain pose. Figure 3.2 shows the coordinate frames of a robot manipulator and an RGB-D sensor package.

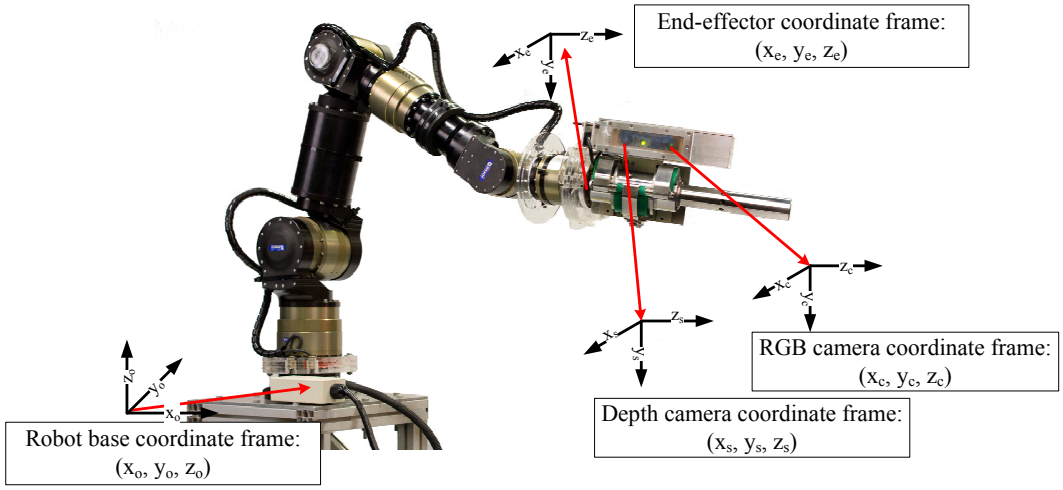


FIGURE 3.2: Coordinate frames of a robot manipulator and an RGB-D sensor package

The hand-eye transform, ${}^{e}T_{s}$, that describes the position and orientation of the depth camera relative to the end-effector is estimated using a calibration approach developed as part of this research and detailed in Appendix A [84]. Given ${}^{o}T_{e}(\vec{Q})$ and ${}^{e}T_{s}$, the transform between the depth camera and the robot base frame ${}^{o}T_{s}$, is calculated as:

$$ {}^{o}T_{s} = {}^{o}T_{e}(\vec{Q}) \times {}^{e}T_{s}. \tag{3.2} $$

${}^{o}T_{s}$ for each viewpoint is used to transform the RGB-D image from the depth camera coordinate frame into the robot base frame that is used as the coordinate frame for the global 3D surface-type map.

3.3 Calculating the Viewing Distance and Viewing Angle for an Image Pixel

The viewing distance and the viewing angle of an image pixel are used in the two algorithms to calculate the diffused reflectance values and the probability map. Figure 3.3 illustrates the viewing distance and viewing angle for a 3D point in the camera coordinate frame (x_c, y_c, z_c) that corresponds to a pixel in the RGB image. The viewing distance, d_c, is defined as the distance from the surface point-to-RGB camera coordinate origin; and the viewing angle, θ_c, is the angle of incidence between the normal of the point $\vec{v}_\eta$, and the straight line direction vector between the point and the RGB camera coordinate origin $\vec{v}_c$. Each pixel in an image will have a corresponding point, and a viewing distance and a viewing angle associated with it. The following process is used to calculate the viewing distance and viewing angle for a pixel in the RGB image.

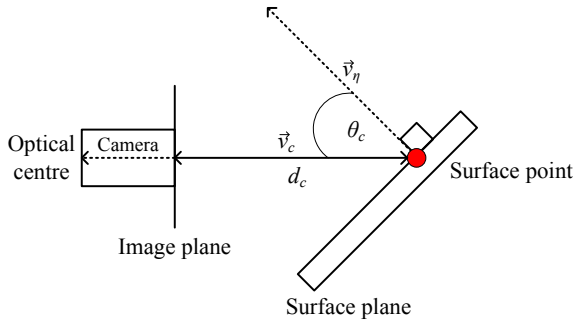


FIGURE 3.3: Viewing distance and viewing angle for a surface point in 3D

First, the depth image, Ω_s, captured by the depth camera is converted using perspective projection [85] into a set of points $(x_{s_i}, y_{s_i}, z_{s_i})$, where $i \in \{1, \ldots n\}$ given n number of pixels in the image. The depth camera's intrinsic parameters [86], which includes the focal lengths F_{s_u} and F_{s_v} and the principal points C_{s_u} and C_{s_v} are used for the perspective

projection, as:

$$x_{s_i} = (i_u - C_{s_u}) \times \frac{\Omega_{s_i}}{F_{s_u}} \tag{3.3}$$

$$y_{s_i} = (i_v - C_{s_v}) \times \frac{\Omega_{s_i}}{F_{s_v}} \tag{3.4}$$

$$z_{s_i} = \Omega_{s_i} \tag{3.5}$$

where i_u and i_v are the image coordinate position for the i^{th} pixel, and Ω_{s_i} is the depth value for the i^{th} pixel. The set of points in depth camera coordinate $(x_{s_i}, y_{s_i}, z_{s_i})$ is transformed into RGB camera coordinate $(x_{c_i}, y_{c_i}, z_{c_i})$ by using the transform ${}^{s}T_{c}$ [86] between the depth camera and the RGB camera.

For the set of points in the RGB camera coordinate, a principle component analysis (PCA)-based approach [87] is applied to identify the surface normals. To calculate a surface normal for a point, a search is performed to identify a subset of surrounding points P within a distance τ_μ. If the number of points in the subset returned from the distance search meets the required amount deemed sufficient (greater than the number of dimensions in Euclidean space), then PCA can be performed. The PCA process includes calculating the covariance matrix $cov(\mathsf{P})$ for the subset of points P, and then analysing the eigenvectors V_λ and the eigenvalues λ for the surface normal. Figure 3.4 shows the eigenvector $V_{min(\lambda)}$, with the smallest eigenvalue $min(\lambda)$. A condition is set to assess whether $V_{min(\lambda)}$ can be taken as the surface normal for the point, and is expressed as:

$$\text{If } \min(\lambda) < \frac{1}{\tau_\alpha}\text{mid}(\lambda) \text{ then } \vec{v}_\eta = V_{min(\lambda)} \tag{3.6}$$

where min() is a function to identify the smallest element in a vector and mid() is a function to identify the element in a vector which has neither the smallest/largest value (for a vector with three elements), and τ_α is a threshold set by the user.

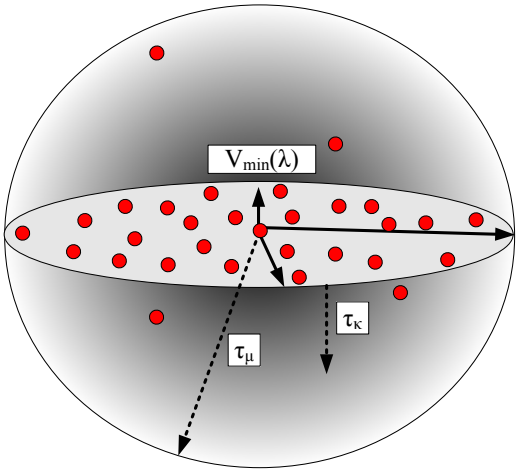


FIGURE 3.4: The surface normal calculated for a 3D point

If the majority (>50%) of the points in P are within a point-to-plane distance of τ_κ to the identified surface plane, then the surface normal is determined to be accurate for the selected point and is labelled as valid. This process is repeated until all points in the set $(x_{c_i}, y_{c_i}, z_{c_i})$ have been either assigned with a surface normal or marked as a point without a valid surface normal (i.e. noise points or points close to corners/edges) because of insufficiency : (a) number of points returned from the distance search of τ_μ is less than the number of dimensions, or (b) the condition in Equation 3.6 was not satisfied, or (c) less than 50% of points in P are within the point-to-plane distance τ_κ.

Given the surface normal for a point, $\vec{v}_{\eta_i}$, and the direction vector starting from the point to the RGB camera coordinate origin, $\vec{v}_{c_i}$, the viewing angle, θ_{c_i}, is calculated as:

$$\theta_{c_i} = \cos^{-1} \frac{\vec{v}_{\eta_i} \cdot \vec{v}_{c_i}}{||\vec{v}_{\eta_i}||\,||\vec{v}_{c_i}||}. \tag{3.7}$$

The viewing distance is calculated as the Euclidean distance between the point to the RGB camera coordinate origin $(x_{c_o}, y_{c_o}, z_{c_o})$ and is expressed as:

$$d_{c_i} = \sqrt{(x_{c_o} - x_{c_i})^2 + (y_{c_o} - y_{c_i})^2 + (z_{c_o} - z_{c_i})^2}. \tag{3.8}$$

The values of d_{c_i} and θ_{c_i} are calculated for each point in the set.

3.4 Surface-type Map in 3D

A 3D grid-based map that is similar in concept to an occupancy grid [88], octomap [89], RGB-D map [90] is used to combine and represent the vision-based (colour and texture) classification results such that it can be used by a robot. In 3D space, a voxel (volumetric pixel) can be used as the individual grid cell that forms a surface-type map. In this way, the classification result of image pixels that have 3D points in close proximity can be combined and represented by a voxel. A surface-type map based on voxels can be used to combine and represent: (a) classification results produced using RGB-D images captured from different viewpoints in the environment; and (b) classification results of an RGB-D image produced by different classifiers. The 3D grid-based map is used in the case study presented in Chapter 6 to combine and represent the (colour and texture) classification results of an environment.

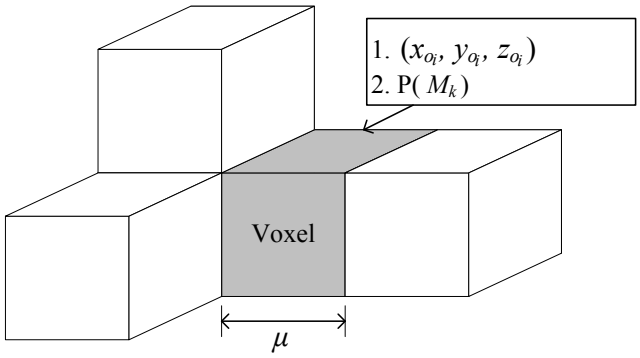


FIGURE 3.5: A grid-based 3D surface-type map used to represent surface-type classification results

Figure 3.5 shows that each voxel is associated with a point in the robot-based coordinate $(x_{o_i}, y_{o_i}, z_{o_i})$ and contains a discrete probability distribution of the surface-types, $P(M_k)$, for $k \in \{1, \ldots_t\}$, given n_t number of surface-types. The point associated with a voxel is the centre position of the voxel's space. Points are defined at intervals according to the voxel size μ. The initial state of $P(M_k)$ for a voxel is a uniform distribution since there is no surface-type *a priori* knowledge.

To assign an image pixel to a voxel, the 3D coordinate point of the pixel is rounded down to the nearest voxel coordinate point. For example, if the voxel size is $\mu = 10$ mm, an

image pixel with the coordinate point (13, 18, 35) will be assigned to the voxel associated with the coordinate point (10, 20, 30). Once the voxel has been identified for the pixel, the conditional surface-type probabilities of the pixel provided by the classification result is updated into the voxel's current surface-type probability distribution $P(M_k)$ using Bayes' rule. The updated surface-type probability distribution of a voxel, $P(M_k|E)$, is calculated using the conditional surface-type probabilities generated by a classifier for the pixel $P(E|M_k)$, the voxel's current surface-type probability distribution $P(M_k)$, and a normalisation constant, $P(E)$ as:

$$P(M_k|E) = \frac{P(E|M_k)P(M_k)}{P(E)}. \tag{3.9}$$

For every new update to the voxel given new observation, $P(M_k)$ is taken as the current $P(M_k|E)$. At any stage of the surface-type map, a voxel's surface-type probability distribution $P(M_k)$ can be used to provide a single surface-type estimate, $\hat{M}_k$ (a decision about the most probable surface-type for the voxel). Maximum a posteriori (MAP) decision rule is used to select the most probable hypothesis as the candidate class as:

$$\hat{M}_k = \underset{M_k}{\operatorname{argmax}} P(M_k). \tag{3.10}$$

3.4.1 Combined Surface-type Map

The classification results produced using a colour-based classifier and a texture-based classifier are combined to generate a combined surface-type map. Firstly, separate surface-type maps are produced from the colour-based classification result and from the texture-based classification result. Then for the corresponding voxels between the two surface-type maps, their surface-type probability distribution are combined as:

$$P_{c,t}(M_k) = \frac{P_c(M_k) + \omega_v \times P_t(M_k)}{\sum_{k=1}^{n_t}(P_c(M_k) + \omega_v \times P_t(M_k))} \tag{3.11}$$

where the combined surface-type probability distribution is $P_{c,t}(M_k)$, the surface-type probability distribution for the colour-based classification result is $P_c(M_k)$, and the surface-type probability distribution for the texture-based classification result is $P_t(M_k)$.

A weighting factor ω_v is applied to $P_t(M_k)$ by using a probability map produced from the algorithm detailed Chapter 5; the weighting factor for a voxel is the average of the probability values for the pixels within the voxel's space.

3.5 Colour Feature Extraction

Chapter 4 will present an algorithm for the extraction of colour feature using RGB-D images and a modified reflectance model. A reflectance model using a single light source is presented and the process to extract three reflectance-based colour features from each colour channel (RGB) of a pixel is provided. A classifier is trained using the extracted reflectance-based colour features to classify surface-types in images with non-uniform illumination. Experiments are conducted to compare the performance of the extracted reflectance-based colour features against other colour-space features, including RGB and the CIELab colour-space component a*b* to classify surface-types under various illumination conditions.

3.6 Classification Results Assessment

Chapter 5 will present an algorithm to produce a probability map to assess texture-based classification results by analysing the image capture conditions (viewing distance, viewing angle and camera optical settings). An analysis is performed to show the effect of image capture condition changes in terms of image qualities (focus quality, spatial resolution, perspective distortion) on the consistency of texture features extracted from images of a surface-type. From the analysis, image capture condition thresholds are defined and used in a function to calculate a pixel's probability of being a specific surface-type. The process using the function to produce a probability map of surface-types in an image is detailed. Experiments are conducted to verify the effect of changes in image capture conditions. An experiment is also conducted to use a probability map to identify regions in an image that have a high probability of being accurately classified.

3.7 Discussion

This chapter has presented the proposed approach to classify surface-types in a complex structural environment under various illumination conditions. The processes in the approach including the use of a robot manipulator to collect RGB-D images, the calculation of viewing distance and viewing angle for each pixel, and the combination and representation of classification results as a surface-type map are detailed in this chapter. It is to note that the parameters and thresholds used for the calculation of viewing angle (size of the subset of points P, τ_μ, τ_α and τ_κ) are empirically selected by considering the resolution and noise level of the depth data and the complexity of the environment. The selected parameters and thresholds are manually evaluated by checking for satisfactory performance for several viewpoints in the environment. Future work will (a) investigate designing a process to optimally select the parameter and threshold values that produces the best performance and (b) develop failure modes to handle the low performance due to incorrectly selected parameters and thresholds. The next two chapters will present the algorithms for colour feature extraction and the algorithm to assess classification results.

Chapter 4

Algorithm for Extraction of Colour Features

When a single light source is used to illuminate the surfaces in a dark environment (Figure 4.1), the captured images may exhibit non-uniform illumination caused by the different position of the light source relative to the surfaces. Non-uniform illumination can adversely affect the classification accuracy of surface-types when colour-based features are used. In order to accurately classify different surface-types in an image that has non-uniform illumination, either the training images needs to completely encapsulate the variance in illumination or an algorithm to extract illumination-invariant colour features is used.

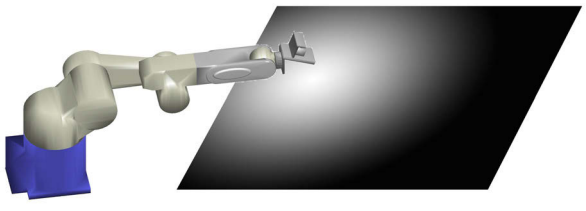


FIGURE 4.1: Robot manipulator with a directional light source illuminating a surface

A widely applied approach to extract illumination-invariant colour features is to convert the standard Red, Green, and Blue (RGB) colour-space into a colour-space that represents the illumination intensity in an individual colour component. Typically, RGB

can be converted into (CIELab) L*a*b*, HSV, HSL, and RG chromaticity colour-spaces to separate illumination intensity into a colour component and provide colour features for classification [57]. Additionally, please refer to Section 2.3 for related research on illumination-invariant features.

The algorithm presented in this chapter extracts colour features in an image that has non-uniform illumination by using the addition of depth (D) information and a reflectance model. The intensity values of an RGB pixel are taken as the irradiance off a surface that is illuminated by a single light source. Assuming Lambertian surface-types, the diffused reflectance values, which describes the ratio of diffused light reflected off a surface, can be extracted from each colour channel of a pixel and used as colour features for classification of the surface-types. In the case of metal such as steel structures, the surfaces can be assumed Lambertian when painted with a matt finish, corroded with rust and/or prepared for painting via grit-blasting. The work presented in this chapter was published in 'Surface-type Classification using RGB-D, Transactions on Automation Science and Engineering' (please refer to Section 1.5.1, point 1).

4.1 Chapter 4 Overview

This chapter presents the algorithm to perform colour feature extraction from an image and the experiments to verify the extracted features. Firstly, a reflectance model of a single light source is provided and used to describe the various illumination conditions on surfaces. The radiometric response function of a camera is discussed and taken into account by modifying the reflectance model.

Figure 4.2 shows the processes of the colour feature extraction algorithm. The preliminary processes of capturing the RGB-D image and calculating the 3D point $(x_{c_i}, y_{c_i}, z_{c_i})$, the surface normal $\vec{v}_\eta$, the viewing direction vector $\vec{v}_c$, and the viewing angle θ_c for an RGB pixel has been provided in Chapter 3. Firstly, the position of the light source in the camera coordinate frame, $\vec{l}$, is calculated using a calibration RGB-D image that contains both specular and diffused reflectance. Once the light source position has been calculated, the reflectance model parameters including light source-to-surface point distance, d_l, and

the angle of incidence of the light source-to-surface point, θ_l, are calculated for the pixels. Finally, given d_l and θ_l, the diffused reflectance values, K_d, for each RGB colour channel where $(K_{d,R}, K_{d,G}, K_{d,B}) \in K_d$, are calculated as the proposed colour features to classify surface-types. The process of calculating d_l and θ_l, and then K_d, is performed for each of the pixels in an image, as detailed in the Algorithm 1 pseudocode.

Experiments are conducted to compare the diffused reflectance values K_d against RGB intensity values and (CIELab) L*a*b* colour component values. Three support vector machine (SVM, refer to Appendix C.2) classifiers are trained with the diffused reflectance values, RGB intensity values, and L*a*b* colour component values to classify for surface-types. The accuracy of the classification results is compared. A portable RGB-D sensor package with a mounted light source has been implemented in the experiments.

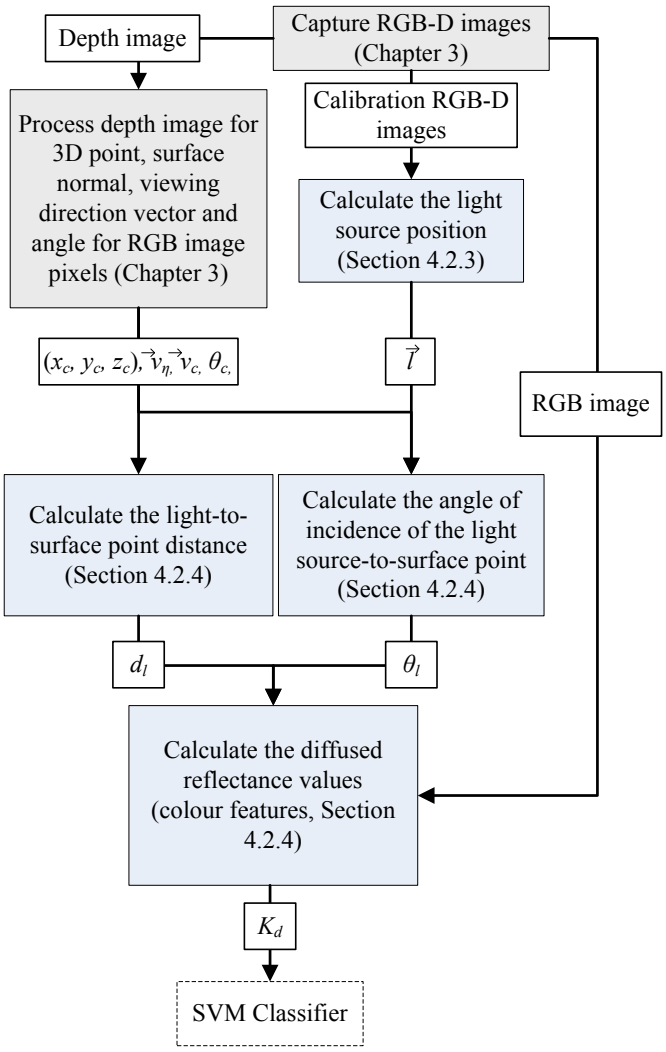


FIGURE 4.2: Colour feature extraction algorithm

Algorithm 1 Calculation of diffused reflectance values (K_d) for an RGB-D image

Require: Light source position, $\vec{l}$, and depth information $(x_{c_i}, y_{c_i}, z_{c_i})$, $\vec{v}_{\eta_i}$

1: **for** pixel $i = 1$ to n in RGB-D image set $\{1, \dots n\}$ **do**
2: **if** pixel i has a valid surface normal $\vec{v}_{\eta_i}$ **then**
3: d_{l_i} = CalculateDistance($\vec{l}$,$x_{c_i}, y_{c_i}, z_{c_i}$)
4: θ_{l_i} = CalculateAngle($\vec{l}$,$x_{c_i}, y_{c_i}, z_{c_i}$,$\vec{v}_{\eta_i}$)
5: **for** colour channel $x = 1$ to 3 in RGB colour-space (R=1, G=2, B=3) **do**
6: K_{d,x_i} = CalculateDiffusedReflectanceValue(d_{l_i},θ_{l_i},I_{r,x_i})
7: **end for**
8: **else**
9: $K_{d,1_i}$,$K_{d,2_i}$,$K_{d,3_i}$ = null values
10: **end if**
11: **end for**

The following is a description of the functions used in the algorithm above:

- CalculateDistance($\vec{l}$,$x_{c_i}, y_{c_i}, z_{c_i}$) calculates the Euclidean distance (d_l) between the light source($\vec{l} = (x_{c_l}, y_{c_l}, z_{c_l})$) and the surface point $(x_{c_i}, y_{c_i}, z_{c_i})$; formula provided in Equation 4.14
- CalculateAngle($\vec{l}$,$x_{c_i}, y_{c_i}, z_{c_i}$,$\vec{v}_{\eta_i}$) calculates the angle of incidence of the light source to the surface point, θ_l; formula provided in Equation 4.13
- CalculateDiffusedReflectanceValue(d_{l_i},θ_{l_i},I_{r,x_i}) calculates the diffused reflectance values, K_d, for each RGB colour channel where $(K_{d,R}, K_{d,G}, K_{d,B}) \in K_d$; formula provided in Equation 4.12

4.2 Diffused Reflectance Values Extraction

This section details the reflectance model selected to describe the illumination of a scene created by a single light source, and the function selected to model the radiometric response of a digital camera. Combining the selected reflectance model and radiometric response

function, a formula is produced that is used to calculate the diffused reflectance values for an RGB pixel as the proposed colour features. The steps to calculate the parameters in the formula are also detailed in this section.

4.2.1 Torrance-Sparrow Reflectance Model

A reflectance model can be used to describe the irradiance of light when it is reflected off a surface, such as the attenuation, specularity and diffusion. The Torrance-Sparrow reflectance model [91] is selected to describe the light reflected off a surface point as a combination of both diffused and specular reflections. The Torrance-Sparrow reflectance model is a general reflectance model and has shown to be able to capture the reflectance properties of a large number of surfaces [92]. Figure 4.3 shows a simple illustration of how light is reflected off a surface, and the characteristics of the diffused reflection and the specular reflection that is produced. In the model, the surface point-to-light source direction vector is $\vec{v}_l$, the surface point normal vector is $\vec{v}_\eta$, and the surface point-to-camera coordinate origin direction vector is $\vec{v}_c$. The diffused reflection describes the light rays that are scattered within the surface, and the specular reflection describes the light rays that are directly reflected from the surface.

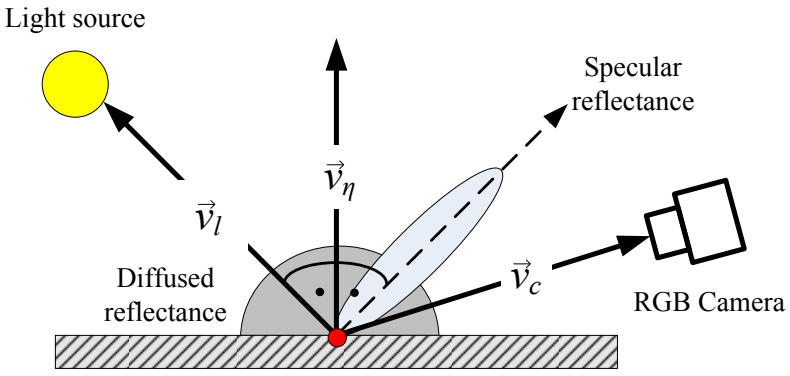


FIGURE 4.3: Diffused and specular reflection

Assuming the Fresnel reflectance is constant and the geometric attenuation factor for a clean air environment is one, the general equation of the Torrance-Sparrow reflectance

model for an image pixel is expressed as:

$$I_r = \frac{I_l}{{d_l}^2}(k_d \cos\theta_l + \frac{k_s}{\cos\theta_c} \exp(-\frac{\alpha^2}{2\sigma^2})). \tag{4.1}$$

The parameters in Equation 4.1 are illustrated in Figure 4.4, where the diffused reflectance ratio is k_d, the specular reflectance ratio is k_s, the light source intensity is I_l, surface-to-camera distance is d_c, light source-to-surface point distance is d_l, reflected light intensity off the surface is I_r, angle between the light source direction vector and the surface normal vector is θ_l, angle between the viewing direction vector and the surface normal vector is θ_c, the angle between the surface normal vector with the bisector of the viewing direction vector and the light source direction vector is α, and the surface roughness is σ.

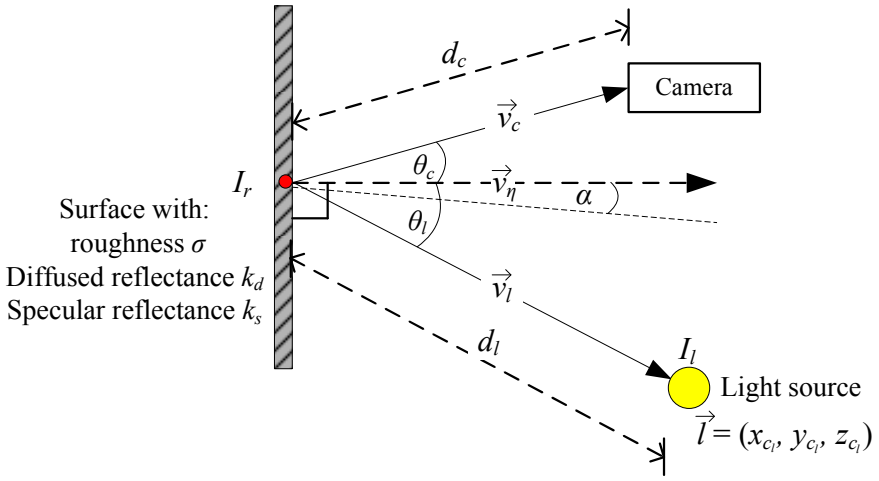


FIGURE 4.4: Parameters of light reflectance model

For a light source intensity I_l, which is a constant value (brightness of the light source is always the same), it is unnecessary to explicitly express the light source intensity parameter I_l in the reflectance model. Therefore, the reflectance values, K_d and K_s, are defined by combining the reflectance ratios, k_d and k_s, with the light source intensity I_l, as:

$$K_d = k_d I_l, \; K_s = k_s I_l \tag{4.2}$$

then Equation 4.1 is rewritten as:

$$I_r = \frac{1}{{d_l}^2}(K_d \cos\theta_l + \frac{K_s}{\cos\theta_c} \exp(-\frac{\alpha^2}{2\sigma^2})). \tag{4.3}$$

It is noted that the reflected light intensity notation, I_r, is a set notation that represents the intensity values for the three RGB colour channels, where $(I_{r,R}, I_{r,G}, I_{r,B}) \in I_r$. In addition, the reflectance values (K_d, K_s) are also set notations used to represent the reflectance values for each RGB colour channel, where $(K_{d,R}, K_{d,G}, K_{d,B}) \in K_d$, and $(K_{s,R}, K_{s,G}, K_{s,B}) \in K_s$. Given that the calculation of the reflectance value is the same for each colour channel, the equations are expressed using the set notations (I_r, K_d, K_s). The calculation process is performed for each colour channel to generate the three separate reflectance values for a pixel.

4.2.2 Radiometric Response Function of a Camera

In many computer vision algorithms, it is assumed that the surface irradiance is linearly related to the RGB intensity values captured by a camera [93][94]. However, a camera will usually have a non-linear radiometric response function [95]; radiometric response function describes how the surface irradiance detected by a pixel in the camera's sensor is converted into RGB image intensity values.

The non-linearity of the radiometric response function needs to be taken into consideration in applications where colour consistency is important (e.g. image mosaics and textures of 3D models [93]). In the proposed colour feature extraction algorithm, the measurement of reflected light intensity off a surface point, I_r, is performed using an RGB camera. As a result, the reflectance values (K_d, K_s) calculated using this reflectance model will be affected by the non-linear radiometric response of the camera. Hence, the radiometric response of a camera needs to be included as part of the reflectance model in order to calculate consistent diffused reflectance values.

It has been proposed in [14] that the radiometric response function can be represented using a polynomial function of n^{th} order depending on the non-linearity of the radiometric

response. Figure 4.5 shows the radiometric response of several different imaging systems, which illustrates that a polynomial function is suitable to model the response function.

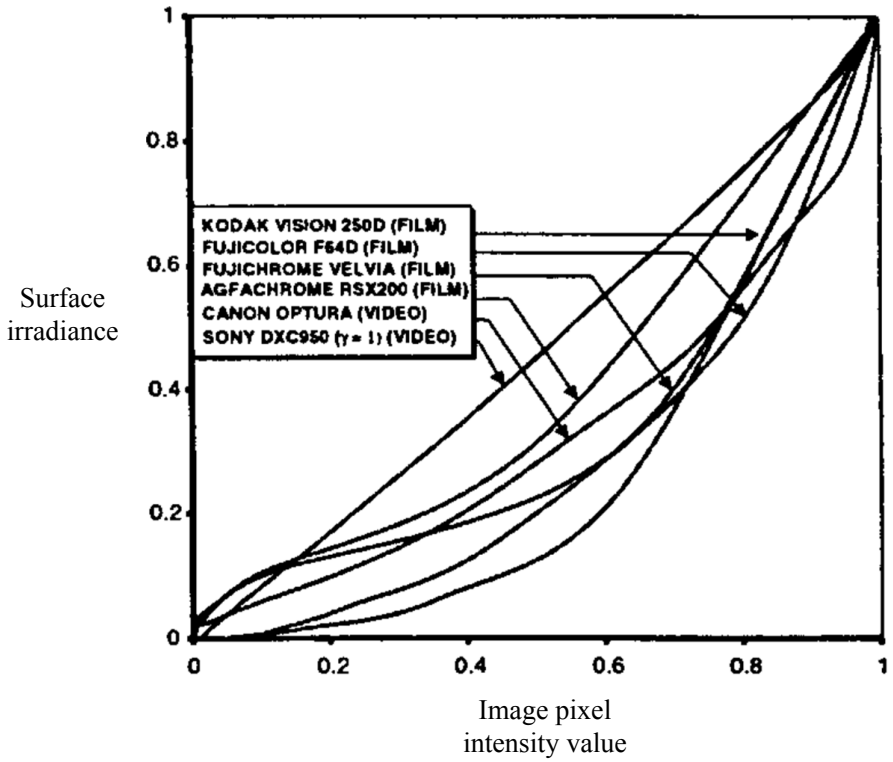


FIGURE 4.5: Response functions of several different imaging systems [14]

In the selected reflectance model, it is assumed that the light source I_l is a constant value and the exposure settings of the camera are fixed. Therefore, the change in reflected light intensity off a surface, as measured by an RGB camera, is only dependent on the distance of the light source to the surface, d_l. The radiometric response of a camera can be included into the reflectance model by modifying the light intensity change relative to the distance d_l, from originally $(\frac{1}{{d_1}^2})$ in Equation 4.3 into a 2^{nd} order polynomial function such that it is modified and rewritten as:

$$I_r = \frac{1}{a_2{d_l}^2 + a_1 d_l + a_0}(K_d \cos\theta_l + \frac{K_s}{\cos\theta_c}\exp(-\frac{\alpha^2}{2\sigma^2})). \tag{4.4}$$

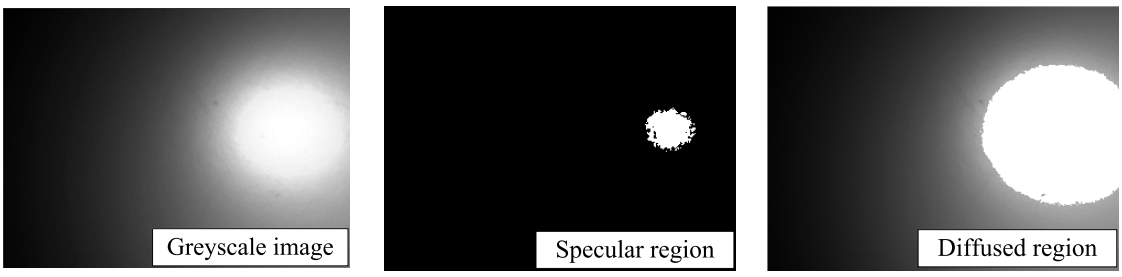
A 2^{nd} order polynomial is identified as the lowest order to provide suitable approximation of the radiometric response for a RGB camera. Calibration experiments have shown that a 2^{nd} order polynomial used in the equation can provide values for a

machine learning algorithm (SVM, naive Bayes) to produce accurate classification. The coefficients (a_2, a_1, a_0) of the polynomial are calculated using a calibration process, where d_l is incrementally increased away from a surface with Lambertian reflectance (only diffused reflectance). The RGB intensity value of a pixel with zero or near zero angle of incidence θ_l, at each increment of d_l, is taken as a sample point. A least squares fitting of a 2^{nd} order polynomial to the sample points is applied to identify the coefficients. As a result, the diffused reflectance values calculated for surface points within the calibrated light source-to-surface range will show improved consistency.

4.2.3 Camera-to-Light Source Position Calculation

The camera-to-light source position, $\vec{l} = (x_{c_l}, y_{c_l}, z_{c_l})$, describes the position of the light source in the camera coordinate frame. It is used to calculate the reflectance model parameters of the light source-to-surface point distance, d_l, and the angle of incidence of the light source to the surface point, θ_l, for each pixel. The camera-to-light source position, $\vec{l}$, can be calculated using the specular and diffused reflectance regions in a calibration image [96].

Figure 4.6*a* shows a greyscale calibration image Ω_c, that can be used to calculate $\vec{l}$, which contains a planar surface that consists of a single surface-type (constant K_d and K_s). Figure 4.6*b* contains a specular reflectance region Ω_{cs} (white) and Figure 4.6*c* contains a diffused reflectance region Ω_{cd} (non-white).



(a) (b) (c)

FIGURE 4.6: *a*) Greyscale of the calibration image Ω_c; *b*) Binary image of specular reflectance region in the calibration image Ω_{cs}; *c*) Diffused reflectance region in the calibration image Ω_{cd}

The specular reflectance region Ω_{cs}, in the calibration image Ω_c, is identified by applying an intensity value threshold, τ_s, as in:

$$\Omega_{cs} = \{i \in \Omega_c | I_{r_i} > \tau_s\}. \tag{4.5}$$

Figure 4.7 shows the centroid pixel in the specular reflectance region, defined as $\vec{p}$; the direction vector from $\vec{p}$ towards the light source position $\vec{l}$, defined as $\vec{v}_{l_p}$; the normal vector for $\vec{p}$, defined as $\vec{v}_{\eta_p}$; and the direction vector from $\vec{p}$ towards the camera coordinate frame origin, defined as $\vec{v}_{c_p}$ (all unit vectors). Because the angle between the surface normal ($\vec{v}_{\eta_p}$), and the bisector of the two direction vectors ($\vec{v}_{l_p}$ and $\vec{v}_{c_p}$) is assumed to be zero for the centroid pixel ($\alpha = 0$), $\vec{v}_{l_p}$ can be calculated as the reflection vector of $\vec{v}_{c_p}$, as:

$$\vec{v}_{l_p} = 2(\vec{v}_{\eta_p} \cdot \vec{v}_{c_p})\vec{v}_{\eta_p} - \vec{v}_{c_p}. \tag{4.6}$$

The direction vector for surface point-to-camera, $\vec{v}_{c_p}$, and the surface normal vector, $\vec{v}_{\eta_p}$, are provided by the depth image (detailed in Chapter 3).

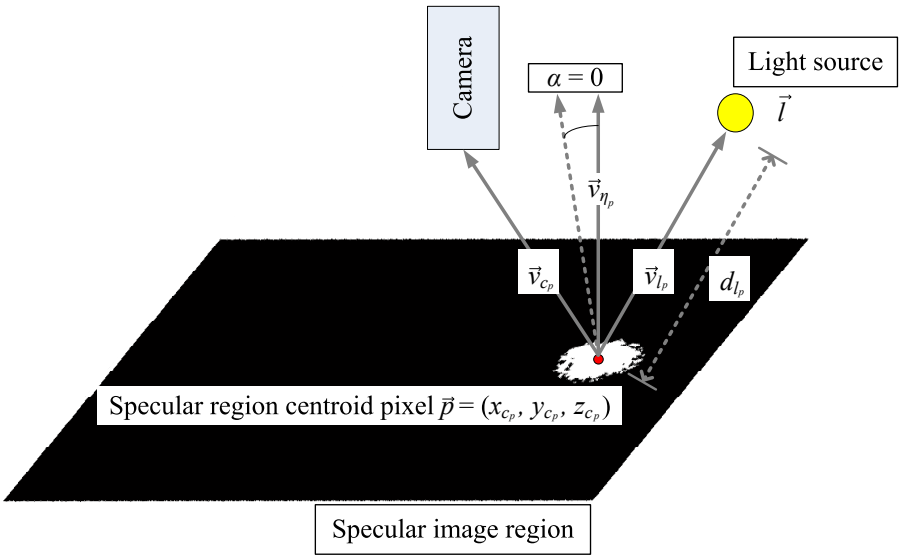


FIGURE 4.7: Light source direction vector estimation using specular centroid pixel

The distance d_{l_p}, between the specular centroid point $\vec{p}$, and the light source position $\vec{l}$, is estimated using the diffused region of the calibration image Ω_{cd} (shown in Figure 4.6*c*). The value of Ω_{cd} is extracted by applying an intensity threshold, τ_d, to the calibration

image Ω_c, as:

$$\Omega_{cd} = \{i \in \Omega_c | I_{r_i} < \tau_d\}. \tag{4.7}$$

A least squares fitting of the diffused reflectance component of Equation 4.4 to Ω_{cd} is performed to calculate d_{l_p}, as:

$$d_{l_p} = \underset{d_n}{\operatorname{argmax}} \sum_{i \in \Omega_{cd}} \left(f(i, d_n) - \frac{1}{N_{cd}} \sum_{i \in \Omega_{cd}} f(i, d_n) \right)^2 \tag{4.8}$$

where:

$$f(i, d_n) = \frac{I_{r_i}(a_2 d_l(i, dn)^2 + a_1 d_l(i, dn) + a_0)}{cos\theta_l(i, d_n)}. \tag{4.9}$$

The number of pixel points in the diffused region is N_{cd}; the optimisation parameter to identify the best fit of diffused reflectance component to the diffused region of the calibration image is d_n; the greyscale intensity value for an i^{th} pixel point is I_{r_i}; the light source-to-surface point distance for an i^{th} pixel point assuming $d_{l_p} = d_n$, is $d_l(i, dn)$; and the angle between the light source direction vector and the surface normal vector for an i^{th} pixel point assuming $d_{l_p} = d_n$, is $\theta_l(i, d_n)$.

The estimated direction vector, $\vec{v}_{l_p}$, and the distance, d_{l_p}, are used to calculate the camera-to-light source position, $\vec{l}$, as:

$$\vec{l} = \vec{p} + d_{l_p} \vec{v}_{l_p}. \tag{4.10}$$

It is noted that depending on factors including light intensity, camera exposure setting and the reflectance of the surface-type, the range of intensities captured in an image can be different. Therefore the thresholds τ_d and τ_s are selected empirically to separate the diffused and specular regions in an image, and are dependent on the intensity range of the captured image. Currently, calibration for the light source position is performed during a commissioning stage which is supervised by an operator. In the event of failure due to inappropriate parameter values, the process is repeated.

4.2.4 Diffused Reflectance Value Calculation - Proposed Colour Features

The diffused reflectance values are calculated and can be used to classify different surface-types. The diffused reflectance component of Equation 4.4 is expressed as:

$$I_r = \frac{1}{a_2 {d_l}^2 + a_1 d_l + a_0}(K_d \cos\theta_l). \tag{4.11}$$

Equation 4.11 is rearranged for the calculation of the diffused reflectance value, K_d, as:

$$K_d = \frac{I_r(a_2 {d_l}^2 + a_1 d_l + a_0)}{\cos\theta_l}. \tag{4.12}$$

To calculate K_d for an image pixel, the light source-to-surface point angle of incidence, θ_l, and light source-to-surface point distance, d_l, needs to be calculated. Figure 4.8 illustrates the calculation of the reflectance model parameters θ_l and d_l for a point corresponding to an image pixel.

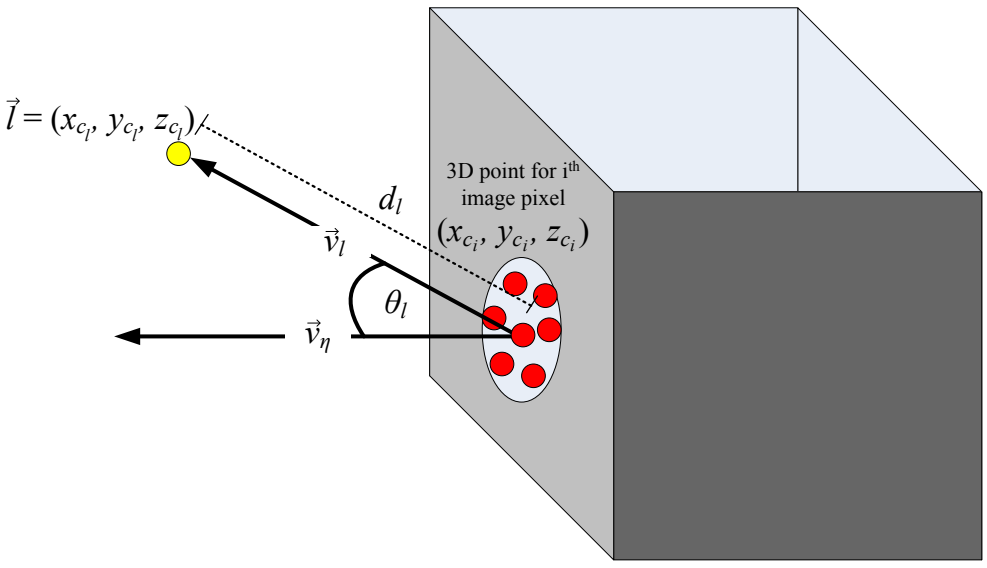


FIGURE 4.8: Calculating θ_l and d_l for a 3D point representing an i^{th} image pixel

The value of θ_l is calculated using the surface normal vector, $\vec{v}_\eta$, and direction vector, $\vec{v}_l$, for the 3D point as:

$$\theta_l = cos^{-1}\frac{\vec{v}_l \cdot \vec{v}_\eta}{||\vec{v}_l||||\vec{v}_\eta||}. \tag{4.13}$$

The light source-to-surface point distance, d_l, is calculated using the light source position, $\vec{l}$, and the point of the pixel with i^{th} index, as:

$$d_l = \sqrt{(x_{c_l} - x_{c_i})^2 + (y_{c_l} - y_{c_i})^2 + (z_{c_l} - z_{c_i})^2}. \tag{4.14}$$

The RGB intensity values for a pixel are used as the reflected light intensity off the surface where $(I_{r,R}, I_{r,G}, I_{r,B}) \in I_r$. As a result, three diffused reflectance values $(K_{d,R}, K_{d,G}, K_{d,B})$ are calculated for an image pixel. The outcome of the algorithm is three diffused reflectance values for each pixel that can be used as the colour features to classify surface-types.

4.3 CIELab L*a*b* Colour-Space Conversion

In this thesis, L*a*b* colour-space conversion is performed on the collected RGB-D images to provide a benchmark comparison against the diffused reflectance values extracted by the algorithm.

The RGB colour-space has been shown to be able to classify surface-types (such as rust) from simple background surfaces [97]. However, non-uniform illumination significantly reduces the classification accuracy when the RGB colour-space is used. Therefore, colour-space conversions have been performed to reduce the misclassification caused by non-uniform illumination. The work in [57] and [11] investigated 14 colour-spaces to find the best configuration for non-uniformly illuminated rust image detection and found that the a*b* colour component from the CIELab colour-space was the best configuration for this application.

The L*a*b* colour-space represents illumination intensity as a separate luminance component. The L* component corresponds to lightness intensity ranging from black to white (0–100), the a* component measures green-red (negative-positive), and the b* measures blue-yellow (negative-positive). To convert RGB into the L*a*b* components, the tristimulus values [98] (X, Y, Z) are calculated as:

$$X = 0.4125 f\left(\frac{R}{255}\right) + 0.3576 f\left(\frac{G}{255}\right) + 0.1805 f\left(\frac{B}{255}\right) \tag{4.15}$$

$$Y = 0.2126f\left(\frac{R}{255}\right) + 0.7152f\left(\frac{G}{255}\right) + 0.0722f\left(\frac{B}{255}\right) \tag{4.16}$$

$$Z = 0.0193f\left(\frac{R}{255}\right) + 0.1192f\left(\frac{G}{255}\right) + 0.9505f\left(\frac{B}{255}\right) \tag{4.17}$$

$$\text{where} f(x) = \begin{cases} 100 \times \left(\frac{x+0.055}{1.055}^{2.4}\right) & \text{, if } x > 0.04045 \\ 100 \times \left(\frac{x}{12.92}\right) & \text{, otherwise.} \end{cases} \tag{4.18}$$

Tristimulus values are calculated using the standards of Observer = 2°, Illuminant = D65. Using the calculated tristimulus values, the values of L*a*b* [57] are calculated as:

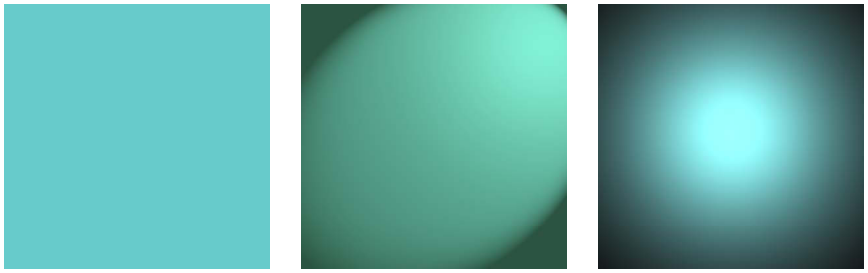
$$L^* = 116f\left(\frac{Y}{Y0}\right) - 16 \tag{4.19}$$

$$a^* = 500\left[f\left(\frac{X}{X0}\right) - \left(\frac{Y}{Y0}\right)\right] \tag{4.20}$$

$$b^* = 200\left[f\left(\frac{Y}{Y0}\right) - \left(\frac{Z}{Z0}\right)\right] \tag{4.21}$$

$$\text{where} f(x) = \begin{cases} x^{\frac{1}{3}} & \text{, if } x > 0.008856 \\ 7.787x + \frac{16}{116} & \text{, otherwise.} \end{cases} \tag{4.22}$$

The effect of non-uniform illumination on different colour-space components is investigated by using images with controlled illumination changes. The controlled illumination changes are achieved by adjusting an original image using simulated lighting effects rendered using digital software (Photoshop, lighting effect: 2 O'clock spotlight and soft omni). Figure 4.9*a* shows the original 400×400 pixels colour image with constant RGB values (103, 202, 203), Figure 4.9*b* shows the original image adjusted to simulate illumination created by a side directional light source, and Figure 4.9*c* shows the original image adjusted to simulate illumination created by a light source directly in front of the image plane.



(a) (b) (c)

FIGURE 4.9: *a*) Original image; *b*) Image adjusted to simulate illumination by a side directional light source; *c*) Image adjusted to simulate illumination by a light source directly in front of the image plane

The three images are converted into the L*a*b colour-space and the distribution of normalised values (i.e. within [0, 1]) in each colour-space component for L*a*b and RGB are shown in histograms. Figure 4.10 shows the histograms for the image with the simulated side directional light source, and Figure 4.11 shows the histograms for the image with the simulated light source directly in front of the image plane. The histograms show that the L*a*b colour-space is able to provide colour features with less variance. The colour-space components of the a* and b* have low distribution variance in comparison with the RGB colour-space components. The distribution variance for the colour component L* is higher in comparison to a* and b* which is expected because it represents the change in illumination intensities in the two images with simulated lighting. It is to note that the histograms for the image with simulated side directional lighting has a tall spike at the low intensity (Figure 4.10), this is due to the rendering process producing uniform colour intensity for the "shaded" area; surface region that is not illuminated by the light source.

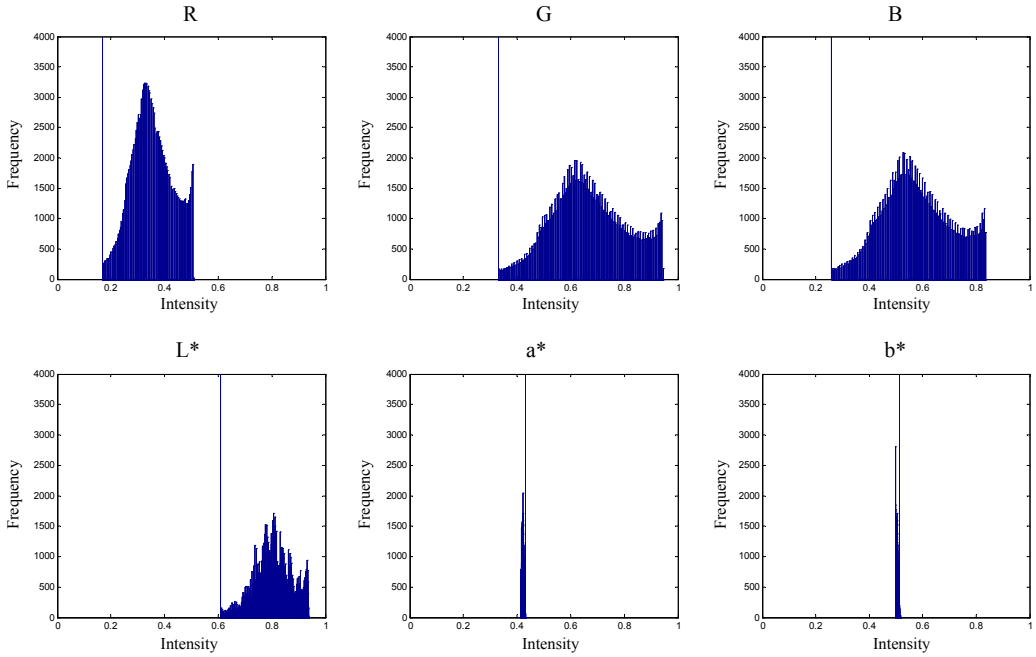


FIGURE 4.10: Histograms of colour-space components for image adjusted to simulate illumination by a side directional light source

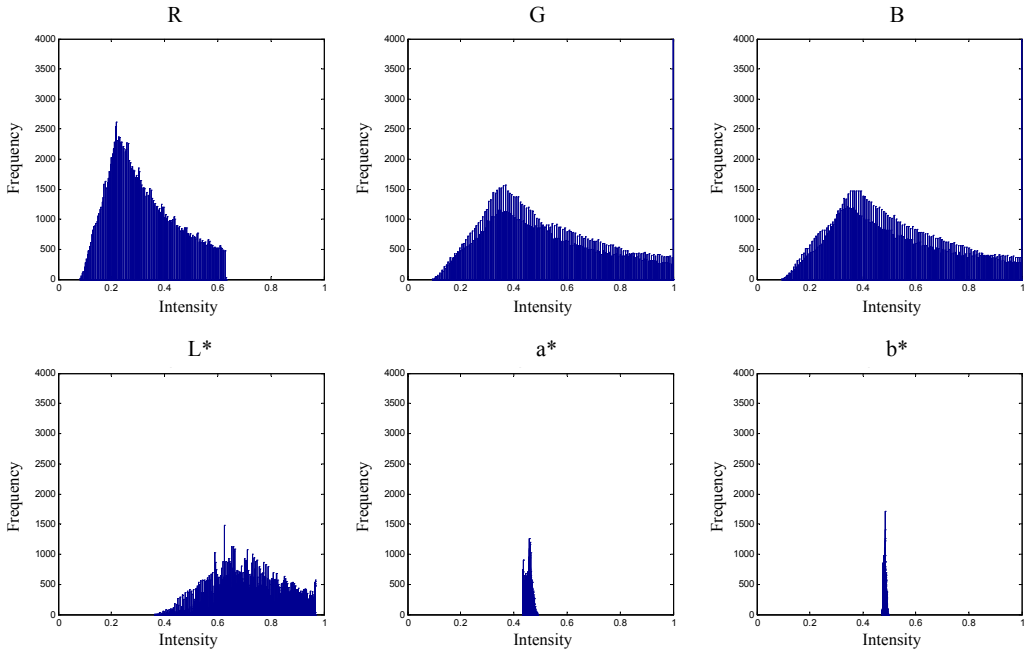


FIGURE 4.11: Histograms of colour-space components for image adjusted to simulate illumination by a light source directly in front of the image plane

Table 4.1 shows the mean and standard deviation for the normalised value distribution of

each colour-space component in the three images. The results also confirm that a* and b* have the lowest distribution variances, where the standard deviations are up to an order of magnitude lower than the RGB colour-space components. In addition, the change of the mean values for a* and b* from the original image, to the two images with simulated lighting is also lower in comparison to RGB. This shows that a* and b* can provide values with higher consistency in the presence of non-uniform illumination. Therefore, they will be used as a benchmark for subsequent experiments with the real image.

TABLE 4.1: Mean and standard deviation of colour-space component distribution

	Original image		Side directional light source		Light source directly in front	
	Mean	StdDev	Mean	StdDev	Mean	StdDev
R	0.403	0	0.3399	0.0883	0.3210	0.1291
G	0.792	0	0.6405	0.1597	0.5257	0.2291
B	0.796	0	0.5512	0.1501	0.5280	0.2298
L*	0.874	0	0.7945	0.0852	0.7304	0.1325
a*	0.424	0	0.4213	0.0038	0.4507	0.0109
b*	0.473	0	0.5037	0.0039	0.4814	0.0037

4.4 Experiment 1: Surface-type Classification of Images Containing a Single Surface Plane with Non-uniform Illumination

In this experiment, RGB-D images of four different surface-types are collected from a laboratory environment (Figure 4.12) using the RGB-D sensor package shown in Figure 4.13. The four surface-types include rusted metal surface, blasted metal surface, timber surface, and painted metal surface. For each surface-type, RGB-D images are collected at four different positions (one for training, and three for testing), as illustrated in Figure 4.14. The training images are captured at a viewing distance to surface of approximately 500 mm and a centre pixel viewing angle of incidence of 0°, and the three testing images are collected at viewing distances ranging between 500–1000 mm, and with viewing angles

in the range of approximately 0–60 degrees for the angle of incidence between the centre pixel ray-cast to the surface normal. Figure 4.15 shows the collected RGB images of the four surface-types that are collected under illumination from an LED light source only. Note that manual contrast adjustments have been applied to the rusted and blasted surface images in order for them to be viewable in Figure 4.15; the experimental classifications were performed using the unadjusted images. The aim of this experiment is to test the performance of the proposed diffuse reflectance features to provide generalisation of the surface-types even under various illumination conditions. This is achieved by training a classifier with limited image data captured from one viewpoint and then classifying images capture from other viewpoints; which changes the illumination on the surface. The classification results of the same surface are compared against the results produced using other colour-based features to observe the relative performance.

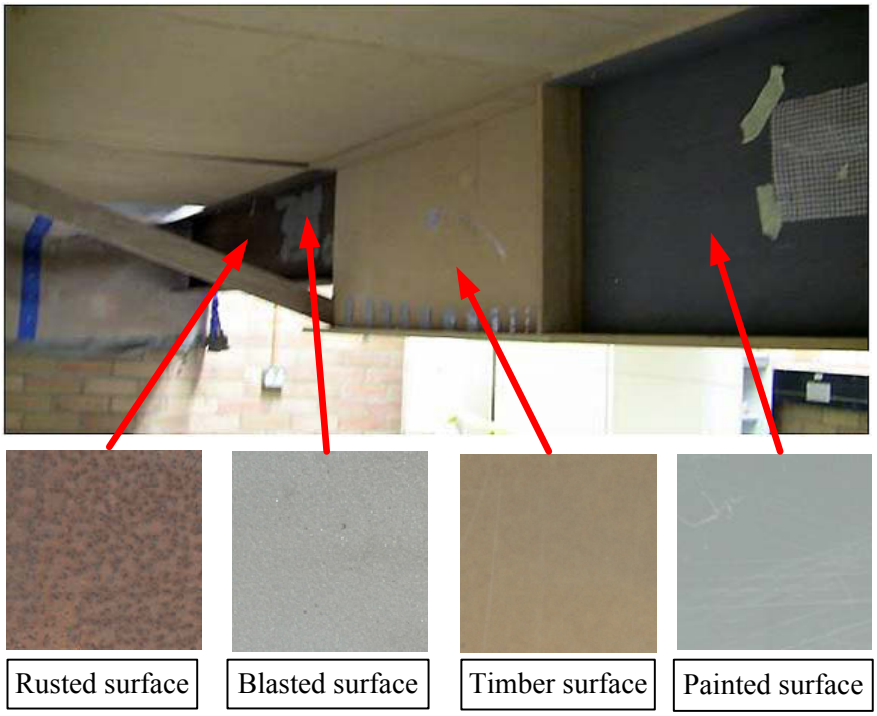


FIGURE 4.12: Experiment environment from which the four surface-types are collected

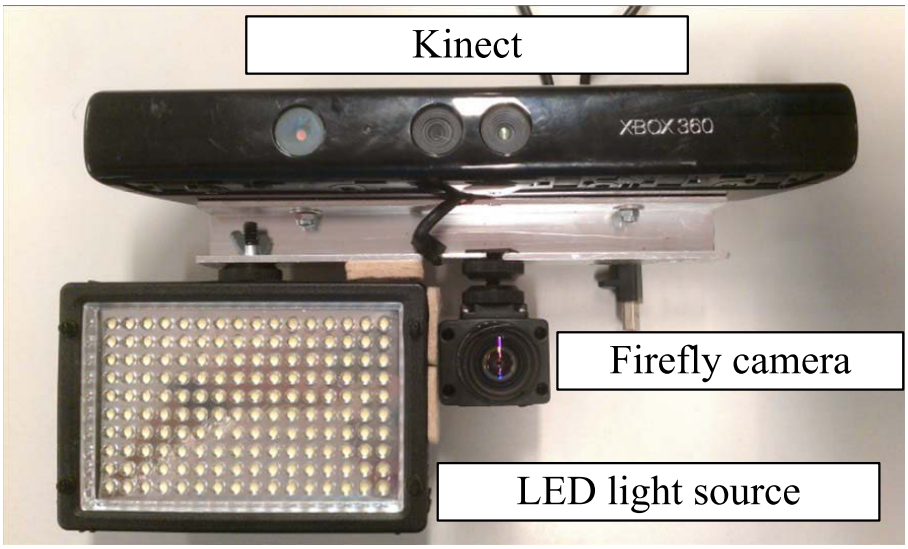


FIGURE 4.13: RGB-D sensor package consisting of a Kinect, Point Grey Firefly camera and LED light source

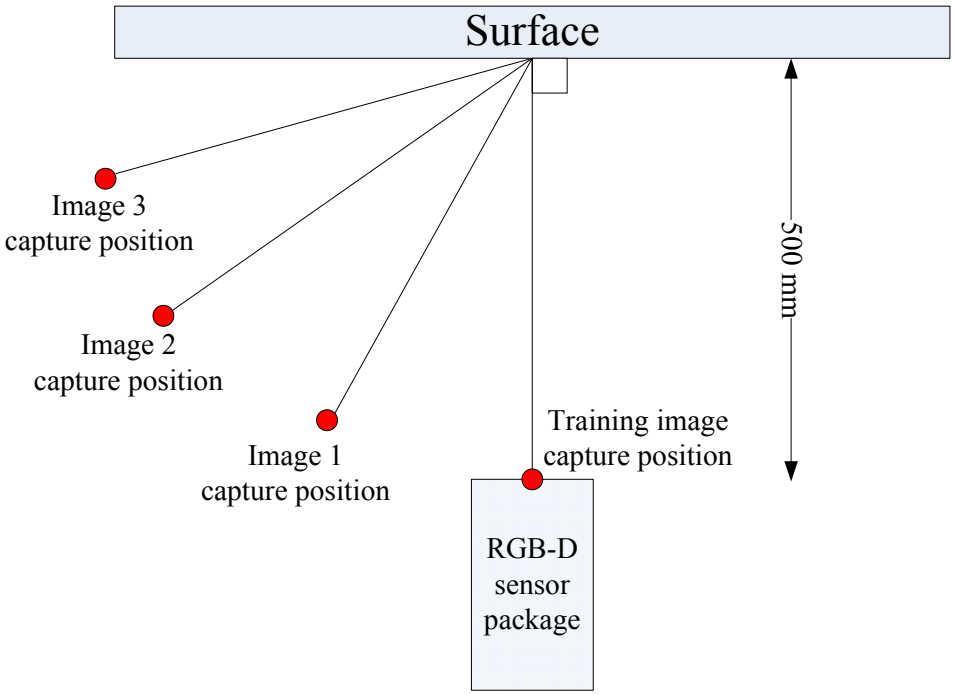


FIGURE 4.14: The different RGB-D image capture positions used to collect images under non-uniform illumination

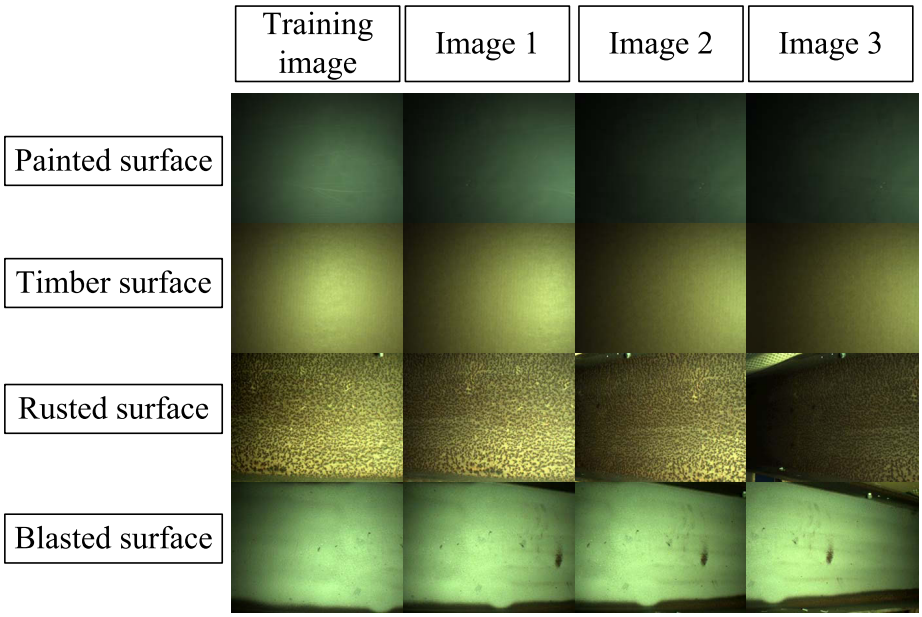


FIGURE 4.15: Training image and three test images

For each RGB-D image, (R,G,B) intensity values, diffused reflectance values $(K_{d,R}, K_{d,G}, K_{d,B}) \in K_d$, and L*a*b* colour-space components (a*, b*) are extracted from each image pixel and used as features for the four surface-type classifications. A multi-class SVM classifier (Appendix C) has been selected to classify the surface-types, where three separate multi-class SVM classifiers are trained with feature vectors (R,G,B), $(K_{d,R}, K_{d,G}, K_{d,B})$ and (a*, b*). The training RGB-D images collected for each surface-type is used to train the three SVM classifiers. The SVM classifiers are validated by classifying the training RGB-D images first, and subsequently applied to classify the three additional images collected for each surface-type.

Figures 4.16, 4.17, 4.18 and 4.19 provide a visualisation of the classification results for each of the surface-types. The colour scheme used in these figures is: blue = painted surface, teal = timber surface, yellow = rusted surface, and red = blasted surface. The change in illumination of images collected for each surface-type progressively increases from the Training image through to Image 3 as a result of the change in the LED light source position and the angle relative to the surface in the image. From Figure 4.16, it is observed that the classification accuracy decreases progressively from the Training image through to Image 3, which corresponds to the progressively increased change in

illumination. This is particularly prominent in the classification result for Image 3 when using RGB features for classification, which shows a large region of misclassification on the left side of the image where the light fall-off is the most significant. Classification results using K_d and a*b* features also show the same misclassification characteristics due to change in illumination, albeit less significant. The classification results shown for the other surface-types in Figures 4.17, 4.18 and 4.19 also exhibit the same misclassification characteristics due to change in illumination; again this is most significant when using RGB features. It is highlighted that the exposure settings on the camera (gain, aperture and shutter speed) are fixed for this experiment such that the changes in illumination intensity are dependent on only the light source's distance and angle to the surface. The high misclassification observed for image 3 (attributed to the illumination intensity of the surface being lower than that of the training data) is therefore a factor of the viewing distance and viewing angle. It is noted that due to the way the RGB-D sensor package is set up, the field of view of the depth camera does not completely cover the field of view of the RGB camera. As a result, the lower part of the RGB image does not have depth values. This is evident in Figure 4.16, which shows the results of the SVM classifier using K_d features with a misclassification of rust (yellow) on the lower part of the images due to null K_d values (given no depth values). Additionally, the size of the uncovered lower part is different for each sample due to the changes in viewing distance and viewing angle between samples, and slight difference when repeating the same viewpoints for capturing different surface-types. Therefore, manual cropping has been performed to omit the lower part of the RGB images. In addition, cropping to Images 2 and 3 for the rusted and blasted surfaces is also performed to omit classification results of unrelated surface-types in the image due to inadequate surface space of the surface-type to cover the whole image frame when the field of view is increased, as shown in Figure 4.15.

Table 4.2 shows the classification accuracy for each RGB-D image (with image cropping applied) using the three SVM classifiers; refer to Appendix E.1 for confusion matrices to all results. Overall, the classifier using K_d features demonstrated the highest accuracy, with accuracy ranging between (96.41–99.98%); followed by the classifier using a*b* features with accuracy ranging between (76.06–96.88%); and then the classifier using RGB features, with accuracy ranging between (15.99–98.70%). For each image that is classified, the

order of accuracy achieved from highest to lowest is K_d, a*b* and RGB, except for the timber surface images. For the timber surface images (training and testing), RGB-based classification results showed higher accuracy than the a*b*-based classification results. RGB features are able to distinguish timber surface with higher accuracy than the a*b* features.

In addition, from Table 4.2 it can be observed that a decrease in classification accuracy occurs from the Training image to Image 3, which corresponds to the increased change in illumination caused by the change in light source position. The most significant decrease in accuracy from the Training image to Image 3 was observed for the classifier trained with RGB features when classifying images of the rusted surface, showing a decrease from 56.03% to 15.99%. For the rusted surface images, the classifiers trained with K_d and a*b* features did not show a significant decrease in accuracy due to the change in illumination, where the results for all images maintained an accuracy above 90%.

In summary, this experiment has demonstrated the use of K_d values to provide more accurate classification of surface-types from images containing a single surface plane. The non-uniform illumination is caused primarily by the increased angle of incidence between the light source and the surface.

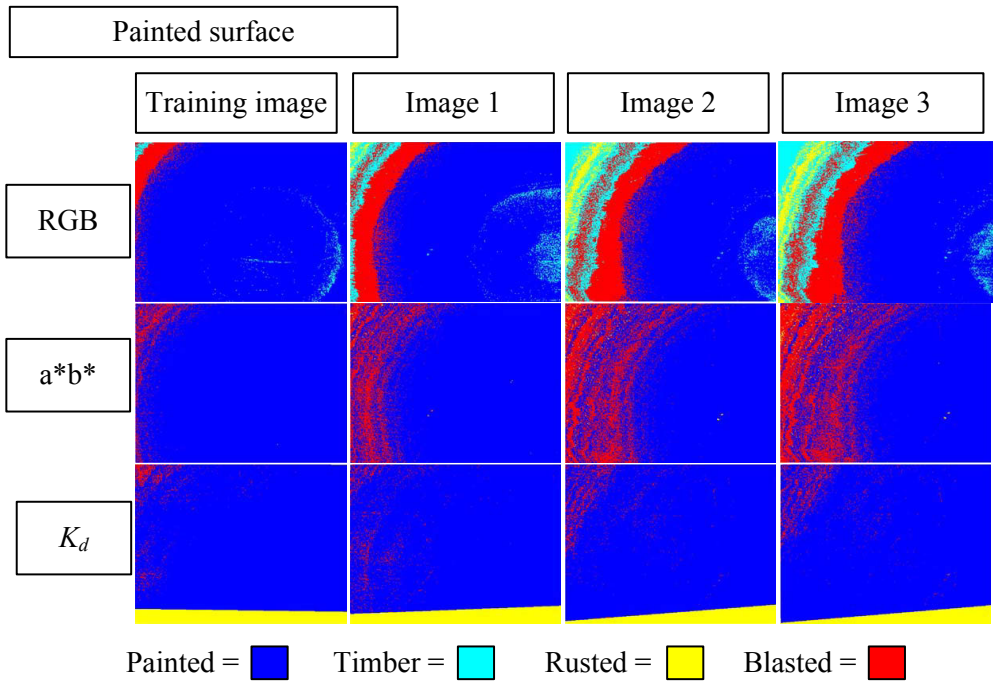


FIGURE 4.16: Painted surface classification results

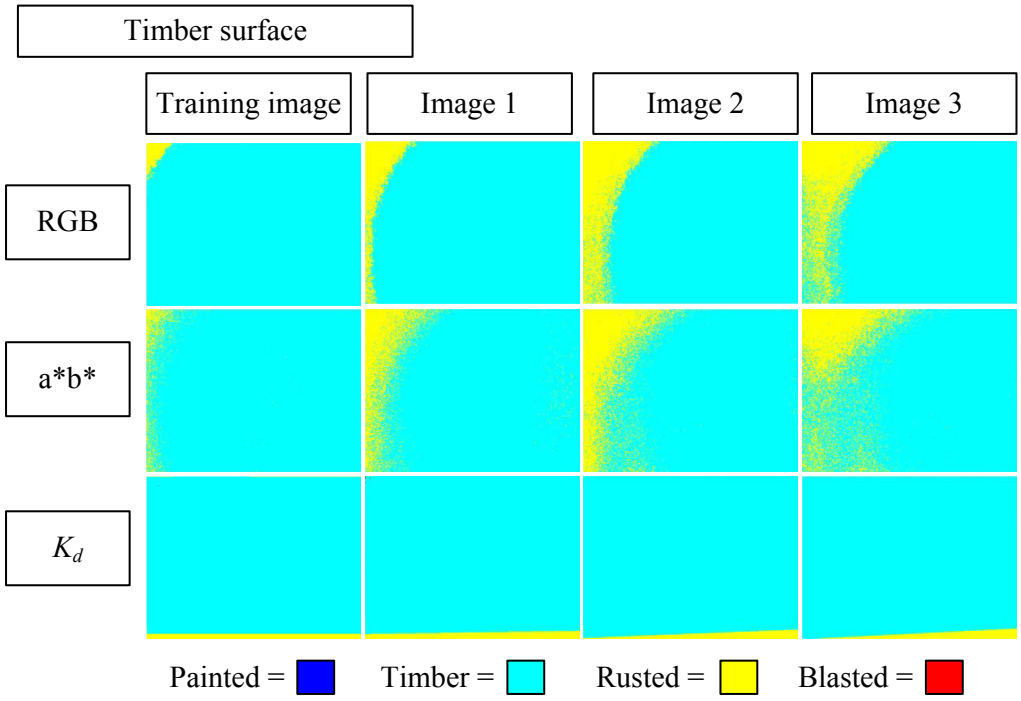


FIGURE 4.17: Timber surface classification results

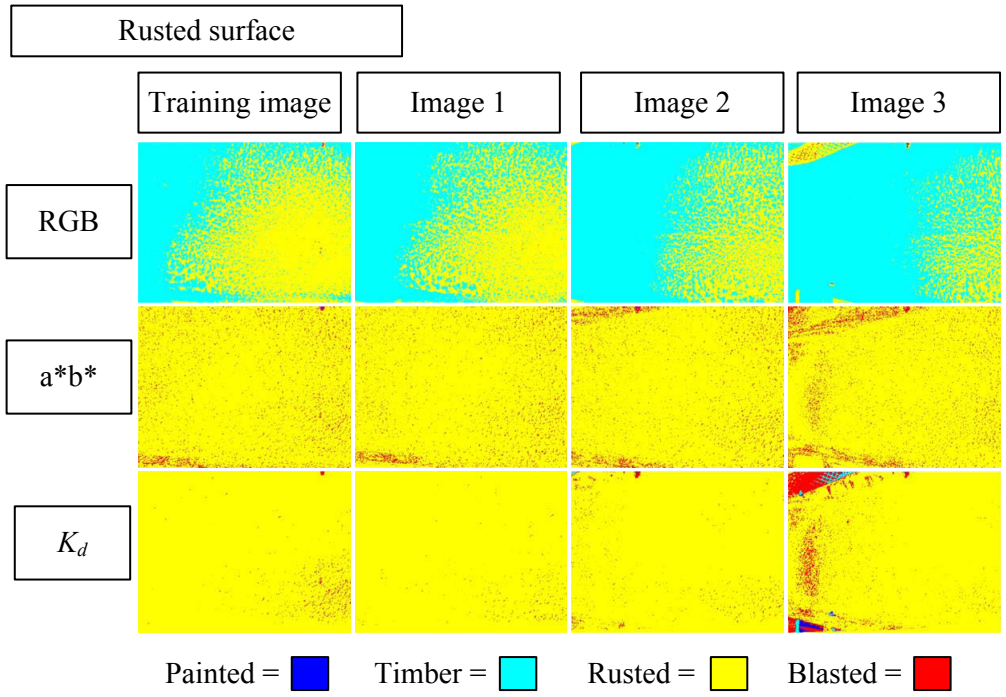


FIGURE 4.18: Rusted surface classification results

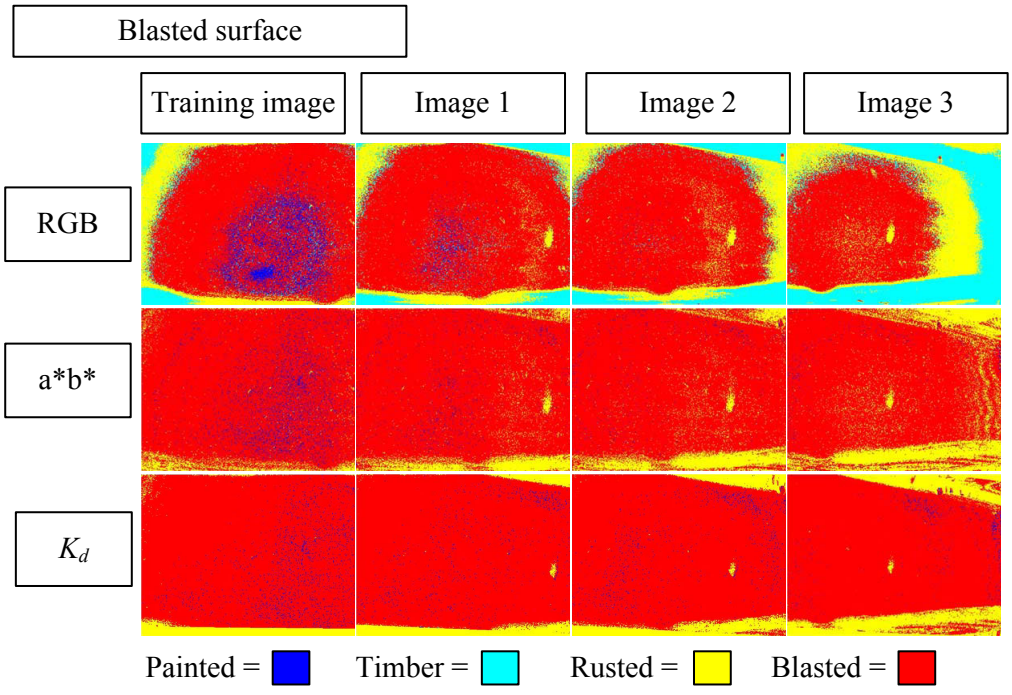


FIGURE 4.19: Blasted surface classification results

TABLE 4.2: Classification results for RGB-D images of a single surface plane

	Accuracy (%)			
	Training image	**Image 1**	**Image 2**	**Image 3**
Painted surface				
RGB	93.90	77.27	62.43	59.34
a*b*	96.21	87.57	78.13	76.06
K_d	97.70	97.19	96.90	96.93
Timber surface				
RGB	98.70	93.08	84.85	83.15
a*b*	96.88	87.93	81.27	81.24
K_d	99.85	99.96	99.89	99.98
Rusted surface				
RGB	54.03	46.34	30.12	15.99
a*b*	96.00	96.355	95.89	94.23
K_d	98.98	99.58	98.91	96.87
Blasted surface				
RGB	76.14	85.82	80.12	56.51
a*b*	92.27	95.61	95.60	91.93
K_d	96.41	98.25	97.94	98.36

4.5 Experiment 2: Surface-type Classification of an Image Containing Multiple Surface Planes with Non-uniform Illumination

In Experiment 2, surface-type classification is performed on an image that contains a complex scene captured in the environment shown in Figure 4.12. Figure 4.20*a* shows an image that contains two perpendicular surface planes with three surface-types, including a blasted metal surface, a rusted metal surface and a timber surface (image is contrast enhanced for viewing purposes). Figure 4.20*b* shows a segmented depth image which

contains the two perpendicular surface planes that are classified in this experiment. The non-uniform illumination in this image is caused by the two surface planes being perpendicular, such that the distance and angle of incidence changes significantly between the two planes. As a result, surface plane 1 (which is further away from the light source than surface plane 2) will have lower illumination intensity but the plane will be lit more uniformly because the angle of incidence with the light source is less. Surface plane 2 will be lit non-uniformly because the change in distance and angle of incidence with the light source is larger.

Surface-type classification is performed on the image (Figure 4.20a) using the three SVM classifiers trained with the training surface-type images in Experiment 1. The classification results produced by the three classifiers are compared by using manually labelled images (ground-truth) for each surface-type. Figure 4.21 shows the binary labelled images for each surface-type on the two surface planes, where the labelled surface-type is shown in white and the non-related surface-types are shown in black in each image.

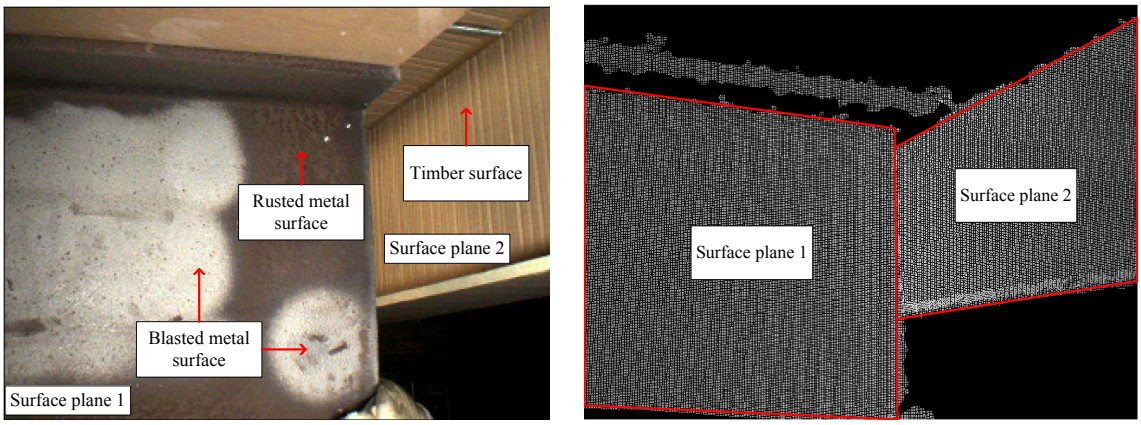
Figure 4.22 shows the visualisation of the surface-type classification results produced by the three SVM classifiers. The colour scheme used in these figures is: teal = timber surface, yellow = rusted surface, and red = blasted surface. Table 4.3 shows the confusion matrices of the surface-type classification results produced by the three SVM classifiers. The confusion matrix shows the classified class (column) produced by an SVM classifier versus the actual class (row) provided by the ground truth labelled images. From the confusion matrix, the classification accuracy of each surface-type can be calculated.

Overall, the SVM classifier trained with K_d features produced the highest classification accuracy for all three surface-types in the test image. From the confusion matrices, it can be seen that the highest classification accuracy achieved for the timber surface is 45.74% using K_d features; followed by the classifiers using RGB and a*b* features with 10.69% and 0.04%, respectively. From the visualisation of the classification results shown in Figure 4.22, the timber surface is misclassified mainly as rusted metal surface for RGB and a*b* feature-based classifiers, and blasted metal surface for the K_d feature-based classifier. For the two other classified surface-types of rusted metal surface and blasted metal surface, a*b* and K_d features produced results of 79.25%, 80.31% and 91.37%, 99.28%, respectively.

The classifier trained with RGB features produced the lowest accuracies in comparison for the rusted metal and blasted metal surfaces with 0.45% and 20.43%. This can be explained by referring to the classification result visualisation for the classifier using RGB features shown in Figure 4.22*a*. It can be seen that the classifier misclassified the majority of the rusted surface as timber surface, and the remaining image regions as rusted metal surface. Therefore, the classifier using RGB features resulted in low accuracy for all three surface-types.

Additional surface-type classification is performed on two more images collected in the environment at different viewpoints (shown in Figure 4.23). Table 4.4 shows the average classification accuracy for each surface-type in the environment. The average classification accuracy of each surface-type is calculated from the classification results of the three images (Figure 4.20*a* and Figure 4.23) produced using the three SVM classifiers. The results in the table indicate that on average, K_d has the highest accuracy, followed by a*b*, and finally RGB.

In summary, this experiment has demonstrated the classification of an image that contains a complex scene where the illumination intensity on the captured surface has changed from the training images used to classify the surface-types. The overall result in this experiment is similar to Experiment 1, which suggests that the diffused reflectance values K_d can provide higher classification accuracy when the captured images contain non-uniform illumination caused by change in light source distance and angle to the surface planes.



(a) (b)

FIGURE 4.20: *a*) Experiment 2 image that contains two surface planes and three surface-types; *b*) Depth image showing the segmented surface planes

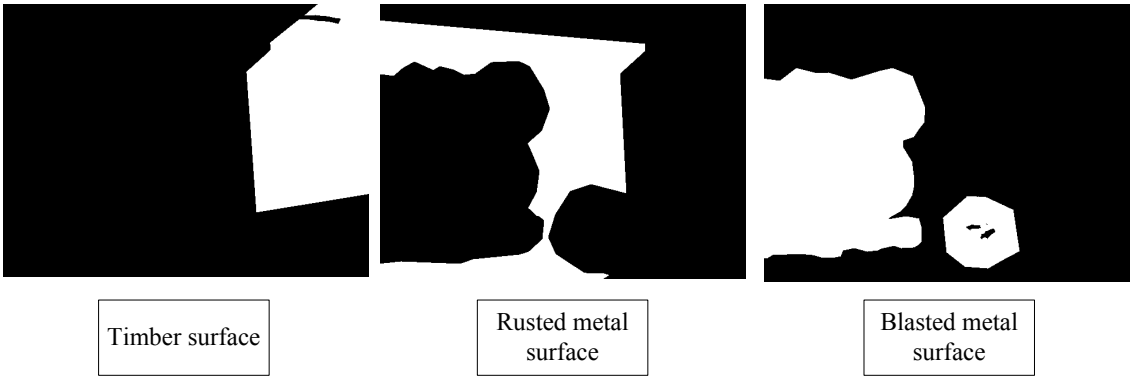
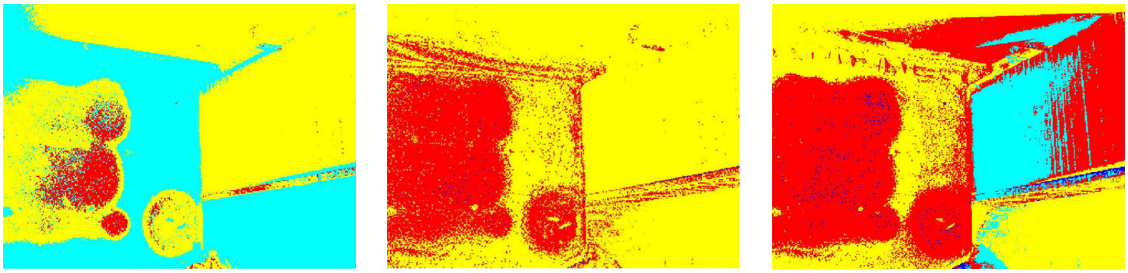


FIGURE 4.21: Binary ground truth labelled images for each surface-type. White is the surface-type, black is not the surface-type



(a) (b) (c)

FIGURE 4.22: *a*) Classification result using RGB features; *b*) Classification result using a*b* features; *c*) Classification result using K_d features. The colour scheme used in these figures are: teal = timber surface, yellow = rusted surface, and red = blasted surface

TABLE 4.3: Confusion matrices for the surface-type classification results produced using the three SVM classifiers

Confusion matrix: RGB				
Surface-types	Timber	Rusted	Blasted	Accuracy (%)
Timber	8823	72769	882	10.69
Rusted	59313	269	2	0.45
Blasted	8002	46969	14119	20.43
Confusion matrix: a*b*				
Surface-types	Timber	Rusted	Blasted	Accuracy (%)
Timber	34	78860	3395	0.04
Rusted	3	47194	12357	79.25
Blasted	0	5925	62737	91.37
Confusion matrix: K_d				
Surface-types	Timber	Rusted	Blasted	Accuracy (%)
Timber	37248	7433	36751	45.74
Rusted	21	47831	11702	80.31
Blasted	0	486	67128	99.28

FIGURE 4.23: Additional images collected in the environment

TABLE 4.4: Average surface-type classification results for images

	RGB	**a* b***	K_d
Average Timber Accuracy (%)	12.23	0.026	44.05
Average Rusted Accuracy (%)	14.13	80.45	82.54
Average Blasted Accuracy (%)	54.43	91.17	96.17
Overall Accuracy (%)	28.79	64.37	78.03

4.6 Discussion

In this chapter, the algorithm to perform extraction of new colour features from an image has been presented. The algorithm uses RGB-D images and a reflectance model to extract three diffused reflectance values for each RGB image pixel. The diffused reflectance values, as another type of colour feature, can be used to classify surface-types in images under various illumination conditions caused by the change in position and angle of a single light source relative to the surfaces shown in the images. Two experiments have been conducted to demonstrate the use of diffused reflectance values to classify images under various illumination conditions. The classification accuracy produced by an SVM classifier trained with diffused reflectance values is compared against two other SVM classifiers trained with RGB colour intensity feature, and CIELab a*b* colour-space features. The experimental results showed that the classifier trained with the diffused reflectance values has produced the highest accuracy amongst the three SVM classifiers.

It is understood that this algorithm requires accurate viewing distance and viewing angle for each image pixel, and it is challenging to calculate surface normals for points on complex structures with multiple small surface planes (i.e. bridge flanges and cross braces) given the clutter of the scene, sensor resolution and/or sensor noise. Therefore, a possible future work will be the use of a priori surface geometry knowledge of the environment to perform surface template matching. By fitting the depth values to precise models of surface structures, it is possible to calculate an accurate distance and surface normal for each pixel point on the surfaces (a detailed discussion is provided in Chapter 7).

In conclusion, this chapter has detailed the algorithm to extract colour features to perform per-pixel surface-type classification. In order to fully utilise the available RGB-D images to produce accurate surface-type classification results, texture features can also be extracted from the image and used as features for surface-type classification. The next chapter presents an algorithm to produce a probability map to assess the accuracy of texture-based classification results.

Chapter 5

Algorithm for Classification Result Assessment

Surface-type classification using texture features first involves the collection of a training image dataset that is a good generalisation of each surface-type. From the collected training image dataset, texture features can be extracted and used to train a classifier that will be used to classify the surface-types.

However, when images are captured in a complex structural environment under different image capture conditions (viewing distance and viewing angle), the appearance of a surface-type can vary texturally from image to image. This change in textural appearance can be attributed to image qualities, including focus quality, spatial resolution and perspective distortion. As such, the texture features extracted from these images can be inconsistent with the training image dataset and the classifier may not be able to accurately classify the surface-types in the images.

This chapter presents an algorithm to calculate a probability map of surface-types in an image, which can be used to assess the texture-based classification results of the image. The approach uses the change in image capture conditions between the image and the training image dataset to calculate the probability values in the map. The capture conditions of an image (viewing distance and viewing angle) are calculated using the depth data provided as part of the collected RGB-D information.

Experiments are conducted in a laboratory environment to verify the effect of image capture conditions on the consistency of the extracted texture features, and the effect of image capture condition changes on classification accuracy. The texture features are analysed using box plots to visualise the effect of image capture condition changes, and are used to train a classifier to classify for surface-types. A naive Bayes classifier is selected given that it builds the class models statistically and can provide a clear association between the box plot visualisation and classification accuracy. A final experiment is conducted to produce probability maps that are used to identify the accurately classified regions in the images. The work presented in this chapter was published in 'An Approach to Identifying Classifiable Regions of an Image taken by Autonomous Robots in Structural Environments, Robotics and Computer-Integrated Manufacturing' (refer to Section 1.5.1, point 2) and 'Image Segmentation for Surface Material-type Classification using 3D Geometry Information, International Conference on Information and Automation' (refer to Section 1.5.2, point 4)

5.1 Algorithm Overview

Figure 5.1 shows the procedure to calculate a probability map for a classified image. The preliminary processes of capturing RGB-D images and calculating the viewing distance (d_c) and the viewing angle (θ_c) for an RGB image have been presented in Chapter 3. For a classifier trained with texture features to classify surface-types, a calibration process is performed to calculate a set of parameter values (τ_n, τ_f, τ_θ, ω_1, ω_2). The calibration process is performed for each surface-type, M_k, for $k \in \{1, \ldots n_t\}$, given n_t number of surface-types. Each surface-type in the training image dataset has a unique set of parameter values ($\tau_{n_{M_k}}$, $\tau_{f_{M_k}}$, $\tau_{\theta_{M_k}}$, $\omega_{1_{M_k}}$, $\omega_{2_{M_k}}$). A unique set of parameter values is necessary for each surface-type, given that the image capture conditions can affect the classifier differently. For example, the classifier may be able to accurately classify surface-type 1 from images collected at a viewing distance in a range of 300 mm to 1000 mm, while surface-type 2 can only be accurately classified in a range of 300 mm to 500 mm.

Once the set of parameter values has been identified for a surface-type, it can be used to calculate the probability value of a pixel being the surface-type. Algorithm 2 shows

the pseudocode for calculating the probability value of image pixels in a classified image. For an image pixel classified as a surface-type class, the parameter values corresponding to that surface-type class are selected and used in a function to calculate the probability value. The process is repeated to calculate a probability value for all pixels in the image.

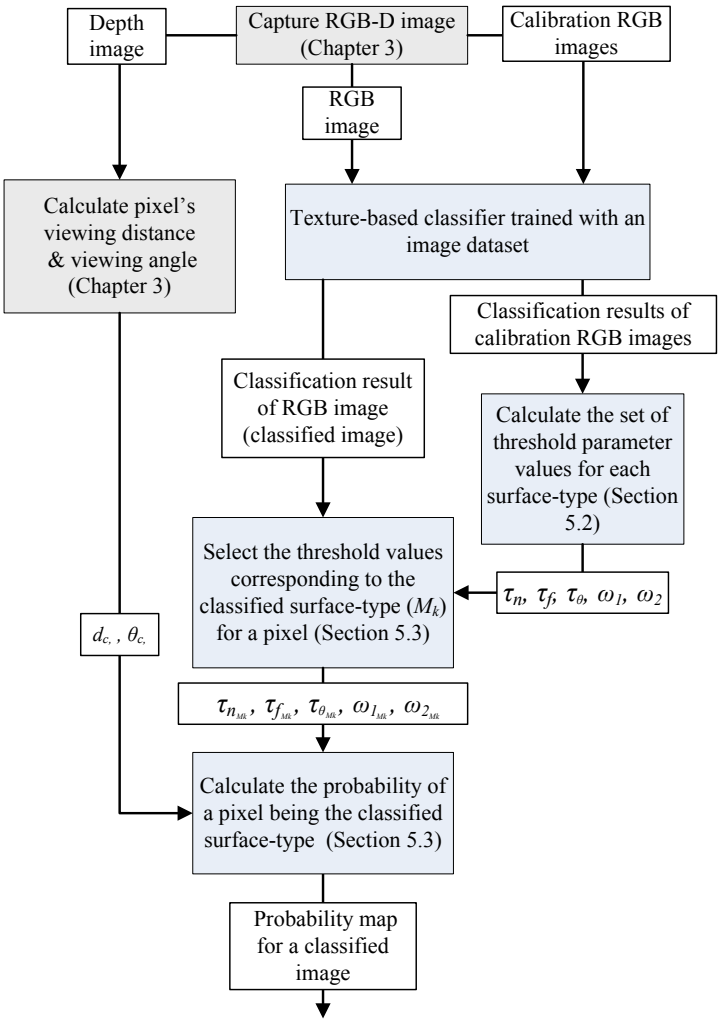


FIGURE 5.1: Algorithm to calculate a probability map of image pixels being a specific surface-type

Algorithm 2 Calculation of the probability value for each pixel in the classification result

Require: Parameter values of surface-types ($\tau_{n_{M_k}}$, $\tau_{f_{M_k}}$, $\tau_{\theta_{M_k}}$, $\omega_{1_{M_k}}$, $\omega_{2_{M_k}}$), and depth information d_{c_i}, θ_{c_i}

1: **for** pixel $i = 1$ to n in RGB-D image set $\{1, \dots n\}$ **do**
2: **if** pixel i has a valid viewing angle θ_{c_i} **then**
3: Get M_k for pixel i from classification result of image
4: $P_{d_{c_i}}$ = CalculateDistanceProbability(d_{c_i},$\tau_{n_{M_k}}$, $\tau_{f_{M_k}}$,$\omega_{1_{M_k}}$)
5: $P_{\theta_{c_i}}$ = CalculateAngleProbability(θ_{c_i},$\tau_{\theta_{M_k}}$,$\omega_{2_{M_k}}$)
6: $P_{d_{c_i},\theta_{c_i}} = P_{d_{c_i}} \times P_{\theta_{c_i}}$
7: **else**
8: $P_{d_{c_i},\theta_{c_i}}$ = null value
9: **end if**
10: **end for**

The following is a description of the functions used in the algorithm above:

- CalculateDistanceProbability(d_{c_i},$\tau_{n_{M_k}}$, $\tau_{f_{M_k}}$,$\omega_{1_{M_k}}$) calculates the probability of a pixel (at viewing distance, d_c) being a surface-type based on the formula provided in Equation 5.17
- CalculateAngleProbability(θ_{c_i},$\tau_{\theta_{M_k}}$,$\omega_{2_{M_k}}$) calculates the probability of a pixel (at viewing angle, θ_c) being a surface-type based on the formula provided in Equation 5.18

5.2 Image Capture Conditions

This section details the image capture conditions (viewing distance and viewing angle) and image qualities (focus quality, spatial resolution and perspective distortion) considered in the algorithm. The effect of image capture conditions on the texture appearance (appearance of a surface as described by the visual texture) of a surface-type in the images and the texture features extracted from these images are discussed. For the purpose of this investigate, we have chosen to use the GLCM texture features

including contrast, correlation, energy and homogeneity (Appendix B.1) to demonstrate the effect of image capture conditions on textural appearance; GLCM texture features have been used in previous literature to classify for different grades of rusting on steel structures [13] [9]. From the discussion, image capture condition threshold parameters are defined for each image quality including distance range for focus quality $\{D_{n_1}, D_{f_1}\}$, distance range for spatial resolution $\{D_{n_2}, D_{f_2}\}$, and viewing angle threshold for perspective distortion τ_θ. The thresholds are used to describe the maximum change in image capture conditions between a classified image and the training image dataset. The thresholds are used to provide the means to assess: (a) the similarity in texture appearance of a surface-type(s) in an image and the training image dataset; and (b) whether the texture appearance of a surface-type in an image has altered to resemble that of another surface-type in the training image dataset (i.e. the texture appearance of out-of-focus blasted metal surface can appear similar to a timber surface). The threshold parameters defined in this section are used in a function to calculate a pixel's probability of being a surface-type.

5.2.1 Focus Quality

Focus quality describes the acutance of an image region defined by geometrical optics, including the plane of focus and the depth-of-field (DOF). Figure 5.2 shows the plane of focus, which is the plane parallel to the image plane with ideal focus quality (i.e. maximum acutance level permitted by the lens quality); and DOF, which is the region in front and behind the plane of focus with acceptable focus quality. For texture-based classification, the DOF range can be used to identify the surfaces in an image with acceptable focus quality to extract texture features for accurate classification. The DOF range $\{D_{n_1}, D_{f_1}\}$ [99] is formulated with the following variables: lens focal length, F_c, lens f-number, δ, the circle of confusion, φ, and the plane of focus-to-camera distance, d_p, as:

$$D_{n_1} = \frac{F_c {d_p}^2}{{F_c}^2 + \delta\varphi(d_p - F_c)} \tag{5.1}$$

$$D_{f_1} = \frac{F_c {d_p}^2}{{F_c}^2 - \delta\varphi(d_p - F_c)}. \tag{5.2}$$

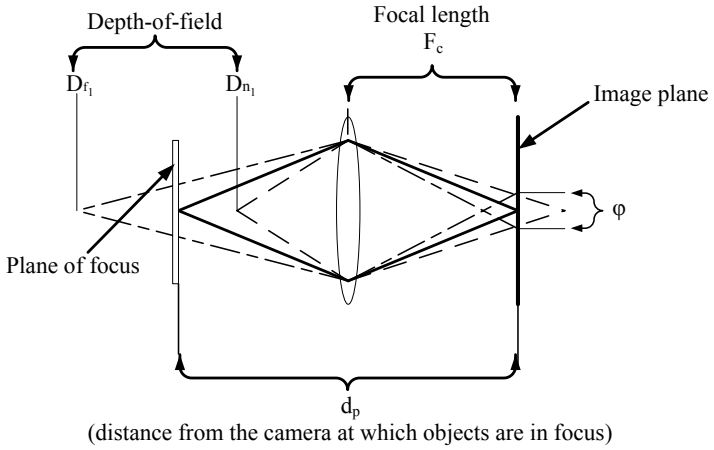
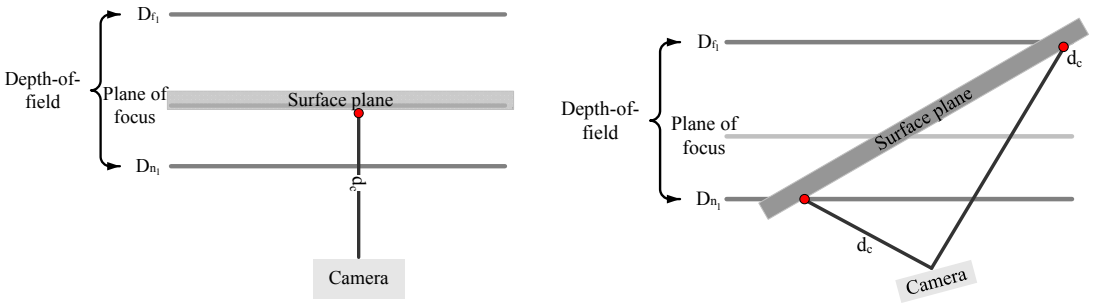


FIGURE 5.2: Plane of focus and depth-of-field diagram

The DOF range Equations 5.1 and 5.2 define the maximum and minimum viewing distances for surface planes captured in an image to have the acceptable focus quality necessary for texture-based classification. Figure 5.3*a* and Figure 5.3*b* show two scenarios where image pixels will have acceptable focus quality. In both scenarios, the viewing distance, d_c, of image pixels corresponding to a surface plane are within the DOF range D_{n_1} to D_{f_1}, as:

$$D_{n_1} \leq d_c \leq D_{f_1} \tag{5.3}$$



(a) (b)

FIGURE 5.3: *a*) Ideal pixel surface position within the DOF range; *b*) Pixel surface positions at the limits of the DOF range

The DOF range can vary from image to image due to optical adjustments made to the lens focal length, F_c, lens f-number, δ, and the plane of focus-to-camera distance, d_p.

However, the circle of confusion, φ, is a fixed value; and in optical terms, it is the value used to determine what is acceptable acutance in an image. It is set according to different image formats. For texture-based classification, φ, can be used to describe the texture acutance necessary for the extraction of texture features. The value of φ, is determined for a texture-based classifier by using a set of calibration images. The calibration images are captured with incrementally increasing plane of focus-to-camera distance (d_p) from the distance at which the camera is focused to, when capturing the training dataset (d_t). Classification is then performed on the calibration images and the results are examined to identify the plane of focus-to-camera distance d_p, where the classification accuracy falls below a predetermined accuracy percentage. This value of d_p, is taken as the maximum viewing distance, D_{f_1}, and a value for φ can be calculated using Equation 5.2 rewritten as:

$$\varphi = \frac{F_c {d_p}^2 - D_{f_1} {F_c}^2}{D_{f_1} \delta (d_p - F_c)}. \tag{5.4}$$

It is to note that either Equation 5.1 or 5.2 can be used for the calculation of the circle of confusion φ, and Equation 5.2 is arbitrarily selected for used.

5.2.2 Effect of Focus Quality on Texture Features

The texture features extracted from images with focus quality change are analysed using a checkerboard image $\Omega_t(u, v)$. Figure 5.4 shows the checkerboard image, where each black or white square is 20×20 pixels.

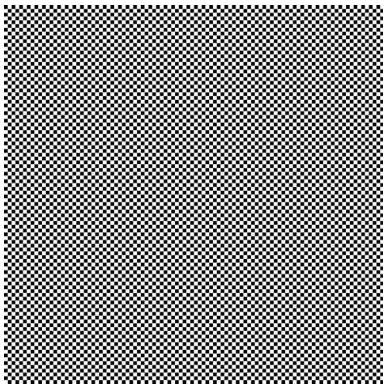


FIGURE 5.4: Checkerboard image $\Omega_t(u, v)$

A 2D Gaussian kernel $g(u, v)$ is convolved with $\Omega_t(u, v)$ to simulate the blurring effect produced by focus quality changes [100]. The 2D Gaussian kernel is expressed as:

$$g(u, v) = \frac{1}{2\pi {\beta_g}^2} \exp^{-\frac{u^2+v^2}{2\beta {g}^2}} \tag{5.5}$$

where β_g, is the variable that affects the amount of blur. A blurred image $\Omega_b(u, v)$ is produced as:

$$\Omega_b(u, v) = g(u, v) * \Omega_t(u, v). \tag{5.6}$$

Figure 5.5 shows 5 out of 11 Gaussian blurred images (1000×1000 pixels) produced by using different values of β_g (ranging from 0 to 10, at increments of 1) in Equation 5.6; applied to the checkerboard image (Figure 5.4). The following steps are performed on each of the 11 Gaussian blurred images: (1) Divide the image (1000×1000 pixels) into 25 equal samples of window size 200×200 pixels (window size is arbitrarily selected); (2) Extract from each sample, GLCM texture features including contrast, correlation, energy and homogeneity (Appendix B.1, GLCM texture features); (3) Plot the distribution of feature values (contrast, correlation, energy and homogeneity) into separate box plots. Figure 5.6 shows the box plots for each texture features. In each box plot diagram, the vertical axis is the feature value for a particular texture feature, and each increment in the horizontal axis corresponds with one of the Gaussian blurred image (11 in total). From the box plot diagrams, it can be seen that the extracted texture feature values have a small distribution range for each image but the values change across the images.

This also shows that the change in focus quality can affect the extracted texture feature values. In order to identify if focus quality change has affected the texture features such that they are inconsistent between an image and the training image dataset, the DOF viewing distance threshold range can be used. It is assumed that for image pixels with viewing distance d_c, that are within the DOF viewing distance threshold range $\{D_{n_1}, D_{f_1}\}$, the effect of focus quality is minimal on texture features.

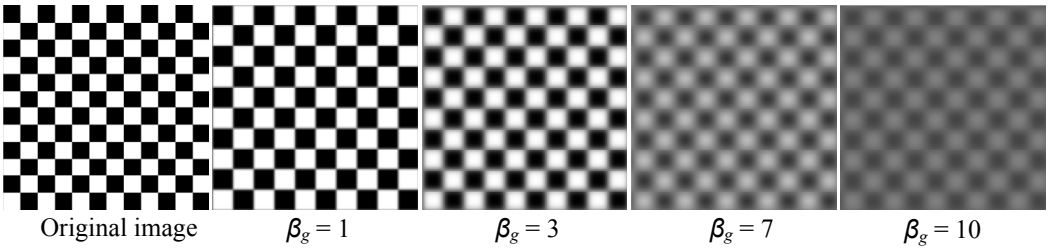


FIGURE 5.5: Gaussian blurred images of the checkerboard

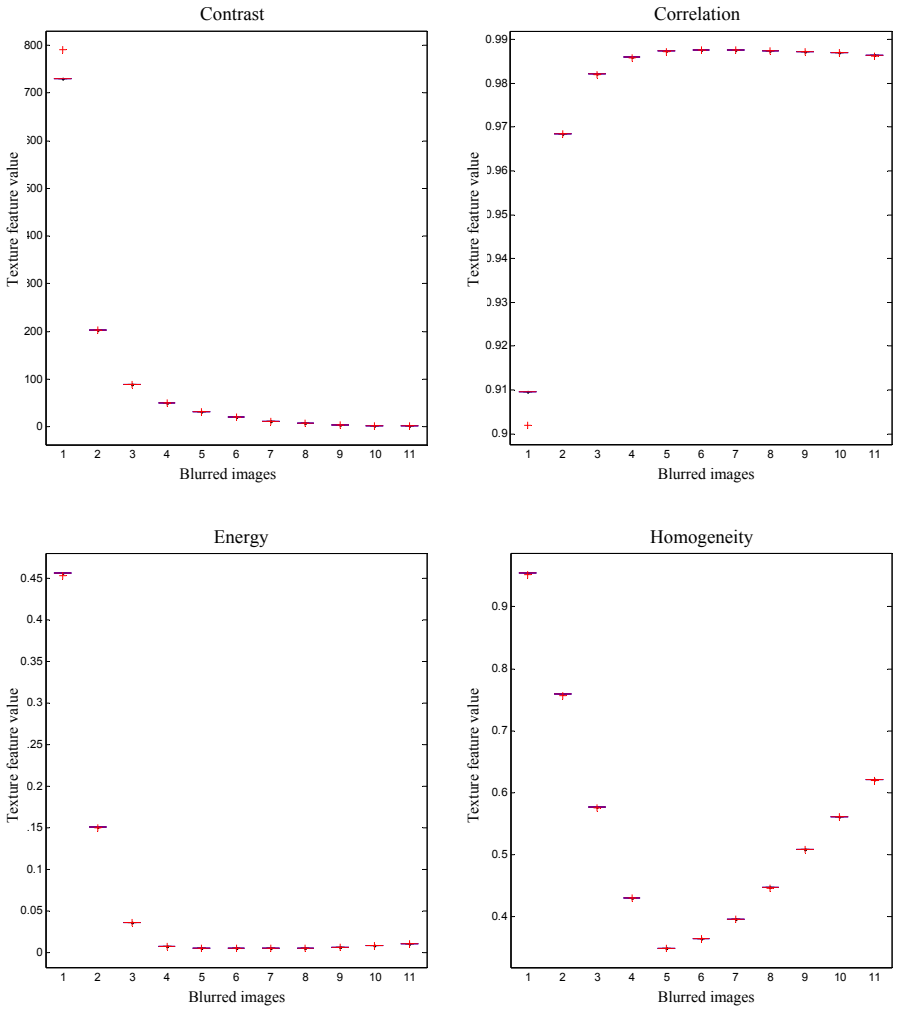


FIGURE 5.6: Box plot diagrams of the texture feature distribution extracted from the blurred images produced using different values of β_g

5.2.3 Spatial Resolution

Spatial resolution describes the number of pixels representing a surface area, i.e. pixels per square millimetre (PPM). Spatial resolution will affect the selection of a window size (n×n pixels) to extract texture features [101]. Therefore, when the window size is fixed, it is necessary to assess the spatial resolution between an image and the training image dataset (collected with a specific spatial resolution) to determine if the image can be accurately classified. In order to assess if the spatial resolution between an image and the training image dataset are different, the viewing distance (d_c) can be used.

Spatial resolution can be expressed in terms of viewing distance (d_c) by using a pinhole camera model analogy, where the image pixels are perspective projected into 3D space. Equations 5.7 and 5.8 are used to project an image pixel into the camera's 3D coordinate frame, where: i_u and i_v are the 2D coordinates of an image pixel, C_{c_u} and C_{c_v} are the camera's principle points, F_{c_u} and F_{c_v} are the focal lengths for the two axes of an image, and z_c, is the distance between the image plane and the surface point corresponding to an image pixel.

$$x_c = (i_u - C_{c_u}) \times \frac{z_c}{F_{c_u}} \tag{5.7}$$

$$y_c = (i_v - C_{c_v}) \times \frac{z_c}{F_{c_v}}. \tag{5.8}$$

Assuming $F_{c_u} \simeq F_{c_v}$ and $d_c \simeq z_c$, the surface area represented by an image pixel increases in proportion to the square of the viewing distance d_c. For example, substituting $F_{c_u} = 1$, $F_{c_v} = 1$, $(i_u - C_{c_u}) = 1$, and $(i_v - C_{c_v}) = 1$, then (x_c, y_c) position of a pixel when $z_c = 1, 2, 3$, etc, is (1, 1), (2, 2), (3, 3) etc. Therefore, the surface area represented by each pixel changes in proportion to the inverse square of the viewing distance d_c (as shown in Figure 5.7), and can be expressed as:

$$PPM \propto \frac{1}{{d_c}^2}. \tag{5.9}$$

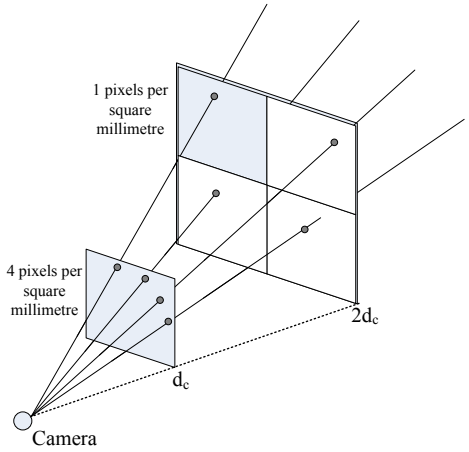


FIGURE 5.7: Example of pixel density on a surface relative to the viewing distance

For an image that is captured at a viewing distance d_c, and where the training image dataset is collected at a viewing distance d_t, the spatial resolution is assumed identical when $d_c = d_t$. A spatial resolution threshold range is defined, where D_{n_2}, is the minimal viewing distance from the training image dataset d_t, and D_{f_2}, is the maximum viewing distance from d_t. An image with a viewing distance of d_c, can be assessed to have similar spatial resolution as the training image dataset, as:

$$D_{n_2} \leq d_c \leq D_{f_2}. \tag{5.10}$$

The viewing distance range $\{D_{n_2}, D_{f_2}\}$ for spatial resolution is determined by using a set of calibration images. The calibration images are captured at various viewing distances (d_c) from the viewing distance used to collect the training image dataset (d_t). Classification is performed on the calibration images and the results are examined to identify the two viewing distance values D_{n_2} and D_{f_2}. From the two values, the one with the closer viewing distance is taken as D_{n_2} and the further viewing distance is taken as D_{f_2}. Within this range $\{D_{n_2}, D_{f_2}\}$, the classification accuracy is above the predefined accuracy threshold.

5.2.4 Effect of Spatial Resolution on Texture Features

The effect of changing spatial resolution on GLCM texture features is analysed by using the texture pattern of a checkerboard, $\Omega_t(u, v)$, shown in Figure 5.4. A scaling coefficient (β_s) is applied to both axes of the checkerboard image to produce images with different spatial scales. The scaling of the pixels from the original image $\Omega_t(u, v)$ into the scaled image $\Omega_t(u', v')$ is expressed as:

$$u' = \|\beta_s u\| \tag{5.11}$$

$$v' = \|\beta_s v\|. \tag{5.12}$$

When upscaling the checkerboard image, a nearest-neighbour interpolation algorithm is applied to generate the values of the new pixels in the upscaled image. Figure 5.8 shows 5 out of the 11 (1000×1000 pixels) images generated using different scaling coefficient values of β_s, ranging from 0.5 to 1.5 at 0.1 increments. Texture features including contrast, correlation, energy and homogeneity are extracted from each scaled image by dividing the image into equal size samples of 200×200 pixels. Figure 5.9 shows the box plot diagrams of value distribution of texture features extracted from the samples. In each box plot diagram, the vertical axis shows the numerical value for the texture feature and the horizontal axis shows the 11 scaled images. From the box plot diagrams, it can be seen that the value distribution for the texture features changes across the different scaled images. This shows that the change in spatial resolution simulated by scaling the texture pattern can affect the consistency of the extracted texture features. Hence, it is necessary to identify if spatial resolution has affected the consistency of texture features. It is noted that scaled image 6 has significantly higher energy and homogeneity values as can be observed from Figure 5.9, this is due to the size of each individual checkerboard black and white cells being 200×200 pixels; which matches with the sampling window size. As a result each sample taken from scaled image 6 contains only black or white pixel values; which results in high energy and homogeneity values.

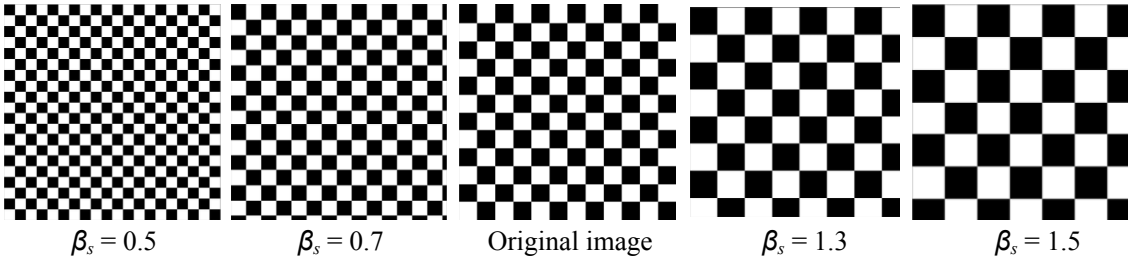


FIGURE 5.8: Upscale and downscale images of the checkerboard image to simulate change in spatial resolution when using a fixed pixel window size to extract texture features

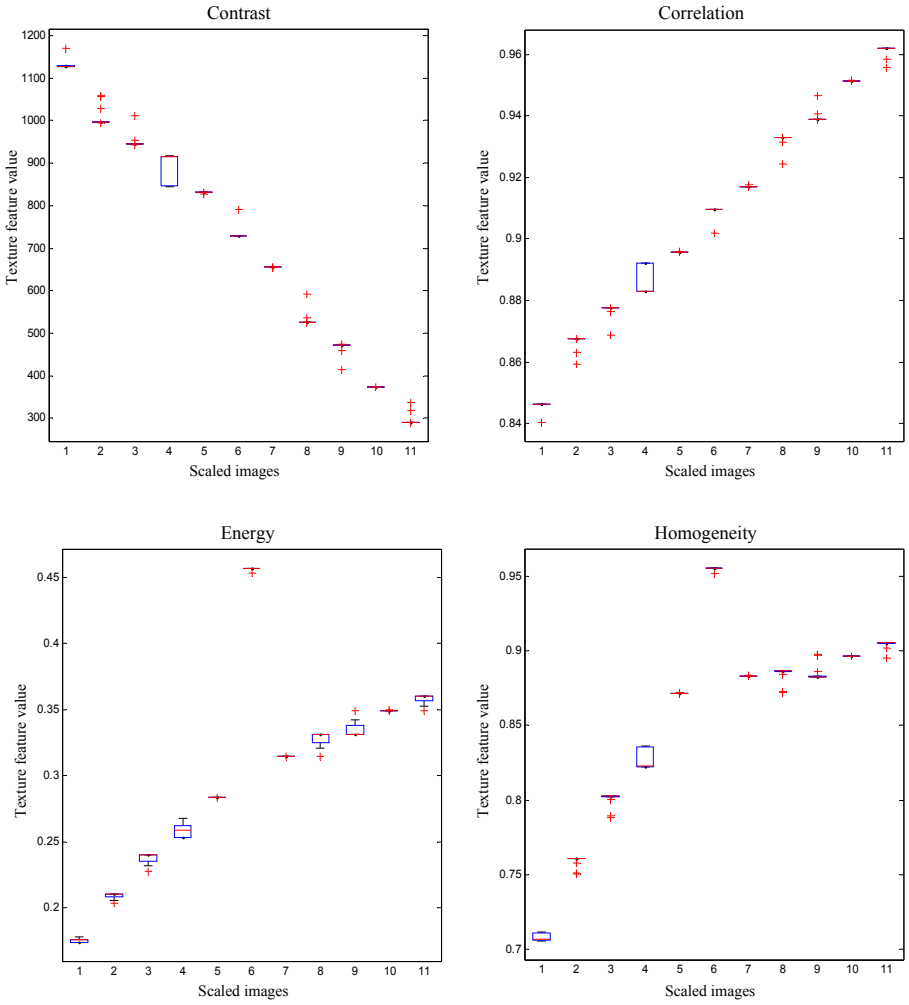


FIGURE 5.9: Box plot diagrams of the texture feature distribution extracted from the scaled images produced using different values of β_s

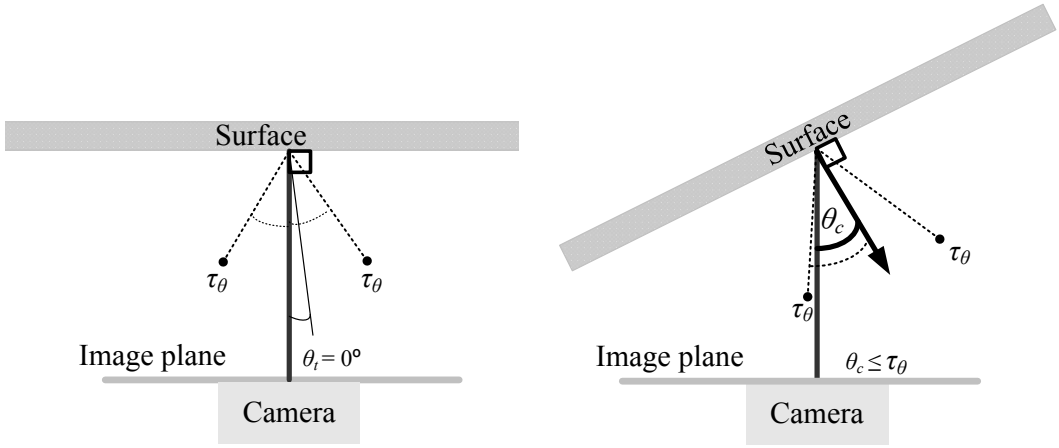
5.2.5 Perspective Distortion

Perspective distortion describes the change in the textural appearance of a surface-type as a result of the viewing angle at which the surface plane is captured by the camera. Perspective distortion in an image can affect the consistency of the extracted texture features, and thus adversely affect the classification accuracy. Therefore, it is necessary to assess the perspective distortion in order to identify its effect on texture features.

Minimal perspective distortion is achieved when the viewing direction of the camera is perpendicular to the surface plane. Therefore, to ensure that perspective distortion is minimised in the training dataset, the viewing angle selected to collect the training image dataset, θ_t, is zero; θ_c is the viewing angle used to capture subsequent testing images. Figure 5.10*a* illustrates the viewing angle $\theta_t = 0°$ that is selected to collect the training image dataset, and also the viewing angle threshold (τ_θ) which defines the maximum viewing angle that can be used to collect subsequent testing images. Testing images collected within the viewing angle threshold (τ_θ) are assumed to have negligible perspective distortion, thus are classifiable by a classifier trained with the training dataset. Figure 5.10*b* shows an example which satisfies this condition, where the viewing angle used to collect the test image is θ_c, is within the viewing angle threshold τ_θ, as:

$$\theta_c \leq \tau_\theta \tag{5.13}$$

In order to determine a suitable value for the viewing angle threshold, τ_θ, which is dependent on the surface-type(s) to be classified, a set of calibration images is used. The calibration images are captured by positioning the camera at various viewing angles, θ_c. Classification is performed on the calibration images using the trained classifier and the results are examined to identify the viewing angle at which the classification accuracy falls below an acceptable (predetermined) accuracy percentage. This value is taken as the viewing angle threshold τ_θ.



(a) (b)

FIGURE 5.10: *a*) The camera viewing angle used to capture the training dataset, θ_t, and the viewing angle threshold, τ_θ; *b*) An example of a camera viewing angle that is within the viewing angle threshold

5.2.6 Effect of Perspective Distortion on Texture Features

The texture feature values extracted from images with perspective distortion are also analysed using the texture pattern from a checkerboard image $\Omega_t(u, v)$ shown in Figure 5.4. Given the simple texture pattern being used (straight parallel lines and homogeneous), perspective distortion caused by viewing angle is sufficiently approximated using a shearing transformation [102]; for more complex patterns planar homography can be applied [103]. A shearing coefficient, β_k, is applied to both axes of $\Omega_t(u, v)$ to generate a distorted image $\Omega_t(u', v')$, as:

$$u' = \|u + \beta_k y\| \tag{5.14}$$

$$v' = \|\beta_k u + v\|. \tag{5.15}$$

Figure 5.11 shows 5 out of 10 (1000×1000 pixels) distorted images produced with different values of β_k, ranging from 0.1 to 0.5 at 0.05 increments. Texture features including contrast, correlation, energy and homogeneity are extracted from each distorted image by dividing the image into equal size samples of 200×200 pixels. Figure 5.12 shows the box plot diagrams of the value distribution for each texture feature. In each box plot diagram, the vertical axis shows the numerical value for the texture feature and the horizontal axis shows the 10 distorted images. From the box plot diagrams, it

can be seen that the extracted texture feature values have a small distribution range, but the values are inconsistent across the distorted images. Therefore, it is necessary to assess if perspective distortion has affected the consistency of texture features. The viewing angle threshold (τ_θ) can be used.

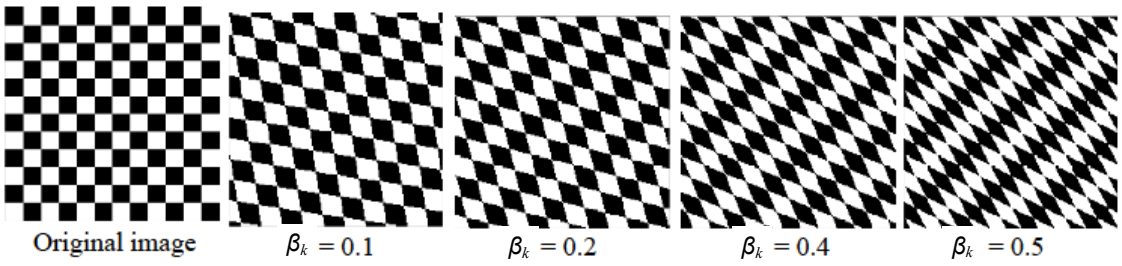


FIGURE 5.11: Distorted images of the checkerboard

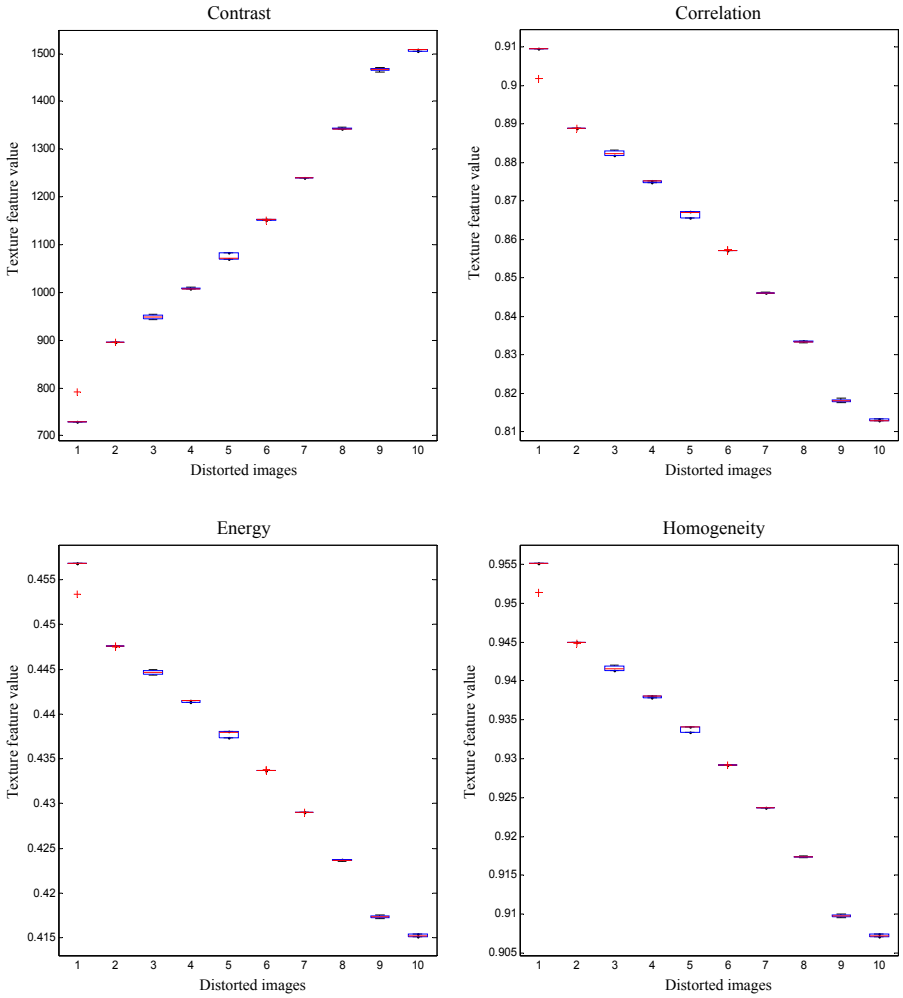


FIGURE 5.12: Box plot diagrams of the texture feature distribution extracted from the distorted images produced using different values of β_k

5.3 Calculation of a Probability Map

From Section 5.2, image capture condition parameters have been defined, including the viewing distance range for DOF $\{D_{n_1}, D_{f_1}\}$, spatial resolution $\{D_{n_2}, D_{f_2}\}$, and the viewing angle threshold for perspective distortion (τ_θ). This section discusses the use of these parameters to calculate the probability for an image pixel being a surface-type. The process of using the function to generate a probability map for a classified image is also detailed. The probability map contains the probability of pixels in an RGB image being a

specific surface-type and can be used to weight the accuracy of classification results when texture features are used to classify.

Combining Equations 5.3 and 5.10, we have:

$$\tau_n \leq d_c \leq \tau_f \tag{5.16}$$

where $\tau_n = max\{D_{n_1}, D_{n_2}\}$, and $\tau_f = min\{D_{f_1}, D_{f_2}\}$. Given the viewing distance range (τ_n, τ_f), the probability value (P_{d_c}) of an image pixel being a surface-type is calculated using a sigmoid function as:

$$P_{d_c} = \frac{1}{1 + e^{w_1(\tau_n - d_c)}} - \frac{1}{1 + e^{w_1(\tau_f - d_c)}} \tag{5.17}$$

where ω_1 is a weighting coefficient to define the sigmoid curve around the viewing distance range (τ_n, τ_f). The transition slope between low and high probability values is affected by the threshold value selected for ω_1. Figure 5.13 shows an example of the sigmoid function used to calculate the probability value, P_{d_c}, for an image pixel.

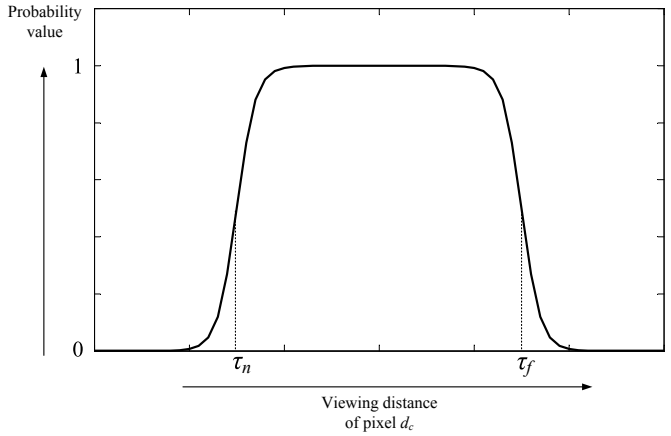


FIGURE 5.13: Sigmoid function to calculate the probability value of a pixel based on the viewing distance

For perspective distortion given the viewing angle threshold τ_θ, the probability value P_{θ_c} of an image pixel being a surface-type is calculated using a sigmoid function as:

$$P_{\theta_c} = 1 - \frac{1}{1 + e^{w_2(\tau_\theta - \theta_c)}} \tag{5.18}$$

where ω_2 is a weighting coefficient to define the sigmoid curve around the viewing angle threshold τ_θ. The transition slope between low and high probability values is affected by the threshold value selected for ω_2. Figure 5.14 shows an example of the sigmoid function used to calculate the probability value P_{θ_c}.

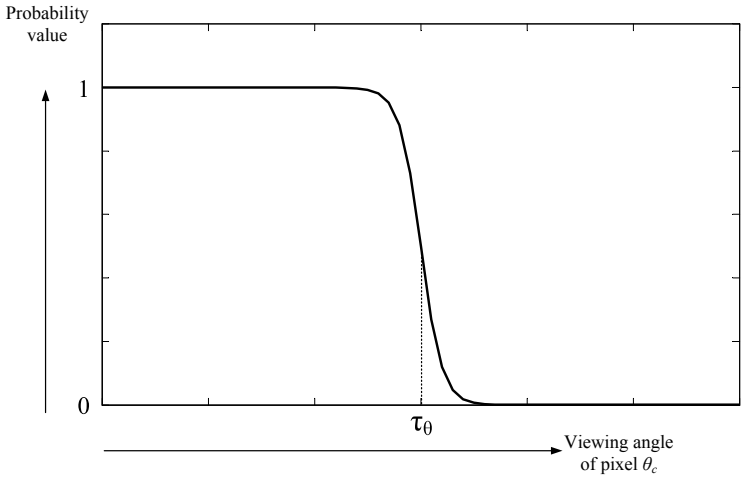


FIGURE 5.14: Sigmoid function to calculate the probability value of a pixel based on the viewing angle

From the probability values calculated based on viewing distance (P_{d_c}) and viewing angle (P_{θ_c}) the combined probability value, P_{d_c,θ_c}, is calculated as:

$$P_{d_c,\theta_c} = P_{d_c} \times P_{\theta_c}. \tag{5.19}$$

The value of P_{d_c,θ_c} is used to represent the probability of a pixel being a surface-type based on image capture conditions. Figure 5.15 shows a visualisation of the calculated probability value P_{d_c,θ_c} with respect to image capture conditions in viewing distance d_c, and viewing angle θ_c.

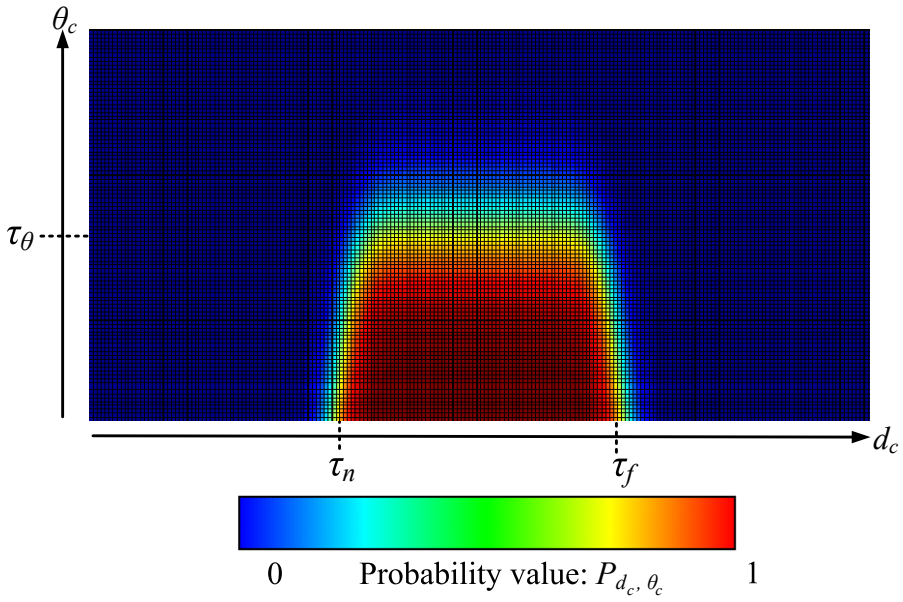


FIGURE 5.15: Visualisation of the probability value P_{d_c,θ_c} with image capture condition changes in viewing distance d_c, and viewing angle θ_c

Each surface-type in a training image dataset requires calibration to produce a unique set of parameter values for calculating the probability value ($\tau_{n_{M_k}}$, $\tau_{f_{M_k}}$, $\tau_{\theta_{M_k}}$, $\omega_{1_{M_k}}$, $\omega_{2_{M_k}}$). This is necessary given that the classification accuracy of each surface-type can be affected differently by image capture condition changes. Figure 5.16 shows the procedure of generating a probability map for a classified image.

For a pixel in an image classified as surface-type M_k, the threshold parameter values for the surface-type ($\tau_{n_{M_k}}$, $\tau_{f_{M_k}}$, $\tau_{\theta_{M_k}}$, $\omega_{1_{M_k}}$, $\omega_{2_{M_k}}$) are used to calculate the probability value. This process is repeated until all the pixels in an image are assessed, and a probability map is generated.

The generated probability map can be used to identify regions in a classified image that have a high probability of being accurately classified. Additionally, a probability map can be used to provide a weighting factor for the classification result when combining into a surface-type map (detailed in Section 3.4.1 and demonstrated in Chapter 6).

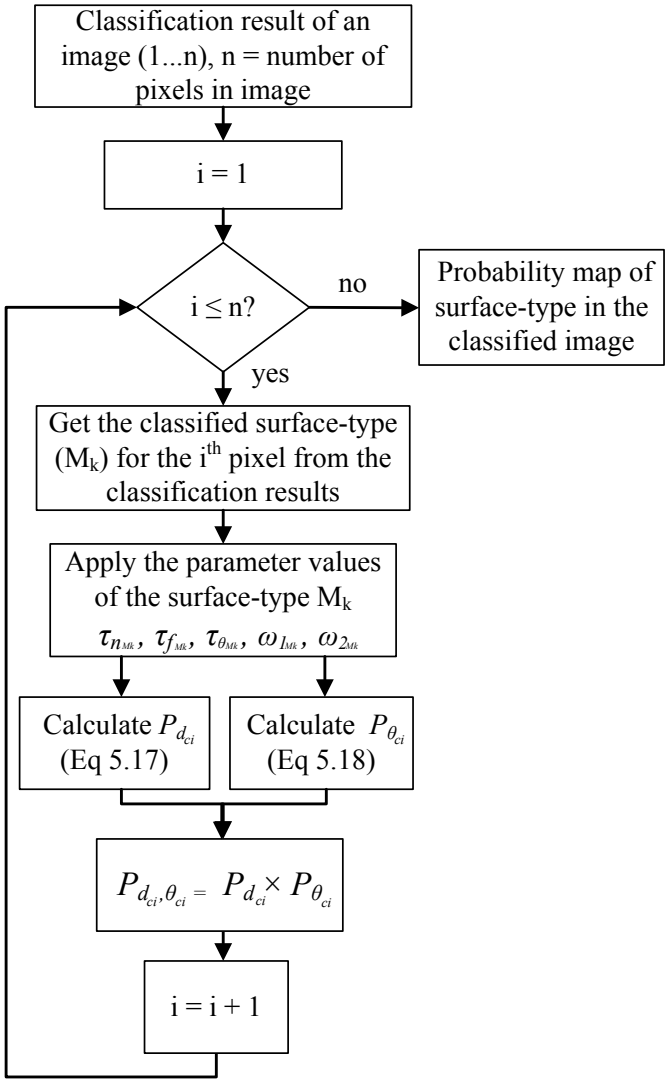


FIGURE 5.16: Procedure for calculating the probability value of the classification results of an image

5.4 Experiments

Three experiments are conducted in this chapter to verify the algorithm. The first experiment is performed to experimentally verify the effect of image capture conditions on the consistency of extracted texture features. The second experiment is performed to demonstrate the effect of image capture conditions on the classification accuracy. The

third experiment is performed to demonstrate the use of probability maps to identify regions in images that have a higher probability of being accurately classified.

5.4.1 Experiment 1: Effect of Image Capture Conditions on Texture Features

In this experiment, the effect of image capture condition in terms of image qualities (focus quality, spatial resolution and perspective distortion) on texture feature extraction is experimentally verified. Figure 5.17 shows the camera setup and the surface-type in this experiment. The images of the surface are captured under different image capture conditions to produce three sets of 1280×960 pixels images (Cropped to 960×960 pixels square images). Each set of images has changes to one image quality, i.e. image set one contains images with only focus distance changes. Texture features including contrast, correlation, energy and homogeneity are extracted from each image by dividing into 100 samples (manually selected size), therefore a sampling window size of 96×96 pixels is used. The value distribution of the extracted texture features are visualised and statistically compared using box plot diagrams. The experiment performed on the three sets of images is outlined separately as follows.

(1) Figure 5.18 illustrates the image capture conditions used to collect a set of images with focus distance changes. The camera is set at a fixed viewing distance $d_c = 100$ mm, and a viewing angle $\theta_c = 0^\circ$ to the surface plane. The plane of focus, d_p, is changed in 10 mm increments between a range of 30 mm to 170 mm by adjusting the focus on the camera lens, where an image is captured at each increment. Figure 5.19 shows the captured set of images with focus distance changes, and Figure 5.20 shows the box plot diagrams of the value distribution of each texture feature. In each box plot diagram, the vertical axis shows the numerical value for the texture feature and the horizontal axis shows the 15 captured images with plane of focus changes ranging from 30 mm to 170 mm corresponding to images 1–15.

From the box plot diagrams shown in Figure 5.20, it can be seen that the extracted texture features are inconsistent across the images, in which the most observable results are demonstrated for the texture features of energy and correlation. For the texture feature

of energy, the mean value for image 1 is 0.0224 and the mean for image 15 is 0.3600. A two-sample t-test (i.e. using Matlab function ttest2) between image 1 and image 15 supports the hypothesis that the distributions come from populations with unequal means. Similarly, for the texture feature of correlation, the mean for image 1 is 0.9680 and the mean for image 15 is 0.5648; and the result of a two-sample t-test between image 1 and image 15 also supports the hypothesis that the two distributions come from populations with unequal means. In summary, given the change in focus distance produced by varying the plane of focus (d_p) in the range of 30 mm to 170 mm, inconsistencies are observed for the extracted texture features across the set of images.

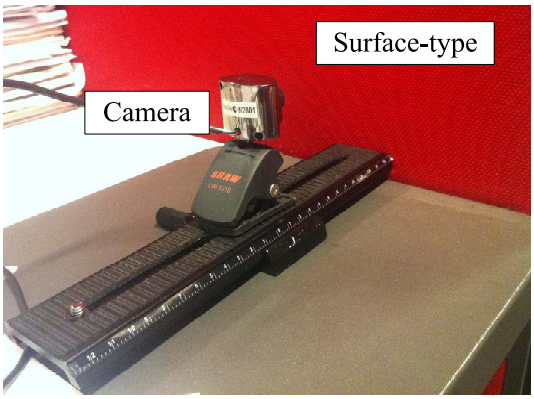


FIGURE 5.17: Experimental setup of camera to capture images of a surface-type

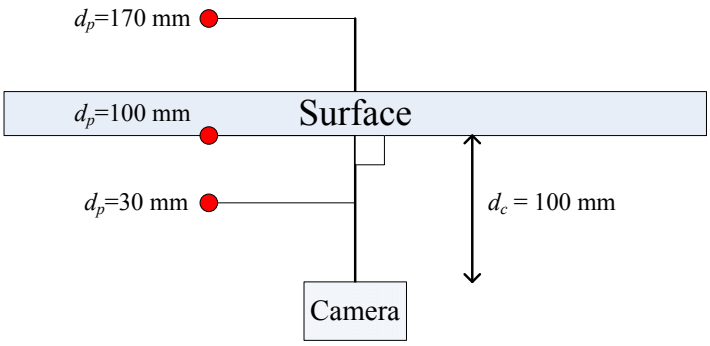


FIGURE 5.18: Image capture conditions used to capture images with focus distance changes

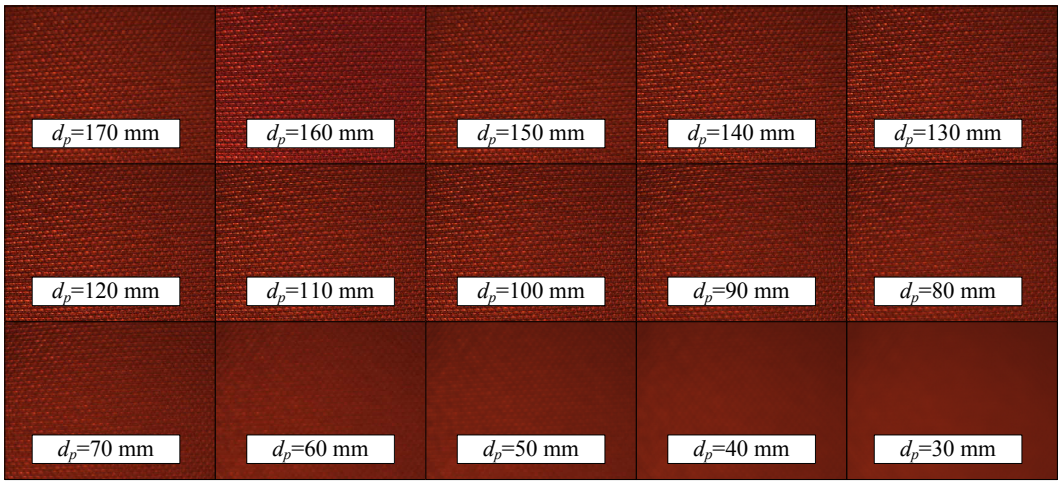


FIGURE 5.19: Set of images with focus distance changes

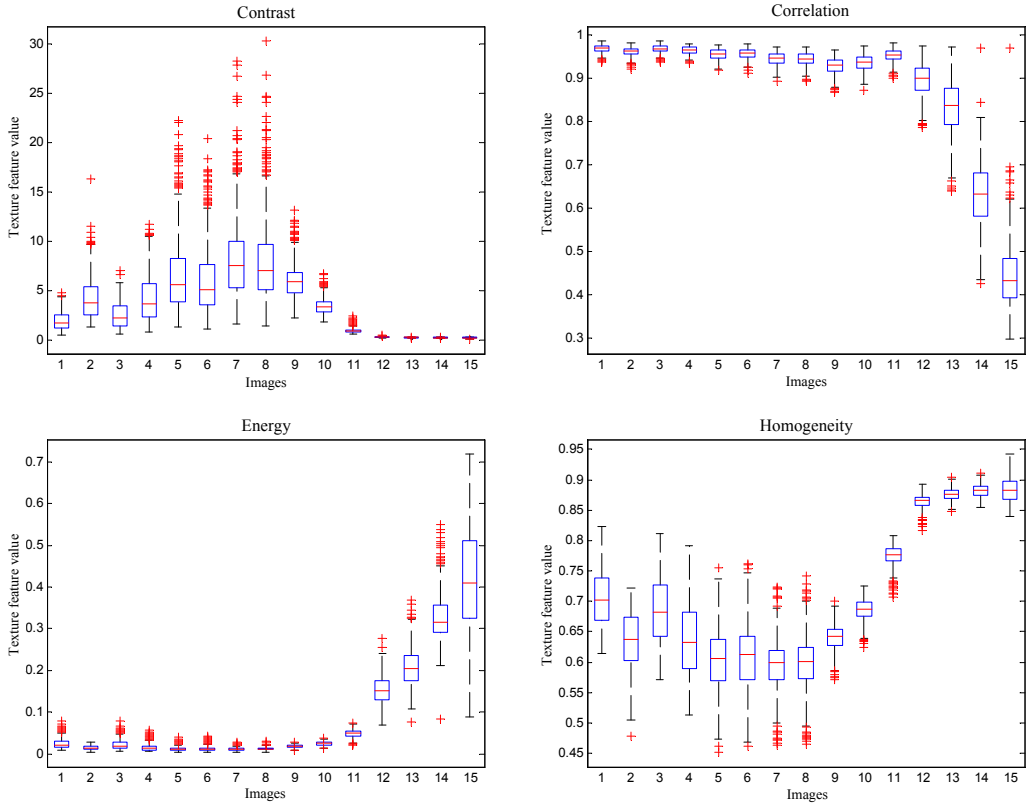


FIGURE 5.20: Box plot diagrams of the texture features distribution extracted from the set of images with focus distance change: horizontal axis shows the images (1–15) corresponding with plane of focus change from (30 mm to 170 mm); and vertical axis shows the values for each texture feature

(2) Figure 5.21 illustrates the image capture conditions used to collect a set of images

with spatial resolution changes. The viewing distance, d_c, and plane of focus, d_p, are concurrently changed in 10 mm increments between the range of 30 mm to 170 mm from the surface plane, and the viewing angle θ_c, is kept constant at 0°. An image is captured at each increment and Figure 5.22 shows the set of images with spatial resolution changes. Figure 5.23 shows the box plot diagrams of the value distribution of each texture feature. In each box plot diagram, the vertical axis shows the numerical value for the texture feature and the horizontal axis shows the 15 captured images with plane of focus change from 30 mm to 170 mm corresponding to images 1–15.

From the box plot diagrams shown in Figure 5.23, it can be seen that the extracted texture features have similar value distribution ranges across the images. The texture features of contrast, energy and homogeneity are observed to have a similar value distribution across the images. For example, for the texture feature of contrast, image 1 has a mean value of 5.7561 and a standard deviation of 1.5976, and image 15 has a mean value of 7.8408 and a standard deviation of 4.2553. However, a two-sample t-test between image 1 and image 15 supports the hypothesis that the distributions come from populations with unequal means. Furthermore, the most significant change in value distribution between images is observed for the texture feature of correlation. For example, the mean and standard deviation for image 1 are 0.8838 and 0.0311 respectively, and for image 15, these values are 0.9530 and 0.176, respectively. A two-sample t-test between image 1 and image 15 also supports the hypothesis that the distributions come from populations with unequal means. In summary, given the change in spatial resolution produced by concurrently varying the viewing distance d_c, and the plane of focus d_p, between the range of 30 mm to 170 mm, it is shown that the value distribution of the extracted texture features appear similar but are statistically inconsistent across the set of images. The value distribution appear similar given that the viewing distance range 30 mm to 170 mm produces insignificant change in spatial scaling/resolution for a window size of 96×96 pixels; the selected window size is manually verified to be able to encapsulate the fractal of the surface-type's texture pattern for all tested spatial resolutions.

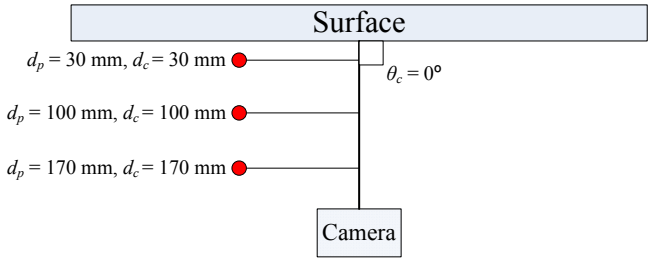


FIGURE 5.21: Image capture conditions used to capture images with spatial resolution changes

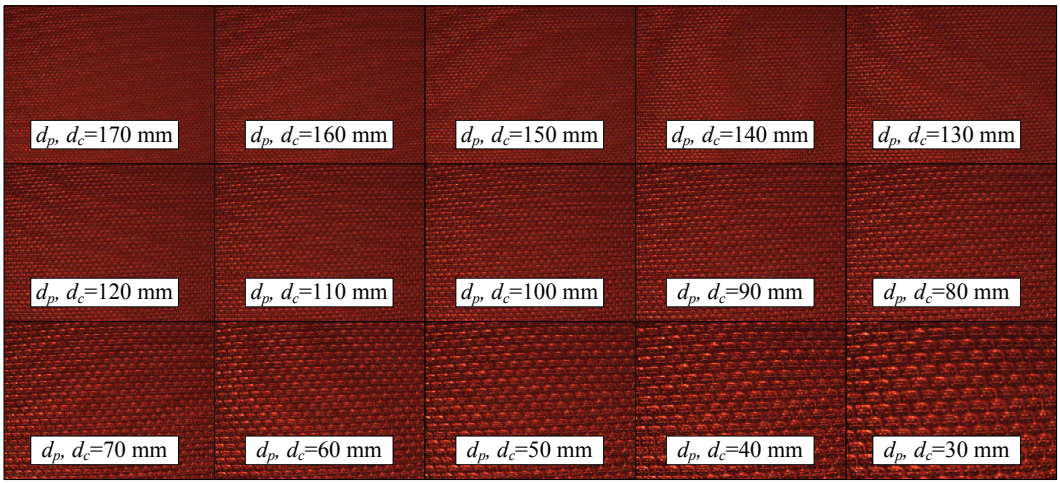


FIGURE 5.22: Set of images with spatial resolution changes

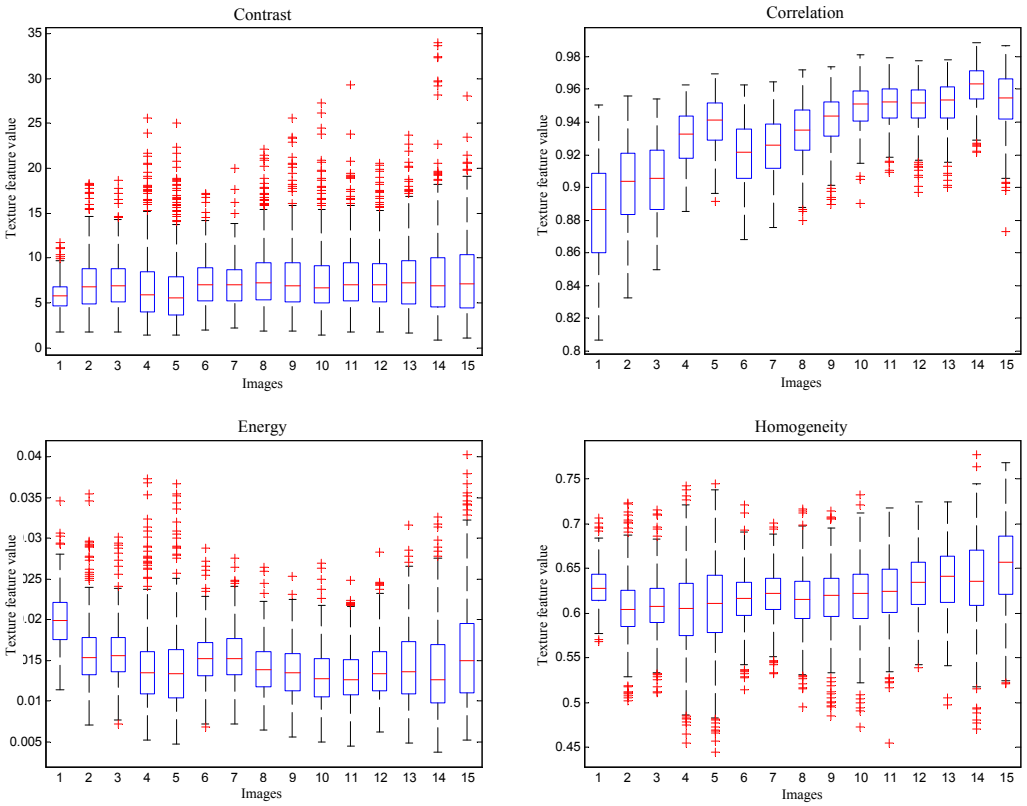


FIGURE 5.23: Box plot diagrams of the texture feature distribution extracted from the set of images with spatial resolution change: horizontal axis shows the images (1–15) corresponding with viewing distance and plane of focus change from (30–170 mm); and vertical axis shows the texture feature values

(3) Figure 5.24 illustrates the image capture conditions used to collect a set of images with perspective distortion. The camera is set with a constant viewing distance of $d_c = 100$ mm, and plane of focus of $d_p = 100$ mm. The viewing angle, θ_c, is changed at 15° increments between the range of 0° and 60°, and an image is captured at each increment. Figure 5.25 shows the captured set of images with perspective distortion and Figure 5.26 shows the box plot diagrams of the value distribution of each texture feature. In each box plot diagram, the vertical axis shows the numerical value for the texture feature and the horizontal axis shows images 1–5 corresponding with the viewing angle change from 0° to 60°.

From the box plot diagrams shown in Figure 5.26, it can be seen that the texture value distributions are inconsistent across the images. The texture distribution ranges are observed to increase from image 1 through to image 5 for all texture features including contrast, correlation, homogeneity and energy. For example, for the texture feature of

contrast, image 1 has mean and standard deviation values of 17.3258 and 2.7177, respectively; and image 5 has mean and standard deviation values of 12.3026 and 10.8181, respectively. The standard deviation of image 5 is approximately three times greater than image 1. A two-sample t-test between image 1 and image 5 supports the hypothesis that the distributions come from populations with unequal means. Overall, given the pattern skewing caused by a change of viewing distance θ_c between the range of 0° to 60°, it is shown that the value distributions of the extracted textures are inconsistent. It is noted that the change in viewing angle also results in viewing distance changes in the images. Hence, the cumulative effect of focus distance change and spatial resolution can also present in this set of images.

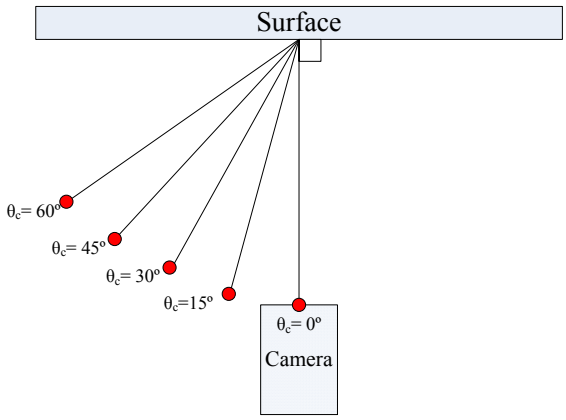


FIGURE 5.24: Image capture conditions used to capture images with perspective distortion

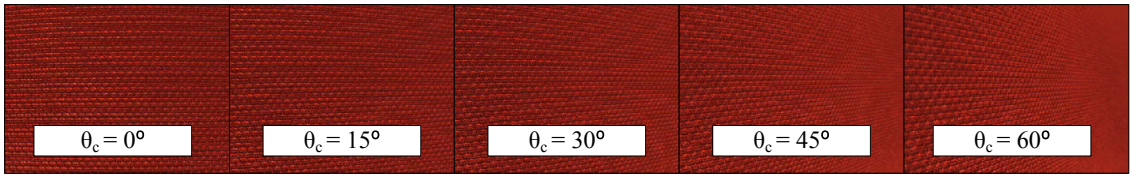


FIGURE 5.25: Set of images with perspective distortion

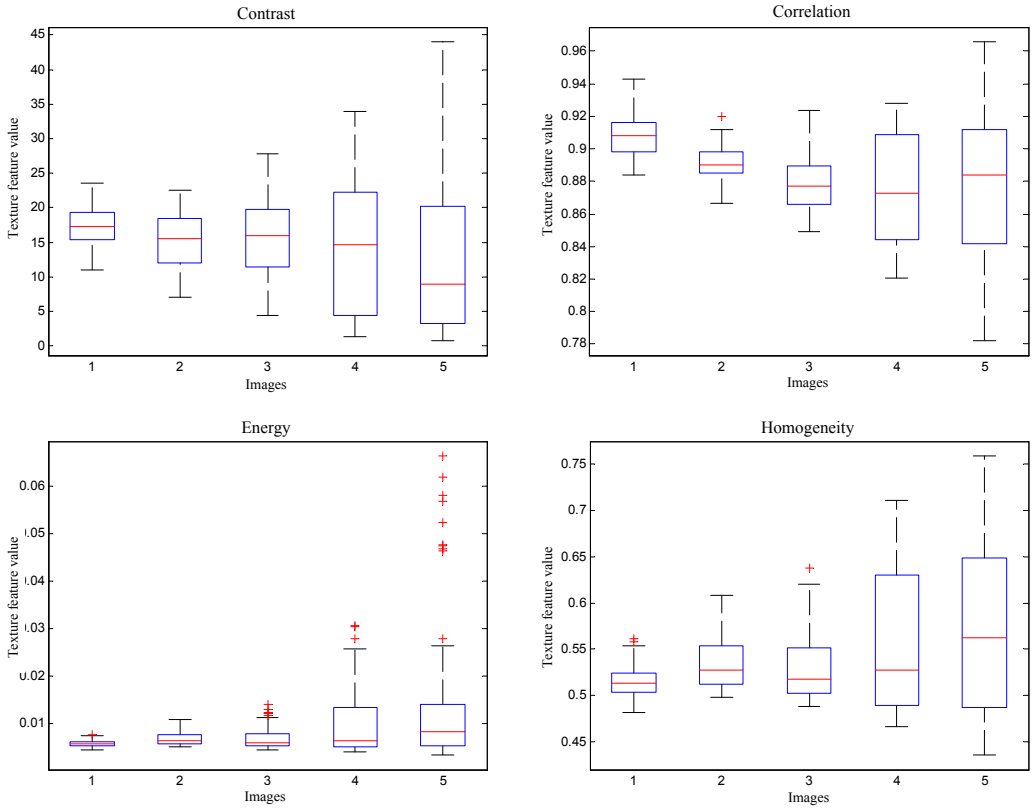


FIGURE 5.26: Box plot diagrams of the texture feature distribution extracted from the set of images with perspective distortion: horizontal axis shows the images 1–5 corresponding with viewing angle change from 0° to 60°; and vertical axis shows the values for each texture feature

Experiment 1 has demonstrated the effects of focus, spatial resolution and perspective distortion on the consistency of extracted texture feature values. It has been shown that the change in image capture conditions has resulted in the extracted texture features having distribution range changes. This can adversely affect the accuracy of surface-type classification, particularly when the range of feature values for the surface-type overlaps with other surface-type classes.

5.4.2 Experiment 2: Effect of Image Capture Conditions on Surface-type Classification

In this experiment, a classifier trained with an image dataset that contains three surface-types captured under known image capture conditions. The classifier is applied

to classify images of the same surface-types captured under different image capture conditions. Experiment 1 has shown that the changed image capture conditions can affect the consistency of the extracted texture features for a surface-type. The objective of this experiment is to verify the effect of image capture conditions on classification accuracy.

Figure 5.27 shows the environment used to collect the three surface-types, including blasted metal surface, rusted metal surface and timber surface. Figure 5.28 shows the different image capture conditions used to capture a set of images for each surface-type. The viewing distance, d_c, is changed in 50 mm increments between the range of 50 mm to 300 mm from the surface plane. At each viewing distance, the viewing angle (θ_c) is also changed in 10° increments between the range of 0° to 50°, resulting in a total of 36 images (1280×960 pixels) captured for each surface-type. For this experiment, the camera's optical settings are fixed and set with a plane of focus $d_p = 100$ mm. Figure 5.29 shows the set of images captured for the blasted metal surface, Figure 5.30 shows the set of images captured for the rusted metal surface, and Figure 5.31 shows the set of images captured for the timber surface.

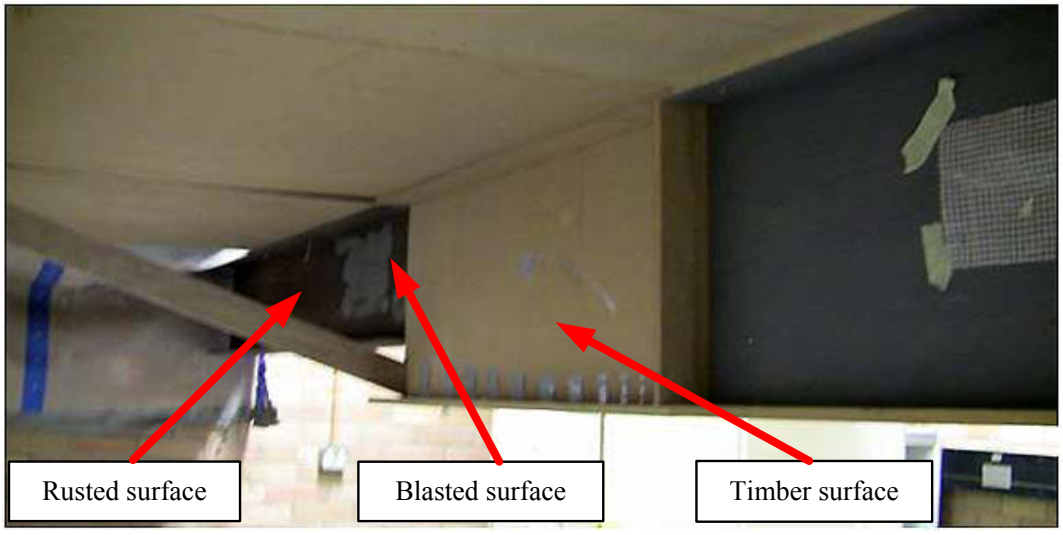


FIGURE 5.27: Experimental environment to capture surface-type images with different image capture conditions

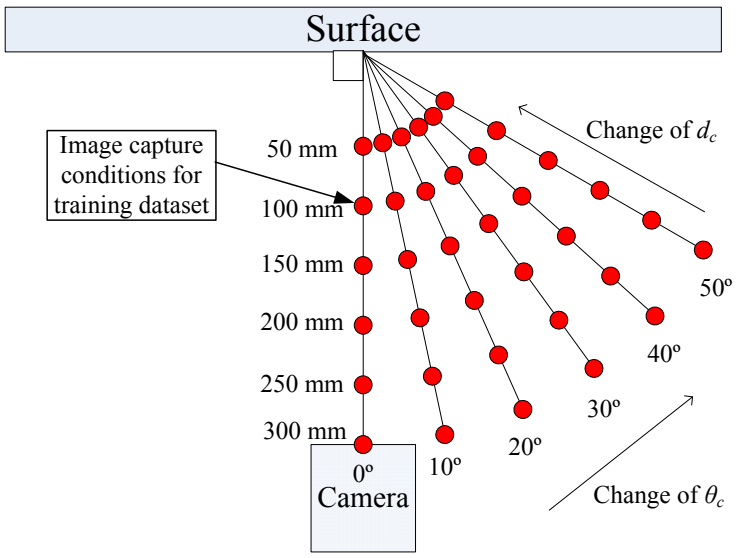


FIGURE 5.28: Image capture conditions used to capture a set of images for each surface-type in the experimental environment

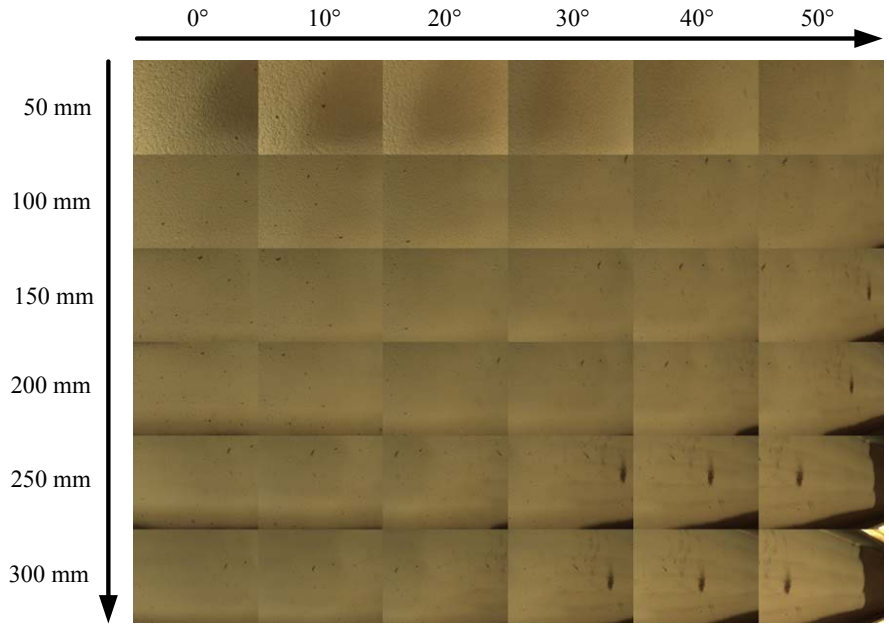


FIGURE 5.29: Set of images of blasted metal surface captured with changes in image capture conditions

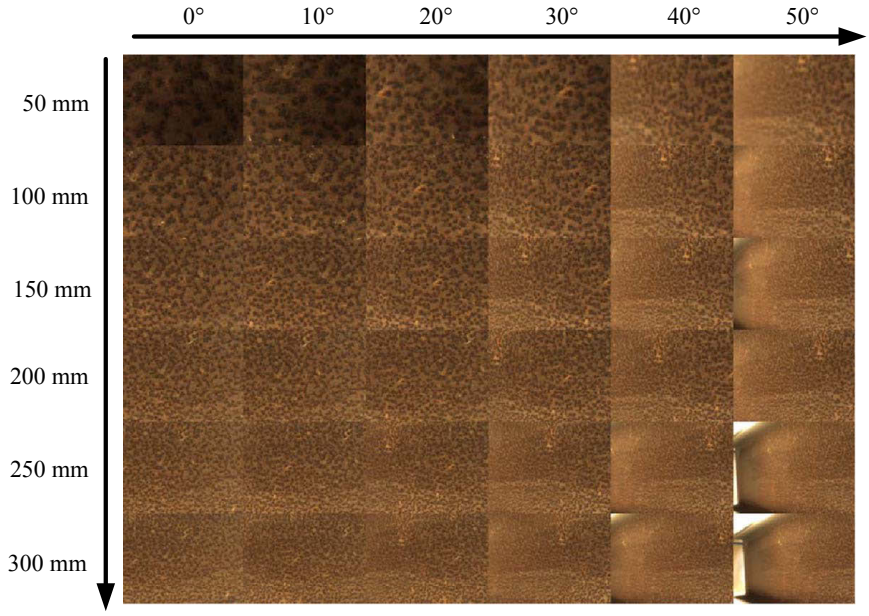


FIGURE 5.30: Set of images of rusted metal surface captured with changes in image capture conditions

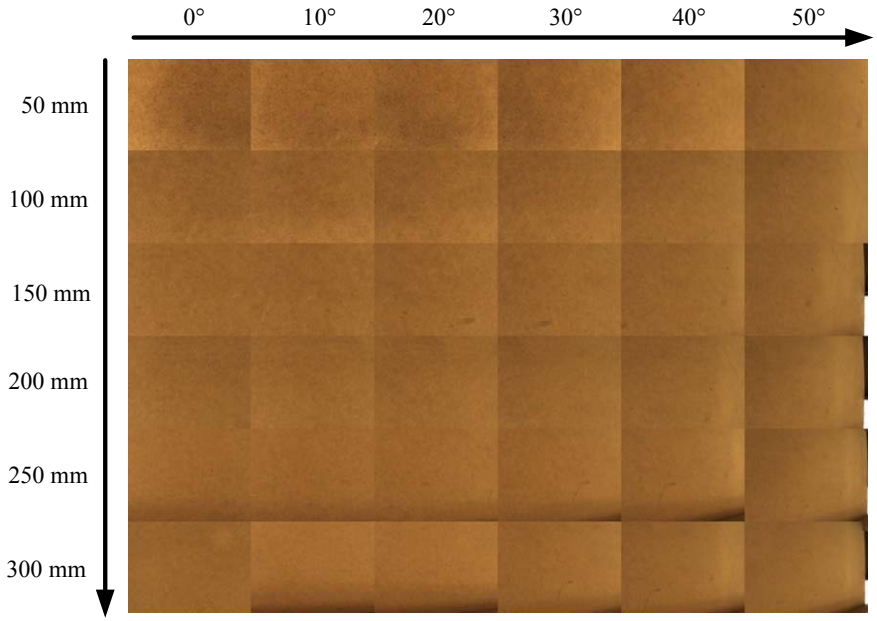


FIGURE 5.31: Set of images of timber surface captured with changes in image capture conditions

Each image is cropped into 960×960 pixels, and is divided into samples with a window size of 96×96 pixels to extract texture features of contrast, correlation, energy and

homogeneity. The four features are used to train a classifier that will be used to classify the three surface-types in this experiment. Figure 5.32 shows the images for each surface-type collected at $d_c = 100$ mm and $\theta_c = 0°$, which are used as the training image dataset. A naive Bayes classifier (Appendix C) is trained using the training image dataset. In addition, a distance-based outlier reject option [104] is used to reject samples with features that are outliers to the training image dataset. All the images in Figure 5.29–5.31, including the training images (Figure 5.32), are classified using the trained naive Bayes classifier. Tables 5.1, 5.2 and 5.3 show the classification accuracy for the images, where the grey cells highlight the classification result of the training images.

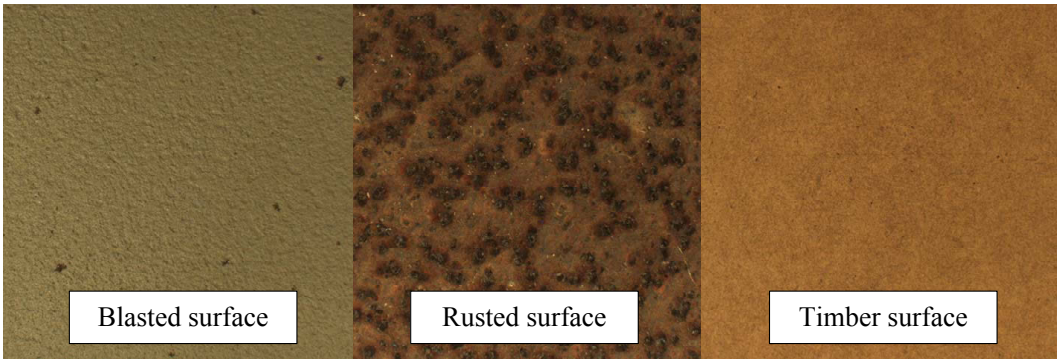


FIGURE 5.32: The images used in the training dataset with image capture conditions of $d_c = 100$ mm and $\theta_c = 0°$

TABLE 5.1: Classification accuracy for blasted metal surface images

	Classification Accuracy (%) of blasted metal surface images					
	0°	**10°**	**20°**	**30°**	**40°**	**50°**
50 mm	73	27	24	14	0	1
100 mm	73	43	10	18	19	0
150 mm	41	37	23	13	13	1
200 mm	31	35	20	7	7	4
250 mm	6	8	26	23	9	8
300 mm	8	4	8	11	6	7

Table 5.2: Classification accuracy for rusted metal surface images

	Classification Accuracy (%) of rusted metal surface images					
	0°	**10°**	**20°**	**30°**	**40°**	**50°**
50 mm	12	0	0	0	0	0
100 mm	100	99	70	14	6	5
150 mm	100	100	100	100	100	93
200 mm	100	100	100	100	100	100
250 mm	100	100	100	100	100	78
300 mm	100	100	99	100	96	65

Table 5.3: Classification accuracy for timber surface images

	Classification Accuracy (%) of timber surface images					
	0°	**10°**	**20°**	**30°**	**40°**	**50°**
50 mm	22	9	35	0	0	0
100 mm	94	87	77	13	0	0
150 mm	74	72	71	70	48	0
200 mm	76	82	60	50	14	0
250 mm	46	59	45	21	0	0
300 mm	14	0	0	0	0	0

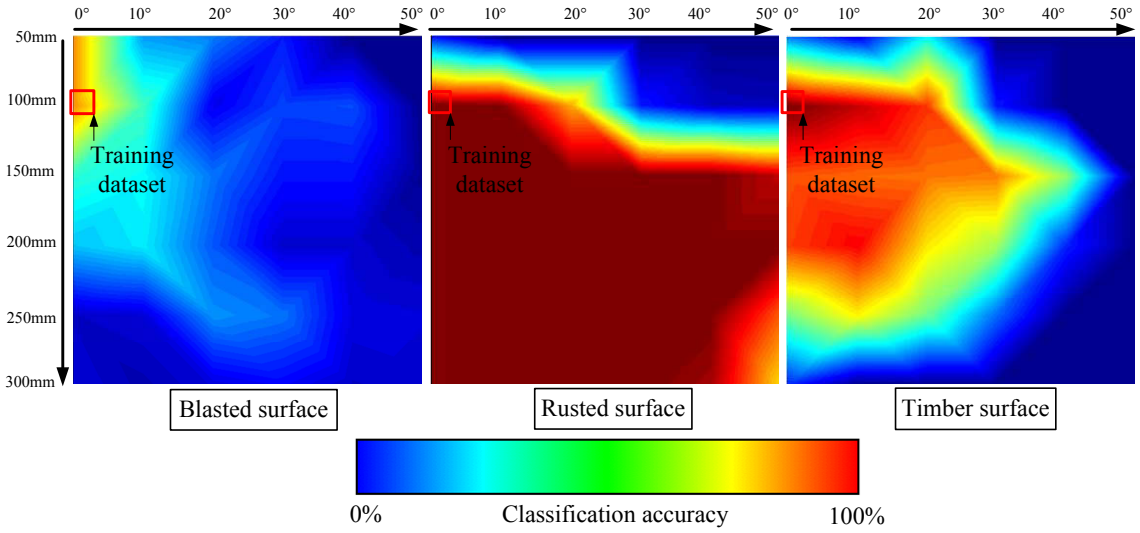


FIGURE 5.33: Visualisation of the classification accuracy for the surface-type images corresponding to the results presented in Tables 5.1, 5.2 and 5.3

Overall, it can be seen from Tables 5.1, 5.2 and 5.3 that the classification accuracy is affected by the change in image capture conditions. The highest classification accuracy is observed for the images used as the training dataset ($d_c = 100$ mm, $\theta_c = 0°$) for all the three surface-type classes: blasted surface achieved 73% accuracy, rusted surface achieved 100% accuracy and timber surface achieved 94% accuracy. The lowest classification accuracy is observed for blasted metal surface (7%) and timber surface (0%) with image capture conditions $d_c = 300$ mm and $\theta_c = 50°$, and for rusted metal surface (0%) with image capture conditions $d_c = 50$ mm and $\theta_c = 50°$. In general, the classification results show that accuracy decreases when the image capture conditions deviates from the training image dataset's, which is ($d_t = 100$ mm, $\theta_t = 0°$). It is interesting to observe that the classification result for the rusted surface showed high accuracy for the majority of images. This could be because the texture features clearly distinguish the rusted surface amongst the three surface-types, such that the classifier can classify rusted surfaces accurately across the tested range of image capture conditions.

Figure 5.33 shows a visualisation of the classification accuracy plotted against viewing distance and viewing angle, generated using the classification results presented in the tables. From the visualisation, it can be observed that the change in classification accuracy relative to image capture conditions is different for each surface-type. This

difference in classification accuracy can be attributed to the capability of the chosen texture features to distinguish between the three surface-types. As observed, the features (contrast, correlation, energy and homogeneity) can clearly distinguish a rusted metal surface and have resulted in high classification accuracy for a large range of image capture conditions. The lowest accuracy was observed for the blasted metal surface with only a small range of image capture conditions with high classification accuracy. This can be observed from the visualisation shown in Figure 5.33. It is noted that the change in classification accuracy relative to image capture conditions is used in the algorithm to determine the parameter values (τ_n, τ_f, τ_θ, ω_1, ω_2) for each surface-type.

5.4.3 Experiment 3: Identification of Accurately Classified Image Region(s)

In this experiment, images are captured from a scene that contains multiple surface planes and are classified to identify for a timber surface-type. A probability map is produced to identify the regions in an image (called "segmented regions") that have a higher probability of being the timber surface-type. The classification accuracy of an image and the segmented regions only are compared. This experiment demonstrates the use of probability maps to identify regions in images that have a high probability of being accurately classified.

Figure 5.34 shows the RGB-D sensor package used in this experiment (Point Grey Firefly and PMD nano), and the complex scene that contains multiple surface planes from which RGB-D images are collected and classified for the timber surface-type. The PMD nano (ToF based depth camera) is used for capturing the depth images in this experiment; given the close proximity with the surfaces (around 300mm) to capture the texture details. The Kinect, which was used in Chapter 4 is not suitable in this situation given that it has a minimum range of 500mm. The PMD nano used an integration time in the range of 20 to 200 micro seconds for this experiment. Figure 5.35 shows the 600×600 pixels training image for the timber surface-type collected with image capture conditions of viewing distance $d_t = 200$ mm and viewing angle $\theta_t = 0°$. The camera's optical settings are fixed for this experiment with $d_p = 200$ mm.

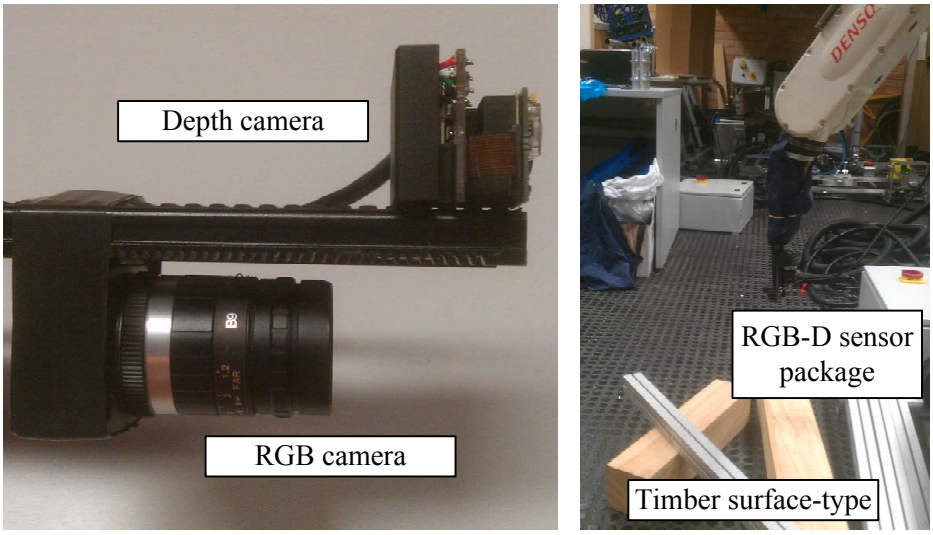


FIGURE 5.34: The RGB-D sensor package used in this experiment and the experiment scene with multiple surface planes

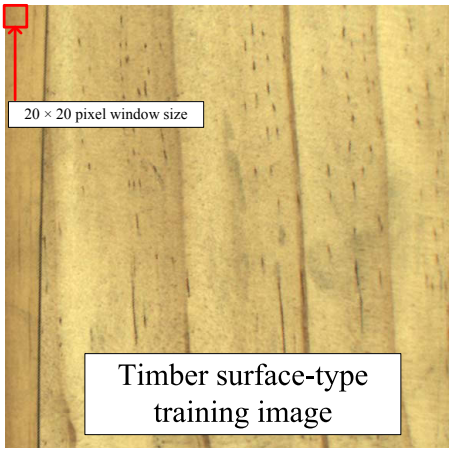


FIGURE 5.35: 600×600 pixels training image of the timber surface-type

The training image is divided into samples of 20×20 pixels window size to extract texture features including contrast, correlation, energy and homogeneity. A single-class naive Bayes classifier is trained with the training samples and is used to classify the timber surface-type. In this experiment, test images are divided into samples of 20×20 pixels window size that are classified by the trained classifier. Figure 5.36 shows (first row) four test images that are captured from the scene (shown in Figure 5.34) with multiple surface planes, and the surface-type classification results for the four test images (second row). The classification results shown in the second row are in binary, where white represents

the samples classified as timber surface-type and black represents the samples classified as non-timber surface-types.

The probability map for the timber surface-type is generated using the parameter values: $\tau_n = 150$ mm, $\tau_f = 250$ mm, $\tau_\theta = 20°$, $\omega_1 = 1$, and $\omega_2 = 1$. Figure 5.36 shows (third row) the regions (white) in each test image with a high probability of being accurately classified as a surface-type. The white regions are identified by thresholding for pixels with probability value P_{d_c,θ_c} P_{d_c}, that are greater than 0.9. It is noted that the threshold value is selected manually from a calibration process in which a set of training samples with a range of probability values are classified; all samples with probability value greater that 0.9 are shown to be accurately classified and thus 0.9 is selected as the threshold value.

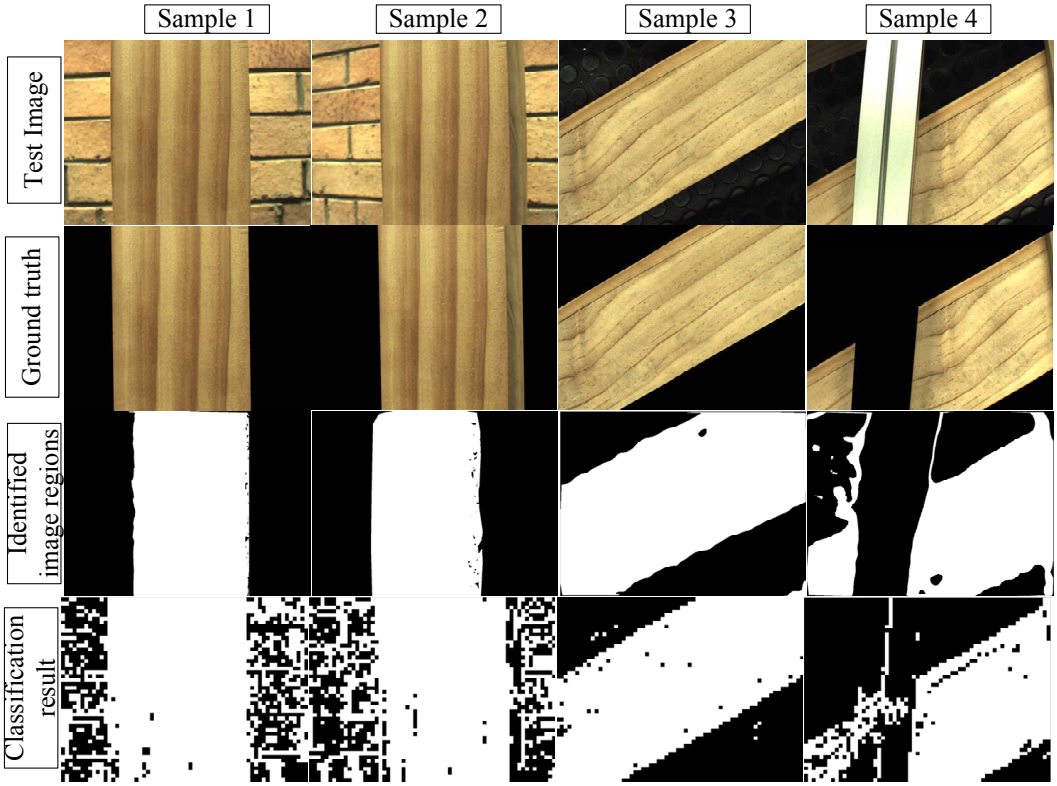


FIGURE 5.36: Row 1 test images; row 2 classification results of test images; row 3 segmented image regions with a high probability of being accurately classified

The test image and the segmented regions of the test image are compared in terms of their accuracies. Tables 5.4, 5.5, 5.6, and 5.7 are the confusion matrices for each of the four test images. Each table contains two separate confusion matrices: one for complete

classification result of the test image, and the other for classification result of the segmented regions only. The rows in a confusion matrix shows the classified class and the columns show the actual class. From the confusion matrices, it can be observed that the classification accuracy of the segmented regions, are higher than the original test images. Classification accuracy is observed to be higher for image 1 by 19.91%, image 2 by 20.94%, image 3 by 9.21%, and image 4 by 17.87%. The higher classification accuracy for the segmented region only, can be attributed to lower misclassification of the timber surface-type.

In conclusion, this experiment showed the use of a probability map to assess regions in an image that have a high probability of being accurate. The identified regions are shown to have lower misclassification and, consequently, have a higher accuracy in comparison to the classification result of the test image.

TABLE 5.4: Confusion matrices for test image 1

Confusion matrices: Test image 1		
Classification result of test image		
Surface-types	**Timber**	**Non-timber**
Timber	672105	10657
Non-timber	245876	300143
		Accuracy: 79.12%
Classification result of segmented regions only		
Surface-types	**Timber**	**Non-timber**
Timber	530840	10657
Non-timber	0	0
		Accuracy: 99.03%

Table 5.5: Confusion matrices for test image 2

Confusion matrices: Test image 2		
Classification result of test image		
Surface-types	**Timber**	**Non-timber**
Timber	648907	65124
Non-timber	232401	278059
		Accuracy: 75.70%
Classification result of segmented regions only		
Surface-types	**Timber**	**Non-timber**
Timber	552287	15880
Non-timber	3334	192
		Accuracy: 96.64%

Table 5.6: Confusion matrices for test image 3

Confusion matrices: Test image 3		
Classification result of test image		
Surface-types	**Timber**	**Non-timber**
Timber	853759	17578
Non-timber	140287	219062
		Accuracy: 87.17%
Classification result of segmented regions only		
Surface-types	**Timber**	**Non-timber**
Timber	821012	15816
Non-timber	15665	17618
		Accuracy: 96.38%

TABLE 5.7: Confusion matrices for test image 4

Confusion matrices: Test image 4		
Classification result of test image		
Surface-types	**Timber**	**Non-timber**
Timber	579420	11740
Non-timber	322823	316659
		Accuracy: 72.81%
Classification result of segmented regions only		
Surface-types	**Timber**	**Non-timber**
Timber	551599	10606
Non-timber	52616	63850
		Accuracy: 90.68%

5.5 Discussion

An algorithm that uses image capture conditions to calculate a probability map has been presented in this chapter. The probability map can be used to identify the accurately classified regions in images. The image capture conditions and image qualities (focus quality, spatial resolution and perspective distortion) has been discussed and analysed to define a set of threshold parameters. The threshold parameters were used to calculate the probability values for an image pixel being a surface-type. The function can be used to produce a probability map for the classification results of an image classified using texture features.

Experiments were conducted in a laboratory environment to verify the effect of image capture conditions on extracted texture feature values, the effect of image capture conditions on classification accuracy, and the use of probability maps to assess for image regions that are accurate. Currently, the probability map is used assess for regions in an image that has a high probability of being accurate. However, it is possible to use the probability map in such a way as to provide the most probable surface(s) of a pixel to

provide additional information for accuracy assessment (Chapter 7 provides a more detail discussion). Furthermore, in this experiment a focusable lens was selected over a pinhole-like lens to provide better low-light performance (bigger maximum aperture) and sharpness across a wider range of viewing distances. In future, an investigation will be conducted to investigate the suitability of using a pinhole lens or a focusable lens depending on the target environment. Additionally, further future work will look into autonomously selecting the parameter values for an environment and investigates the failure modes if the values are not selected properly.

From the algorithm presented in chapter 4, colour features can be extracted to produce per-pixel classification result of a test image, and from the algorithm present in this chapter, a probability map is produced for an image classified based on texture features. The next chapter will present a case study on a robotic system that uses the algorithms presents in chapters 4 and 5 to build a surface-type map in 3D.

Chapter 6

Case Study

This chapter presents a case study of the surface-type classification approach detailed in Chapter 3 that uses the algorithms presented in Chapter 4 and Chapter 5. In this case study, a laboratory experimental environment is set up to enable an RGB-D sensor package mounted onto the end-effector of a robot manipulator to collect images from different viewpoints. The RGB-D images collected from each viewpoint are classified using classifiers trained with colour-based and texture-based features to produce two sets of classification results. From the colour-based and texture-based classification results, a combined surface-type map in 3D can be generated.

6.1 Experiment Setup

6.1.1 Setup of RGB-D Sensor Package

The experiments are conducted using an RGB-D sensor package with an attached light source, as shown in Figure 6.1. The sensor package consists of a Microsoft Kinect, a Point Grey Firefly 1.2 megapixel camera with a 6 mm lens (RGB camera), and a single LED light source. It is noted that a single LED light source was selected in this sensor package over the previous LED array light source used in Chapter 4 due to design constraints of size and weight on end-effector attachment. The single LED light source and the LED

array light source can both be assumed as a point light source; albeit having different illumination intensities. In the RGB-D sensor package, the IR camera (depth camera) from the Kinect is used to provide a 640×480 pixels depth image with a field of view of 57° horizontally and 43° vertically. The Firefly camera is used to provide a 1280×960 pixels RGB image with a field of view of 56° horizontally and 43° vertically (the field of view specifications are for a 1/2-inch sensor size, therefore they are slightly narrower for the Firefly's 1/3-inch sensor size at approximately 43° horizontally and 33° vertically).

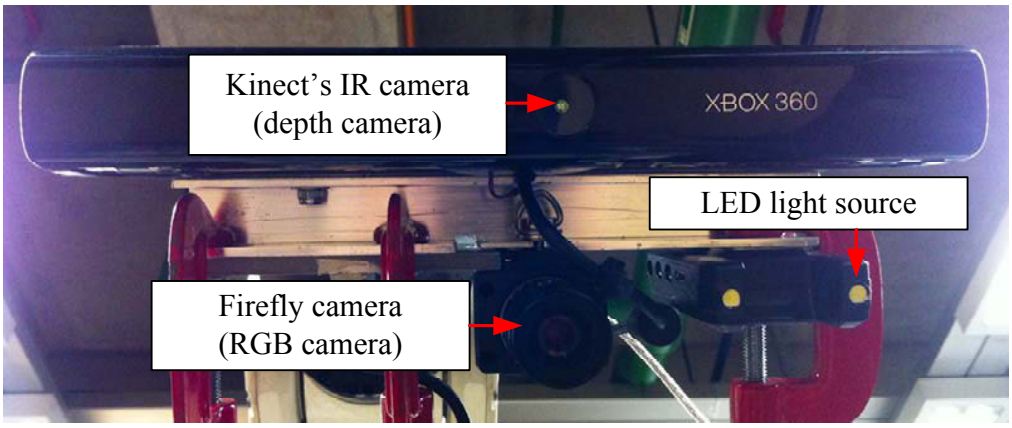


FIGURE 6.1: RGB-D sensor package: Firefly camera, Kinect, and LED light source

6.1.2 Robot Manipulator

The robot manipulator used in this case study is the Denso VM-6083 manipulator arm with six degrees-of-freedom. Figure 6.2 shows the Denso VM-6083 robot manipulator where the RGB-D sensor package can be attached to the end-effector. In this configuration, it is possible to precisely position the RGB-D sensor package at a wide range of viewpoints in a complex environment. The Denavit and Hartenberg (D-H) parameters and limitations of the Denso VM-6083 are provided in Table 6.1. Using the D-H parameters, a kinematic model can be generated to represent the position of the end-effector relative to the robot manipulator base for a given set of joint angles.

TABLE 6.1: Denavit and Hartenberg [15] parameters, joint types and limits for the Denso VM-6083 manipulator arm

Link	Twist α_i	Length a	d Offset	State at Zero	Min (deg)	Max (deg)
1	$-\frac{\pi}{2}$	0.18	0.475	0	-170	170
2	0	0.385	0	$-\frac{\pi}{2}$	-90	135
3	$\frac{\pi}{2}$	-0.1	0	$\frac{\pi}{2}$	-80	165
4	$-\frac{\pi}{2}$	0	0.445	0	-185	185
5	$\frac{\pi}{2}$	0	0	0	-120	120
6	0	0	0.084	0	-360	360

FIGURE 6.2: Denso VM-6083 robot manipulator

6.1.3 Calibration

Three different calibration processes are performed to identify the coordinate frame transformation between: (1) the Firefly camera and the IR camera; (2) the IR camera and the robot manipulator end-effector (hand-eye); and (3) the Firefly camera and LED light source. The transformations are used in the surface-type classification approach to generate depth information in the RGB camera coordinate frame for the algorithms and to produce the surface-type map in the robot base coordinate frame. The three calibration processes are outlined below.

(1) Camera calibration is performed between the Firefly camera and the Kinect's IR camera to enable perspective projection of the depth image produced by the IR camera into 3D points in space, and then transformation and perspective projection into the

Firefly camera coordinate frame. In this case study, the cameras are calibrated using the Matlab toolbox provided here `http://www.vision.caltech.edu/bouguetj/calib_doc/` based on the checkerboard method [86]. Figure 6.3 shows the 25 checkerboard images used for calibration and Figure 6.4 shows the calibrated extrinsic transformation between the camera coordinate frames that will be used to transform the 3D points and to generate depth information in the Firefly camera coordinate frame, and Table 6.2 shows the calibrated extrinsic parameter values.

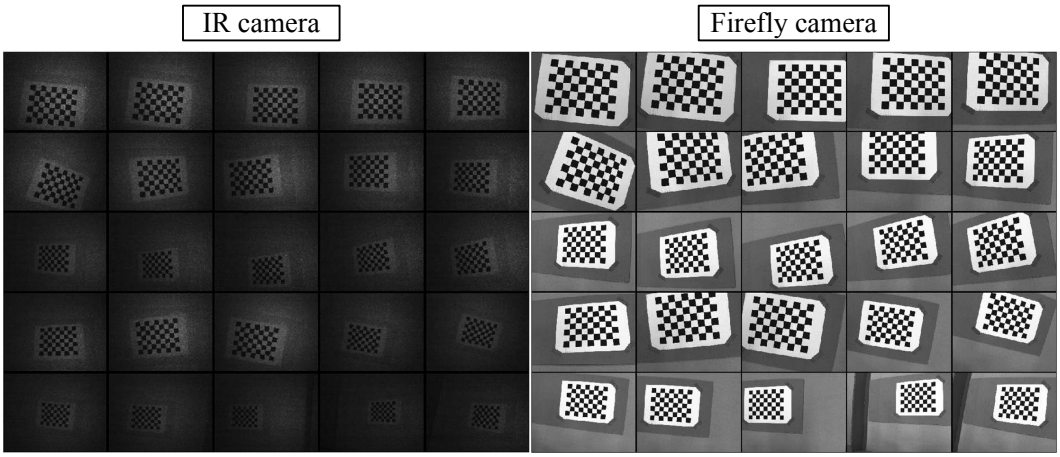


FIGURE 6.3: 25 checkerboard images captured by the IR camera (left) and the Firefly camera (right) for intrinsic and extrinsic calibration

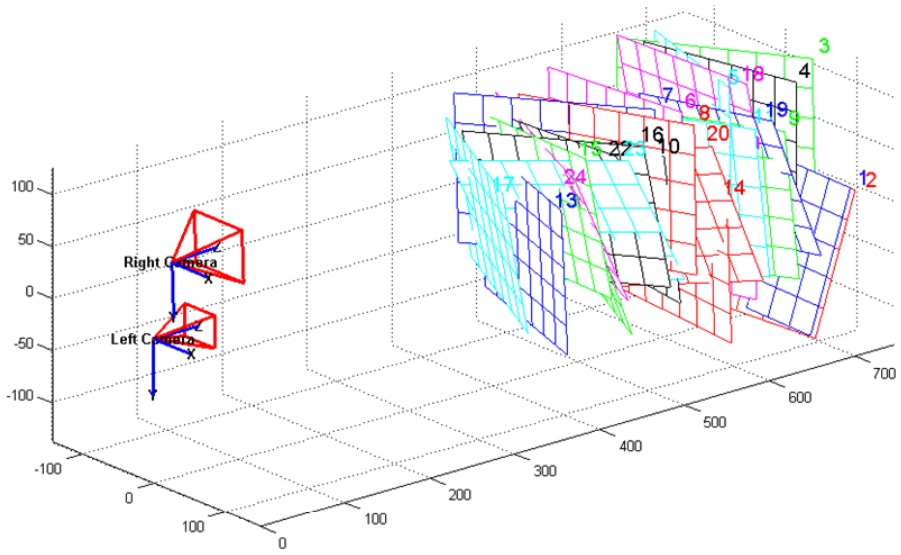


FIGURE 6.4: Extrinsic transformation between the Firefly camera and the IR camera coordinate frames

TABLE 6.2: Calibration parameter values for IR to Firefly camera

Rotation vector (Radian)	-0.03674	-0.02362	0.00861
Rotation reprojection error (±)	0.00412	0.00496	0.00023
Translation vector (Radian)	3.48739	77.41613	1.42057
Translation reprojection error	0.11491	0.12650	0.71923

(2) Hand-eye calibration of the IR camera coordinate frame to the robot manipulator base coordinate frame is performed to enable the transformation of the collected RGB-D images into the robot base frame. The hand-eye calibration method used is presented in Appendix A, and Figure 6.5 shows the calibration IR image and depth image produced from the IR camera. Figure 6.6 shows the robot manipulator in the pose used to capture the calibration image and a simulation of the robot manipulator with a point cloud transformed into the robot base frame using the IR camera-to-robot base transformation produced from hand-eye calibration.

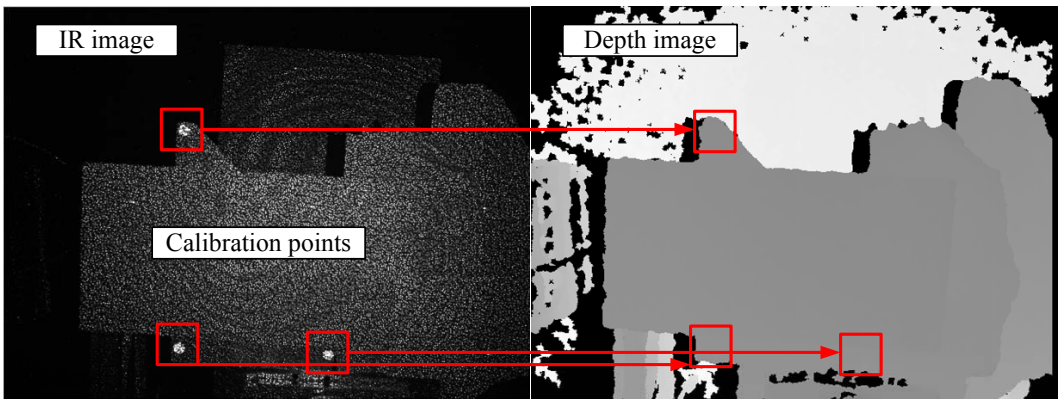


FIGURE 6.5: IR and depth images used to identify the calibration points to perform hand-eye calibration

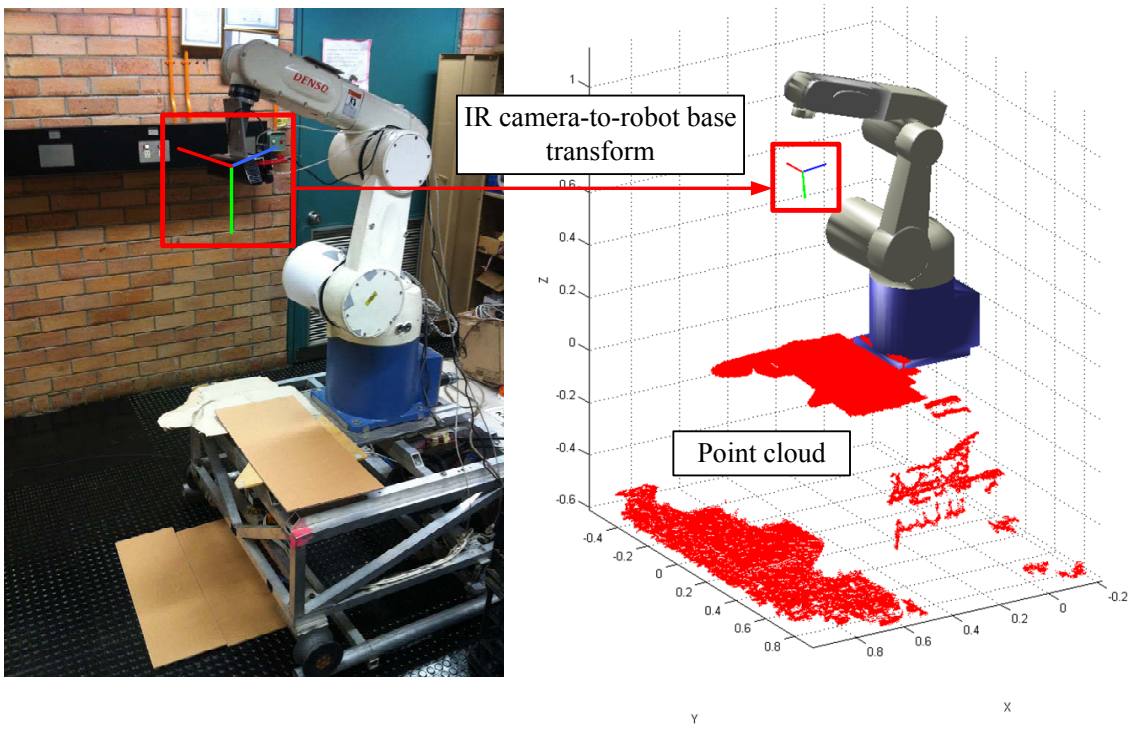


FIGURE 6.6: Real robot manipulator and a simulation of the robot manipulator with a point cloud transformed into the robot base coordinate frame

(3) Light source calibration is performed to identify the position of the LED light source relative to the Firefly camera coordinate frame. Figure 6.7 shows the calibration image, the binary image of the specular region, and the image showing only the diffused region. The specular and diffused regions are segmented by using threshold values of $\tau_d = \tau_s = 255 \times 0.7$. Figure 6.8 shows the light source position identified using the centroid of the specular region (specular point) in the calibration image and a reflectance model.

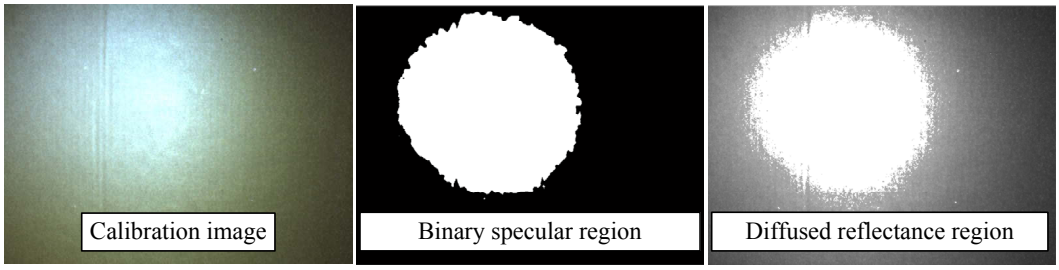


FIGURE 6.7: Calibration images used to identify the light source position relative to the Firefly camera coordinate frame

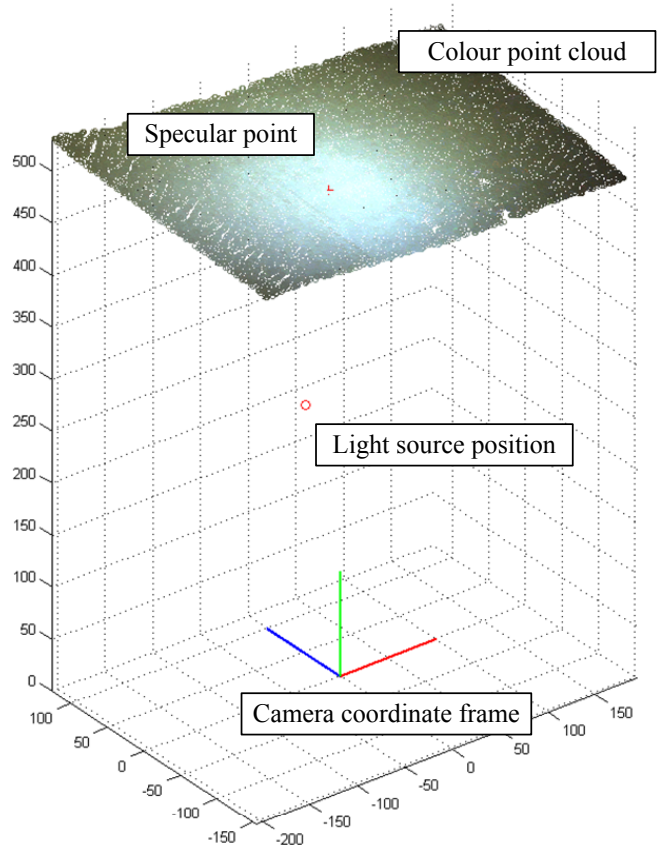


FIGURE 6.8: The calibration image perspective projected into 3D and light source position relative to the Firefly camera coordinate frame

6.1.4 Method to Evaluate the Accuracy of Classification Results using a Surface-type Map

To evaluate the accuracy of surface-type classification results, a benchmark surface-type map can be used. A benchmark surface-type map is produced by manually labelling the surface-types in the RGB-D image. Each pixel from the labelled RGB image is perspective projected into 3D, and the labelled surface-type of pixels stored in voxels (μ = 10 mm) to produce a benchmark surface-type map, as detailed in Section 3.4. In this way, a benchmark surface-type map that shows a surface area can be used to evaluate the accuracy of surface-type maps produced from colour and texture-based classification results. Figure 6.9 shows the setup of the environment; and Figure 6.10 shows a benchmark surface-type map of the environment (left), which can be used to evaluate the

accuracy of the surface-type map produced from classification results (right). Accuracy evaluation is performed by comparing the voxels in corresponding 3D locations between the benchmark and the evaluated surface-type maps. A voxel in the evaluated surface-type map is considered as accurate if the most probable surface-type stored in the voxel matches with the benchmark map's labelled surface-type. The overall accuracy of an evaluated surface-type map is taken as the total number of voxels evaluated to be accurate divided by the total number of voxels in the evaluated surface-type map.

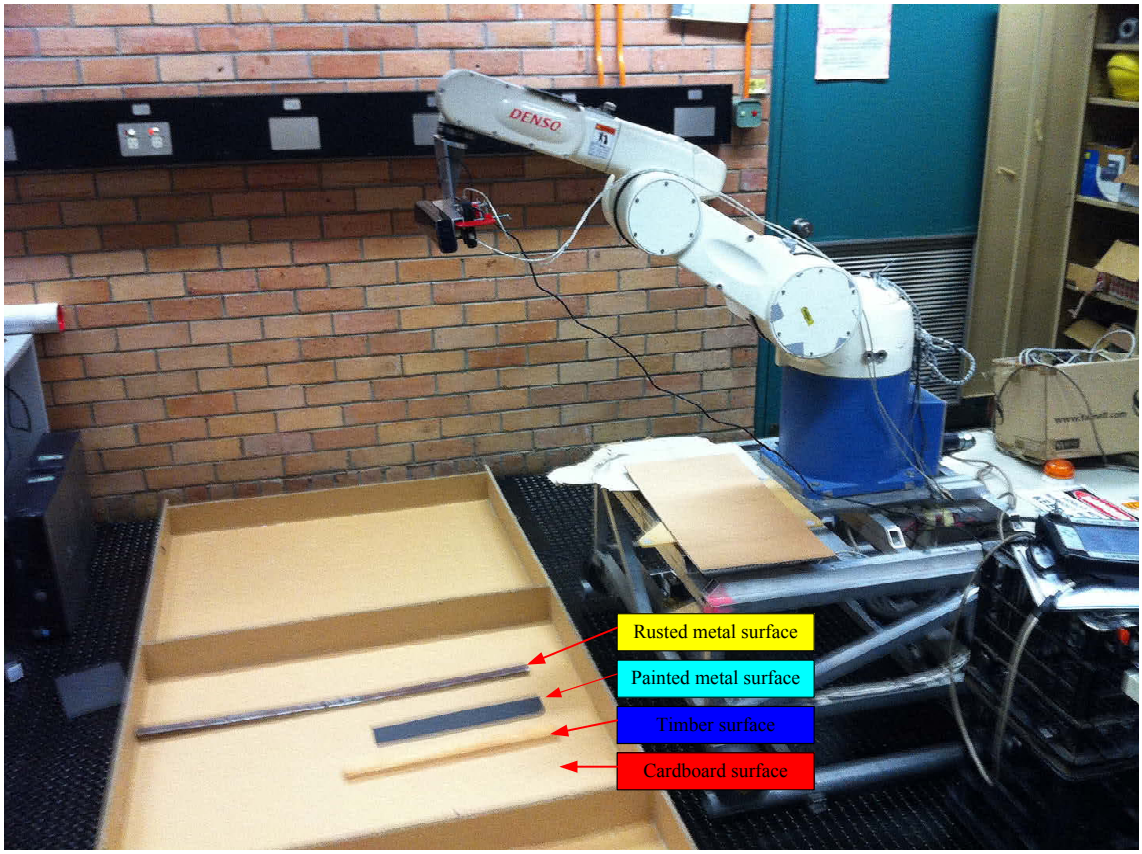


FIGURE 6.9: Setup of the environment to generate a benchmark surface-type map

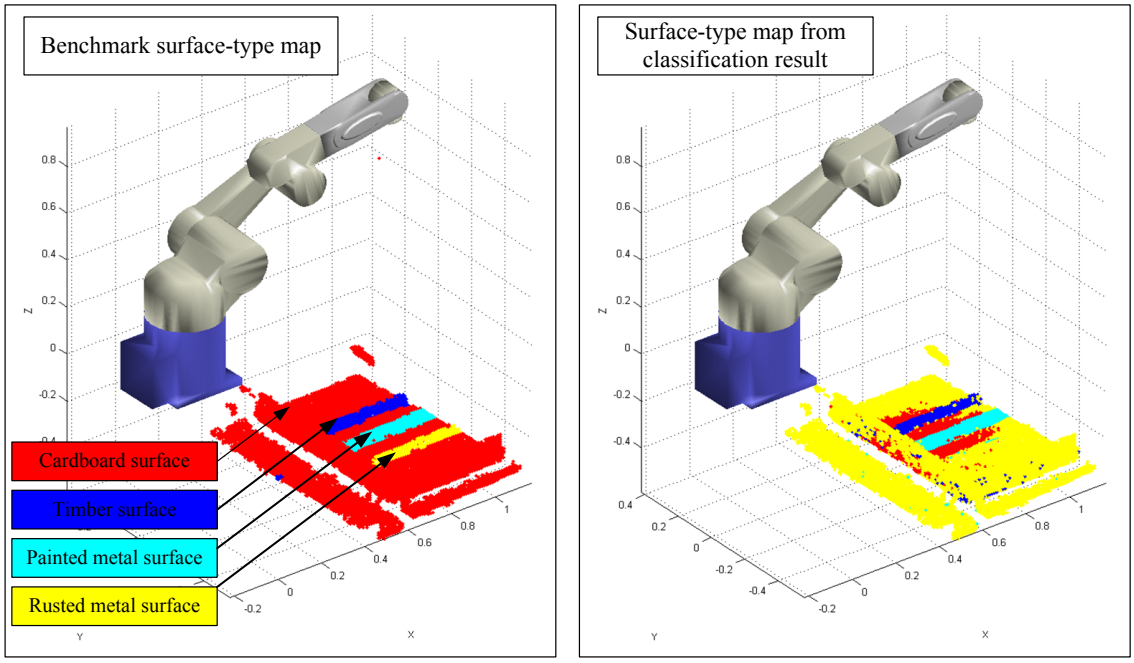


FIGURE 6.10: Benchmark surface-type map, and surface-type map generated from classification results

6.1.5 Training Surface-type Classifiers

In this case study, a training image dataset is collected using a viewing distance $d_c = 650$ mm, viewing angle $\theta_c = 0°$ from the surface plane, a fixed camera optical/exposure setting ($d_p = 650$ mm) for the Firefly camera, and a constant illumination provided by the LED light source. Figure 6.11 shows the four original 1280×960 pixels images collected by the Firefly camera of each surface-type (rusted metal beam, painted metal surface, timber and cardboard), and Figure 6.12 shows the 40×700 pixels regions cropped from each image to form the training image dataset used to train surface-type classifiers (four-class classifiers). It is noted that the cardboard surface-type has not been previously investigated in prior chapters.

Colour and texture features are extracted from the training image dataset to train surface-type classifiers for the purpose of comparison. For this case study, SVM is selected as the classifier to produce the classification results given the higher accuracy it is able to achieve over the naive Bayes classifier; this is determined by a preliminary comparison of the two classifiers using the training image dataset. Three sets of colour

features are extracted to train three separate multi-class SVM classifiers, which include RGB, a*b*, and $K_d \in (K_{d,R}, K_{d,G}, K_{d,B})$. The camera radiometric function parameters selected to extract K_d are $a_2 = 0$, $a_1 = 1$, $a_0 = 0$, based on a set of calibration images captured at distance d_l (650–1300 mm). For texture, features are extracted using local binary patterns (LBP) with a sample window size of 20×20 pixels (Appendix B) to train a multi-class SVM classifier. The LBP has been selected to provide robustness against non-uniform illumination in the images [105] and also against rotation variance [106].

Given the low dimension of the colour features, the radial basis function (RBF) kernel is selected for the SVM classifier; and for the texture features that have a higher dimension, a linear kernel is selected for the SVM classifier. Cross-validation is performed on the training image dataset to estimate the performance of the surface-type classifiers trained with the different features. Cross-validations [107] including resubstitution and hold-out (two-fold validation) are performed to evaluate the four classifiers (three colour-based and one texture-based). Hold-out validation is performed by dividing the extracted training samples into two equal size subsets (subset 1 and subset 2) to be used as both training and testing data.

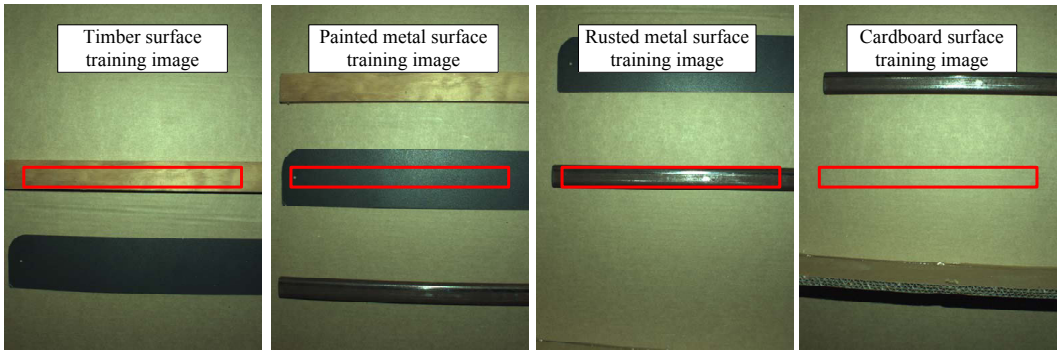


FIGURE 6.11: 1280×960 pixels training images collected for each surface-type

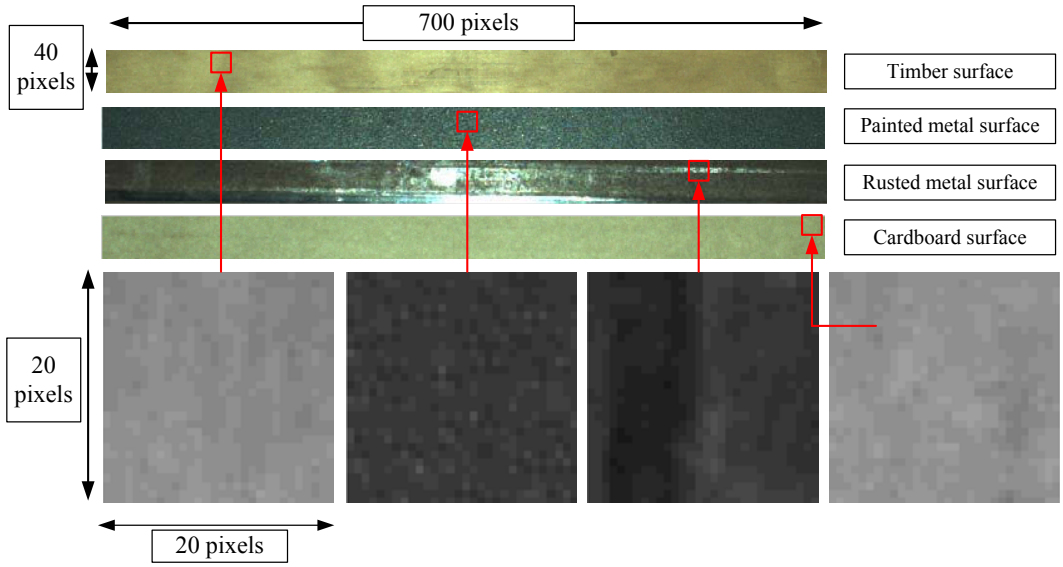


FIGURE 6.12: Training image dataset used to extract features to train surface-type classifiers

Tables 6.3 to 6.6 show the results of cross-validation for each of the four classifiers trained with different features. The diagonal cells (top-bottom) show the classification accuracy from resubstitution validation, and the remaining cells show the classification accuracy for hold-out validation when using all, subset 1 and subset 2 in different combinations of training and testing. From the results, it can be seen that the three colour-based classifiers have similar accuracy performance and the texture-based classifier has the highest resubstitution classification accuracy. However, hold-out validation shows that the texture-based classifier may be overfitting the model to the training data, given the lower accuracy observed in comparison to the other classifiers. For the case study, the four classifiers trained with all the samples extracted from the training image dataset will be used.

TABLE 6.3: Classification accuracy for SVM classifier trained with RGB features

		Classification accuracy (%) for RGB features		
		Training		
		All	**Subset 1**	**Subset 2**
Testing	**All**	99.34	91.95	93.83
	Subset 1	98.95	99.56	87.76
	Subset 2	99.74	84.33	99.91

TABLE 6.4: Classification accuracy for SVM classifier trained with a*b* features

		Classification accuracy (%) for a*b* features		
		Training		
		All	**Subset 1**	**Subset 2**
Testing	**All**	97.59	95.79	96.25
	Subset 1	96.44	97.52	93.32
	Subset 2	98.75	94.05	99.18

TABLE 6.5: Classification accuracy for SVM classifier trained with K_d features

		Classification accuracy (%) for K_d features		
		Training		
		All	**Subset 1**	**Subset 2**
Testing	**All**	96.95	95.42	96.21
	Subset 1	95.71	96.36	93.82
	Subset 2	98.19	94.49	98.60

TABLE 6.6: Classification accuracy for SVM classifier trained with LBP features

		Classification accuracy (%) for LBP features		
		Training		
		All	**Subset 1**	**Subset 2**
Testing	**All**	100	82.86	80.36
	Subset 1	100	100	60.71
	Subset 2	100	65.71	100

6.2 Experiment 1: Surface-type Classification with Viewing Distance Change

Experiment 1 demonstrates the surface-type classification approach when using the robot manipulator to position the RGB-D sensor package that maintains consistent viewing angle to the surface. Shown in Figure 6.13 is the experiment setup that contains surface-types including a rusted metal beam, painted metal surface, timber and cardboard. Figure 6.13 also shows the difference in illumination condition on the surfaces, provided by the ambient lighting of the laboratory (left) and only the LED light source (right). The image set is collected under the illumination conditions provided by only using the LED light source.

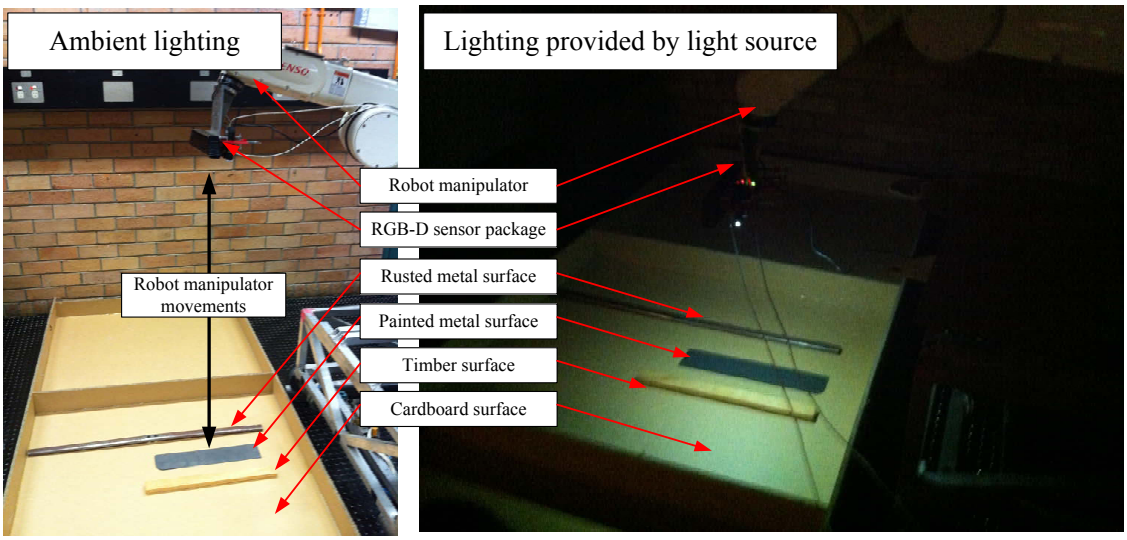


FIGURE 6.13: Experiment 1 setup of the laboratory environment

Figure 6.14 shows a set of 16 images collected from the environment by positioning the sensor package at different viewpoints using the robot manipulator. The camera exposure/optical settings remains unchanged from the training image dataset (from Section 6.1.5). Images 1–5 shown in Figure 6.14 are collected using the same viewing distance and viewing angle as the training image dataset (d_c = 650 mm, $\theta_c = 0°$). Only the observed part of the surface changes in images 15 (random positions; rotation and translation were not recorded), which has resulted in an image occluding a surface-type (i.e. image 1 is missing the timber surface). Images 6–16 in the set are collected at increasing viewing distances away from the surface plane while maintaining the viewing angle ($\theta_c = 0°$). The robot manipulator is used to position the RGB-D sensor package at 50 mm viewing distance increments for images 6–16, resulting in a final viewing distance of 1300 mm for image 16 from the surface plane.

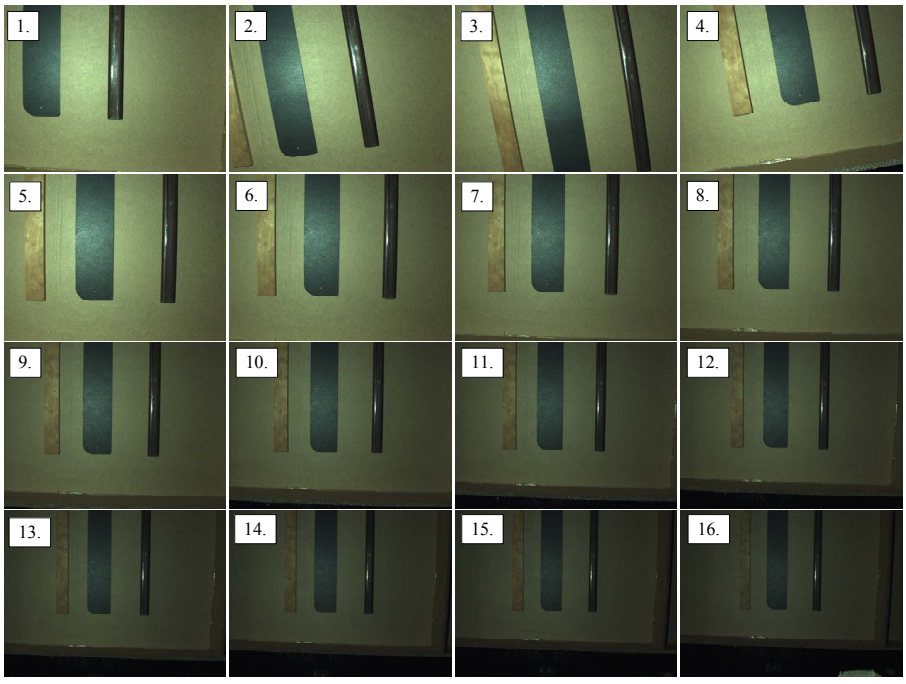


FIGURE 6.14: Images collected of the environment

Features including RGB, a*b*, K_d and LBP are extracted from each image using the same conditions that are used in Section 6.1.5 for training the classifiers. The four classifiers trained with the training image dataset in Section 6.1.5 are applied to classify the images. Figure 6.15 shows classification results produced by the classifier trained

with RGB features, Figure 6.16 shows the classification results produced by the classifier trained with a*b* features, Figure 6.17 shows the classification the results produced by the classifier trained with K_d features, and Figure 6.18 shows the classification results produced by the classifier trained with LBP texture features. Figure 6.19 shows the visualisation of the probability maps produced for the texture-based classification results (from Figure 6.18). The probability maps are produced using the parameters $\tau_n = 500$ mm, $\tau_f = 800$ mm, $\tau_\theta = 30°$, $\omega_1 = 1$, $\omega_2 = 1$, for all the surface-types.

Combined classification results are produced by combining the colour (K_d, Figure 6.17) and texture (LBP, Figure 6.18) classification results. For each image, the classification results for K_d and LBP, and a probability map are used to produce a combined surface-type map (Section 3.4.1). Figure 6.20 shows the combined classification results and Figure 6.21 shows the surface-type maps for Image 1, of K_d, LBP and the combined classification result.

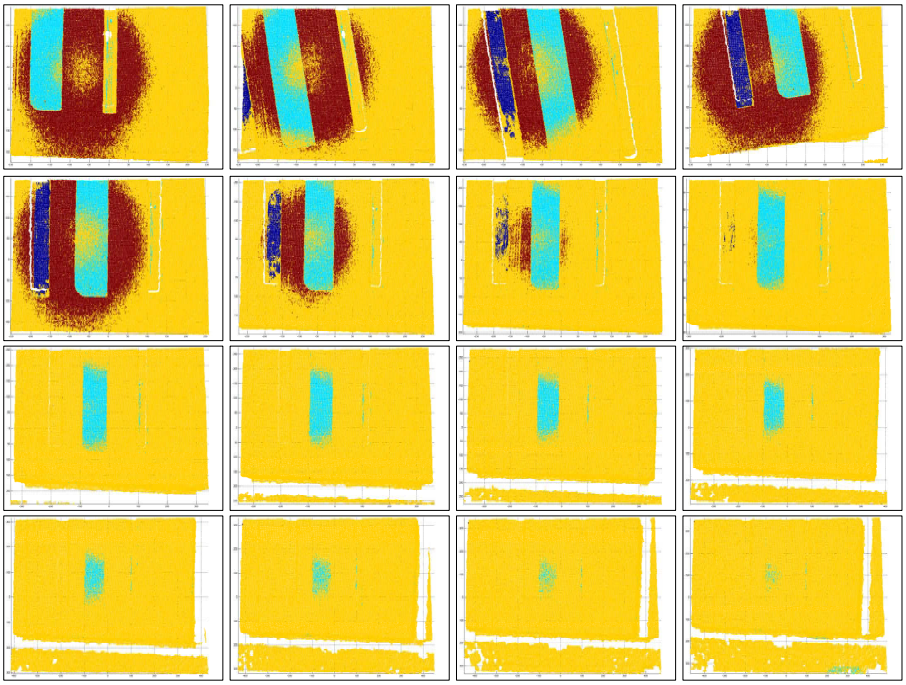


FIGURE 6.15: Classification results using classifier trained with RGB features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

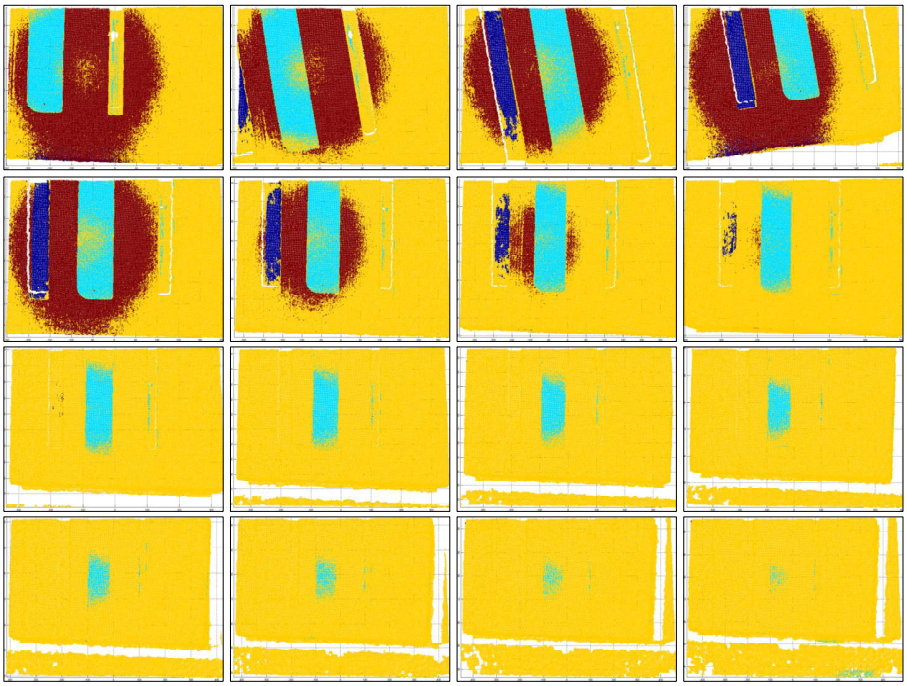


FIGURE 6.16: Classification results using classifier trained with a*b* features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

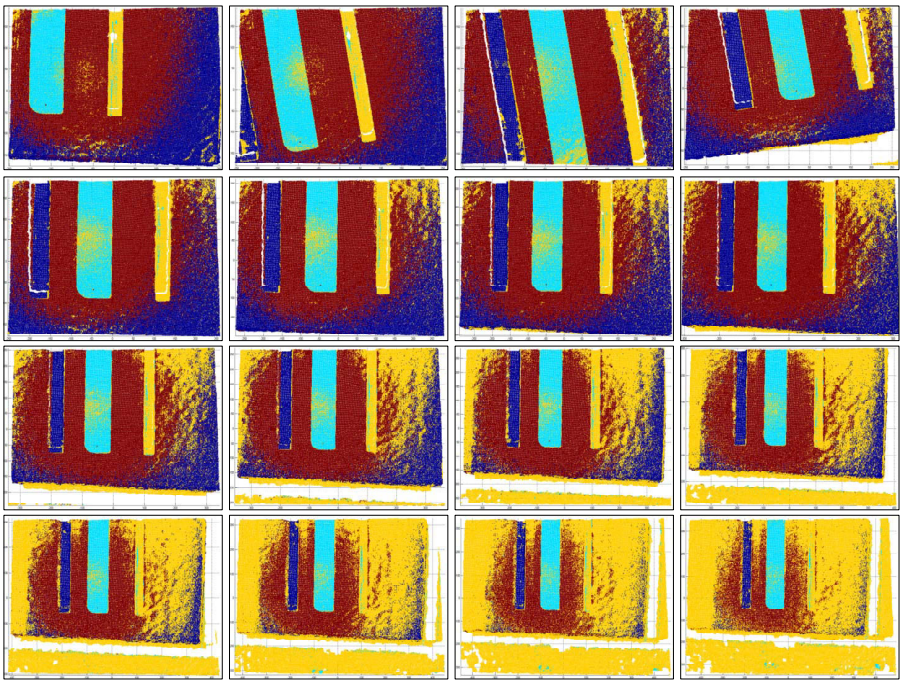


FIGURE 6.17: Classification results using classifier trained with K_d features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

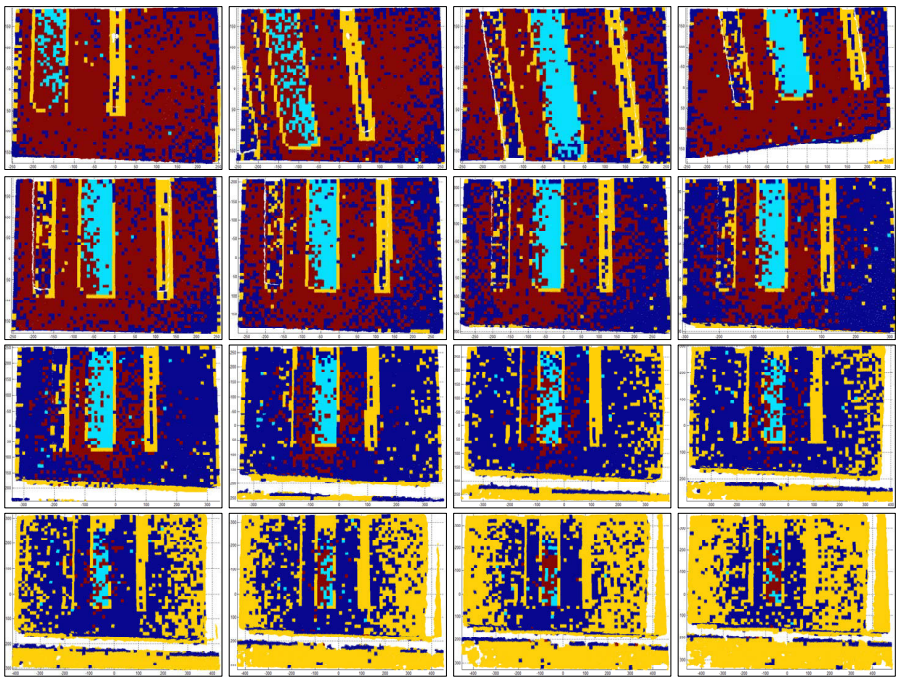


FIGURE 6.18: Classification results using classifier trained with LBP features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

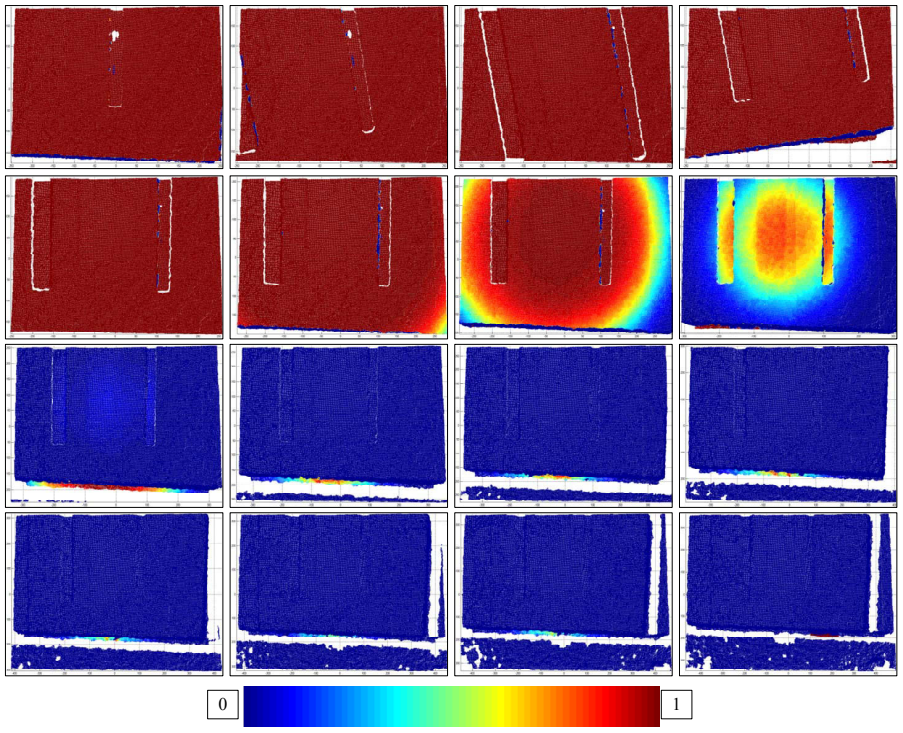


FIGURE 6.19: Probability maps for texture-based classification results

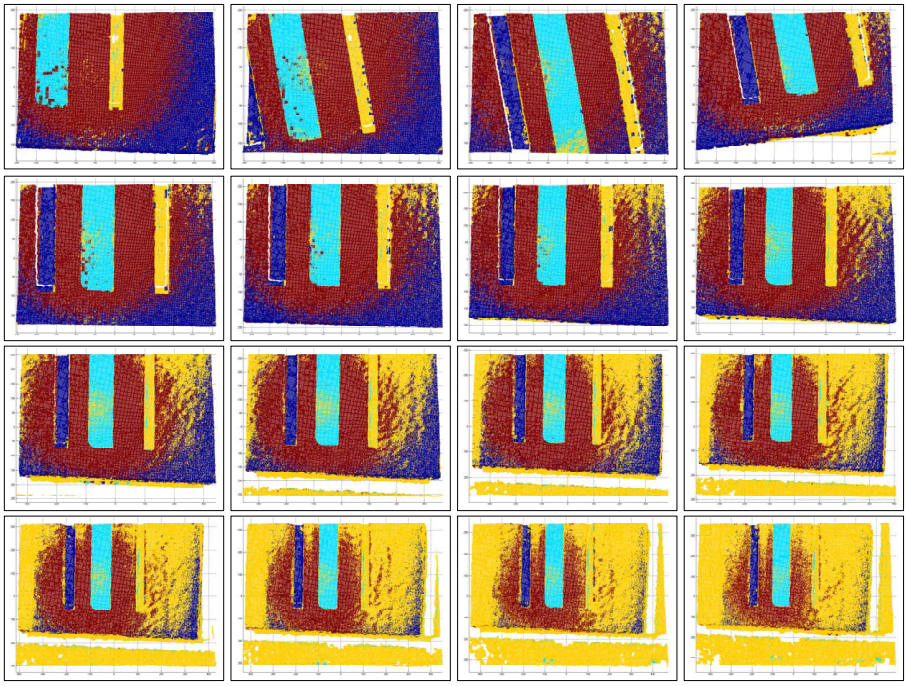


FIGURE 6.20: Classification results by combining K_d and LBP classification results: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

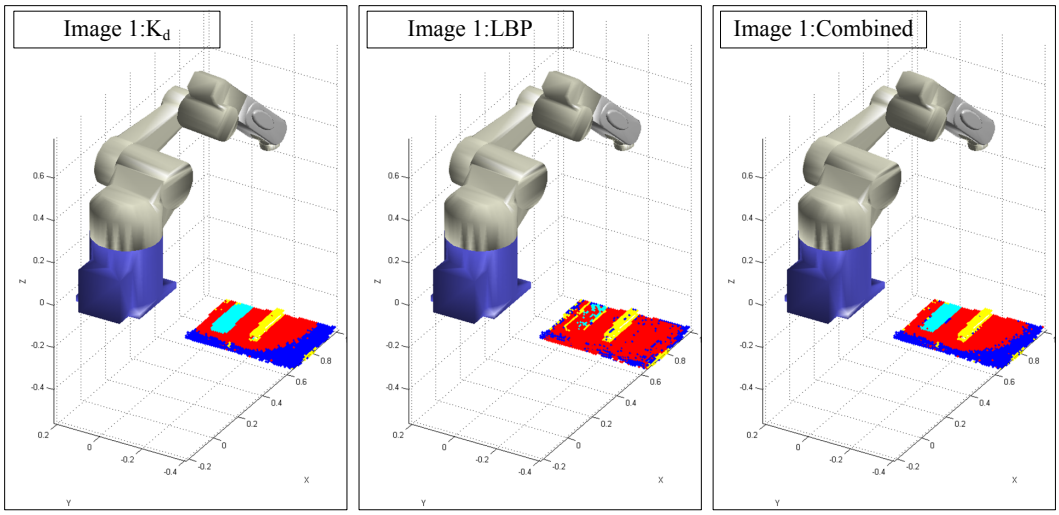


FIGURE 6.21: Image 1 surface-type maps generated using classification results from K_d, LBP and Combined: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

The overall accuracy of the classification result for each image is evaluated by comparing against a manually labelled ground truth of the image. In the experiment, this is

performed by converting into surface-type maps and then comparing against a benchmark surface-type map produced for this experiment (example shown in 6.10 and explained in Section 6.1.4). Figure 6.22 shows a column chart of the classification accuracy for each image. The horizontal axis shows the 16 images, and the vertical axis shows the accuracy percentage achieved by each set of features. From the chart, it can be observed that the classification produced using K_d features has the highest accuracy when compared with the classification produced using RGB and a*b* colour features for all 16 images. Please refer to Appendix E.2 for more details on the classification results of each surface-type; presented using confusion matrices.

Examining the classification accuracy for the images 1–5 which have the same image capture conditions as the training image dataset, RGB-based classification achieved accuracies between 42.30% and 52.9%, a*b*-based classification achieved accuracies between 44.21% and 56.19%, and K_d-based classification achieved accuracies between 57.89% and 66.55%. For images 6–16, where the viewing distance changes, RGB-based classification achieved accuracies between 2.69% and 30.82%, a*b*-based achieved accuracies between 2.70% and 33.15%, and K_d-based classification achieved accuracies between 22.24% and 64.98%. The results show that K_d features in comparison to RGB and a*b* can produce higher classification accuracy for both images with the same image capture conditions as the training image dataset, and for images with various illumination conditions.

For the texture-based classification using LBP features, higher accuracies were achieved for images 1–5 in comparison with the K_d-based classification. However, when the camera viewing distance is changed, the accuracy of the LBP-based classification fell below the K_d-based classification. For images 1–5, LBP-based classification achieved accuracies between 58.89% and 72.17%; and for images 6–16 between 12% and 66.07%.

For the combined results, which use the K_d-based and LBP-based classification results, classification accuracy was achieved in the range of 58.47% to 67.77% for images 1–5, and 22.24% to 65.37% for images 6–16. The combined classification results demonstrated improved accuracy over the K_d-based classification for images 1–8, and were equal in

accuracy for images 9–16. When compared against the LBP-based classification, the combined classification results demonstrated improved accuracy for images 2, 3 and 7–16.

In summary, this experiment has demonstrated the use of K_d to improve colour-based classification accuracy of images captured by an RGB-D sensor package positioned by the robot manipulator. In addition, this experiment demonstrated the use of K_d-based and LBP-based classification results with a probability map to produce combined classification results with improved accuracies.

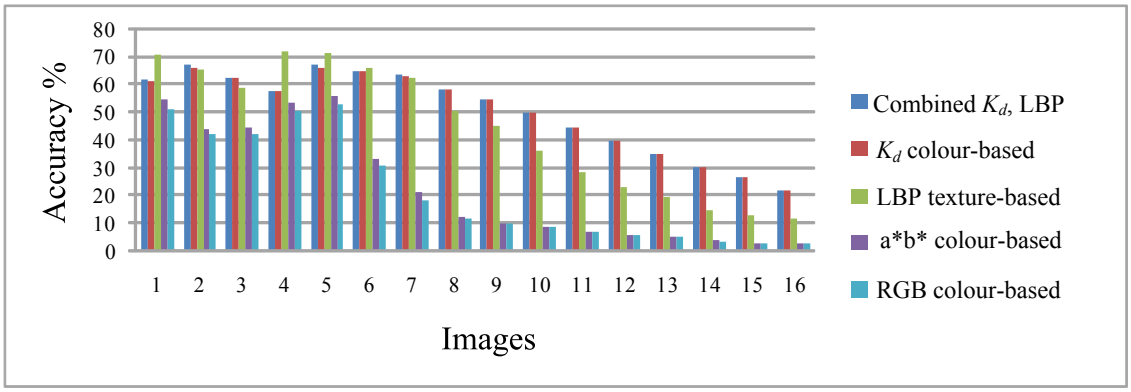


FIGURE 6.22: Classification accuracy for each viewpoint using the classifiers trained with different features

6.3 Experiment 2: Surface-type Classification with Viewing Distance and Viewing Angle Change

Experiment 2 demonstrates the surface-type classification approach when using the robot manipulator to position the RGB-D sensor package with viewing distance and viewing angle changes. Figure 6.23 shows the sweeping motion used to position the RGB-D sensor package from viewpoint 1 to viewpoint 6, used to collect images of surface-types including a rusted metal beam, painted metal surface, timber and cardboard. It is noted that the rotation motion used in this case study differs from the motion used to position the RGB camera in Chapter 5 (Experiment 2, rotated around the cameras optical axis). This is because the RGB-D sensor package in this case study has multiple cameras, and thus multiple optical centres also exist. Therefore instead of arbitrarily choosing one cameras optical axis to rotate around, rotation is performed using the last robot joint. Figure

6.24 shows the six images collected at position intervals between and including viewpoint 1 and viewpoint 6. The camera exposure/optical settings remain unchanged from the settings used to collect the training image dataset. Image 1 corresponding to viewpoint 1 is collected at a viewing distance $d_c = 650$ mm and viewing angle to surface $\theta_c = 0°$. From viewpoint 1, the angle of the 5^{th} joint is changed in 10° increments and an image is captured at each increment with the final position being viewpoint 6 that is used to collect image 6. The collected images have changes to both viewing distance and angle.

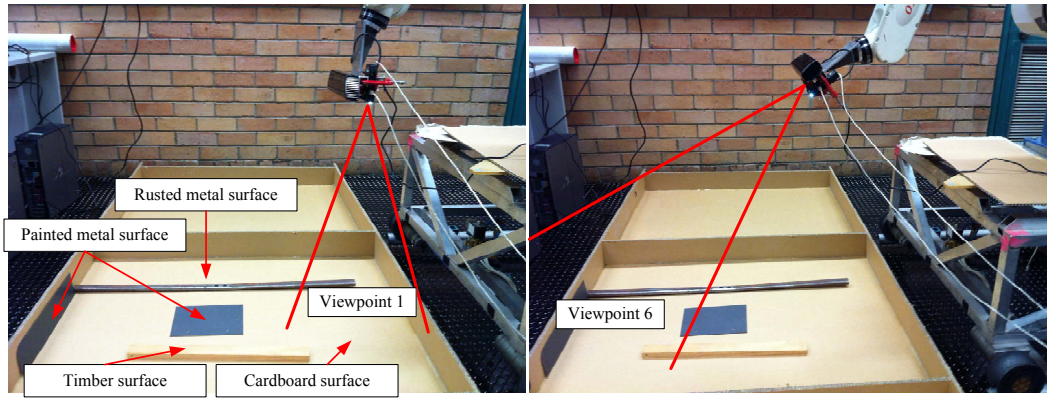


FIGURE 6.23: Experiment 2 setup of the laboratory environment

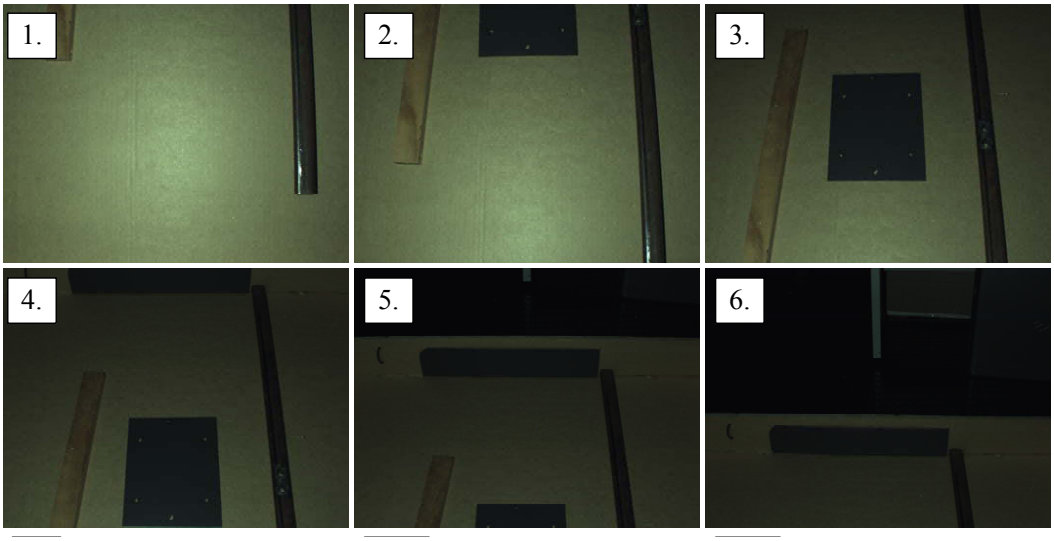


FIGURE 6.24: Images collected of the environment

Similar to Experiment 1, features including RGB, a*b*, K_d and LBP are extracted from each image. The four classifiers trained with the training image dataset from Section 6.1.5

are used to classify each of the images. Figure 6.25 shows the RGB-based classification results, Figure 6.26 shows the a*b*-based classification results, Figure 6.27 shows the K_d-based classification results, and Figure 6.28 shows the LBP-based classification results. Figure 6.29 shows the visualisation of the probability maps produced for the LBP-based classification results.

Combined classification results are produced by combining the colour (K_d, Figure 6.27) and texture (LBP, Figure 6.28) classification results. For each image, the classification results for K_d and LBP, and a probability map are used to produce a combined surface-type map. Figure 6.30 shows the combined classification results and Figure 6.31 shows the surface-type maps for K_d, LBP and the combined classification results of image 1. It is to note that images 5 and 6 exhibits depth sensor noise and this is observable in Figure 6.25 to Figure 6.30. This noise appears as speckle points with gaps of no data on top half of the two images; this is a characteristic of the Kinect sensor when no depth reading is returned for a surface point.

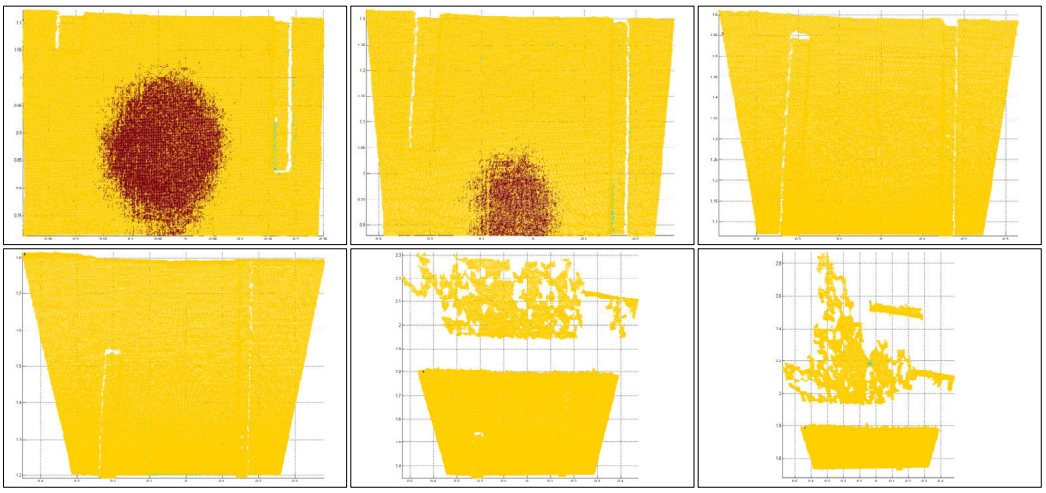


FIGURE 6.25: Classification results using classifier trained with RGB features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

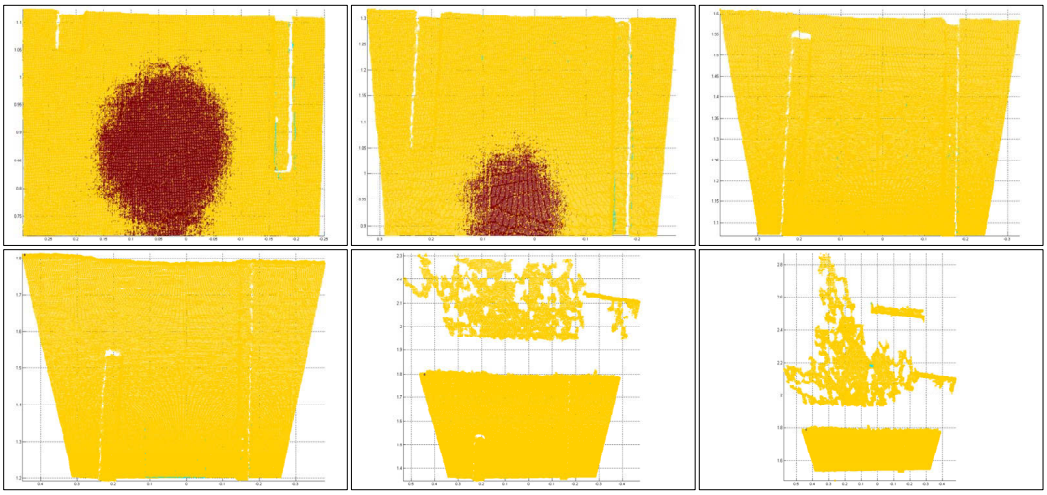


FIGURE 6.26: Classification results using classifier trained with a*b* features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

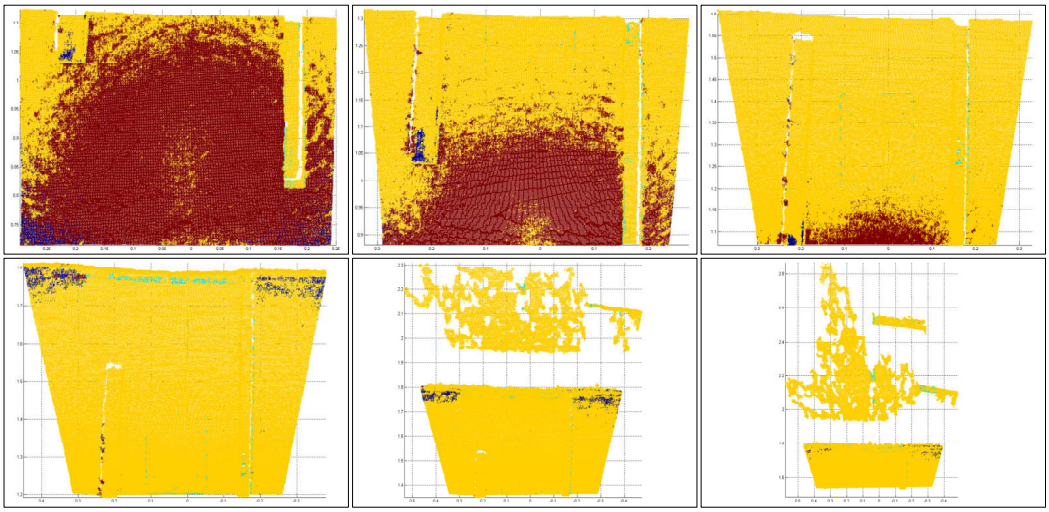


FIGURE 6.27: Classification results using classifier trained with K_d features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

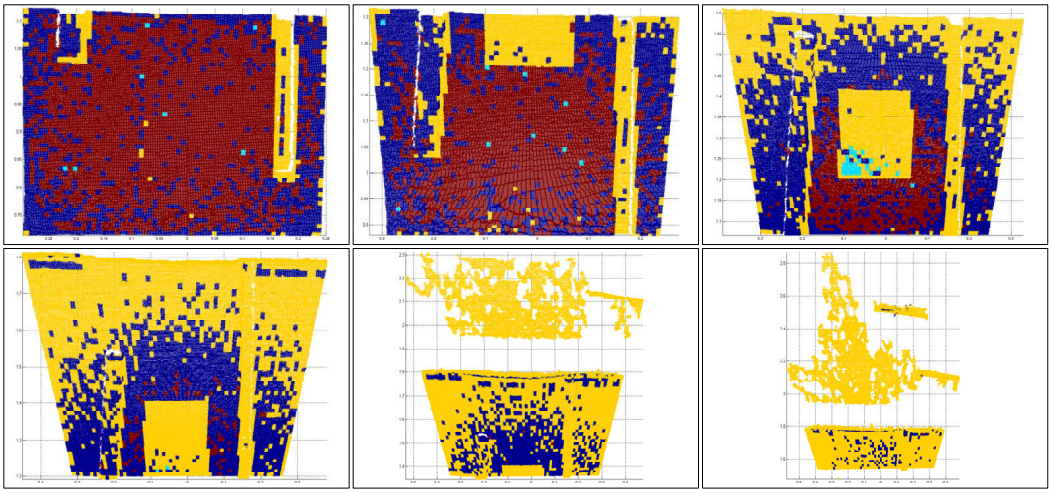


FIGURE 6.28: Classification results using classifier trained with LBP features: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

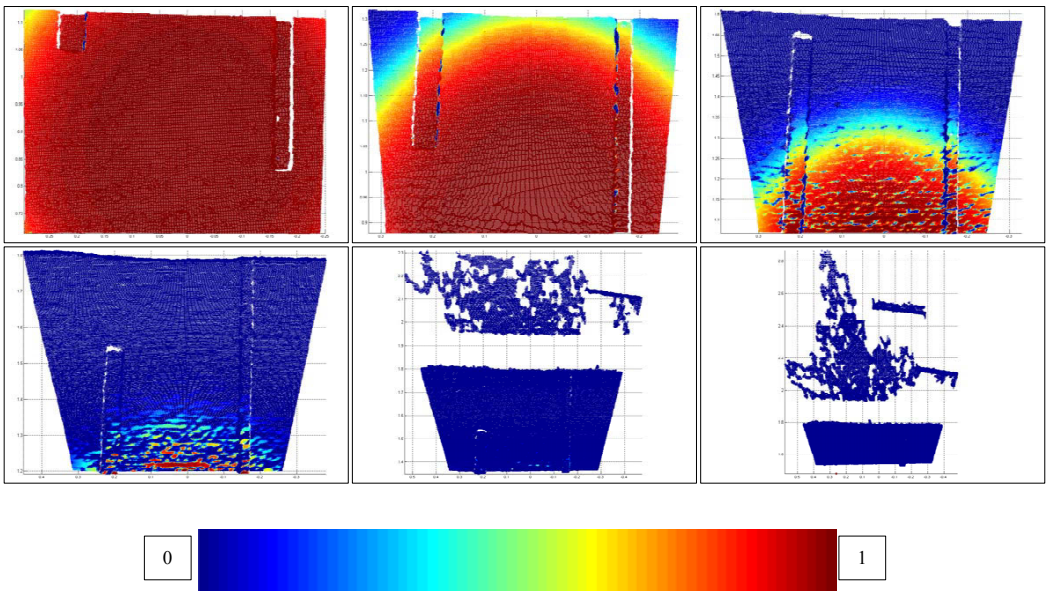


FIGURE 6.29: Probability maps for texture-based classification results

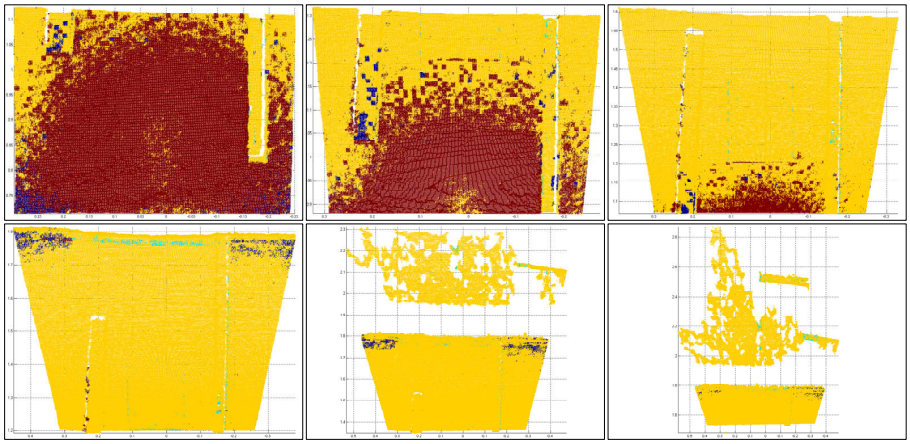


FIGURE 6.30: Classification results produced by combining K_d and LBP classification results: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

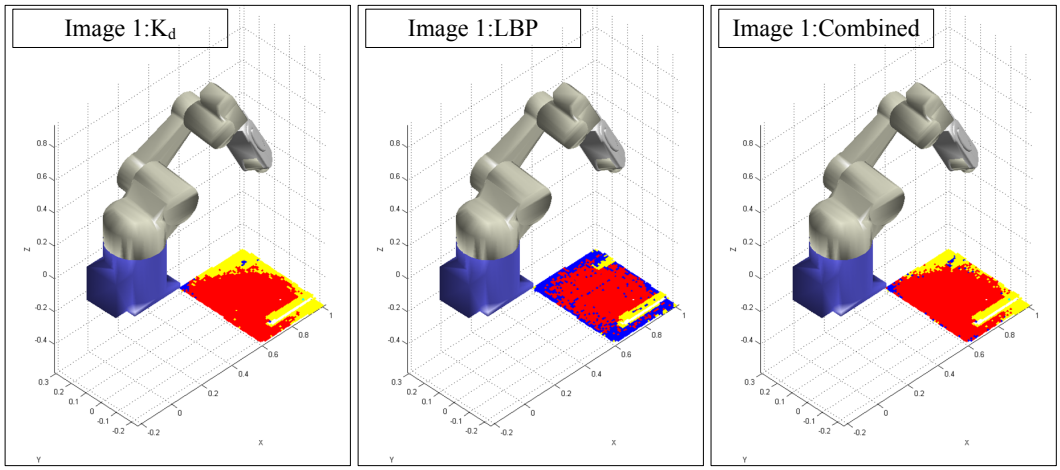


FIGURE 6.31: Image 1 surface-type maps generated using classification results from K_d, LBP and Combined: Timber surface is dark blue, painted metal surface is sky blue, rusted metal surface is yellow and cardboard is red

The accuracy of the classification results are also evaluated by converting them into surface-type maps such that they can be compared against a benchmark surface-type map. Figure 6.32 shows a column chart of the classification accuracy of each image. Please refer to Appendix E.3 for more details on the classification results of each surface-type; presented using confusion matrices. The horizontal axis shows the six images, and the vertical axis shows the accuracy percentage achieved for each set of features. From the chart, it can be observed that the classification produced using K_d features has the highest accuracy when compared against RGB and a*b* colour

classification for all six images. The results show that K_d features in comparison to RGB and a*b* can produce higher classification accuracy for the images.

Examining the colour-based classification of the images, RGB-based classification achieved accuracies between the range of 1.48% to 28.75%, a*b*-based classification achieved accuracies between 1.48% and 33.43%, and K_d-based classification achieved accuracies between 1.50% and 78.15%. The results indicate that K_d features extracted from images with various illumination conditions can produce higher accuracy in comparison with RGB and a*b*. Examining the texture-based classification of the images, LBP-based classification achieved accuracies between 3.11% and 74.37%; and examining the combined classification results of K_d and LBP, an accuracy range of 1.50% to 78.40% was achieved. The results indicate that the combined classification results improved or were equal in accuracy to the K_d-based classification. When compared against the LBP-based classification, the combined results showed improved accuracy for image 1, but lower accuracies for the remaining images. This is because the LBP-based results had low weighting in the contribution to produce the combined classification results for images 3 to 6 due to the low weighting values provided by the probability maps (Figure 6.29).

In summary, this experiment has demonstrated the use of K_d to improve colour-based classification of images captured by an RGB-D sensor package positioned by the robot manipulator. In addition, this experiment demonstrated the use of a probability map to combine the classification result from an LBP-based classification with a K_d-based classification. The combined classification results showed improved or equal accuracy to the K_d-based classification and image 1 for LBP-based classification.

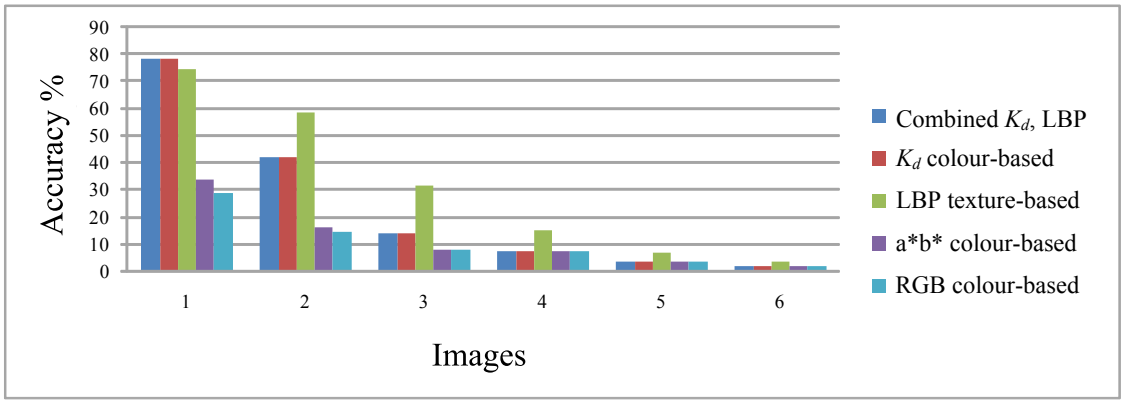


FIGURE 6.32: Classification accuracy for each viewpoint using the classifiers trained with different features

6.4 Discussion

The case study has demonstrated the surface-type classification approach using an RGB-D sensor package mounted to the end-effector of a robot manipulator. Two experiments were performed to demonstrate surface-type classification when the robot manipulator is used to execute different types of movements. Experiment 1 showed the use of movements where the priority was to position the RGB-D sensor package with linear movements such that a constant viewing angle is maintained. This movement option resulted in large changes in robot poses and is suitable when there are no obstacles in the environment for collision. Experiment 2 showed the use of a sweeping movement by moving a single joint to position the RGB-D sensor package. The images contained surfaces with changes to both viewing distance and viewing angle. This movement option can be used when the robot manipulator is at the reach limits or when there is an obstacle.

It was also observed that due to the training image dataset being collected by cropping from the centre region of each image (Figure 6.11), the illumination fall-off in the corners of the images has not been provided in the training image dataset. Hence, all the images classified in Experiments 1 and 2 showed high misclassification for the corner edges. To enable practical use of the approach, a more extensive training image dataset will need to be collected and evaluated, and an additional restriction will also need to be placed on the viewing distance and viewing angle. This is because the light intensity will eventually be

too low for classification, and the selected reflectance model is appropriate for viewing angle of incidences below 60° [108]. For future work, analysis will be performed to determine the relation between illumination fall-off observed in the corners of the images and poor classification results. Furthermore, future work will extend into analysing the factors of illumination fall-off, including a cameras vignetting and the light fall-off characteristics of the selected light source. In this way it is possible to design an improved sensor package which has minimal illumination fall-off at the edge of the images; for the specified operating viewing distance and angle range at which data will be captured from.

Chapter 7

Conclusions

This thesis has presented an approach to classify surface-types in a complex structural environment under various illumination conditions. Two algorithms have been developed: (a) a colour feature extraction algorithm that can improve classification accuracy of an image under various illumination conditions; and (b) an algorithm to generate a probability map that can be used to assess classification results. The two algorithms have been used in the surface-type classification approach to produce colour-based and texture-based classification results that are combined and represented in a surface-type map in 3D.

7.1 Summary of Contributions

7.1.1 A Surface-type Classification Approach using RGB-D Images

The presented approach to classify surface-types using RGB-D images provides a solution to classify a complex environment under various illumination conditions. The approach combines the use of a vision-based classification method with the addition of depth information to improve the accuracy of surface-type classification. The approach also proposes the use of a robot manipulator as the means to position an RGB-D sensor package in complex environments in order to collect RGB-D images at different observation viewpoints. In this way, surface-type classification results can be combined

by using the 3D position of the pixel in a global frame (robot base coordinate frame). Consequently, an awareness of the surrounding environment can be incrementally built up by making observations from different viewpoints and then combining the information gained from the viewpoints into a global surface-type map.

7.1.2 An Algorithm for Colour Feature Extraction

An algorithm that uses an RGB-D image and a modified reflectance model has been presented to extract colour features that can be used to classify surface-types in images captured under various illumination conditions. The existing Torrance-Sparrow reflectance model has been selected and modified with a camera radiometric function to derive for the colour features. The algorithm estimates the relative distance and angle between a point on a surface and the light source. Using the estimated position information and a reflectance model, a set of diffused reflectance values are calculated for each RGB image pixel. These values can be used as colour features to classify surface-types. It was demonstrated that the extracted features showed improvements in surface-type classification over existing colour conversion approaches including the CIELab L*a*b colour-space. An existing classifier (SVM) was utilised to compare the different feature sets.

7.1.3 An Algorithm for Classification Result Assessment

An algorithm for calculating a probability map to assess the classification results produced by a texture-based classifier has been presented. This algorithm can be used for images that are captured in a complex structural environment under various image capture conditions (viewing distance and viewing angle), such that the appearance of a surface-type can vary texturally from image to image. The probability map contains the probability of pixels in an RGB image being a specific surface-type and can be used to assess the classification results of an RGB image classified using texture features. The probability value is calculated based on the difference in image capture conditions between an image pixel and the training image dataset. The probability map can be applied to identify the regions in an image that have a high probability of being accurately classified, and can be used to provide a weighting factor when combining

texture-based classification results into a surface-type map. Existing techniques for extracting texture features including GLCM (Chapter 5) and LBP (Chapter 6), and classification tools Naive Bayes (Chapter 5) and SVM (Chapter 6) have been used to demonstrate the outcomes in the algorithm.

7.1.4 Practical Contribution

The thesis has implemented the surface-type classification approach on a robotic system designed for steel bridge maintenance. The robotic system consists of a six degrees-of-freedom Denso VM-6083 robot manipulator with an RGB-D sensor package mounted to the end-effector. The RGB-D sensor package setup consists of a Point Grey Firefly RGB camera, a Microsoft Kinect and a light source. Experiments have been conducted to explore an environment for surface-types. Different types of robot manipulator movements were used to collect the RGB-D images for surface-type classification. The performance of the surface-type classification approach was evaluated by comparing the classification results generated using different colour features (RGB, a*b*, K_d) and texture features (LBP). It is important to highlight that various existing programming libraries and software packages have been used to develop the propose approach and do not form part of the contribution. The following is a list of the key softwares and libraries that have been used:

1. Matlab functions (including function to convert from RGB to CIELab colour space)
2. Photoshop for resizing, scaling, and simulated light function
3. OpenNI library and mex functions for Matlab `https://github.com/OpenNI/OpenNI`
4. LIBSVM `www.csie.ntu.edu.tw/~cjlin/libsvm`
5. Camera calibration `http://www.vision.caltech.edu/bouguetj/calib_doc/`

7.2 Discussion of Limitations

The application of a vision-based system to inspect surfaces in a structurally complex 3D environment is one of the challenges in robotics research. This thesis has addressed the challenging issues of classifying surface-types in a structurally complex environment under various illumination conditions. However, to fully realise a machine vision system capable of operating in a harsh environment, such as in bridge maintenance, the following issues need to be further addressed:

- The presented colour feature extraction algorithm is currently limited to planar surfaces with Lambertian reflectance that can be illuminated using a light source and captured using an RGB camera. Surface-types that may not be classified correctly by this presented algorithm include transparent surfaces, highly reflective surfaces and non-reflective surfaces, because the diffused reflectance values may not be extracted. However, the lack of diffused reflectance off a surface-type can be used as a feature to increase the types of surface-types that are currently classifiable. In addition, the specular reflectance component of an image is another potential feature that can be used to extract information on surface-types and needs to be further investigated.

- Calculation of viewing distance and viewing angle for complex shaped surfaces in a field environment is a challenging task when using RGB-D images. The current generation of RGB-D cameras based on structured-light (Kinect, Xtion, Primesense) and ToF (Kinect2, PMD Nano) are all capable of providing dense point cloud representation of surfaces. However, the point cloud can be noisy, especially in an environment such as a steel bridge grit-blasting environment. For example, the PMD Nano ToF sensor used in Chapter 5 (also applicable to active stereo vision sensors) can suffer from multi-path interference depending on the viewing distance and viewing angle of the sensor to a surface in the environment; thus usable for only a limited viewing distance and viewing angle range that has been tested to produce good depth readings. As a result, it is difficult to calculate the viewing distance and viewing angle for each RGB pixel as required by the two algorithms developed. A potential strategy to overcome this limitation is to use *a priori* knowledge about the surface geometry in the sensed environment. Given *a priori* knowledge, it is possible to fit template

surfaces to the point cloud. By using the point cloud to associate a surface shape (i.e. from a library of CAD shapes) to the position and orientation in the environment, a noise-less geometry of the surface can be produced. From the noise-less geometry of the surface, it is possible to calculate the viewing distance and viewing angle for pixels captured of that surface.

- The use of the RGB-D information to re-render texture on surfaces as if it were captured at specific image capture conditions is another possibility for texture-based classification. In work such as [56], the idea of re-rendering the texture of a surface-type as if it was captured at a specific viewing distance and angle has been suggested. Given the quality of RGB-D images produced by the current generation of sensors, it is possible to modify the texture appearance of a surface-type to match it with the training image dataset, such that classification accuracy can be improved. However, a limitation of re-rendering surfaces, which can address the issue of spatial scaling/resolution and perspective distortion, is that it assumes all the surfaces that appear in an image are in focus. Surfaces in an image that do not originally have focus quality (i.e. out-of-focus) will not benefit from re-rendering. Therefore, the probability map produced by the algorithm in Chapter 5, which considers focus quality, can be applied to determine whether re-rendering the texture appearance of a surface will improve classification accuracy.

- The current use of a probability map is to assess if the surface-type of each pixel has been classified accurately. However, the current probability map provides a conservative probability value of a pixel being a specific surface-type. For example, the threshold range to calculate the probability value (based on viewing distance and viewing angle) can be set to a smaller and more conservative range than the actual range, which can be used to classify surface-types accurately. Therefore, classification results that are actually accurate may be discarded given the selected threshold. Currently, the probability map is used to provide an assessment of the classification results. Further investigation will be necessary to examine how to use the probability map to assess classification results for improved surface-type classification accuracy.

  One possibility is to generate the probability map to provide probability values for each pixel of being each surface-type. Then the most probable surface-type(s) for each pixel

can be identified and used as additional information to make an assessment on whether the classification result for a pixel is accurate. This can be further improved by capturing the training image for surface-types at different image capture conditions (currently all training images are captured with the same image capture conditions). For example, if the training image for surface-type 1 is captured at a viewing distance of 100 mm, and surface-type 2 at a viewing distance 200 mm, a probability map generated for an image captured at viewing distance 100 mm can identify for each pixel that surface-type 1 is most probable. The use of this most probable surface-type information can be combined with the current use of a probability map to assess the classification result. This needs to be investigated as part of future work.

- The optimal use of the combination of features (colour-based and texture-based) and the machine learning algorithms, including the naive Bayes and SVM, have not been explored in this thesis. It is understood that the selected classifiers require parameter tuning to maximise accuracy. Parameter tuning such as pixel window size selection, feature reduction, dataset normalisation, and kernel selection, can all be performed to improve accuracy. However, for the presented algorithms, the selected classifiers were used to demonstrate the relative accuracy improvement gained from using different features. Therefore, parameter tuning of the classifier has not been a primary focus in this research.

7.3 Future Work

Based on the research of this thesis, a number of future works have been identified. These areas are beyond the scope of this current thesis, but it is hoped that these topics will inspire future work in this interesting area of field robotics.

- At the current stage, the computational performance of the approach when implemented in robotic grit-blasting in a steel bridge has not been evaluated. As part of future work suitable hardware for a small form-factor sensor package and processing unit will be selected, fitted to a grit-blasting robot, and computational performance tests will be conducted to determine whether the approach meets the online inspection challenge.

- It is currently challenging to calculate accurate surface normals for points on non-planar surfaces given the clutter of the scene and sensor noise. Therefore, potential future work of surface template matching to improve the accuracy of the viewing distance and viewing angle for each pixel is possible.

- A robot manipulator can be used to position the sensor package in various positions, which is very useful in complex environments. It is also possible to implement 3D point matching algorithms to identify the coordinate frame of the RGB-D sensor package relative to a global frame. Given an application environment suitable for 3D point matching algorithms, it is possible to implement the surface-type classification approach in a portable manner without the need for a robot manipulator.

- In terms of improving the time-efficiency of the approach, it is possible to investigate a viewpoint selection algorithm that can reduce the number of observations required to explore an environment for surface-types. Provided with a geometric map of the environment prior to surface-type exploration and the means to position the sensor package at exact positions and orientations, it is possible to generate a minimal number of viewpoints that can be used to completely inspect an environment for surface-types.

- Future work for bridge maintenance applications should extend this approach to address the issues of dust and vibration, where grit-blasting creates a dust laden environment and vehicle traffic on the top deck creates vibration on the steel structure.

Appendix A

IR Camera Hand-eye Calibration

Appendix A is an extract from the published works [84] on an automated and cost-effective method of hand-eye calibrating an IR depth camera to the end-effector of a robot manipulator. The work is additional to the approaches and algorithms presented in this thesis that utilises a RGB-D sensor package mounted to a six degree-of-freedom robot manipulator to collect RGB-D image. This Appendix is intended as supplementary material for the setup of an IR depth camera to the end-effector of a robot manipulator for surface-type classification.

A.1 Methodologies

There are three main assumptions that must be made to calibrate the hand-eye transform ${}^{e}T_{s}$ for an IR depth camera mounted to a robot manipulator. Firstly the model of the robot manipulator must be accurate [109] [110]. Secondly, the intrinsic of the IR depth camera must be known [111]. Finally, reflector discs must be installed such that the robot base-to-object relationship can be established using the technique [112]. Figure A.1 shows the overview of the hand-eye calibration approach. A set of robot manipulator poses are selected so the IR depth camera on the end-effector is positioned and orientated to capture IR images containing all the reflector discs. The pixels representing the reflector discs in the IR image have significantly higher intensity values than the pixels of the surrounding

environment, and thus the location in the image can be extracted. A 3D point cloud is generated from the depth image and IR camera's intrinsic model. Using the IR image and 3D point cloud, a feature point in 3D can be selected from the centre of each reflector disc using a combined approach of image processing, plane fitting, and circle fitting techniques. The camera-to-robot base transform ${}^{o}T_{s}$ is calculated through feature correspondence by matching the extracted feature points (in the camera frame) with the established ground truth location (in the robot base frame) of the feature points. The hand-eye transform ${}^{e}T_{s}$ is then calculated using the camera-to-robot base transform and the end-effector-to-robot base transform ${}^{o}T_{e}(Q)$ is calculated from the model of the robot manipulator.

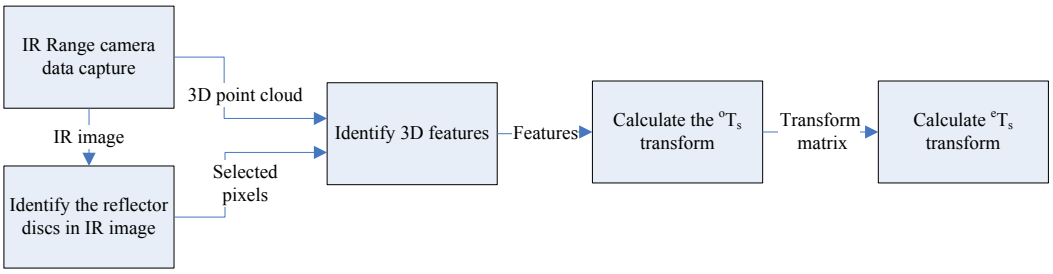


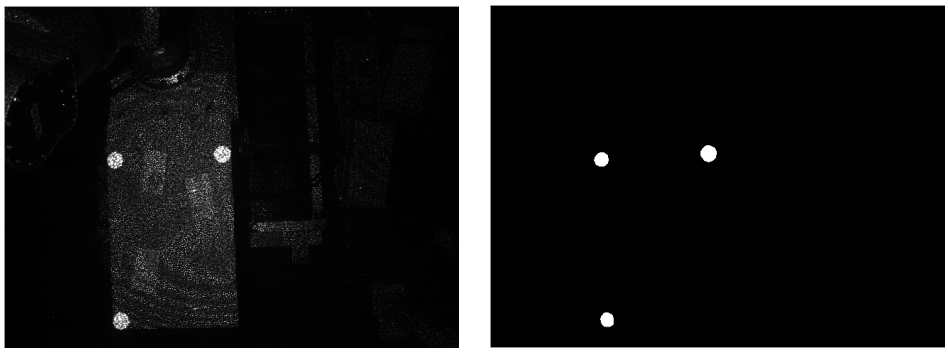
FIGURE A.1: Overview of the process for hand-eye calibration

A.2 Feature Points Identification

3D features corresponding to reflector discs are extracted from the IR image and the 3D point cloud. The captured IR image of the reflector discs (shown in Figure A.2*a*) is processed using a median filter to isolate the reflector disc pixel from the projected IR pattern. An incrementally increasing filter kernel is used to remove the projected IR pattern gradually from the IR image and terminates when only the reflector disc blobs remain in the IR image. Region fill is then applied to the filtered image to remove the holes in the reflector disc blobs and then the image is converted from greyscale to black & white by thresholding the intensity values. The resulting binary image is shown in Figure A.2*b*. Each of the pixel in the blobs are indexed to the corresponding 3D coordinate in the point cloud to represent the reflector discs' location in 3D as shown in Figure A.3*a*.

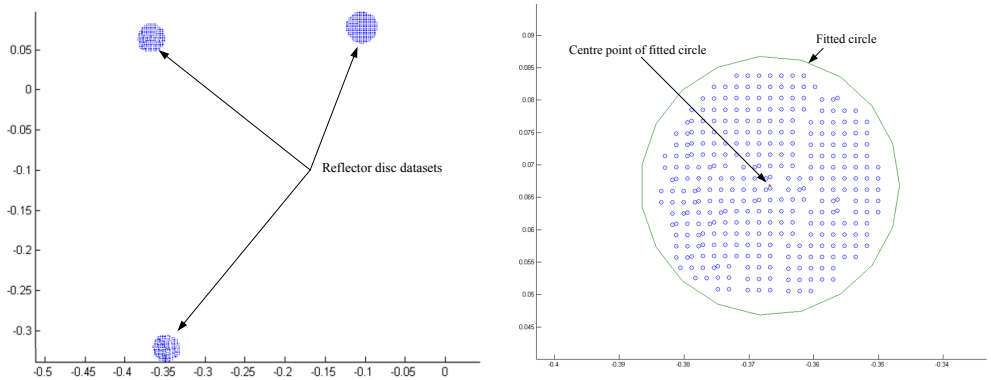
$$\beta = \arccos(V_n \bullet V_x) \tag{A.1}$$

$$\gamma = \arccos(V_n \bullet V_y) \tag{A.2}$$



(a) (b)

FIGURE A.2: *a*) IR image; *b*) Binary image of reflector discs



(a) (b)

FIGURE A.3: *a*) Datasets of points in 3D representing the reflector discs; *b*) Circle fit on a dataset

A geometric least-squares fitting of a circle [113] is applied to extract the centre point [X_c, Y_c], of the fitted circle as shown in Figure A.3*b*. Since the radius, r is known *a priori*, the following error function is minimised to identify [X_c, Y_c],

$$F(X_c, Y_c) = \sum_{j=1}^{N} \sqrt{(X_j - X_c)^2 + (Y_j - Y_c)^2} - r \tag{A.3}$$

The depth value of the centre point, Z_c is calculated using the plane coefficients from the dataset D_j and X_c, Y_c,

$$Z_c = \frac{aX_c + bY_c + d}{-c} \tag{A.4}$$

The 3D feature point is calculated for each reflector disc and used to match against the known ground truth location of the reflector discs relative to the robot base, further discussed in section A.3.

A.3 Camera-to-robot Transform through 3D Feature Matching

In order to calculate the camera-to-robot base transform, ${}^{o}T_{s}$, a feature matching technique is used to relate the IR depth camera frame with the robot base frame. The robot base-to-object relationship describing the object in the robot base frame is assumed to be static and in a position and orientation that can be easily measured, and the camera-to-object relationship describing the object in the IR depth camera frame is dynamic. As shown in Figure A.4*a*, the transform ${}^{o}T_{s}$ describes the translation and rotation required to match the object in the IR depth camera frame to the measured location of the object in the robot base frame.

M number of reflector discs placed in a specific configuration on a flat plane are used to determine the correspondence between the two coordinate frames. The location of the reflector discs relative to the robot base is known accurately by moving an end-effector pointer tool to the reflector disc positions as in [112] and the camera-to-object relationship is obtained using the method described in Section A.2. Feature point matching is performed using Singular Value Decomposition (SVD) [114] to find the least-square fit of feature points in the IR depth camera frame, f_i with the feature points in robot base frame, f_i' where $i = 1, 2, ...M$, using the equation

$$f_i' = {}^{o}R_s f_i + {}^{o}t_s + N_i \tag{A.5}$$

where ${}^{o}R_s$ is the rotation matrix, ${}^{o}t_s$ is the translation vector and N_i the noise vector. The least-square solution to ${}^{o}R_s$ and ${}^{o}t_s$ is taken as the camera-to-robot base homogeneous transform ${}^{o}T_{s}$,

$${}^{o}T_{s} = \begin{bmatrix} & {}^{o}R_s & & {}^{o}t_s \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.6}$$

A.4 Hand-Eye Transform and Point Cloud Registration

In order to register a point cloud from the IR depth camera frame to the robot base frame, the hand-eye transform ${}^{e}T_{s}$ is calculated. As shown in Figure A.4*b*, the hand-eye transform is constant between the camera-to-robot base frame and the end-effector-to-robot base frame for any given robot manipulator pose. Using the camera-to-robot base transform ${}^{o}T_{s}$ (Section A.3) and the end-effector-to-robot base transform ${}^{o}T_{e}(Q)$ calculated from the corresponding robot manipulator pose with joint angles, $Q = [q_1, q_2, ...q_6]^T$ using the D-H model of the robot manipulator,

$$ {}^{o}T_{e}(Q) = \prod_{i=1}^{6} {}^{i-1}T_{i}(q_i) \tag{A.7} $$

the following expression is solved for the hand-eye transform.

$$ {}^{e}T_{s} = {}^{o}T_{s} \times {}^{o}T_{e}(Q)^{-1} \tag{A.8} $$

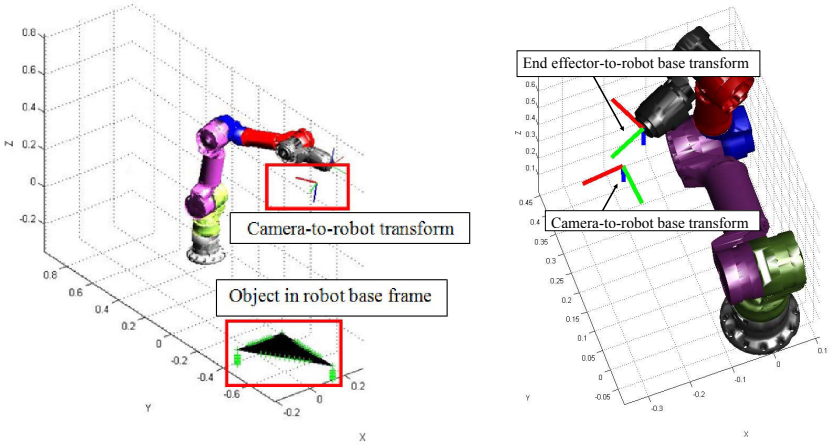
Using the calculated ${}^{e}T_{s}$, the registration of a point cloud P_k in the IR depth camera frame to the robot base frame, P'_k where $k = 1, 2, ...N$, is achieved by determining the camera-to-robot base transform ${}^{o}T_{s}(Q)$ for the end effector-to-robot base transform ${}^{o}T_{e}(Q)$, specific to a robot manipulator pose with joint angles, $Q = [q_1, q_2, ...q_6]^T$,

$$ {}^{o}T_{s}(Q) = {}^{o}T_{e}(Q) \times {}^{e}T_{s} \tag{A.9} $$

where

$$ [P'_k, \mathbf{1}] = [P_k, 1] \times {}^{o}T_{s}(Q) \tag{A.10} $$

It is noted that the vector of ones is the same length as the point cloud. The hand-eye transform is used to register point clouds, from the IR depth camera frame into the robot base frame. A rich geometry map of the environment can be constructed using a single IR depth camera by moving the manipulator into different poses [7].



(a) (b)

FIGURE A.4: *a*) An IR depth camera attached to a robot manipulator observing the calibration plate; *b*) Camera-to-robot base frame and end-effector-to-robot base frame

A.5 Limitations and Concluding Note

Currently the hand-eye calibration approach is limited by the viewing distance. Due to intensity saturation of pixels in the IR image at close viewing distances, an accurate depth reading of the reflector disc is not possible. Thus, the feature extraction scheme could be modified to select 3D dataset points from the surface surrounding each reflector disc as opposed to the surface on each reflector disc. The depth returns for pixels surrounding each reflector disc does not suffer from intensity saturation; therefore feature extraction should be possible for a closer viewing distances. A redesign of the calibration plate is necessary to provide the surface plane around each reflector disc.

Appendix B

Texture Features

B.1 Grey Level Co-occurrence Matrix

Grey Level Co-occurrence Matrix (GLCM) texture features [10] are used in this thesis to demonstrate the effect of image capture condition changes on texture consistency and surface-type classification accuracy. Comparative studies have demonstrated that GLCM features can provide high classification accuracy results [71][115]. More specifically, GLCM texture features for steel surface classification has been demonstrated in [9][70].

A GLCM is created based on a spatial relationship defined between a reference and neighbour pixel, which can include the direction (i.e. 0°, 45°, 90°) and the number of offset pixels between the reference and neighbour. Given a spatial relationship between pixels, the occurrence of different combinations of intensity levels between the compared reference and neighbour pixels are tallied and recorded in the corresponding position of the GLCM. The following GLCM features can be extracted: contrast, correlation, energy and homogeneity.

Contrast is defined as the sum of square variances where u and v are the respective row and column value of the matrix and D is the normalised co-occurrence value in each GLCM element,

$$\text{Contrast} = \Sigma_{u,v}|u-v|^2 D(u,v) \tag{B.1}$$

a contrast value of 0 indicates a uniform image.

Correlation measures the linear dependency of grey levels between neighbouring pixels where μ_u and μ_v are the GLCM column and row mean values, and σ_u and σ_v are the GLCM column and row standard deviations. The correlation value indicates the quality of the linear relationship, where a value of 1 indicates a perfectly correlated image,

$$\text{Correlation} = \Sigma_{u,v} \frac{(u - \mu_u)(v - \mu_v)D(u, v)}{\sigma_u \sigma_v} \tag{B.2}$$

Energy is calculated as the sum of squared co-occurrence values, D. The energy value indicates the textural uniformity observed from the repetition of pairs of pixel in the image. An energy value of 1 indicates a constant image.

$$\text{Energy} = \Sigma_{u,v} D(u, v)^2 \tag{B.3}$$

Homogeneity is the inverse difference moment measuring the distribution closeness of elements in the GLCM to the matrix diagonal. The weighting of values decreases exponentially away from the diagonal of the matrix,

$$\text{Homogeneity} = \Sigma_{u,v} \frac{D(u, v)}{1 + |u - v|^2} \tag{B.4}$$

B.2 Local Binary Patterns

Local Binary Patterns (LBP) is a texture operator which labels the pixels of an image by thresholding the neighbourhood of each pixel with the value of the centre pixel [47]. LBP has been demonstrated to be robust towards monotonic grey scale changes caused by illumination variations [105][116], hence used in this thesis to provide texture features to classify surface-types in images with non-uniformly illuminated surfaces

Figure B.1 shows how the LBP operator is applied by comparing a pixel to its eight surrounding neighbour pixels and threshold labelling the comparison result as one if the

pixel is greater than the neighbour's value, and zero otherwise. The resulting set of labels for the neighbour pixels are interpreted as an 8-bit binary number that can be converted into a decimal value. The calculation of the decimal value for a pixel can be expressed as,

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^P \tag{B.5}$$

where

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{B.6}$$

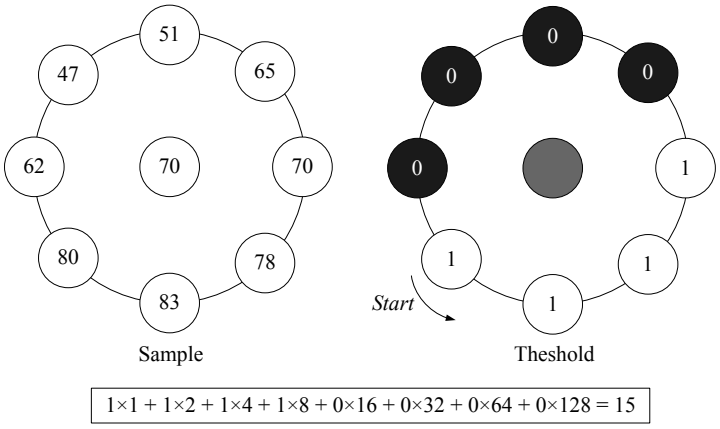


FIGURE B.1: The calculation of a decimal value for a pixel using the LBP operator

where g_c, is the centre pixel and g_p, are the neighbour pixels. For a sample window size (e.g 20 $\times$ 20 pixels) selected to extract texture features, the LBP decimal value is calculated for each pixel to compute a histogram showing the frequency of each LBP decimal value occurring (8-bit = 256 possible LBP decimal values). The histogram can be used as the feature vector to describe the texture in the sample window. In this thesis, a histogram is produced that contains the frequency of 59 LBP decimal values (uniform LBP, neighbours = 8, radius = 1, mapping = 8 [105]) and is used as the feature vector to classify surface-types.

Appendix C

Multi-class Surface-type Classifier

The following is a summary of the machine learning algorithms used to perform surface-type classification including the naive Bayes classifier and the Support Vector Machine (SVM) classifier. The naive Bayes classifier is selected to provide a simple probabilistic approach to classify large feature vectors and the SVM is used to demonstrate a powerful classification approach based on the Vapnik-Chervonenkis theory [34].

C.1 Naive Bayes Classifier

A naive Bayes approach assumes feature independence, and an initial uniform probability for all surface-type defined within the environment [117][118][119]. Given a particular surface-type, $M_k \epsilon M$ where M contains all the surface-types for a given environment, the probability that the surface-type is M_k is given by,

$$P(M_k|f) = \frac{P(f|M_k)P(M_k)}{P(f)} \tag{C.1}$$

The denominator in Equation (C.1) is invariant across all surface-types, and is considered as a normalisation parameter so Equation (C.1) can be rewritten as,

$$P(M_k|f) \propto P(f|M_k)P(M_k) \tag{C.2}$$

Feature independence is assumed for the surface-type features, f_n. Thus, the classifier can be expressed as,

$$P(M_k|f) \propto \prod_{t=1}^{n} P(f_t|M_k)P(M_k) \tag{C.3}$$

where $\mathrm{P}(f_t|M_k)$ is the conditional probability describing the probability of the surface-type feature, f_t given the surface-type, M_k. The class prior, $\mathrm{P}(M_k)$ is defined to be uniform across all the surface-type that exists in the environment.

The maximum *a posteriori* decision rule is applied for the selection of the most probable hypothesis as the candidate class,

$$m = max_m(P(M = m|f)) \tag{C.4}$$

Additionally, a reject option is applied to the candidate class to safeguard against misclassification if the labelling confidence of the most probable hypothesis is low.

C.2 Support Vector Machines

The support vector machine (SVM) is a popular classifier and has been applied in various image recognition tasks [13][120]. It has been shown to robustly perform (two-class) classification performance using a limited training dataset. However, a multi-class SVM classifier is complicated to formulate and can be intractable. To achieve multi-class classification with SVM, the approach of combining several binary SVM classifiers is used. Several approaches are available including; one-versus-all method using winner-takes-all strategy (WTA-SVM), one-versus-one method using max-wins voting (MWV-SVM), and error correcting output codes with pairwise coupling by combining posterior probabilities of individual SVMs PWC-PSVM. An empirical study [121] comparing the different approaches concludes that WTA-SVM and MWV-SVM are competitive with no clear superiority, while PWC-PSVM consistently outperforms these two methods. In this thesis, we have chosen to use an implementation of PWC-PSVM described in [122] to perform classification of different surface-types in an image.

Appendix D

Surface Preparation Guideline

D.1 Description

SSPC VIS2: Provides a method to evaluate the degree of rusting on painted steel (or iron) surfaces. Particularly useful for evaluating rust on test panels or on localized areas of structures. A scale and description of rust grades are given, and 27 full-colour photographs and the corresponding black-and-white rust images illustrating the maximum percentage of rusting allowed for each rust grade from rust grade 9 to rust grade 1 for three different rust distributions are included. Published in 2000 by SSPC SSPC VIS5: This guide contains a series of colour photographs that represent various conditions of unpainted steel surfaces prior to and after surface preparation by wet abrasive blast cleaning (WAB) (slurry blasting). SSPC Reference Photographs are intended to supplement the written standards. This guide can be used to clarify the acceptable level of flash rusting allowed prior to the application of protective coatings.

Technical Reference Section

Surface Preparation Grades Of Blast Cleaned Steel

Assessment of degree of removal of rust, mill scale and other visual contaminants (inspected without magnification)

IMPORTANT NOTE: Surface cleanliness is not a measure of surface profile – see the following pages for notes on surface profile

EXPLANATORY DETAILS

Rust Grade A
Steel with the millscale layer intact and very minor, or no rusting.

Rust Grade B
Steel with spreading surface rust, and the millscale commenced flaking.

Rust Grade C
Rusty Steel with the millscale layer flaked and loose, or lost, but only minor occurrence of pitting.

Rust Grade D
Very rusty steel with the millscale layer all rusted and extensive occurrence of pitting.

Blast Class 1 (SP-7/N4)
Very light, whip over blast clean with removal of loose surface contaminants.

Blast Class 2 (SP-6/N3)
Substantial blast clean with wide spread, visible contaminant removal and base metal colour appearing.

Blast Class 2 1/2 (SP-10/N2)
Intense blast clean leaving shaded grey metal with only tiny, isolated flecks or strips of visible contaminants.

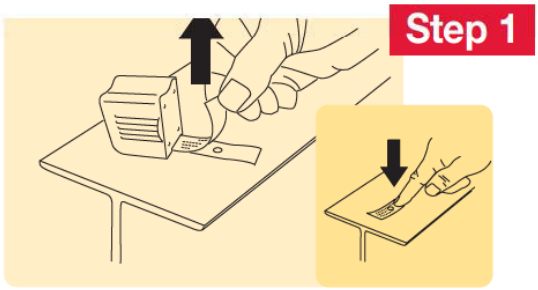
Blast Class 3 (SP-5/N1)
Complete blast clean with consistent, metal colour all over and no visible contaminants.

NOTE: All blasting preparation grades must be free of oil, grease and dirt.

DISCLAIMER: The grades of rust and blast cleaned surfaces 'information', described and illustrated on these pages are a guide only and do not claim to be approved nor complying nor substitutable for or by any surface preparation standard. It is responsibility of the reader and/or users of this 'information' to separately determine and verify the specifications and/or methods and/or assessment of surface preparation as indicated or directed or required by or in any work specifications or standards. Blastmaster expressly disclaims any liability for the use or misuse of this 'information'.

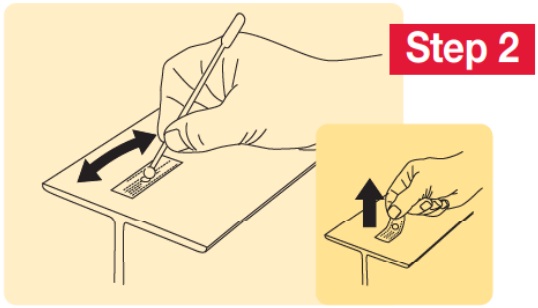
Measuring Surface Profile

USING THE REPLICA TAPE METHOD TO MEASURE AVERAGE MAXIMUM PEAK-TO-VALLEY PROFILE HEIGHT (Rz)



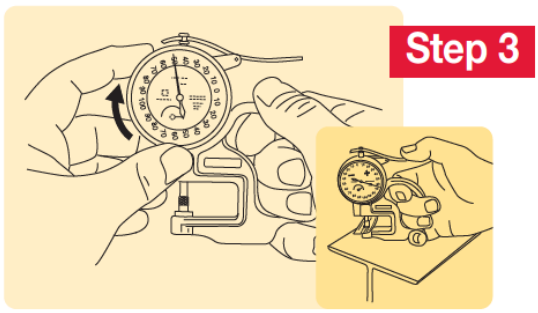
Test Preparation

- Select a representative test site free of dust, dirt and pitting.
- Choose the appropriate grade of Testex tape – refer to Inspection Instruments page 187 for details of the various scale measurement ranges.
- Peel a test tape strip from the roll – a 'bullseye' marker dot will remain on the slip paper.
- Apply the tape to the test surface – rub over the tape with a finger to ensure it is firmly adhered.



Perform the Test

- Using moderate to firm pressure, rub the test window with the round-tip burnishing tool - refer page 187.
- Take care not to dislodge the test tape (caused by bumping the tool against the edge of the circular cutout window).
- Burnish the test window until it has uniformly darkened – the colour indicates the profile has been impressed into the test tape.
- Peel the test tape strip from the surface.



Measuring the Test Result

- Use a dial thickness gauge with the correct specifications (ie accuracy, anvil spring pressure and anvil size) for replica tape – refer to Inspection Instruments page 187.
- Clean the anvils and check/adjust the zero point.
- After cleaning and checking the gauge zero point, adjust the dial to minus 50 microns (this compensates for the thickness of the tape carrier film and allows the profile measurement to be read directly from the gauge).
- Centre the test tape between the anvils, gently allow the anvils to close on the tape, and note the reading on the dial.
- Take several readings to establish accuracy. (Reposition the tape in the anvils between each reading.)

SOURCES OF ERROR

There are four major sources of error in determining the profile of a blast cleaned surface, which can be minimised with the following suggestions.

1. Inherent Profile Variation in the surface – perform at least 3 tests per 10 square metres of area.

2. Contaminant particles in the anvils or tape – select a clean surface; clean and check the anvils; examine the test tape; double check any questionable readings. To indicate the size significance of seemingly tiny contaminants, please note that human hair is approx. 50 microns thick.

3. Improper Gauge – a good gauge has an accuracy of ±5 micron, closing force of 1.5N and at least one anvil 0.25" (6.3mm) diameter.

4. Deficient Impressing Technique – use a profile training tool to verify the burnishing technique is correct. See page 187.

Appendix E

Confusion Matrices

This section presents classification result of images used in Chapter 4: Experiment 1, Chapter 6: Experiment 1 and Chapter 6: Experiment 2

E.1 Chapter 4: Experiment 1

The following forty-eight confusion matrices shows the classification result achieved for images of a surface-type when using the following feature vectors (RGB, a*b* and K_d).

The presented tables are clearly labelled with the surface-type of the image sample, the reference name used for the image, the feature vector used and the different classes of surface-types.

Painted surface: Training image				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	258583	3506	0	13272
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Training image				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	264938	0	7	10416
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Training image				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	269042	0	91	6228
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 1				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	214505	19964	1600	41529
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 1				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	243104	0	231	34263
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 1				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	269785	1	47	7765
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 2				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	178144	38199	10608	58386
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 2				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	222962	0	1069	61306
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 2				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	276510	3	0	8824
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 3				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	170465	43786	13228	59782
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 3				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	218499	0	1324	67438
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Painted surface: Image 3				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	278470	6	34	8751
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Training image				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	291208	3832	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Training image				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	285860	9180	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Training image				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	294601	435	4
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 1				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	273401	20322	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 1				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	258281	35442	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 1				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	293616	89	18
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 2				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	251351	44855	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 2				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	240729	55477	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 2				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	295899	307	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 3				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	246104	49871	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 3				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	1	240462	55510	2
Rusted	0	0	0	0
Blasted	0	0	0	0

Timber surface: Image 3				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	295945	30	0
Rusted	0	0	0	0
Blasted	0	0	0	0

Rusted surface: Training image				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	5	132741	156187	139
Blasted	0	0	0	0

Rusted surface: Training image				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	57	4	277514	11497
Blasted	0	0	0	0

Rusted surface: Training image				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	10	0	286125	2937
Blasted	0	0	0	0

Rusted surface: Image 1				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	154588	133592	71
Blasted	0	0	0	0

Rusted surface: Image 1				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	31	2	277746	10472
Blasted	0	0	0	0

Rusted surface: Image 1				
Confusion matrix: K				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	3	287060	1188
Blasted	0	0	0	0

Rusted surface: Image 2				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	1	203818	87879	71
Blasted	0	0	0	0

Rusted surface: Image 2				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	73	0	279776	11920
Blasted	0	0	0	0

Rusted surface: Image 2				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	48	91	288606	3024
Blasted	0	0	0	0

Rusted surface: Image 3				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	11	222623	42374	10
Blasted	0	0	0	0

Rusted surface: Image 3				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	75	1	254113	10829
Blasted	0	0	0	0

Rusted surface: Image 3				
Confusion matrix: K				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	76	97	256741	8104
Blasted	0	0	0	0

Blasted surface: Training image				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	28787	14179	22511	209028

Blasted surface: Training image				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	18140	0	3072	253293

Blasted surface: Training image				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	9113	0	727	264665

Blasted surface: Image 1				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	4840	11690	21731	231565

Blasted surface: Image 1				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	6017	0	5814	257995

Blasted surface: Image 1				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	4540	0	168	265118

Blasted surface: Image 2				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	777	11076	39619	207491

Blasted surface: Image 2				
Confusion matrix: a*b*				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	3608	0	7776	247579

Blasted surface: Image 2				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	5164	0	160	253639

Blasted surface: Image 3				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	3	36272	65031	131643

Blasted surface: Image 3				
Confusion matrix: RGB				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	3	36272	65031	131643

Blasted surface: Image 3				
Confusion matrix: K_d				
Surface-types	Painted	Timber	Rusted	Blasted
Painted	0	0	0	0
Timber	0	0	0	0
Rusted	0	0	0	0
Blasted	3243	0	567	229139

E.2 Chapter 6: Experiment 1

The following five confusion matrices shows the classification result achieved for each surface-type when using the following feature vectors ($\{K_d$,LBP$\}$, K_d, LBP, a*b* and RGB). Each confusion matrix contains classification result for all sample instances extracted from the 16 images used in this experiment.

Image 1 - 16				
Confusion matrix: Combined K_d,LBP				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	2762	0	431	354
Painted	51	4178	190	441
Rusted	52	58	2679	277
Cardboard	8411	309	30499	21053

Image 1 - 16				
Confusion matrix: K_d, colour-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	2764	0	350	433
Painted	2	4184	237	437
Rusted	22	69	2655	320
Cardboard	8753	358	30606	20555

Image 1 - 16				
Confusion matrix: LBP, texture-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	2291	0	757	499
Painted	1220	2361	251	1028
Rusted	1230	0	1706	130
Cardboard	37511	2794	3144	16823

Image 1 - 16				
Confusion matrix: a*b*, colour-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	706	2	2682	157
Painted	0	2501	2155	204
Rusted	0	19	2991	56
Cardboard	74	218	56419	3561

Image 1 - 16				
Confusion matrix: RGB, colour-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	624	2	2579	162
Painted	0	2486	2172	202
Rusted	0	32	2975	59
Cardboard	43	260	56751	3218

E.3 Chapter 6: Experiment 2

The following five confusion matrices shows the classification result achieved for each surface-type when using the following feature vectors ($\{K_d$,LBP$\}$, K_d, LBP, a*b* and RGB). Each confusion matrix contains classification result for all sample instances extracted from the 6 images used in this experiment.

Image 1 - 6				
Confusion matrix: Combined K_d,LBP				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	52	2	870	15
Painted	0	16	2797	2
Rusted	20	8	1659	14
Cardboard	426	233	26406	3218

Image 1 - 6				
Confusion matrix: K_d, colour-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	41	2	886	10
Painted	0	16	2799	0
Rusted	3	12	1668	18
Cardboard	353	236	26894	3493

Image 1 - 6				
Confusion matrix: LBP, texture-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	687	1	211	38
Painted	2424	167	77	147
Rusted	1286	15	373	27
Cardboard	18189	4225	2372	6190

Image 1 - 6				
Confusion matrix: a*b*, texture-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	0	0	937	0
Painted	0	12	2803	0
Rusted	0	2	1699	0
Cardboard	0	20	29909	1047

Image 1 - 6				
Confusion matrix: RGB, texture-based				
Surface-types	Timber	Painted	Rusted	Cardboard
Timber	0	0	937	0
Painted	0	5	2810	0
Rusted	0	4	1697	0
Cardboard	0	19	30107	850

Bibliography

[1] S. Se and P. Jasiobedzki. Photo-realistic 3D model reconstruction. *International Conference on Robotics and Automation*, pages 3076 – 3082, Orlando FL, 2006.

[2] K. Lai, L. Bo, X. Ren, and D. Fox. Detection-based Object Labeling in 3D Scenes. *International Conference on Robotics and Automation*, pages 1330 – 1337, Saint Paul, MN, 2012.

[3] Blastmaster. Catalogue: Equipment, Components and Technical references. 2006.

[4] J. Martinez-Alajarin, J. D. Luis-Delgado, and L. M. Tomas-Balibrea. Automatic system for quality-based classification of marble textures. *IEEE Transactions on Systems, Man, and Cybernetics*, 35:488 – 497, 2005.

[5] A. W. K. To, G. Paul, and D. K. Liu. Image segmentation for surface material-type classification using 3D geometry information. *International Conference on Information and Automation*, pages 1717 – 1722, Harbin, 2010.

[6] G. Paul, S. Webb, D. K. Liu, and G. Dissanayake. A Robotic System for Steel Bridge Maintenance: Field Testing. *Australasian Conference on Robotics and Automation*, pages 1 – 8, Brisbane, Australia, 2010.

[7] G. Paul, N. Kirchner, D. K. Liu, and G. Dissanayake. An effective exploration approach to simultaneous mapping and surface material-type identification of complex three-dimensional environments. *Journal of Field Robotics*, 26:915 – 933, 2009.

[8] L. Q. Zhu and Z. Zhang. Auto-classification of insect images based on color histogram and GLCM. *International Conference Fuzzy Systems and Knowledge Discovery*, pages 2589 – 2593, Yantai, Shandong, 2010.

[9] C. Unsalan and A. Ercil. Automated Inspection of Steel Structures. *Springer - Verlag Recent Advances in Mechatronics*, pages 468 – 480, 1999.

[10] R. M. Haralick, K. Shanmugam, and H. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:610 – 621, 1973.

[11] P. H. Chen, H. K. Shen, C. Y. Lei, and L. M. Chang. Support-vector-machine-based method for automated steel bridge rust assessment. *Automation in Construction*, 23:9 – 19, 2012.

[12] O. Wang, P. Gunawardane, S. Scher, and J. Davis. Material classification using BRDF slices. *International Conference on Computer Vision and Pattern Recognition*, pages 2805 – 2811, Miami, 2009.

[13] E. Momma, Y. Kimura, H. Ishii, T. Ono, M. Ayoja, and T. Higuchi. Rust classification using image analysis of steel structures. *International Conference on Sensing Technology*, pages 409 – 413, Tainan, Taiwan, 2008.

[14] T. Mitsunaga and S. K. Nayar. Radiometric self calibration. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, Fort Collins, CO, 1999.

[15] M. Spong and M. Vidyasagar. *Robot Dynamic and Control*. John Wiley & Sons, Inc., 1st edition, 1989.

[16] Transport Roads & Maritime Services. Road Maintenance Contestability Reform Program: Industry Briefing. 2012.

[17] C. H. Hare. Protective coatings for bridge steel. *Technical Report 0-309-04421-9, Transportation Research Board*, 1987.

[18] N. Kirchner and G. Paul. Bridge maintenance robotic arm: mechanical technique to reduce the nozzle force of a sandblasting rig. *Journal of Wuhan University of Technology*, pages 12 – 18, 2006.

[19] D. K. Liu, G. Dissanayake, P. B. Manamperi, G. Fang, N. Kirchner, G. Paul, S. Webb, P. Chotiprayanakul, and J. Xie. A Robotic System for Steel Bridge Maintenance: Research Challenges and System Design. *In Proceedings of the Australasian Conference on Robotics and Automation*, 2008, Canberra, Australia.

[20] G. Paul. *Autonomous exploration and mapping of complex 3D environments by means of a 6DOF manipulator*. PhD thesis, University of Technology, Sydney, 2010.

[21] M. Golfarelli, D. Maio, and S. Rizzi. Correction of dead-reckoning errors in map building for mobile robots. *IEEE Transactions on Robotics and Automation*, 17(1): 37 – 47, 2001.

[22] S. Ogawa, K. Watanabe, and K. Kobayashi. 2D mapping of a closed area by a range sensor. *In Proceedings of the 41st SICE Annual Conference*, 2:1329 – 1333, 2002.

[23] F. Lu and E. Milios. Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans. *Computer Vision and Pattern Recognition*, pages 935 – 938, Seattle, WA, 1994.

[24] B. Pitzer, S. Kammel, C. DuHadway, and J. Becker. Automatic reconstruction of textured 3D models. *International Conference on Robotics and Automation*, pages 3486 – 3493, Anchorage, AK, 2010.

[25] I. Kim, D. Kim, Y. Cha, K. Lee, and T. Kuc. An embodiment of stereo vision system for mobile robot for real-time measuring distance and object tracking. *International Conference on Control, Automation and Systems*, pages 1029 – 1033, Seoul, 2007.

[26] J. Oh and C. Lee. Development of a stereo vision system for industrial robots. *International Conference on Control, Automation and Systems*, pages 659 – 663, Seoul, 2007.

[27] M. V. Rohith, G. Somanath, D. H. Norris, J. J. Gutierrez, and C. Kambhamettu. A camera flash based projector system for true scale metric reconstruction. *Workshop on Applications of Computer Vision*, pages 1 – 8, Snowbird, UT, 2009.

[28] J. Aue, D. Langer, B. Muller-Bessler, and B. Huhnke. Efficient segmentation of 3D LIDAR point clouds handling partial occlusion. *IEEE Intelligent Vehicles Symposium*, pages 423 – 428, Baden, 2011.

[29] D. Neumann, F. Lugauer, S. Bauer, J. Wasza, and J. Hornegger. Real-time RGB-D mapping and 3-D modeling on the GPU using the random ball cover data structure. *International Conference Computer Vision Workshops*, pages 1161 – 1167, Barcelona, 2011.

[30] L. Sabattini, A. Levratti, F. Venturi, E. Amplo, C. Fantuzzi, and C. Secchi. Experimental comparison of 3D vision sensors for mobile robot localization for industrial application: Stereo-camera and RGB-D sensor. *International conference Control Automation Robotics Vision*, pages 823 – 828, Guangzhou, 2012.

[31] A. Ranganathan and J. Lim. Visual place categorization in maps. *International Conference Intelligent Robots and Systems*, pages 3982 – 3989, San Francisco, CA, 2011.

[32] J. Park, M. H. Baeg, J. Koh, K. Park, and M. Baeg. A new object recognition system for service robots in the smart environment. *International Conference Control, Automation and Systems*, pages 1083 – 1087, Seoul, 2007.

[33] D.G. Lowe. Object recognition from local scale-invariant features. *International Conference on Computer Vision*, 2:1150 – 1157, Kerkyra, 1999.

[34] A. Oliva and A. Torralba. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *International Journal of Computer Vision*, 42:145 – 175, 2001.

[35] C. Gabriella, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. *Workshop on Statistical Learning in Computer Vision*, pages 1 – 22, 2004.

[36] M. Yazdchi, M. Yazdi, and A. G. Mahyari. Steel Surface Defect Detection Using Texture Segmentation Based on Multifractal Dimension. *International Conference on Digital Image Processing*, pages 346 – 350, Bangkok, 2009.

[37] H. Hu, Q. Gu, and J. Zhou. HTF: a novel feature for general crack detection. *International Conference on Image Processing*, pages 1633 – 1636, Hong Kong, 2010.

[38] H. Chen, W. Sheng, N. Xi, M. Song, and Y. Chen. Automated robot trajectory planning for spray painting of free-form surfaces in automotive manufacturing. *International Conference on Robotics and Automation*, 1:450 – 455, 2002.

[39] A. McCrea, D. Chamberlain, and R. Navon. Automated inspection and restoration of steel bridges–a critical review of methods and enabling technologies. *Automation in Construction*, 11(4):351 – 373, 2002.

[40] E. Wu, Y. Ke, and B. Du. Noncontact Laser Inspection Based on a PSD for the Inner Surface of Minidiameter Pipes. *IEEE Transactions on Instrumentation and Measurement*, 58(7):2169 – 2173, 2009.

[41] N. Kirchner, D. K. Liu, and G. Dissanayake. Surface Type Classification With a Laser Range Finder. *Journal Sensors*, 9:1160 – 1168, 2009.

[42] T. Bo, J. Kong, X. Wang, and C. Li. Surface Inspection System of Steel Strip Based on Machine Vision. *First International Workshop on Database Technology and Applications*, pages 359 – 362, Wuhan, Hubei, 2009.

[43] L. A. O. Martins, L. C. Padua, and P. E. M. Almeida. Automatic detection of surface defects on rolled steel using Computer Vision and Artificial Neural Networks. *Conference on IEEE Industrial Electronics Society*, pages 1081 – 1086, Glendale, AZ, 2010.

[44] S. H. Hashemi, M. A. Moradpour, E. Karimi, and M. Kazemi. The use of machine vision inspection for out of roundness measurement in steel pipes. *Conference on Pattern Recognition and Image Analysis*, pages 1 – 5, Birjand, 2013.

[45] X. and Y. Chaolin. On-line defect detection algorithms for surface inspection of hot rolled strips. *International Conference on Mechanic Automation and Control Engineering*, pages 2350 – 2353, Wuhan, 2010.

[46] Y. J. Liu, J. Kong, X. Wang, and F. Jiang. Research on image acquisition of automatic surface vision inspection systems for steel sheet. *International Conference on Advanced Computer Theory and Engineering*, 6:189 – 192, 2010.

[47] T. Ojala, M. Pietikinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51 – 59, 1996.

[48] Y. He and N. Sang. Robust Illumination Invariant Texture Classification Using Gradient Local Binary Patterns. *International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping*, pages 1 – 6, Xiamen, 2011.

[49] E. Hayman, B. Caputo, M. Fritze, and J. O. Eklundh. On the significance of real-world conditions for material classification. *In Proceedings of the European Conference on Computer Vision*, pages 253 – 266, Czech Republic, 2004.

[50] O. G. Cula and K. J. Dana. 3D texture recognition using bidirectional feature histograms. *International Journal of Computer Vision*, 1:33 – 60, 2004.

[51] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 8:1265 – 1278, 2001.

[52] R. E. Broadhurst. Statisical estimation of histogram variation for texture classification. *International Workshop on Texture Analysis and Synethis*, pages 25 – 30, Beijing, China, 2005.

[53] G. Finlayson, M. S. Drew, and C. Lu. Intrinsic Images by Entropy Minimization. *Computer Vision - ECCV*, 3023:582 – 595, 2004.

[54] M. Varma and A. Zisserman. A Statistical Approach to Material Classification Using Image Patch Exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:2032 – 2047, 2009.

[55] O. Drbohlav and M. J. Chantler. Illumination-invariant texture classification using single training images. *International Workshop on Texture Analysis and Synthesis*, pages 31 – 36, 2005.

[56] X. Llado, M. Petrou, and J. Marti. Surface texture recognition by surface rendering. *Optical Engineering*, 44, 2005.

[57] P. H. Chen, Y. C. Yang, and L. M. Chang. Automated bridge coating defect recognition using adaptive ellipse approach. *Automation in Construction*, 18(5): 632 – 643, 2009.

[58] P. H. Chen, H. K. Shen, C. Y. Lei, and L. M. Chang. Fourier-Transform-Based Method for Automated Steel Bridge Coating Defect Recognition. *The Proceedings of the Twelfth East Asia-Pacific Conference on Structural Engineering and Construction*, 14:470 – 476, 2011.

[59] P. H. Chen, Y. C. Yang, and L. M. Chang. Illumination adjustment for bridge coating images using BEMD-Morphology Approach (BMA). *Automation in Construction*, 19(4):475 – 484, 2010.

[60] M. Bryson, M. J. Roberson, O. Pizarro, and S. B. Williams. Colour-Consistent Structure-from-Motion Models using Underwater Imagery. *Robotics: Science and Systems*, pages 1 – 8, Sydney, Australia, 2012.

[61] A. Yamashita, M. Fujii, and T. Kaneko. Color Registration of Underwater Images for Underwater Sensing with Consideration of Light Attenuation. *International Conference on Robotics and Automation*, pages 4570 – 4575, Roma, Italy, 2007.

[62] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, and T. Hawkins. A lighting reproduction approach to live-action compositing. *ACM Transactions on Graphics*, 21(3):547 – 556, 2002.

[63] M. Louw and F. Nicolls. Surface Classification Via BRDF Parameters using Population Monte Carlo for MRF Parameter Estimation. *International Conference on Computer Graphics and Imaging*, pages 145 – 154, Innsbruck, Austria, 2010.

[64] J. Gu and C. Liu. Discriminative illumination: Per-pixel classification of raw materials based on optimal projections of spectral BRDF. *Conference on Computer Vision and Pattern Recognition*, pages 797 – 804, Providence, RI, 2012.

[65] J. Jachnik, R.A. Newcombe, and A.J. Davison. Real-time surface light-field capture for augmentation of planar specular surfaces. *International Symposium Mixed and Augmented Reality*, pages 91 – 97, Alanta, GA, 2012.

[66] A. Hartl, J. Grubert, D. Schmalstieg, and G. Reitmayr. Mobile interactive hologram verification. *International Symposium on Mixed and Augmented Reality*, pages 75 – 82, 2013.

[67] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786 – 804, 1979.

[68] R. R. Todd and J. M. Hans. A Review of Recent Texture Segmentation and Feature Extraction Techniques. *Journal-CVGIP-Image Understanding*, 57(3):359 – 372, 1993.

[69] T. Mihran and A. K. Jain. Handbook of pattern recognition and computer vision. *World Scientific Publishing Co., Inc.*, pages 235 – 276, 1993.

[70] C. Unsalan. Pattern recognition methods for texture analysis case study: steel surface classification. Master's thesis, Bogazii University, 1995.

[71] J. S. Weszka, C. R. Dyer, and A. Rosenfeld. A Comparative Study of Texture Measures for Terrain Classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(4):269 – 285, April 1976.

[72] M. Tuceryan and A. K. Jain. Texture Segmentation Using Voronoi Polygons. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:211 – 216, 1989.

[73] B. Schachter, A. Rosenfeld, and L. S. Davis. Random mosaic models for textures. *IEEE Transactions on Systems, Man, and Cybernetics*, 8:694 – 702, 1978.

[74] S. Mallat. A Theory for Multiresolution Signal Decomposition: the Wavelet Representation. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 11: 674 – 693, 1989.

[75] M. R. Turner. Texture discrimination by Gabor functions. *Biological Cybernetics*, 55(2-3):71 – 82, 1986.

[76] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* Henry Holt and Co., Inc., New York, NY, USA, 1982.

[77] L. Jacobson and H. Wechsler. The Wigner Distribution as a Tool for Deriving an Invariant Representation of 2-D Images. *Pattern Recognition and Image Processing*, pages 218 – 220, Las Vegas, Nevada, 1982.

[78] C. Song, F. Yang, and P. Li. Rotation Invariant Texture Measured by Local Binary Pattern for Remote Sensing Image Classification. *International Workshop on Education Technology and Computer Science*, 3:3 – 6, 2010.

[79] A. P. Bradley, P. T. Jackway, and B. C. Lovell. Scale-Space Texture Analysis. *Digital Image Computing: Techniques and Applications*, pages 68 – 73, 1995.

[80] S. Yan, S. Sayad, and S. T. Balke. Image quality in image classification: Adaptive image quality modification with adaptive classification. *Computers and Chemical Engineering*, 33(2):429 – 435, 2009.

[81] S. Yan, S. Sayad, and S. T. Balke. Image quality in image classification: Design and construction of an image quality database. *Computers and Chemical Engineering*, 33:421 – 428, 2009.

[82] F. Xuan, Z. H. Wei, L. Xiao, and T. Zhan. Image quality assessment based on geometrical structural orientation. *International Conference on Control and Electronic Engineering*, pages 43 – 46, Changchun, 2010.

[83] Y. Yi, X. Yu, L. Wang, and Z.Yang. Image Quality Assessment Based on Structural Distortion and Image Definition. *International Conference Computer Science and Software Engineering*, 6:253 – 256, Wuhan, Hubei, 2008.

[84] A. W. K. To, G. Paul, D. Rushton-Smith, D. K. Liu, and G. Dissanayake. Automated and Frequent Calibration of a Robot Manipulator-mounted IR Range Camera for

Steel Bridge Maintenance. *International Conference on Field and Service Robotics*, pages 1 – 14, Matsushima, Japan, 2012.

[85] R. M. Haralick and L. G. Shapiro. Computer and Robot Vision. *Addison-Wesley Longman Publishing Co., Inc.*, 1991.

[86] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. *In Proceedings of the International Conference on Computer Vision*, 1:666 – 673, 1999.

[87] G. Paul, D. K. Liu, and N. Kirchner. An Algorithm for Surface Growing from Laser Scan Generated Point Clouds. *Robotic Welding, Intelligence and Automation, Springer-Verlag*, 362:481 – 491, 2007.

[88] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46 – 57, 1989.

[89] A. Hornung, K. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: an efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 34:189 – 206, 2013.

[90] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard. An evaluation of the RGB-D SLAM system. *International Conference on Robotics and Automation*, pages 1691 – 1696, Saint Paul, MN, 2012.

[91] K. E. Torrance and E. M. Sparrow. Theory of off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, pages 1105 – 1114, 1967.

[92] A. S. Georghiades. Incorporating the Torrance and Sparrow model of reflectance in uncalibrated photometric stereo. *International Conference on Computer Vision*, 2: 816 – 823, Nice, France, 2003.

[93] S. J. Kim and M. Pollefeys. Robust Radiometric Calibration and Vignetting Correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (4):562 – 576, 2008.

[94] S. Lin, J. Gu, S. Yamazaki, and H. Y. Shum. Radiometric calibration from a single image. *Computer Vision and Pattern Recognition*, 2:938 – 945, 2004.

[95] K. Shafique and M. Shah. Estimation of the radiometric response functions of a color camera from differently illuminated images. *International Conference on Image Processing*, 4:2339 – 2342, 2004.

[96] K. Hara, K. Nishino, and K. lkeuchi. Light source position and reflectance estimation from a single view without the distant illumination assumption. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):493 – 505, 2005.

[97] S. Lee, L. M. Chang, and M. Skibniewski. Automated recognition of surface defects using digital color image processing. *Automation in Construction*, 15(4):540 – 549, 2006.

[98] R. C. Gonzalez and R. E. Woods. Digital Image Processing. *2nd Ed. Prentice Hall*, 2001.

[99] A. P. Pentland. A new sense for depth of field. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 9(4):523 – 531, 1987.

[100] H. Hu and G. D. Hann. Low Cost Robust Blur Estimator. *International Conference on Image Processing*, pages 617 – 620, 2006.

[101] D. Chen, D. A. Stow, and P. Gong. Examining the effect of spatial resolution and texture window size on classification accuracy: an urban environment case. *International Journal of Remote Sensing*, 25:2177 – 2192, 2004.

[102] A. W. Paeth. Graphics gems: A Fast Algorithm for General Raster Rotation. *Academic Press Professional, Inc.*, pages 179 – 195, 1990.

[103] F. Kahl, S. Agarwal, M. Chandraker, D. Kriegman, and S. Belongie. Practical Global Optimization for Multiview Geometry. *International Journal on Computer Vision*, 79:271 – 284, 2008.

[104] H. L. Capitaine and C. Frlicot. Classification with reject options in a logical framework: a fuzzy residual implication approach. *European Society of Fuzzy Logic and Technology Conference*, pages 855 – 860, Lisbon, Portugal, 2009.

[105] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971 – 987, 2002.

[106] N. P. Doshi and G. Schaefer. Rotation-invariant local binary pattern texture classification. *ELMAR*, pages 71 – 74, Zadar, 2012.

[107] R. R. Picard and R. D. Cook. Cross-Validation of Regression Models. *Journal of the American Statistical Association*, 79:575 – 583, 1984.

[108] K. Ikeuchi and K. Sato. Determining reflectance properties of an object using range and brightness images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(11):1139 – 1153, 1991.

[109] J. Ji, L. Sun, and L. Yu. A New Pose Measuring and Kinematics Calibrating Method for Manipulators. *International Conference on Robotics and Automation*, pages 4925 – 4930, Roma, 2007.

[110] A. Omodei, G. Legnani, and R. Adamini. Three methodologies for the calibration of industrial manipulators: Experimental results on a SCARA robot. *Journal of Robotic Systems*, 17(6):291 – 307, 2000.

[111] H. Malm and A. Heyden. Simplified intrinsic camera calibration and hand-eye calibration for robot vision. *International Conference on Intelligent Robots and Systems*, 1:1037 – 1043, 2003.

[112] M. Dinham and G. Fang. A low cost hand-eye calibration method for arc welding robots. *International Conference on Robotics and Biomimetics*, pages 1889 – 1893, 2009.

[113] W. Gander, G. H. Golub, and R. Strebel. Least Squares Fitting of Circles and Ellipses. *BIT Numerical Mathematics, Springer*, pages 558 – 578, 1994.

[114] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-Squares Fitting of Two 3-D Point Sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 698 – 700, 1987.

[115] J. Y. Tou, Y. H. Tay, and P. Y. Lau. A Comparative Study for Texture Classification Techniques on Wood Species Recognition Problem. *International Conference on Natural Computation*, 5:8 – 12, 2009.

[116] T. Ojala, M. Pietikinen, and T. Menp. A generalized Local Binary Pattern operator for multiresolution gray scale and rotation invariant texture classification. *Advances in Pattern Recognition*, pages 397 – 406, 2001.

[117] B. Ren and L. Cheng. Research of Classification System Based on Naive Bayes and MetaClass. *International Conference on Information and Computing Science*, pages 154 – 156, 2009.

[118] Z. Qin. Naive Bayes Classification Given Probability Estimation Trees. *International Conference on Machine Learning and Applications*, pages 34 – 42, 2006.

[119] H. J. Huang and C. N. Hsu. Bayesian classification for data from the same unknown class. *IEEE Transactions on Systems Man, and Cybernetics*, 32:137 – 145, 2002.

[120] J. Dong, Y. Duan, and Z. Yang. Three-Dimensional Surface Texture Classification Based on Support Vector Machines and Wavelet Packets. *International Symposium on Intelligent Information Technology Application*, 3:124 – 127, 2008.

[121] K. B. Duan and S. S. Keerthi. Which is the best multiclass SVM method? An empirical study. *International Workshop on Multiple Classifier Systems*, pages 278 – 285, 2005.

[122] T. K. Huang, R. C. Weng, and C. J. Lin. Generalized Bradley-Terry Models and Multi-Class Probability Estimates. *Journal of Machine Learning Research*, 7:85 – 115, 2006.