

PLACE RECOGNITION SYSTEM FOR LOCALIZATION OF MOBILE ROBOTS

Thesis

Submitted in Partial Fulfillment of the Requirements Of
BITS F421T: THESIS

BY

Raghavender Sahdev

(2011A7TS257H)

B.E. (Hons) Computer Science Engineering

Under the supervision of

Dr. John K. Tsotsos (Professor), York University, Canada

&

Dr. Aruna Malapati (Assistant Professor), BITS Pilani, India

Submitted in Partial Fulfillment of the Requirements Of

BITS F421T: THESIS



BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI HYDERABAD
CAMPUS

(April 2015)

A Project Report
On
**PLACE RECOGNITION SYSTEM FOR LOCALIZATION OF MOBILE
ROBOTS**

BY
RAGHAVENDER SAHDEV

2011A7TS257H
B.E. (Hons.) Computer Science Engineering

Under the supervision of
Professor JOHN K. TSOTSOS

and Co-Supervision of

Dr. Aruna Malapati

Computer Science Department

SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS OF

BITS F421: THESIS PROJECT



York University

(Toronto, Canada)

At the Department of Electrical Engineering and Computer Science

(April 2015)

ACKNOWLEDGEMENT

I would like express my gratitude towards Professor Tsotsos for providing me not only with the opportunity of carrying out my bachelors' thesis under his supervision, but also providing me with everything that I required throughout the course of my thesis. I will forever be thankful to him for his guidance, kindness and support provided during the thesis. It gives me great pride to have completed my bachelors' thesis under his supervision. I would also like to thank Dr. Aruna Malapati for being my co supervisor for the thesis. I am grateful to her for all the support she has provided me for carrying out my thesis in Canada.

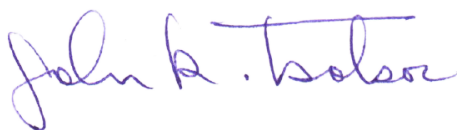
I express my warm thanks to Mr. Amir Rasouli for helping me with any technical problem that I faced. I would also like to thank Ms. Yulia Kotseruba, Mr. Calden Wloka and Mr. Asheer Bachoo for providing me with valuable insights at times when I was stuck.

Additionally I take this opportunity to thank all the members of the lab for proving a friendly atmosphere which made the completion of the thesis way more fun and made my stay at York University, Canada a memorable one.

My special thanks go to my parents and brother for always supporting me in my studies in a foreign country and providing me with all the unconditional love and support during the course of my stay.

CERTIFICATE

This is to certify that the Thesis entitled, Place Recognition System for Localization of Mobile Robots and submitted by – Raghavender Sahdev ID No. 2011A7TS257H in partial fulfilment of the requirement of BITS F421T Thesis embodies the work done by him under my supervision



Signature of the Supervisor

Name **John K. Tsotsos**

Designation **Professor**

Date: **April 30, 2015**

Signature of the Co- Supervisor

Name

Designation

Date:

LIST OF SYMBOLS AND ABBREVIATIONS USED

Symbols used

- Θ - theta
- λ - lambda
- φ - psi is the phase offset
- Υ - gamma is the spatial aspect ratio
- Σ, σ - sigma

Abbreviations used

- HOUP – Histogram of Oriented Uniform Patterns
- LBP – Local Binary Patterns
- SVM – Support Vector Machines
- KNN – K Nearest Neighbour
- VME – Virtual ME
- SIFT – Scale Invariant Feature Transform
- SURF – Speeded up Robust Features
- ROS – Robot Operating System
- OSS – One Shot Similarity
- LDA – Linear Discriminant Analysis

THESIS ABSTRACT

Thesis Title: Place Recognition System for Localization of Mobile Robots

Supervisor: Professor John K. Tsotsos, York University, Canada

Co-Supervisor: Dr. Aruna Malapati, BITS Pilani, Hyderabad Campus, India

Semester: Second

Name of Student: Raghavender Sahdev ID No. 2011A7TS257H

This thesis focuses on development of a Place Recognition and Categorization system for a mobile Robot. This work is motivated by the paper on Histogram of Oriented Gradients by Fazl Ersi and Tsotsos, 2012. The Robot learns places from experience and then recognizes previously observed topological places in known environments and categorizes previously unseen places in new environments. This system has been practically tested with a novel stereo dataset that has been developed to validate the theoretical results of the proposed system. A HOUP descriptor has been developed which is used to represent an image and then appropriate classifiers have been used to perform the classification tasks. It is shown in the report that our developed system not only performs well on the existing datasets but also performs remarkably well on the dataset that has been developed by us.

CONTENTS

1. Introduction.....	1
2. Relevant Prior Work.....	1
3. HOUP descriptor.....	3
3.1. Oriented Band Pass Filter.....	3
3.2. Gabor Filter.....	4
3.3. Local Binary Patterns.....	5
3.3.1. Gray Scale and Rotation Invariance.....	7
3.3.1.1. Achieving Gray Scale Invariance.....	7
3.3.1.2. Achieving Rotation Invariance.....	8
3.4. HOUP descriptors Comparison.....	12
4. Sub Division and Feature Selection.....	13
5. Scene Representation.....	14
6. Experiments.....	15
6.1. The UIUC Dataset.....	15
6.2. KTH Dataset	16
7. Our Dataset.....	19
7.1. Experimental Setup.....	19
7.1.1. Experimental Scenario.....	20
7.1.2. Robot Platform.....	23
7.2. Experimental Results.....	26
7.2.1. Same Robot, Same Lighting Conditions.....	26
7.2.2. Same Robot, Different Lighting Conditions.	27
7.2.3. Different Robot, Same Lighting Conditions..	28
7.2.4. Different Robot, Different Lighting Conditions.....	28
8. Programming Platforms.....	29

A. Conclusion.....	30
B. References.....	31

1. INTRODUCTION

Autonomous Mobile Robots have been studied by a large number of researchers. One of the most important capabilities is Robot Localization. Robot Localization refers to answering the question for the robot “Where am I?” Localization in general has 2 aspects *qualitative* and *quantitative*. The qualitative aspect of Localization refers to knowing where the robot is qualitatively. For example - In a building the robot should know that it is on a particular floor in room number 12 (which may be a seminar room, lab, kitchen, etc.) The Quantitative aspect of Localization allows the robot to have the knowledge about its coordinates in the particular room with reference to a standard point.

In this report our focus is to deal with the qualitative aspect of Localization. We focus on Topological Place Recognition and Topological Place Categorization. Topological Place Recognition gives the robot the ability to recognize previously seen places/environments and classify them into their respective class whereas Topological Place Categorization allows the robot to learn from a specified set of places and recognize previously unseen environments and places. We here used vision as a tool to solve this task. We have implemented a HOUP descriptor [1] in this report and use it as a tool to generate descriptors required to perform the recognition and categorization tasks. For a given image sub block, a HOUP descriptor is produced by passing the sub block through a Gabor filter oriented in different orientations. The output of the Gabor filter is then used to generate Local Binary Patterns similar to those used in ones proposed by Ojala [2]. These patterns reflect the textural features in the image (curved edges, flat regions, dark spots, bright spots, etc.). After generating the various features for a number of image sub blocks, feature selection is performed based on the notion of a Kernel Alignment Technique [3] Christianini. The most informative features are selected from the pool of features and taken into consideration for the training dataset. A similarity measure for the HOUP descriptors has been developed based on the One Shot Similarity (OSS) Kernel. [4] This ensures robustness against perceptual aliasing. Perceptual aliasing refers to the occurrence of visually similar image sub blocks in multiple classes of images. For example a patch of sky could be present in multiple classes (suburb, mountain, coast, highway, etc.).

2. RELEVANT PRIOR WORK

Most of the work done by early researchers on place categorization focused on the use of laser range finders to perceive the environment. Such methods make a rough estimate of the geometric layout of the surrounding. Examples of this work include that of Mozoz et al. (2005) [17] where they address the problem of classifying places in the environment of a mobile robot into semantic categories. Their approach uses the AdaBoost algorithm which trains a set of classifiers for place recognition based on laser range data. They apply their approach to distinguish between rooms, corridors, and hallways. Another notable work is that of Zender et al. [18] wherein they create conceptual representations of human-made indoor environments using mobile robots. The concepts described by Zender refer to spatial and functional properties of typical indoor environments. His model is based on composition of layers representing maps at different levels of abstraction. The complete system is integrated in a mobile robot endowed with laser and vision sensors for place and objects recognition. Their system also incorporates a linguistic framework that actively supports the acquisition process, and which is used for situated dialogue.

Although the use of laser scans has been widely used in the past and has displayed high performance, it has some disadvantages. It is often restricted to recognizing specific type of places with similar geometric structure. However if such a recognition system based on laser range scans is asked to distinguish between places with similar geometric structure with a different appearance, it fails. This problem lead to the development of information rich sensors like ‘*vision*’ (cameras). Vision very effectively handles this problem. Rottmann et. al. (2005) [19] proposed a method which combines laser range features and visual features to enable the robot to support a great variety of place categories. These features were used in a supervised leaning approach to label different locations using boosting. A Hidden Markov Model was applied to increase the robustness of the final classification. Other examples using vision sensors include the global scene recognition method of Olivia and Torralba (2001), which uses the Discrete Fourier Transform to encode spectral information of the image. The spectral signals from the non over-lapping sub-blocks are then compressed to produce the image representation. It is basically a low dimensional representation of the scene that is termed as the spatial envelope often referred to as the gist of an image [16]. The work was further extended by Torralba et al. (2003) by using wavelet based image decomposition instead of Discrete Fourier Transform to produce more compact and precise image representations. They evaluated the performance of their scene representation method for recognizing place categories collected by a mobile system, and reported reasonable accuracy in recognizing place categories such as “Conference Room”, “Corridor” and “Office”, which have a lower range of intra class variations.

Other works used for global scene classification referred to as landmark based approaches suggests using local image features to represent and classify the scenes. Local features characterize limited areas of the image and they often provide more robustness against common image variations. One of the most famous descriptors being used for describing the local features in an image is the Scale Invariant Feature Transform (SIFT) of Lowe (2004) [9]. It has dominated the field in the last 10 years and has been widely used by researchers throughout the world. Lazebnik et al. (2006) describes a method for recognizing scene categories based on approximate global geometric correspondence. It works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region. The resulting “spatial pyramid” is a simple and computationally efficient extension of an orderless bag of features image representation, and it shows significantly improved performance on challenging scene categorization tasks. The spatial pyramid framework also offers insights into the success of several recently proposed image descriptors, including Torralba’s ‘*gist*’ and Lowe’s SIFT descriptors.

Apart from the ones mentioned above, various other descriptors have been proposed, the Speeded up robust features (SURF) have been used to form descriptors for an image by Bay H et al.[10]. The SURF descriptor is partly inspired by SIFT. The standard version of SURF is several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT. SURF is based on sums of 2D Haar wavelet responses and makes an efficient use of integral images. Dallal and Triggs proposed the famous Histogram of Oriented Gradients (HOG descriptors) for detection of pedestrians [11]. Another famous work is that of Viola and Jones [13] which has been widely used for face detection.

This work focuses on re-implementing the paper of Fazl-Ersi and Tsotsos [1] on Histogram of Oriented Uniform Patterns for robust Place Recognition and Categorization.

3. HOUP descriptor

Histogram of Oriented Uniform Patterns (HOUP) is a distribution based descriptor as suggested by the name itself. The initial image representation that is used to build the histogram describes the frequency content of the image; it can also be viewed as a descriptor based on spatial frequency. The following figure gives a general overview of the process of generating a HOUP descriptor for an image-

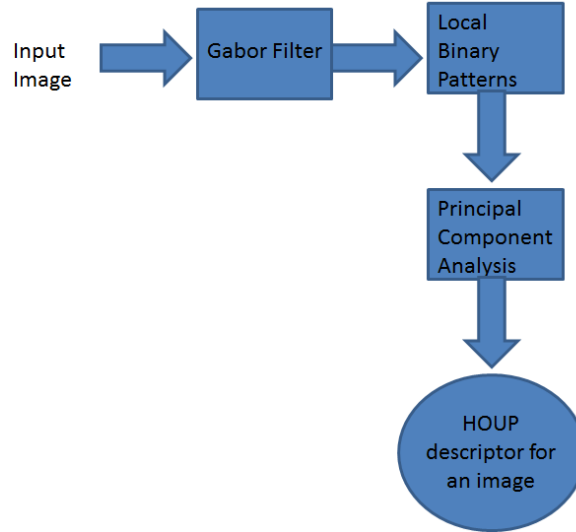


Figure 1. Initially the image is convolved with a gabor filter, then LBPs are generated for each orientation ($59 \times 6 = 354$), PCA then reduces the dimensionality to 70 dimensions. The result is a 70 dimensional HOUP descriptor for an image / image region.

The following sections present the algorithm used for generating the HOUP descriptors.

3.1 Oriented Band Pass filters

Research has provided empirical evidence that the human visual system uses oriented band pass filters in its vision system to observe scenes. We use this motivation to produce the intermediate stage in generation of the HOUP descriptors. The responses of oriented band pass filters have been useful in numerous computer vision applications such as texture analysis, edge detection, image data compression, motion analysis and image recognition. Among different oriented filters, Gabor Filters have received considerable attention, due to the ability to approximate certain cells present in the visual cortex of mammals. It has also been shown that these filters possess optimal localization properties in both spatial and frequency domain, and thus are well suited for texture analysis and encoding. Related work has been done by Torralba (2002) for encoding images for developing a context based Vision System for place and Object Recognition [12]. In this work too, we use a similar method of initially encoding the image by passing it through a Gabor filter.

3.2 The Gabor Filter

Gabor Filters are widely used in the field of Image Processing and Computer Vision for texture analysis, feature extraction, disparity estimation, etc. These filters are special types of filters which only allow a certain band of frequencies to pass through and reject the others. The filter can be mathematically represented as:

$$g(x, y; \lambda, \theta, \varphi, \sigma, \gamma) = \exp \left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2} \right) \exp \left(i \left(2\pi \frac{x'}{\lambda} + \varphi \right) \right) \quad (1)$$

where

$$x' = x \cos \theta + y \sin \theta$$

and

$$y' = -x \sin \theta + y \cos \theta$$

(2)

Where

- θ , theta is the orientation of the normal to the parallel stripes of a Gabor function.
- λ , lambda represents the wavelength of the sinusoidal factors
- φ , phi is the phase offset
- γ , gamma is the spatial aspect ratio
- σ , sigma is the standard deviation of the Gaussian envelope
- x and y are the coordinates of the pixels in the image

Important Parameters of the Gabor Filter

Theta – This parameter is the most important one, it decides what kind of features the filter responds to. Example given theta=0, implies the filter is responsive to horizontal features only. In this work theta = $n\pi/6$ where $n = 0, 1, 2, 3, 4$ and 5 and have the output generated at 6 different orientations. Varying the orientations is what we call the oriented band pass filter's output.

Sigma – This parameter controls the width of the Gaussian envelope used in the Gabor kernel.

Gamma – gamma is the spatial aspect ratio. It controls the ellipticity of the Gaussian. When gamma = 1, the Gaussian envelope is circular.

After generating the Gabor kernel we convolve the image with the kernel and get the filtered image.

$$v_k(x) = \left| \sum_{x'} i(x') g_k(x - x') \right| \quad (3)$$

Here $v_k(x)$ is the output of the convolved image with the Gabor filter $g_k(x - x')$ at a specific frequency and orientation. $i(x')$ is the input image to the Gabor filter.

For computing an intermediate stage of the Houp descriptor for an image sub block, we convolve the image with a Gabor filter as shown above. Gabor filters are generated at 6 different orientations and each orientation's output is then passed to a local binary pattern [2]. Detailed analysis is performed on gabor coefficients and their joint distribution using local binary patterns. This is to aggregate encoded information at different locations into a low dimensional image representation. The suggested aggregation method based on the uniform pattern boosts the discriminative power and generalizability of the representations; it produces scene representations with lower dimensions than most of the existing methods.

Gabor Filters parameter selection

The selection of the parameters of the gabor filter is an important task which needs to be addressed. There does not exist any definite mechanism for selection of the parameters of the gabor filter. The parameter values depend on the dataset which is being used so there are no generalized values set for the gabor filter parameters which produce the best possible output. Here the following parameters for the gabor filter need to be selected. θ (theta), λ (lambda), φ (psi), γ (gamma) and σ (sigma). We are considering 6 different orientations for the filter, so the value of theta is $n\pi/6$ where $n = 0, 1, 2, 3, 4$ and 5 . φ (psi), the phase offset is set as zero. The remaining parameters λ (lambda), γ (gamma) and σ (sigma) have been chosen by searching the entire 3D space generated by λ , γ and σ . The value of the parameters that give the highest accuracy have been used in this work. The values of gabor filters parameters that are being used for the place categorization are different from those being used for place recognition as these are dependent on the dataset.

3.3 Local Binary Patterns

Uniform Patterns are a specific type of Local Binary Patterns proposed by Ojala 2002 [2] for grayscale texture classification. The method is based on recognizing that certain local binary patterns termed as 'uniform' are fundamental properties of local image texture, and their occurrence histogram proves to be a very powerful texture feature. Ojala derives a generalized grayscale and rotation invariant operator presentation that allows for detecting the 'uniform' patterns for any quantization of the angular space and for any spatial resolution and presents a method for multi-resolution analysis. The approach of Ojala [2] is very robust in terms of grayscale variations, since the operator by definition is invariant against any monotonic transformations of the gray scale. The proposed method of local binary patterns is also computationally simple as the operator can be implemented with a few operations in a small neighbourhood and a lookup table.

Two-dimensional textures have been studied and found to have many potential applications in the field of remote sensing, biomedical analysis and industrial surface inspection, but only a few examples of successful exploitation of texture exist. A major problem is that textures in the real world are often not uniform, due to variations in orientations, scale, or other visual appearance. Gray scale invariance is an important issue due to uneven illumination or great within-class variability.

Most of the existing approaches to texture classifications – generalized co-occurrence matrices [a], polarograms [b], texture anisotropy [c] indirectly assume that the unknown samples to be classified are identical to the training samples with respect to spatial scale, orientation and grayscale properties. However, real world textures can occur at arbitrary spatial resolutions and rotations and

they may be subjected to varying illumination conditions. This led to research incorporating the properties of invariance with respect to grayscale, rotation and spatial scale.

One of the earliest works to incorporate invariance of all the three properties (rotation, spatial scale and gray scale) is that on Zernike Moments by Wang and Healey [7]. In this report we use the local binary patterns proposed by Ojala [2] which is a theoretically and computationally simple approach. It is robust in terms of gray scale variations and which is shown to discriminate a large range of rotated textures efficiently. A gray scale and rotation invariant texture operator based on local binary patterns has been used here. Starting from a joint distribution of gray values of a circularly symmetric neighbour set of pixels in a local neighbourhood, an operator which is by definition invariant against any monotonic transformations of the gray scale has been derived by Ojala [2]. Rotational invariance is achieved by recognizing that this gray scale invariant operator incorporates a fixed set of rotation invariant patterns.

The most important property of using the local binary patterns (LBPs) is that certain LBPs termed as 'uniform' represent the fundamental properties of the local image texture and they help in generating a generalized gray scale and rotation invariant operator for detecting these 'uniform' patterns. The term 'uniform' refers to the uniform appearance of the local binary patterns, i.e. there are a limited number of transitions or discontinuities in the circular presentation of the pattern. These 'uniform' patterns provide a vast majority, sometimes over 90%, of the 3x3 texture patterns in examined surface textures. The most frequent 'uniform' binary patterns correspond to primitive micro-features such as edges, corners and spots, hence they can be regarded as **feature detectors** that trigger for the best matching pattern.

The texture operator being used here allows for detecting 'uniform' local binary patterns at circular neighbourhoods of any quantization of the angular space and at any spatial resolution. Consider a generalized case based on a circularly symmetric neighbour set of P members on a circle of radius R, the operator is denoted as $LBP_{P,R}^{riu2}$. The superscript riu2 has been used here to abbreviate rotation invariance and uniform transitions of at most 2. This is explained in equation (11). Parameter P controls the quantization of angular space, whereas R determines the spatial resolution of the operator. In this report P=8 and R=1 have been used for generating the HOUP descriptor.

The discrete occurrence histogram of the 'uniform' patterns (which is the responses of the $LBP_{P,R}^{riu2}$ operator) computed over an image or a region of image is a very powerful texture feature. The structural and statistical approaches are effectively combined by computation of the occurrence histogram; the local binary pattern detects microstructures (edges, lines, dark/bright spots, flat areas, etc.) whose underlying distribution is estimated by the histogram. Image texture has 2 properties spatial structure (pattern) and contrast (the amount of local image texture). These two form an interesting pair with respect to gray scale and rotation invariant texture description: spatial pattern is affected by rotation, contrast is not, and vice versa, where contrast is affected by the gray scale, spatial pattern is not. Consequently as long as we want to restrict ourselves to pure gray scale invariant texture analysis, contrast is of no interest as it depends on the gray scale.

The $LBP_{P,R}^{riu2}$ operator is an excellent measure of the spatial structure of local image texture, but it discards the other property of 'contrast' as it depends on the gray scale. In this report we use the local binary patterns aimed at achieving gray scale and rotational invariance only.

3.3.1 Gray Scale and Rotation Invariant LBPs

Here we present the derivation of the gray scale and rotation invariant texture operator. We begin by defining texture T in a local neighbourhood of a monochrome texture image as the joint distribution of the gray levels of P ($P > 1$) image pixels:

$$T \approx t(g_c, g_0, g_1, \dots, g_{P-1}) \quad (4)$$

Where gray value g_c corresponds to the gray value of the center pixel of the local neighbourhood and g_p ($p=0, \dots, P-1$) corresponds to the gray values of P equally spaced pixels on a circle of radius R ($R > 0$) that form a circularly symmetric neighbour set. If the coordinates of g_c are $(0,0)$, then the coordinates of g_p are given by $(-R\sin(2\pi p/P), R\cos(2\pi p/P))$. Figure 1 illustrates circular symmetric neighbour sets for various (P, R) . The gray values of the neighbors which do not fall exactly in the pixels are computed by interpolation.

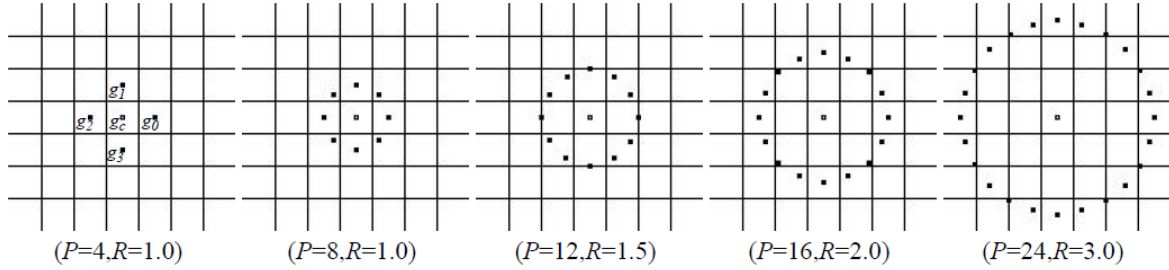


Figure 1. Circularly symmetric neighbour sets for different (P, R) .

In this work we consider the case of $P=8, R=1$ as shown above. Here we consider a 3×3 neighbourhood and subtract each pixel intensity from the centre pixel. Then if the sign of the resultant is positive we give it a value of 1 else we assign it a value of 0. We then generate the Local Binary Pattern by generating a decimal number. We then convert that number to a binary number which is a string of 0s and 1s. Following steps elucidate the process of generating a LBP for a pixel:

3.3.1.1 Achieving Gray Scale Invariance

Initially we subtract the gray value of the center pixel g_c from the gray values of the circularly symmetric neighbourhood g_p ($p=0, \dots, p-1$). This does not result in losing any information.

$$T \approx t(g_c, g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c) \quad (5)$$

Next we assume $g_p - g_c$ is independent of g_c ; this allows us to factorize the above equation as follows:

$$T \approx t(g_c) t(g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c) \quad (6)$$

Practically an exact independence is not required; hence the factorized distribution is just an approximation of the joint distribution. This small approximation results in achieving invariance with respect to shift in gray scale. The distribution $t(g_c)$ in equation (6) describes the overall luminance

of the image, which is unrelated to local image texture, and consequently does not provide useful information for texture analysis.

As the above equation (6) is independent of $t(g_c)$, we discard it:

$$T \approx t(g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c) \quad (7)$$

The above equation is a highly discriminative texture operator. It records the occurrences of various patterns in the neighbourhood of each pixel in a P dimensional histogram. For constant regions, the differences are zero in all directions. On a slowly sloped edge, the operator records the highest difference in the gradient descent and zero values along the edge, and for a spot the difference are high in all directions.

Signed differences $g_p - g_c$ are not affected by changes in mean luminance, hence the joint difference distribution is invariant against gray scale shifts. Here invariance with respect to the scaling of gray scale is achieved by considering just the sign of differences instead of their exact values:

$$T \approx t(s(g_0 - g_c), s(g_1 - g_c), \dots, s(g_{P-1} - g_c)) \quad (8)$$

$$\text{where} \quad s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (9)$$

By assigning a binomial factor of 2^p for each sign ($s(g_p - g_c)$), equation (8) is transformed into a unique number as follows:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_0) 2^p \quad (10)$$

Here in this thesis report, we set $P=8$ as we consider a 3×3 neighbourhood for a pixel for the task of generating the HOUP descriptor. The name 'Local Binary Pattern' reflects the functionality of the operator, i.e. a local neighbourhood (in our case 3×3) is thresholded at the gray value of the center pixel into a binary pattern. $LBP_{P,R}$ operator is by definition invariant against any monotonic transformation of the gray scale, i.e. as long as the order of the gray value in the image stays the same, the output of the $LBP_{P,R}$ operator remains constant. This is because of the function $s(x)$ which only considers the sign depending on the difference of each neighbouring pixel with the center pixel. This is what achieves the invariance against any monotonic transformation of gray scale.

3.3.3.2 Achieving Rotation Invariance

The $LBP_{P,R}$ operator produces 2^P different output values, corresponding to the 2^P different binary patterns that can be formed by the P pixels in the neighbour set. When the image is rotated, the gray values g_p will correspondingly move along the perimeter of the circle around g_0 . Since g_0 is

always assigned to the gray values of the element (0,R), to the right g_c rotating a particular binary pattern naturally results in a different $LBP_{P,R}$ value. This does not apply to patterns comprising of only 0's (or 1's) which remain constant at all rotation angles. To remove the effect of rotation, i.e. to assign a unique identifier to each rotation invariant local binary pattern we define:

$$LBP_{P,R}^{ri} = \min \{ROR(LBP_{P,R}, i) \mid i = 0, 1, \dots, P-1\} \quad (11)$$

$$LBP_{P,R}^{ri} = \min \{ROR(LBP_{P,R}, i) \mid i = 0, 1, \dots, P-1\} \quad (11)$$

Where $ROR(x, i)$ performs a circular bit-wise right shift on the P-bit number x i times. In terms of image pixels equation (11) simply corresponds to rotating the neighbour set clockwise so many times that a maximal number of the most significant bits, starting from g_{P-1} are 0.

$LBP_{P,R}^{ri}$ quantifies the occurrence statistics of individual rotation invariant patterns corresponding to certain micro-features in the image hence the patterns can be considered as feature detectors. Figure 2 illustrates the 36 unique rotation invariant local binary patterns that can occur in the case of $P=8$, i.e. $LBP_{P,R}^{ri}$ can have 36 different values. For example pattern #0 detects bright spots, #8 detects dark spots and flat areas, and #4 detects edges.

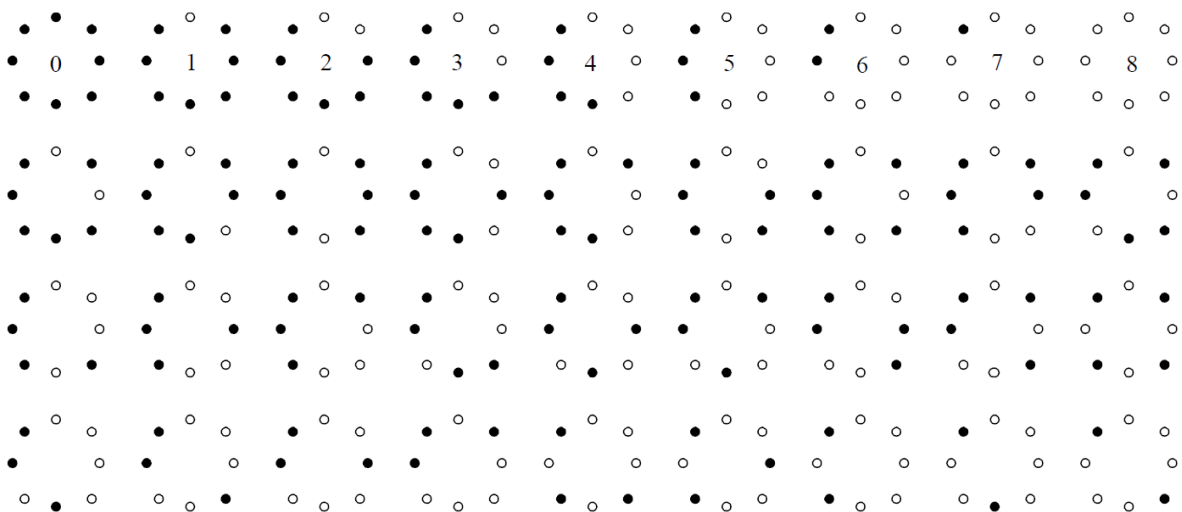


Figure 2 The 36 unique rotation invariant binary patterns that can occur in a circularly symmetric neighbour set of $LBP_{P,R}^{ri}$. Black and White circle correspond to bit values of 0 and 1 in the 8 bit output of the operator. The first row contains the nine 'uniform' patterns, and the numbers inside them correspond to their unique $LBP_{P,R}^{ri2}$ codes.

In the above figure (Fig 2) it can be seen that only the first row (#0 to #8) contains the uniform patterns as they have at most 2 transitions from 0 to 1 or 1 to 0.

To formally define the 'uniform' patterns, a uniformity measure $U(\text{'pattern'})$, which corresponds to the number of spatial transitions (bitwise 0/1 changes) in the 'pattern' has been used. For example, patterns 00000000_2 and 11111111_2 have U value of 0, while the other seven patterns in the first row of Fig. 2 have U value of 2, as there are exactly two 0/1 transitions in the pattern. Similarly, other 27

patterns have U value of at least 4. We designate patterns that have U value of at most 2 as ‘uniform’ and propose the following operator for gray scale and rotation invariant texture description instead of $LBP_{P,R}^{ri}$:

(12) (13)

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & \text{if } U(LBP_{P,R}) \leq 2 \\ P+1 & \text{otherwise} \end{cases} \quad (11)$$

where

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)| \quad (12)$$

Superscript $riu2$ reflects the use of rotation invariant ‘uniform’ patterns that have U value of at most 2. By definition exactly $P+1$ ‘uniform’ binary patterns can occur in a circularly symmetric neighbor set of P pixels. Equation (12) assigns a unique label to each of them, corresponding to the number of ‘1’ bits in the pattern ($0 \rightarrow P$), while the ‘non-uniform’ patterns are grouped under the ‘miscellaneous’ label ($P+1$). In Figure 2 the labels of the ‘uniform’ patterns are denoted inside the patterns. In practice the mapping from $LBP_{P,R}$ to $LBP_{P,R}^{riu2}$, which has $P+2$ distinct output values ($0, 1, 2, \dots, P+1$), is best implemented with a lookup table of 2^P elements which is 256 elements in our case.

The final texture feature employed in texture analysis is the histogram of the operator outputs (i.e. pattern labels) accumulated over a texture sample. The reason why the histogram of ‘uniform’ patterns provides better discrimination in comparison to the histogram of all individual patterns comes down to differences in their statistical properties. The relative proportion of ‘non-uniform’ patterns of all patterns accumulated into a histogram is so small that their probabilities cannot be estimated reliably. Inclusion of their noisy estimates in the (dis)similarity analysis of sample and model histograms would deteriorate performance.

When we use a 3×3 neighbourhood ($P=8, R=1$), we get a look-up table of 2^P ($2^8 = 256$) elements. Only 58 of the 256 total patterns generated are uniform. This means only 58 patterns have a transition of 0 to 1 or 1 to 0 at most two times. The 58 uniform patterns correspond to the decimal numbers as follows:

Decimal Representation	Binary Number
0	00000000
255	11111111
1	00000001
2	00000010
4	00000100
8	00001000
16	00010000

Decimal Representation	Binary Value
32	00100000
64	01000000
128	10000000
3	00000011
6	00000110
12	00001100
24	00011000
48	00110000
96	01100000
192	11000000
129	10000001
63	00111111
126	01111110
252	11111100
249	11111001
243	11110011
231	11100111
207	11001111
159	10011111
254	11111110
253	11111101
251	11111011
247	11110111
239	11101111
223	11011111
191	10111111
127	01111111
248	11111000
241	11110001
227	11100011
199	11000111
143	10001111
31	00011111
62	00111110
124	01111100
7	00000111
14	00001110
28	00011100
56	00111000
112	01110000
224	11100000
193	11000001
131	10000011
240	11110000
225	11100001
195	11000011
135	10000111
15	00001111
30	00011110

Decimal Representation	Binary Value
60	00111100
120	01111000

Table 1: the 58 possible uniform patters involving at most two transitions of 0/1 or 1/0

The 59th dimension is the sum of all other non-uniform patterns and is taken to be as the dimension representing the non-uniform patterns. So in total we have 59 dimensional image representations for an image sub block. We then consider computing the Histogram of Oriented Uniform Patterns for each output of the oriented band pass filter. As we took into account 6 different orientations, we get $59 * 6 = 354$ dimensional representations for an image sub block. This dimensionality is then reduced by selecting the first N principal components in such a way that the sum of chosen eigen values of the principal components accounts for more than 95% of the sum of all components. In our experiments the values of N is set to be 70 as it accounts for more than 95% of the sum of eigen values in most cases. So we select the first 70 principal components to act as representations for an image. Hence we have a 70 dimensional representation of an image sub block which we term as the “HOUP” descriptor for the image sub block. This is what we call one candidate feature for the image.

3.4 HOUP descriptors comparison

Comparing the HOUP descriptors

Numerous comparison metrics exist for comparing descriptors. However most of the methods do not take into account the problem of perceptual aliasing. Perceptual Aliasing refers to the occurrence of the visually similar image sub block in different categories. Following diagrams can help visualize the problem of perceptual aliasing –



Figure 2 : Above 3 images illustrate the problem of perceptual aliasing. Different images have similar image sub blocks.

Clearly here it can be seen that the particular image sub block appears visually similar and occurs in different classes. A conventional similarity measure would assign a higher similarity between the descriptors of these image sub blocks. Our work tackles the problem of perceptual aliasing by using a simple variant of the One Shot Similarity (OSS) measure by Wolf [4]. Given a pair of HOUP descriptors, the Linear Discriminant Analysis (LDA) algorithm is used to learn a model for each of the descriptors (as single positive samples) against a set of examples A. Each of the two learned models is applied on the other descriptor to obtain a likelihood score. The two estimated scores are then combined to compute the overall similarity score between the two descriptors:

$$s_n(x_I^n, x_I^n) = (x_I^n - \mu_A)^T S_A^{-1} \left(x_I^n - \frac{x_I^n + \mu_A}{2} \right) + (x_J^n - \mu_A)^T S_A^{-1} \left(x_I^n - \frac{x_J^n + \mu_A}{2} \right) \quad (13)$$

Here S_A and μ_A are the mean and co-variance of A.

In the OSS measure of Wolf [4], A is the training dataset excluding the samples from the classes that are currently being compared. Whereas in our work we replace the set A with the entire training set, because it is possible to take into consideration the distinctiveness of the descriptors, which is the key to achieve robustness against perceptual aliasing. If two descriptors are similar to each other but are indistinctive and relatively common in the dataset (example the descriptors extracted from repetitive features in the environment like sky patches), they receive a low similarity score. This is because the individual models that have been learned for the two descriptors cannot separate them well from the typical examples in A and therefore return a lower similarity scores when applied to one another. On the other hand, when two descriptors are distinctive but have lower similarity than the examples of perceptual aliasing, they are still assigned a higher similarity score because they can be separated better from the examples in A.

4. SUB DIVISION and FEATURE SELECTION

Here we divide an image into sub blocks to generate different features which would provide an informative representation of the image. We divide the given image into 1x1, 2x2, 3x3, 4x4 and 5x5 blocks. So in total we have 55 sub blocks * 3 frequencies = 165 candidate features. Now a HOUP descriptor for each image sub block / Gabor frequency is computed. It is observed that highest accuracy is achieved when using the 3x3 sub division scheme, we get 9 features each of 70 dimensionality.

Feature selection refers to selecting the most informative features among the pool of features generated by the sub division scheme. Here a method for feature selection based on Kernel Alignment is used. This was introduced by Christianini [3]. It aims at providing a similarity measure between a kernel and a target kernel function:

$$A(K_1, K_2) = \frac{\langle K_1, K_2 \rangle_F}{\sqrt{\langle K_1, K_1 \rangle_F \langle K_2, K_2 \rangle_F}} \quad (14)$$

Here $\langle K_1, K_2 \rangle_F$ is the Frobenius dot product and $A(K_1, K_2)$ is the alignment between the Kernels.

A target kernel function K_T is defined as $K_T(I, J) = 1$, if I and J belong to the same class, else 0.

For each candidate feature n (Image sub block or Gabor Frequency), its corresponding HOUP descriptors extracted from the training images form a kernel K_n by using the similarity measure of the OSS [7] score as described in the previous section, it is encapsulated in a parameterized sigmoid function:

$$K_n(I, J) = \frac{1}{1 + \exp(-\sigma_n s_n(x_I^n, x_J^n))} \quad (15)$$

$s_n(x_I^n, x_J^n)$ is the similarity between the n th descriptor extracted from the images I and J , and σ_n is the kernel parameter, chosen to maximize the alignment $A(K_1, K_2)$.

The informative feature selection works in the following manner;

$$Q_l = \arg \max_{K_i \in P_l} \min_{K_j \in R_l} (A(K_i \cdot K_j, K_T) - A(K_j, K_T)) \quad (16)$$

Here Q_l is the l^{th} feature to be selected; P_l is the pool of candidate features, R_l is the set of selected features up to iteration l , Q_l is the feature to be selected in iteration l , and $K_i \cdot K_j$ is the joint kernel produced by combining s_i and s_j . Following steps illustrate the process of feature selection

- Compute alignment for each of Kernels as defined in equation 14.
- For the first feature we select the feature whose kernel gives the maximum alignment.
- We then select a second feature from the pool of features P ; But the selection criteria at this stage cannot be the alignment with the target kernel anymore because if the second feature selected is also highly aligned with the first feature, the alignment of the combined kernel would remain the same. On the other hand if the 2 selected features are not aligned to each other, the combined kernel produced by the 2 features would be even more aligned to the target kernel
- So the feature should maximize $A(K_n \cdot K_1) - A(K_1, K_T)$, where K_1 is the kernel matrix of the first selected feature.
- The third feature should maximize $\min [\{A(K_n \cdot K_1) - A(K_1, K_T)\}, \{A(K_n \cdot K_2) - A(K_2, K_T)\}]$, where K_1 and K_2 are the first 2 selected features.
- We then proceed to searching for the next feature, f_3 whose kernel delivers the maximum amount of additional alignment with respect to each of the previously selected features. Iteratively we proceed as described above for the successive features (f_4, f_5, \dots) as per the expansion of the feature selection equation (16).
- Feature Selection Algorithm terminates when there is a negligible change in alignment between the kernels.

Feature selection has been implemented but is not computationally feasible for a practical system for a mobile robot. Moreover it is computationally very expensive. This analysis is based on the execution of codes developed in Matlab.

5. SCENE REPRESENTATION

We further divide the image into 1x1, 2x2, 3x3, 4x4 and 5x5 blocks and we then compute a HOUW descriptor for each of the image sub blocks. We will then have a total of 55×3 frequencies = 165 candidate features. We then adopt the method of feature selection for selecting the most

informative features from the above pool of features. Feature Selection is implemented using a method of Target Kernel Alignment proposed by Christianini [5] 2002 as discussed in the previous section.

Initially we tried to avoid the feature selection part and simply considered the 5 (1x1, 2x2, 3x3, 4x4 and 5x5 sub division) cases individually; it was observed that highest accuracy was achieved when considering the 3x3 sub division scheme.

So we initially went with considering 9 features for an image. Two types of experiments were conducted for validating the practical significance of our proposed descriptor –

- Topological Place Categorization – the UIUC dataset was used
- Topological Place Recognition – KTH IDOL dataset was used

For the Topological place categorization the LIBSVM algorithm IS used with a variant of the OSS kernel as discussed in previous section. The libsvm tool [6] is used with the default parameters with the exception of c and w which are set to 0.6 and 1.3 respectively.

For the Topological Place Recognition the 1 Nearest Neighbour (1-NN) classifier is used to perform the classification task. Correlation is used with the 1-NN classifier.

6. EXPERIMENTS

6.1 The UIUC dataset

The UIUC dataset [8] has been developed by Olivia and Torralba (2001), Fei-Fei and Perona (2005) and Lazebnik et al. (2006). This is one of the most commonly used databases for scene recognition in the field of Computer Vision. The dataset consist of 15 scene categories – “Suburb, Living Room, Forest, Mountain, Open Country, Street, Store, Bedroom, Industrial, Highway, Coast, Inside City, Office, Tall Building and Kitchen.” Each class contains 210 – 410 images. Same images from the dataset can be seen in Fig. 3



Figure 3: Scene Categorization images from the UIUC Database

The standard procedure for experimenting with this dataset is randomly selecting 100 images for training and rest for testing. We here use the same standard protocol used for the dataset; we use 100 images selected from each category for training and use the remaining images in the dataset for testing. The procedure described in Fazl-Ersi [1] uses the feature selection algorithm to select the most informative features, Fazl-Ersi mentions that on an average 43 features are selected from the pool of the 165 candidate features. This leads to $43 * 70 = 3010$ dimensional representations to describe a single image. Fazl-Ersi compares the accuracy of his method with other state of the art methods and performs better than them. In his paper [1], he mentions that feature selection selects almost all 1x1 and 2x2 grids whereas 48% and 23% of the 3x3 and 4x4 blocks are selected.

All images have been resized to 256x256 as most of the images are closer to this number. We here use the $3 \times 3 = 9$ features to represent an image which leads to $9 * 70 = 630$ dimensional representation for an image. The LIBSVM tool [5] is used as the classification algorithm for classifying the images. The LIBSVM tool with a variant of the OSS kernel is used as the underlying kernel measure similar to that in the original paper [1]. We here use the LIBSVM tool [5] with a linear kernel to perform the task of classification of the scene categories. We then incorporate the OSS^+ kernel to be used as a predefined kernel with the LIBSVM algorithm.

LIBSVM Tool

The Support Vector Machine algorithm has been used as a classifier. The Support Vector Machine is one of the best available kernel based classifiers.

One of the most important tasks involved while using a classifier is to have an appropriate method to normalize the data depending on the classifier used. For example when using the SVM algorithm with a radial basis (rbf) kernel, the performance of the system would be very unacceptable unless the data is appropriately normalised. Similarly while using the LIBSVM tool with the OSS^+ kernel, the data has been scaled to be between 0 and 1. This is a crucial part and boosts the accuracy by 10 to 15 per cent as opposed to using it without scaling. Additional benefits and the difference that scaling can make for a building a successful system can be found in [14] where in the author mentions about various examples where scaling the data shoots up the accuracy by even 30%.

Accuracy

We achieve an accuracy of 72 % by using the 9 features with the linear kernel; After employing the OSS^+ kernel, we get an accuracy of 75.33% as opposed to the 86 % accuracy achieved in the original paper. We are off by 10 % for visual place categorization.

Reasons for low accuracy:

- We here have not implemented feature selection because of its computational inefficacy
- We have a very compact representation of a single image 9 features as opposed to the 43 features being used in the original work

It is expected that after implementing the above points, the accuracy should be comparable to that stated in the paper [1]. However we argue about the infeasibility of having the feature selection algorithm to be implemented as it takes a large amount of time for finding the informative features.

6.2 KTH IDOL dataset

This dataset is well known for Topological Place Recognition. It was created by Pronobis et al. (2006) [15]. The purpose of this experiment is different from the previous one this one is a recognition task not a categorization one. However this is also challenging as it provides images of different places under varying lighting conditions. This dataset is built in an office environment and has images belonging to 5 places – *“kitchen, corridor, one person office, two person office and a printing area”*. The images have been captured by 2 robots Minnie and Dumbo under 3 different lighting conditions – night, sunny and cloudy. The images captured from the two robots appear quite different as cameras are mounted at different heights. The dataset contains 24 image sequences captured by the 2 robot in 3 different lighting conditions and each lighting condition has 4 image sequences. Each image sequence has 850 to 1200 images. The entire dataset can be downloaded from <http://www.cas.kth.se/IDOL/>.

As in the original paper [1] and Wu and Rehg (2011), we use the first two image sequences in our experiments. We perform 3 types of experiments

- Same Robot Same Lighting conditions
- Same Robot Different Lighting Conditions
- Different Robot Same Lighting Conditions

We here use the sub division scheme of diving each image into 1x1, 2x2, 3x3, 4x4 and 5x5 sub blocks leading to generation of 1, 4, 9, 16 and 25 features. For all the experiments above under varying lighting conditions, we use just the 3x3 sub division scheme as opposed to implementing the feature selection algorithm (to select some specific features) which is computationally very expensive and not worth the effort. We found that using the 3x3 sub division scheme produces features which give the highest accuracy. Other sub division scheme produces features that give lower accuracies. It was found that a particular image can be represented by 3x3 x 70 dimensions = 630 dimensional representation. So each image is described by a feature vector that is 630 numbers. After generating the feature vectors for all the images, the training dataset is then passed through a Nearest Neighbour Classifier.

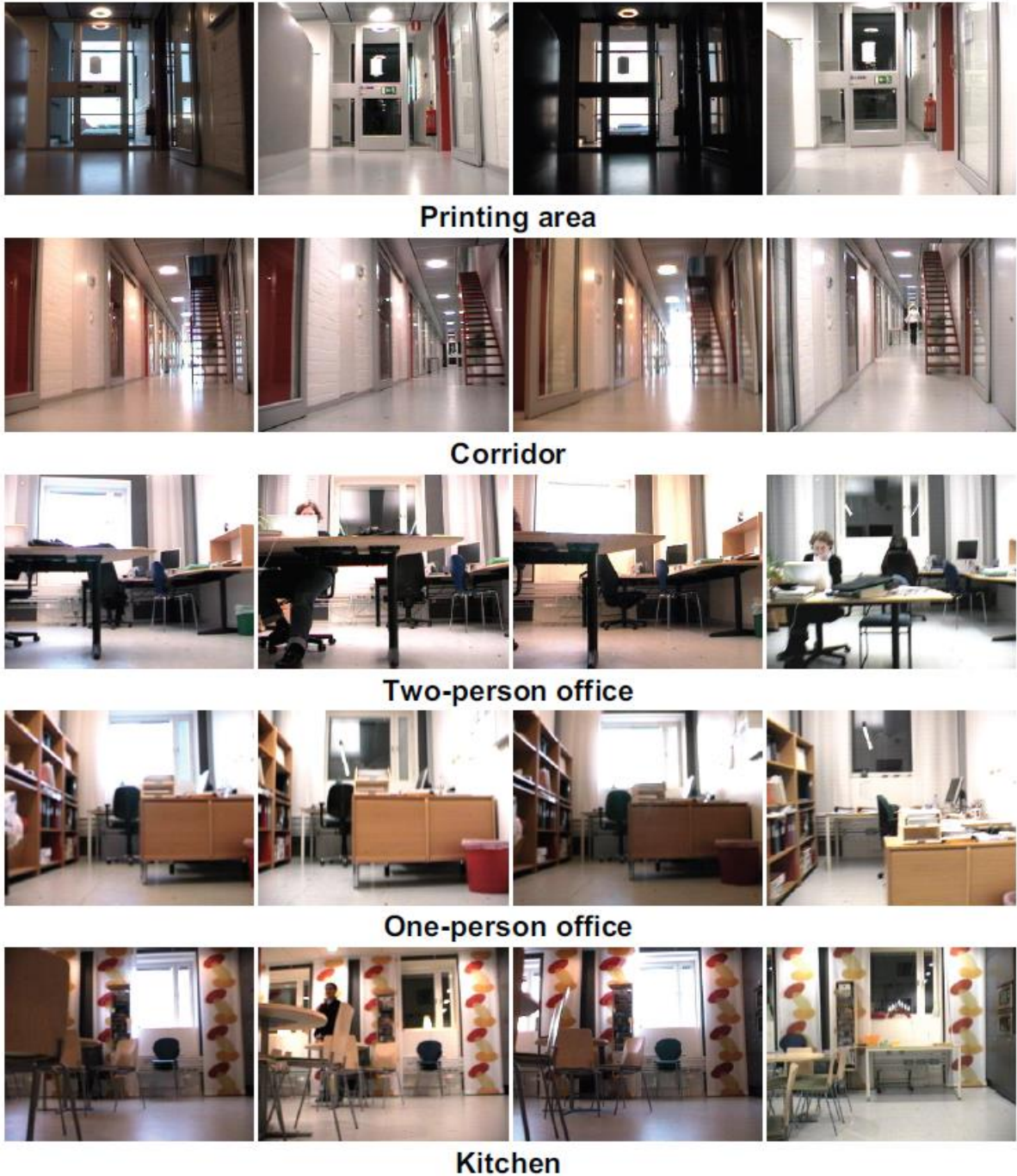


Figure 4: Sample images from the KTH IDOL topological place recognition database, with five places. For each place, the first three sample images (from left to right) were captured by Dumbo robot at three different lighting conditions (cloudy, night and sunny, respectively) and the fourth sample was captured by the Minnie robot at the night lighting condition. All sample images for each place were captured from relatively same pose.

The Classifier used – 1 Nearest Neighbour

The 1 Nearest Neighbour Classifier has been used here. Various distance metrics such as Euclidean distance, correlation, cosine distance, city block metric, Chebychev distance and Spearman distance have been tried with the generated dataset. The classification results were best reported when using

the spearman distance. Two important things need to be paid attention to while generating the HOUP descriptor using the method as illustrated in Figure 1 and while using the classifier. First the value of Gabor filters parameter play a crucial role in the generation of the feature vector for each image. Using good parameters can often shoot up the accuracies by 10 % more than that obtained by using parameters that do not lead to producing a good representation. In our case as explained in the section on Gabor Filters, the values of parameters have been chosen by doing a brute force search of the entire space and selecting the ones that lead to the highest classification accuracy. Secondly the nearest neighbour classifier needs to be paid attention to; one should select the the distance metric that best suits their dataset. In this experiment it was found that the Spearman distance was the best which suited the dataset. So the gabor filter parameter and choosing the distance measure play an important role in optimizing the performance of the place recognition system.

Experimental Results

The following table shows the average accuracies by taking into account all possible combinations for the experiment

Experiment	Train	Test	Lighting	Performance			
			Wu & Reh	Wu & Reh, 2011	Pronobis, 2006	Fazl-Ersi, Tsotsos	Sahdev, Tsotsos
1	Minnie	Minnie	Same	95.35	95.51	96.61	95.38
	Dumbo	Dumbo	Same	97.62	97.26	98.24	97.22
2	Minnie	Minnie	Different	90.17	71.90	92.01	85
	Dumbo	Dumbo	Different	94.98	80.55	95.76	88*
3	Dumbo	Minnie	Same	77.78	66.63	80.05	72.46
	Minnie	Dumbo	Same	72.44	62.20	75.43	75.48

Table 2: Results obtained from the KTH IDOL dataset.

*This value is approximate, out of the 24 possible cases; 12 have been taken into account.

In the paper by Fazl-Ersi, Tsotsos, 2012[1], for experiment 1,2 and 3 the feature selection algorithm selected 9, 13 and 23 features. Naturally number of selected features increased with the difficulty of the experiments with maximum being for the third one.

Reasons for the lower accuracies than the best available Fazl-Ersi, Tsotsos[1]:

- The accuracies reported above are from 9 features by using the 3x3 sub division scheme
- We have used only 9 features to be used for all the three type of experiments. It can be seen that we get comparable results to the ones reported by Fazl-Ersi, Tsotsos [1]. We are off on average by around 6 %.
- This is due to the fact that Feature selection has not been used to select the most informative features.
- We have not implemented feature selection because our final place recognition system has to be used on an actual mobile robot and we want it to run the proposed system in this

report in a realistic situation. We want the training to be done in a less amount of time so that it is practically feasible.

Feature selection was implemented for some experiments and it did increase the accuracies by approximately 3-4%. But realizing the practical infeasibility of the feature selection algorithm, it was decided to not use it. After implementing the feature selection algorithm the accuracies are expected to increase and move closer to those of Fazl-Ersi and Tsotsos 2012.

7. OUR DATASET

Several Datasets exist for Visual Place Recognition such as the one described above [15]. Siagian and Itti, 2007 developed the USC dataset for topological place recognition. Most of the existing datasets for Visual Place recognition are monocular datasets and do not provide much information with regard to depth. We in this work have developed a dataset comprising of 11 different classes described in the next sub section. The dataset developed is a binocular dataset which has 3 image representations of a scene – the left image, the right image and the depth map.

7.1 Experimental Setup

In this section we describe the experimental scenario and the data acquisition devices employed for the evaluation of our visual place recognition system. We tested it on two mobile robot platforms, “Pioneer” and “Virtual ME”. The robot platforms used for data acquisition are shown in Figure 4. This dataset has been generated keeping in mind to have a dataset that can be publically used by researchers. It is a challenging novel stereo dataset acquired in two different lighting conditions.



Figure 4: Virtual ME (left) and Pioneer (right). For Virtual ME height of camera above floor is 117 cms; for Pioneer its 88 cms.

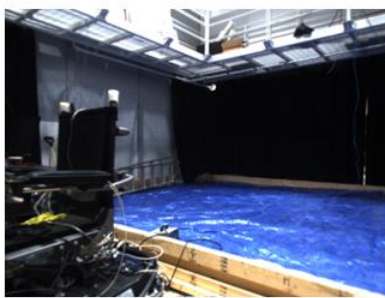
7.1.1 Experimental Scenario

The experiments were conducted on the third floor of the Lassonde Building at York University, Canada. Data has been captured from eleven different places; each place represents a different type of functional area that is commonly observed in a university building devoted to research. Following places have been used to acquire the data:

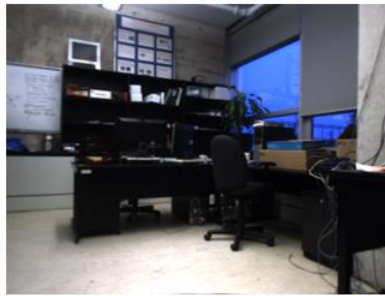
- Arena - this is the place where most of the robots of the lab are kept and various experiments are conducted. A Mars type of environment is present which is covered with a blue sheet to prevent dust.
- Work Place – This is the part of the lab where students work on the assigned work places.
- Ash Room – This is the Lab Managers room.
- Living Room – This is the living room present in the lab
- Corridor – corridors present in the building
- Lab2 – This is the second type of Lab.
- Plant Room – This room is a sub part of the second lab where a plant is present
- Lounge – This place is the computer science department’s graduate lounge for students.
- Prof Room – This is a typical professor’s room
- Seminar Room – This is a seminar room in the computer science department at York University.
- Wash Room – This is the wash room present at york

Some of the places are treated as a different entity while some are separated by cardboard room dividers or curtains to mark off different parts of a big lab. Arena, Workplace, Ash Room and the living Room are different places of one big lab. Lab2, Plant Room and Professor Room are in the second big lab having the three different places as described above. Lounge, Seminar Room and Wash Room are three separate entities used to capture images and generate the dataset. Corridor is a place that essentially links the various places (labs, lounge, washrooms, seminar rooms, etc.) together. Example picture of the eleven places can be seen in Figure 5.

As already mentioned the visual dataset was developed using the two robot platforms under two different lighting conditions day (when the natural sun light dominates) and night (when the rooms light has a significant effect on the place). The image acquisition was spread over a period of two weeks to generate the dataset. In this way we captured the visual variability that might have occurred.



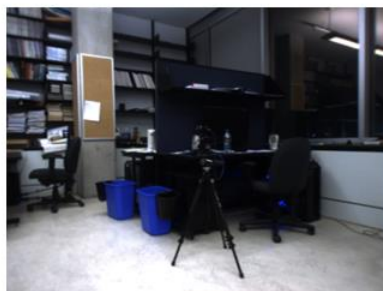
Arena



Ash Room



Corridor



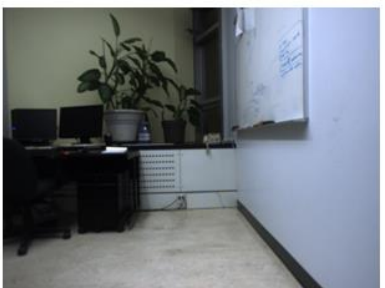
Lab2



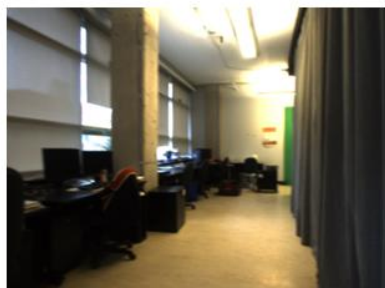
Living Room



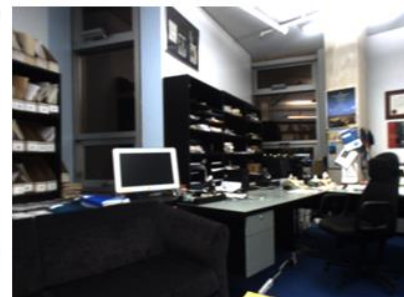
Lounge



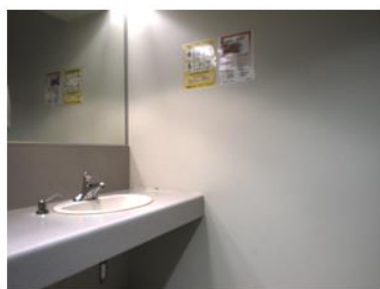
Plant Room



Work Place



Professor Room



Wash Room



Seminar Room

Figure 5: The eleven different places using which the dataset was generated

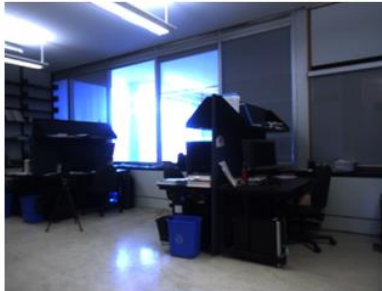
7.1.2 Robot Platforms

Both robots the White Bot '*Virtual ME*' and the red coloured bot '*Pioneer*' are equipped with a directed perception pan tilt unit and a point grey stereo camera bumble bee. However as can be seen in Figure 4, the cameras are mounted at different heights. On *Pioneer* the camera is 88 centimetres above the ground level, whereas on *Virtual me* it is 117 centimetres above the floor. All images were acquired with a resolution of 640 x 480 pixels, with the camera fixed at an upright position. The camera had the freedom to rotate on the spot for Pioneer robot; for virtual me the robot rotated on the spot which gave an indirect effect of having the camera rotate on the spot. The robot (virtual me) and pioneer's camera rotated in order to look around during the acquisition process.

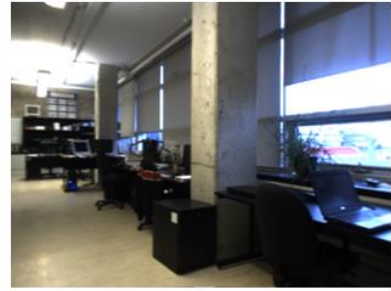
We followed the same procedure during image acquisition with both robot platforms. The robots were manually driven (speed approximately 0.5 meters per second) through all the eleven places while continuously acquiring images at the rate of approximately 3 frames per second. For the different illumination conditions (day and night), the acquisition procedure was performed twice, resulting in two image sequences acquired one after the other giving a total of 4 sequences across a span of two weeks. Example images can be seen in Figure 6. Due to manual control the path of the robot was slightly different for every sequence. Each image sequence consists of 1800 to 2000 images with 60 – 200 images belonging to each place. Currently the process of labelling the places is not automated; it was after acquisition labelled manually by renaming the files depending on the place the robot was in at that particular time. Each image was then labelled as belonging to one of the eleven places based on the position from where it was taken. For example the robot while standing on the exit of '*Ash Room*' views the living room is labelled as '*Ash Room*' because it took the image while it was in the place – '*Ash Room*'. Similarly for Robot standing in '*lab2*' looking at the '*Plant Room*' is labelled as '*lab2*' because it is physically in the place – '*lab2*'.



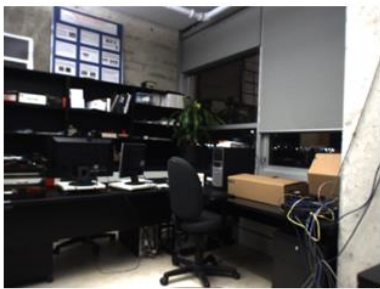
Ash Room (virtual me, day)



Lab (virtual me, day)



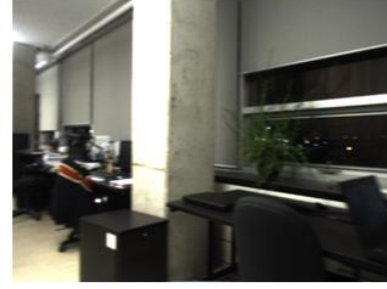
Work Place (virtual me, day)



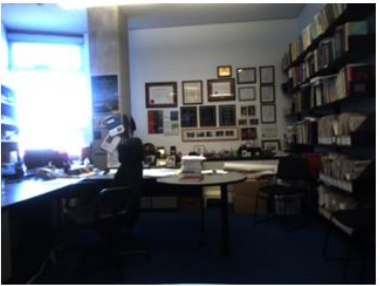
Ash Room (virtual me, night)



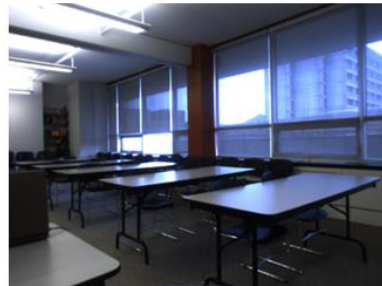
Lab (virtual me, night)



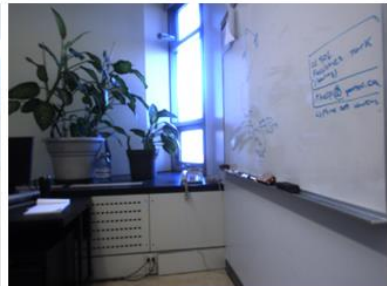
Work Place (virtual me, night)



Prof Room (virtual me, day)



Sem Room(virtual me, day)



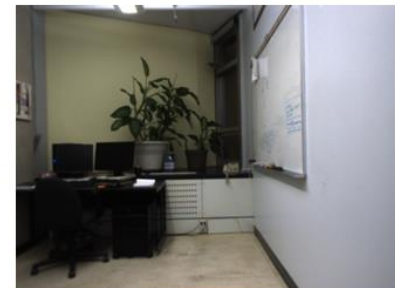
Plant Room (virtual me, day)



Prof Room (virtual me, night)



Sem Room(virtual me, night)



Plant Room (virtual me, night)

Figure 6a. Images acquired by virtual ME under different illumination conditions



Living Room (virtual me, day)



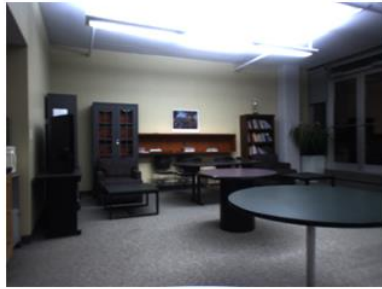
Lounge (virtual me, day)



Corridor (virtual me, day)



Living Room (virtual me, night)

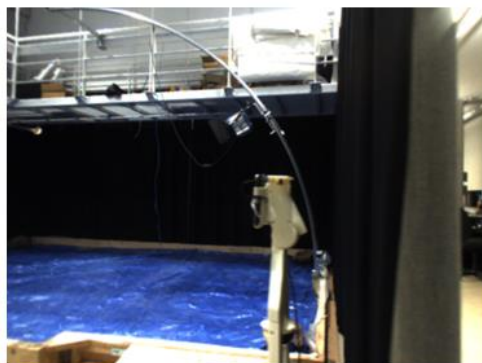


Lounge (virtual me, night)

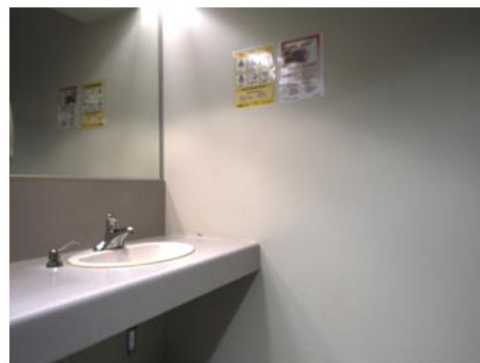


Corridor (virtual me, night)

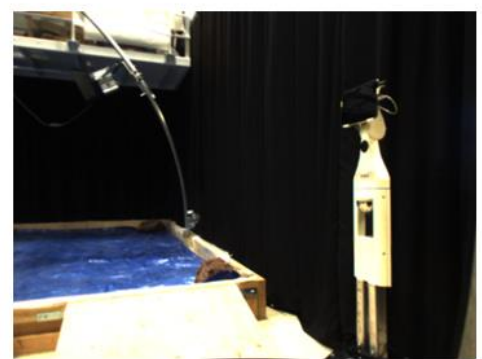
Figure 6b. Images acquired by virtual ME under varying illumination conditions



Arena (virtual me)



Wash Room (virtual me)



Arena (pioneer)



Wash Room (pioneer)

Figure 6c: Images acquired by different robots displaying the variability introduced thereafter

7.2 EXPERIMENTAL RESULTS

We conducted four sets of experiments in order to evaluate the performance of our system and test its robustness to different types of variations. We present the results in the successive subsection and give an illustrative summary through graphs. We started with a set of reference experiments evaluating our method under stable illumination conditions (A). Next we increased the difficulty of the problem and tested the robustness of the system to changing illumination conditions (B) as well as to other variations that may occur in real-world environments. Next we moved on to see whether a model trained on images acquired from one device (robot) can be useful for solving the localization / recognition problems with a different device (robot) in similar illumination condition (C). Finally we modelled a system that would use images trained on one device under a specific lighting condition and test on a different device under different lighting condition (D). We obtain encouraging results for all the 4 types of experiments conducted as can be seen in the next section.

For the different image sequences different number of images for each place were present in all image sequences of the two robots in two lighting conditions. As mentioned in the previous section on the KTH IDOL dataset, a similar approach was used to generate the Houp descriptor for an image sub-block. The same sub-division scheme of 3x3 was used giving rise to $9 \times 70 = 630$ dimensional representation of each image. The classification algorithm being used is also the same as that used for the KTH IDOL dataset; here too we use the 1 nearest neighbour with the Spearman distance metric. For all the four types of experiments mentioned above same Gabor filter parameters and sub division scheme was used. Here too the feature selection algorithm was avoided due to its practical infeasibility in our work for mobile robot localization.

Here as described above, we consider 4 different types of experiments conducted. Following types of experiments were conducted:

- A. Same Robot Same Lighting Conditions
- B. Same Robot Different Lighting Conditions
- C. Different Robot Same Lighting Conditions
- D. Different Robot Different Lighting Conditions

A. Same Robot and same Lighting Conditions

In order to evaluate our method under stable lighting conditions, we trained and tested the system on pairs of image sequences acquired one after the other using the same robot. Although the lighting conditions for both training and test images were in this case very similar, the algorithm had to tackle other kinds of variability such as viewpoint changes caused mainly by the manual control of the robot. The results of the performed experiments are presented in Fig. 8. For each platform and type of lighting conditions used for training, the first bar of first set and second bar of second set presents an average classification rate over the two possible permutations of the image sequences in the training and test sets. On average, the system classified properly 98.2% of the images acquired with Virtual ME and 98.5% of images acquired with Pioneer. After carefully observing the images that were classified incorrectly a majority of them were the ones that occurred while transitioning from one place to another. This can be explained by the fact that the images were not labelled according to their content but to the position of the robot at the time of acquisition. Since these

experiments were conducted with the sequences captured under similar conditions, we treat them as a reference for other results.

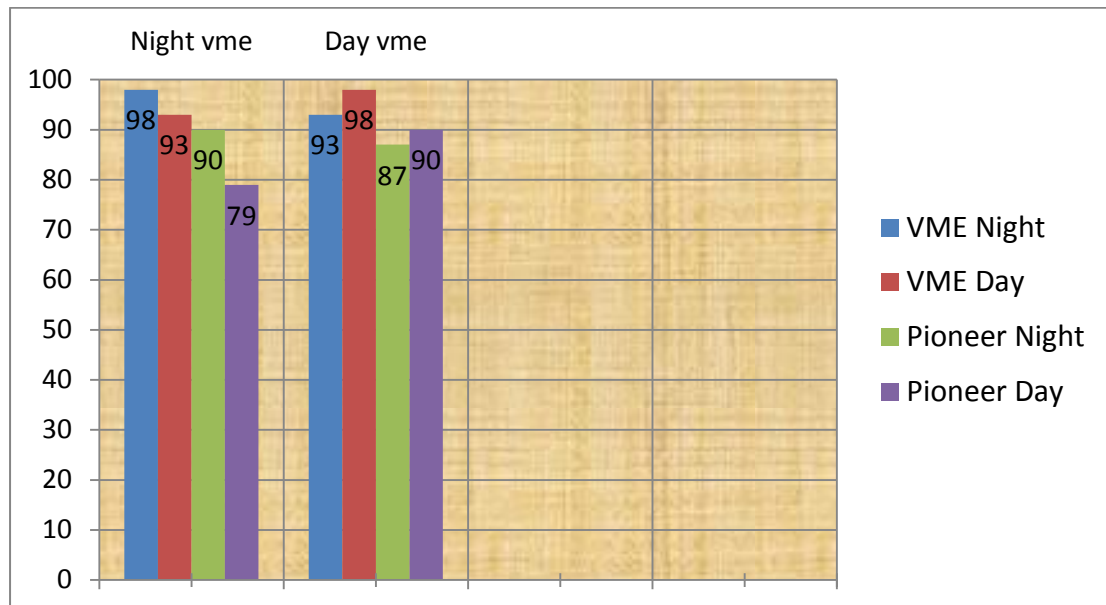


Figure 8: Training on images acquired by Virtual Me. First part shows classification accuracies obtained by training on night images acquired by Virtual ME. Second one shows accuracies obtained by training on day images by Virtual Me

B. Same Robot Different Lighting Conditions

We then conducted a series of experiments aiming to test the robustness of our method to changing lighting conditions as well as to other variations caused by normal activities in the rooms. As with the previous experiments, the same device was used for both training and testing. This time, however, the training and test sets consisted of images acquired under different illumination conditions. Figure 9 shows the average results of the experiments with Training on the first sequence, testing on the second sequence, and vice versa. Second bar (red coloured) of the first set and third bar (green) of second set reflects the accuracy for this experiment.

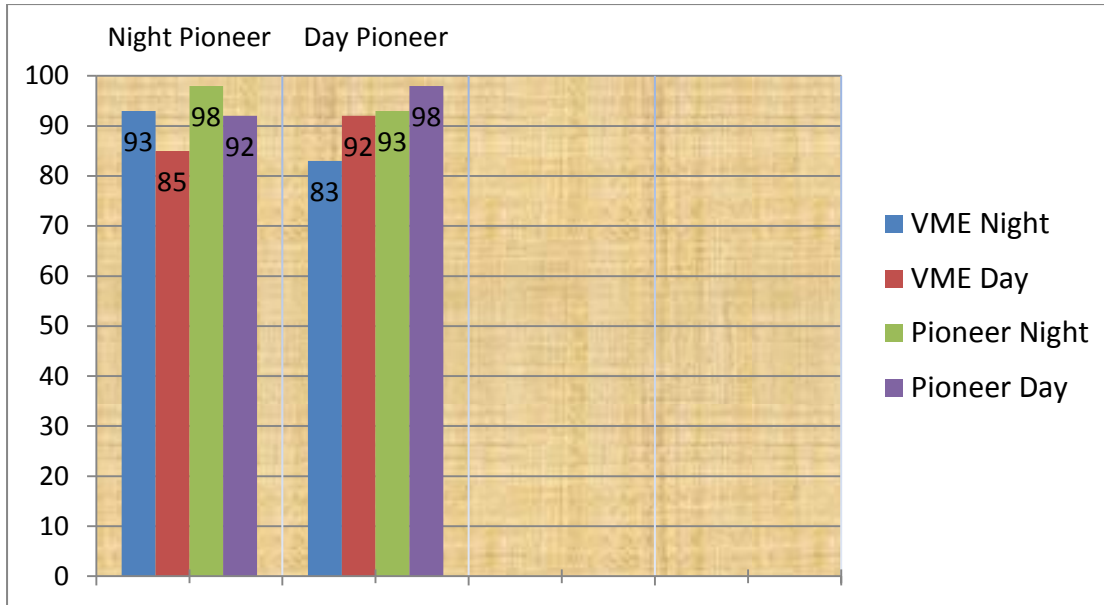


Figure 9: Training on images acquired by Pioneer. First part shows classification accuracies obtained by training on night images acquired by Pioneer. Second one shows accuracies obtained by training on day images by Virtual Me

C. Different Robot Same Lighting Conditions

This experiment was designed to test the portability of the acquired model across different robot platforms. For that purpose we trained and tested the system on images acquired under similar lighting conditions using different robots. We started with the experiments with both robot platforms. We trained the system on the images acquired using either Virtual ME or Pioneer and tested with the images captured with the other robot. We conducted the experiments for all lighting conditions. The main difference between the platforms from the point of view of our experiments lies in the height at which the cameras are mounted. The results presented in Figure 8 and 9 indicate that our method was still able to classify up to about 90% of images correctly. The system performed better when trained on the images captured with Virtual ME. This can be explained by the fact that the lower mounted camera on Pioneer provided less diagnostic information. It was also observed that in general the additional errors occurred when the robot was positioned close to the walls or furniture. In such cases the height at which the camera was mounted influenced the content of the images the most.

D. Different Robot Different Lighting Conditions

This experiment was performed last and was a challenging one as it had to model a scenario taking into account different robots and different illumination conditions. Here the training was done using either Virtual ME or Pioneer in day or night, but the testing was done not only using the other robot but also opposite lighting conditions. Our Place Recognition system is able to successfully perform under such conditions and we get an accuracy of 82 %. This accuracy is quite low as compared to the previous experiments but that is due to the challenging nature of the experiments being carried out. Additional Summary about the accuracies reported in this and the previous sections have been neatly summarized in Table 3.

EXPERIMENTAL RESULTS SUMMARY

Experiment	Training Set	Testing Set	Lighting Conditions	Accuracy
1	Pioneer	Pioneer	Same	98
	Virtual ME	Virtual ME	Same	98
2	Pioneer	Pioneer	Different	93
	Virtual ME	Virtual ME	Different	93
3	Pioneer	Virtual ME	Same	92
	Virtual ME	Pioneer	Same	92
4	Pioneer	Virtual ME	Different	82
	Virtual ME	Pioneer	Different	85

Table 3: Accuracies reported by our system on the dataset generated in this report.

E. Concluding Remarks regarding our dataset and the place recognition system

As can be clearly seen in table 3, our place recognition system performs very well using a standard stereo camera. In this thesis single images are used for the validating the proposed system. A stereo dataset has been generated but we use only left camera's images from it. Our system is not only robust to changes in lighting but also to the variability in different viewpoints introduced by acquiring images by the manual control of the robot. As the system is to be used in a practical environment we kept the training dataset small by using just one of the image sequence for training and the other image sequences for testing. It has been observed by experiments if two image sequences from different lighting conditions are used as the training set the accuracy increases by 3-4 %.

8. Programming Platform Used

Matlab

Matlab has been used as the major programming platform for developing the major part of the software for the place recognition system. It must be noted that the use of parallel pools has been done for the system which boosts up the speed of the algorithm manifold. It should also be noted that the code has not yet been ported from Matlab to C++ after which the computational efficiency of the system is expected to increase.

ROS Connectivity

The Robot Operating System (ROS) was used to build the dataset using the stereo camera Bumblebee. The programming platform used for developing the dataset was C++.

A. CONCLUSION

This thesis presented a vision based place recognition system for the qualitative localization of a mobile robot. We started out by describing a HOUP descriptor and validated its efficiency through a series of experiments in the subsequent sections. Appropriate classifiers were chosen and used for the place categorization and the place recognition systems. One of the significant contributions of this thesis is the generation of a novel stereo dataset for doing place recognition. Future work could include using the information from the disparity / depth maps with the descriptor to increase the performance of the system. We soon plan to enhance the system to make it generalized where in it is left in an unknown environment and different places are labelled by a person remotely operating the rover and initially labelling the places and later the robot is able to successfully localize itself in the unknown environment. Due to the high accuracies reported in the thesis. It is believed that such a system is practically feasible and efficient.

B. REFERENCES

1. **A Journal Paper:** FaFazl-Ersi and Tsotsos (2012) Histogram of Oriented Uniform Patterns for Robust place recognition and categorization. In: The International Journal for Robotics Research.
2. **A Conference Paper:** Ojala T, Pietikainen M and Maenpaa T (2002) Multi resolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(7): 971-987
3. **A Conference Paper:** Christianini N, Shawe-Taylor J, Elisseeff A, et al. (2002) On kernel target alignment
4. **A Conference Paper:** Wolf L, Hassner T and Taigman Y (2009) The one shot similarity kernel. In: *proceedings of IEEE international conference on computer vision*
5. **A Conference Paper:** Chang C and Lin C (2001) LIBSVM : a library for support vector machines.
6. **A Conference Paper:** Lazebnik S, Schmid C and Ponce J (2006) Beyond bag of features: Spatial Pyramid Matching for recognizing natural scene categories. In *proceedings of the IEEE international conference on computer vision and pattern recognition*, pp 2169 – 2178
7. **A Conference Paper:** L Wang and G. Healey (1998) Using Zernike moments for the illumination and geometry invariant classification of multispectral texture. In: *IEEE Transactions on Image Processing*, pp 196 -203.
8. **A Journal Paper:** Torralba A (2003) Contextual Priming for object detection. *International Journal of Computer Vision* 53(2): 169-191
9. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*.
10. **A Conference Paper:** Bay et al. (2006) Speeded Up Robust Features, ECCV
11. **A Conference Paper:** Dalal and Triggs, Histogram of Oriented Gradients, (2005) CVPR
12. **A Conference Paper:** Torralba A, Murphy KP, Freeman WT, et al. (2003) Context Based vision system for place and object detection. In: *proceedings of IEEE international conference of Computer Vision*, p273
13. **A Conference Paper:** Viola P and Jones M (2001) Rapid Object Detection using a Boosted Cascade of Simple features, In: *proceedings of the IEEE international conference on computer vision and pattern recognition*, pp. 511-518.

14. **A Conference Paper:** Chih-Wei Hsu, Chih Chung Chang, Chih-Jen Lin, A Practical Guide to Support Vector Classification, 2003
15. **A Conference Paper:** Pronobis A, Caputo B, Jesfelt P, et al. (2006) A discriminative approach to robust visual place recognition. In: *proceedings of International conference on robots and systems*, pp. 3829-3836
16. **A Journal Paper:** Olivia A and Torralba A (2001) Modelling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision* 42(3): 145-175
17. **A Conference Paper:** Martinez-Mozos O, Stachniss C and Burgard 2 (2005) Supervised learning of Places from Range Data using AdaBoost. In: *proceedings of international conference on robotics and automation*.
18. **A Conference Paper:** Zender H, Martinez-Mozos O, Jensfelte P, et al. (2008) Conceptual spatial representations for indoor mobile robots. *Robotics and Automation Systems* 56(6): 493:502.
19. **A Conference Paper:** Rotmann A, Martinez-Mozos O, Stachniss C, et al. (2005) Semantic Place classification of indoor environments with mobile robots using boosting. In: *proceedings of the national conference on artificial intelligence*, pp. 1306-1311.