Flinders University

College of Science & Engineering

Master Thesis ENGR 9700(A-D)

Evaluation of Bio-inspired Pre-processing to Improve Object Classification

(Thesis submitted to the College of Science and Engineering in partial fulfilment of the requirements for the degree of Master of Engineering(Electronics) at Flinders University, Adelaide, Australia)

Name: Ditish Maharjan (2226435)

Supervisor: Dr. Russell Brinkworth

Submission Date: 02/06/2021

Contents

List of Figures

List of Tables

Acronyms

HDR High Dynamic Range

LDR Low Dynamic Range

MV Machine Vision

CV Computer Vision

TMO Tone Mapping Operator

HVS Human Vision System

VSS Visual System Simulator

SRP Scene Reproduction

BSQ Best Subjective Quality

bpp bits per pixel

OCR Optical Character Recognition

CCD Charge-Coupled Device

CMOS Complementary Metal Oxide Semiconductor

MOS Metal Oxide Semiconductor

ADC Analog to Digital Converter

CNN Convolution Neural Network

iTMO inverse Tone Mapping Operator

LPF Low Pass Filter

SNR Signal to Noise Ratio

RGB Red Green Blue

SIFT Scale-invariant feature transform

HOG Histogram of oriented gradients

SVM Support vector machine

TanH Hyperbolic tangent function

ReLU Rectified linear unit

YOLO You Only Look Once

R-CNN Region Based Convolution Neural Network

HD High Defination

RPN Region Proposal Network

FPN Feature Pyramid Network

R-FCN Region based Fully Convolutional Network

RoI Region of Interest

AP Average precision

mAP Mean average precision

IoU Intersection of Union

TP True positive

FP False positive

TN True negative

FN False negative

AUC Area under curve

GFTT Good Features To Track

SURF Speeded Up Robust Features

FAST Features from Accelerated Segment Test

FLANN Fast Library for Approximate Nearest Neigbour Search

RR repeatability rate

UniSA University of South Australia

ACST Australian Central Standard Time

fps frames per second

CVAT Computer Vision Annotation Tool

ORB Oriented FAST and Rotated BRIEF

FAST Features from Accelerated Segment Test

BRIEF Binary Robust Independent Elementary Features

YOLO You Only Look Once

SSD Singe Shot MultiBox Detector

AR Average Recall

mAR Mean Average Recall

BF Brute Force

FLANN Fast Library for Approximate Nearest Neighbors

Executive Summary

High Dynamic Range (HDR) imaging helps to overcome the limited dynamic range of traditional imaging system, i.e. Low Dynamic Range (LDR) imaging. LDR imaging systems are capable of capturing less than three orders of dynamic information of a scene. Because of this limited dynamic range, LDR imaging systems cannot capture details in scenes with both dark and very bright regions in a single scene. This has been a limiting factor for implementing the LDR imaging system in Machine Vision (MV) applications such as agriculture, surveillance, autonomous driving.

The limitation of traditional imaging systems does not affect HDR imaging with its extended dynamic range. However, HDR imaging for its increased information content requires more storage, longer transfer time and computation power for its use of floating-point data to represent the dynamic range compared to 8-bit LDR images. Tone Mapping Operator (TMO) are used to dynamically compress the higher dynamic range of HDR images to 8-bit LDR images while still preserving some details. While there are multiple state-of-the-art TMOs available for such purpose, most of them have been designed with subjective metrics used for human consumption. There is a novel TMO designed with metrics to improve noise suppression, enhance image contrast and edge detection and reduce image flicker based on a biological inspiration from blowfly called bio-inspired TMO. The metrics used for bio-inspired TMO are focused on information content rather than artistic recreation for human consumption and hence more suited for MV applications.

This thesis is undertaken to evaluate the performance of bio-inspired TMO MV application of Object classification and Localisation on dynamic images. For evaluation, multiple datasets were captured in normal and low light condition on different camera setup using three HDR cameras: Monochrome, Colour and Low-Resolution Colour camera. The HDR images captured for each case were pre-processed using bio-inspired TMO to create PRC datasets. For comparison, the captured HDR images were also used to create LDR datasets by applying a gamma correction of 2.0 followed by histogram equalisation to create RAW datasets. Each of these datasets were annotated to create ground truth data. Faster R-CNN was used as the object detector to generate predictions on the datasets. These predictions were compared with the ground truth annotation data using evaluation metrics of PASCAL VOC.

Upon evaluating the results, it showed that on overall bio-inspired TMO helped to increased object detector's performance. However, this increment was enjoyed only by datasets captured using Monochrome cameras. The gain for low resolution colour camera was marginal while in case of colour cameras the performance of object detector was found to decrease when using bio-inspired pre-processed datasets. Overall, applying bio-inspired TMO on datasets captured by Monochrome cameras showed the maximum improvement in low light conditions, however the improvements were minimal in normal lighting conditions.

Declaration

I, Ditish Maharjan, acknowledge in accordance with the Flinders University's policy on plagiarism that the content of this report is of my own and nobody else's.

Acknowledgements

I would like to express my sincere gratitude and thanks to my supervisor Dr. Russell Brinkworth for his support, motivation, guidance, knowledge and patience. I am thankful to the School of Engineering at Flinders University for it's assistance and support.

I would also like to thank Paulo Santos for his guidance, suggestion and knowledge from lectures which were of tremendous help during the thesis.

Lastly, I am thankful to my family and friends for their continuous support.

Chapter 1

Introduction

1.1 Project Background

The application of Machine Vision (MV) extends to various fields such as factory automation, agriculture, autonomous driving, surveillance, object detection and vehicle tracking. All these applications are subjected to a wide range of illumination conditions. The reliability of MV technology depends on the ability of the imaging system to adapt against such changing lighting condition.

The traditional imaging system, i.e. LDR can capture roughly three orders [8] of the dynamic range of light from a scene. This limits the ability of LDR images to capture details of a scene when exposed to a wide range of lighting conditions. The range of lighting conditions refers to low light and extremely bright light conditions and a combination of both resulting in bright and dark regions. Such limitations of LDR imaging significantly impact its use in MV applications where the details are compromised due to change in lighting conditions. HDR imaging partially overcomes these limitations by combining multiple images taken with different exposure times into a single image, thereby preserving details of a scene when exposed to a wide range of illumination conditions.

LDR imaging to represent the dynamic range of light in a scene uses fixed integers of 8 bits per pixel (bpp). Comparatively, HDR images employ floating-point numbers to represent the dynamic range of light in a scene [9]. HDR imaging can represent the entire 12 orders [8] of the dynamic range of light. The increased information in HDR images requires more storage, memory per pixel, which increases the time for computation and transmission [10; 11]. Furthermore, most display devices are built for 8-bit data and are incompatible with floating data format of HDR [10]. These factors severely limit the use of HDR imaging for MV applications. However, Tone Mapping Operator (TMO) [12], which dynamically compresses the floating-point data format of HDR image to the fixed

8-bit data format of LDR image while still preserving the details of original scene can be utilised.

While most TMOs are designed with intent of artistic recreation of the original scene fit for human consumption using subjective metrics such as Visual System Simulator (VSS), Scene Reproduction (SRP), and Best Subjective Quality (BSQ) [13; 14]. A novel bio-inspired TMO developed by Griffiths [15] was designed with non-subjective novel metrics such as motion artefacts, noise suppression and flicker for MV applications. However, the bio-inspired TMO [15] has only been implemented for static images and lacks comparisons in its implementation in dynamic settings where either the scene or camera itself is in motion.

Henceforth, this thesis evaluates the performance of bio-inspired TMO pre-processed images in dynamic settings for the MV application of object classification and localisation. The settings for evaluation include two dynamic settings: when the camera is static, and the objects in the scene are in motion and when the camera and object both are in motion. Furthermore, each scene has been captured by three cameras: Monochrome, Colour and Low-Resolution Colour camera for comparison of the improvement in Normal and Low light conditions. For the evaluation, multiple bio-inspired TMO pre-processed, and non-processed LDR image datasets are compared for increased object detection using a two-stage object detector Faster R-CNN [5].

1.2 Goal and Objectives

This thesis aims to extend the evaluation of bio-inspired TMO on the dynamic sequence of images by analysing the performance of an Object detector for object classification and localisation on datasets pre-processed with bio-inspired TMO and exposed to Normal and Low illumination conditions. Besides this, the thesis evaluates the performance of Object detector on datasets captured using Monochrome, Colour and a row based multi-exposure Colour (Low Resolution) cameras.

Chapter 2

Background Theory

2.1 Low Dynamic Range (LDR) and High Dynamic Range (HDR) Imaging

2.1.1 Introduction

Dynamic range is the ratio of the largest and smallest values of a quantity under measurement. In terms of the imaging system or photography, the dynamic range of light is the ratio of maximum (bright) and minimum (dark) measurable light intensities of a scene. Photography is the process of recording a scene by capturing all of the various intensities of light in the scene. The light-capturing process is done by using sensors such as Charge-Coupled Device (CCD), Complementary Metal Oxide Semiconductor (CMOS) or by exposing light onto photosensitive material. While the process of capturing an image has become more accessible, the goal of photography has always been to capture the dynamic range of light and the features of the scene while minimising the difference between the captured image and the actual scene [16]. However, traditional commercial cameras are limited in their ability to capture the full dynamic range of light. It was realised early on details of a scene with complicated lighting, i.e. increased dynamic range due to the presence of a bright (light source) and a darker object, could not be captured by exposing the scene only once because of the limited dynamic range of the photosensitive material. Based on this realisation, to capture the full dynamic range of certain scene with complicated lighting condition, HDR images have been developed as early as the 1850s by French photographer Gustave Le Gray. He developed the HDR image by exposing the sky and sea in two monochromatic negatives and combining them as seen in Figure 2.1.

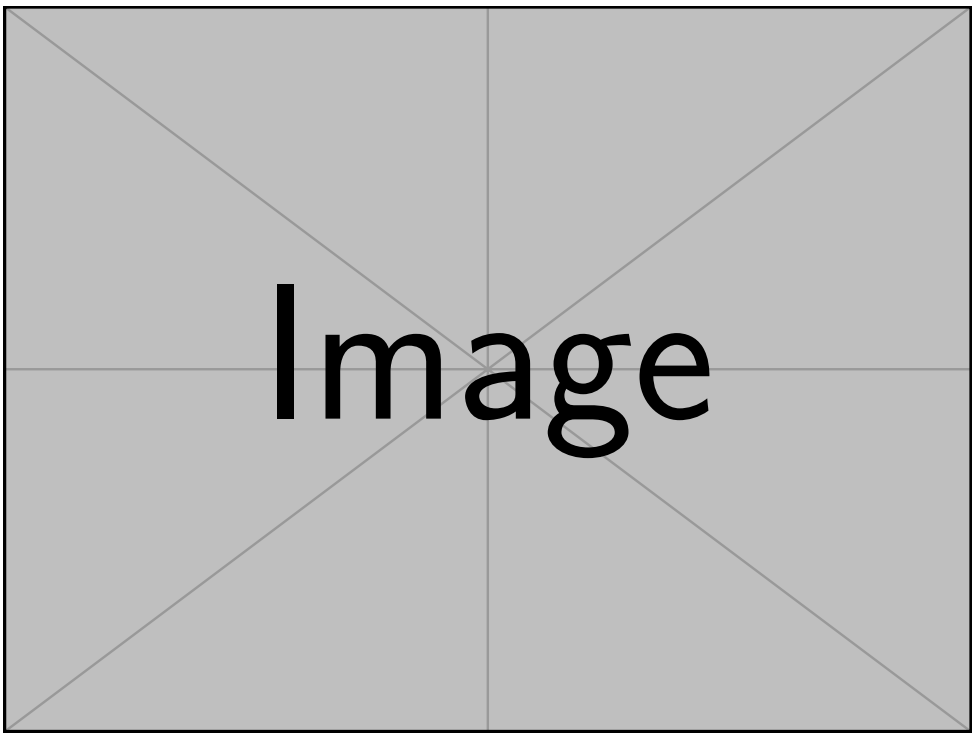


Figure 2.1: "The Brig" taken in 1856 by Gustave Le Gray is one of the first known HDR images [1]

In this thesis, LDR images are defined as images where the dynamic range of light is represented in 256 levels using an 8-bit fixed-integer per pixel. And HDR images are defined as images with a higher dynamic range of light represented as 12-bit floating data per pixel.

2.1.2 HDR Image Creation

Using multiple LDR images, each with different exposure is one of the commonly used ways to create an HDR image. Some of the other methods for creating HDR images are discussed below:

Film Based HDR

The creation of HDR images started as early as 1856 when Gustave created an HDR image titled "The Brig" by combining two different negative films exposed to capture the dynamic range of cloud and sea [1]. The process was manual and resulted in an image as seen in Figure 2.1 where the dynamic range of sea and cloud has been captured in one image.

Charles Wyckoff did the other significant development in film based HDR images for the US Department of Air Force. His patent [17] revealed that by adding multiple emulsion

layers over the silver-halide film where each layer of the emulsion was sensitive to a different wavelength of light. Multiple scene exposures were then simultaneously captured using different exposure rate, which was used to construct a high dynamic image. His method did not require long exposure times and could capture dynamic scenes [17; 15]. This technique was used to photograph nuclear explosions [18].

Single Exposure Based HDR

A departure from the use of multiple images exposed differently to obtain a HDR image was the use of a single image with one exposure to construct a HDR image. Fernandez-Berni et al. [19] developed such single-capture HDR technique that can capture a wide range of illumination. This method used standard CMOS image sensor for image capture. The local adaptation derived as suggested by [20], where the pixel value was derived from its neighbouring pixel's mean value, which enhanced image details. This value was reduced based on the global average illumination value in natural vision system [21]. The advantage of this method was that a single exposure was sufficient to generate an HDR image and saw a reduction of motion blur compared to the multi-exposure technique.

Besides the above technique, certain high-end modern CMOS cameras can capture a greater dynamic range using a singe-shot exposure method [22].

Single Exposure using Deep Learning-Based HDR

HDR images can be generated from a single low dynamic range LDR input by employing a deep CNN to estimate information lost due to saturation of camera sensor such as bright, dark parts of image [23]. This method of using a single exposed LDR for generating HDR is referred to as inverse Tone Mapping Operator (iTMO). While most iTMO cannot reproduce the saturated pixels to produce visually convincing HDR images, this approach was able to do so. The base of this approach is a CNN, which has been trained on a large dataset of HDR images that were augmented for simulation sensor saturation. The approach was limited in its ability to reconstruct missing information in images with more extensive dynamic ranges.

Row Based Multi-exposure

This method is a mixture of single and multiple exposure method for HDR creation. Instead of using multiple images, this method captures a single image in which exposure time is changed line by line or row by row. This results in alternating rows with short and long exposure times. These alternating short and long exposure rows are then merged

to generate an HDR image [24]. The image generated by such a method, however, has reduced resolution because of the merger.

Multiple-Exposure Based HDR

Mann and Picard [9] attempted to construct HDR image by taking multiple images with different exposures and combining them. The steps of this process are [9; 25]:

- Multiple images are taken from a camera at a fixed location with multiple exposures
- Camera response curve estimated from multiple exposed images using self-calibration
- Images are linearised by using inverse of response curve
- Linearised images are merged

These are the basic procedure for capturing HDR image using multiple-exposure and used in many conventional algorithms. Over the years, Ward [26] presented a new method to align images and Kang et al. [27] proposed gradient-based optical flow estimation. These helped reduce the problem of the motion of object or camera shake and ghosting artefacts [27] because of image combination [26].

2.2 Tone Mapping

2.2.1 Introduction

Tone mapping is the process of rendering high contrast and wide colour gamut scenes to a limited contrast and colour representation for a destination medium. Most display devices only support 8 bpp images, i.e. LDR and are unable to display the HDR content. HDR display technologies do exist, but these are expensive, and they have their limitations in dynamic range. And even for displaying HDR content in HDR displays some form of tone-mapping is required [13].

Most Computer Vision (CV) algorithms have been designed primarily of LDR content in mind, and switching to floating-point HDR can reduce their performance or be incompatible [28]. Besides this, compared to LDR, using HDR content requires at least four times more storage capacity, transmission bandwidth and computation-time. All these factors significantly hamper implementing a HDR system on an embedded platform.

These situations could be mitigated by using tone-mapping techniques. The tone-mapped content, although reduced in the dynamic range compared to the original HDR content, still holds details that could not have been achieved had such content been captured using standard LDR [29; 30].

2.2.2 Categories of TMO

TMOs based on how the image is processed can be classified into two types:

1. **Global Operators (Spatially Uniform)** — Global TMOs use a single, non-linear and spatially uniform mapping function for all the pixels in a image [31]. Once an optimal function value is determined for an image, the same non-linear transformation is applied to all the pixels in that image [31]. Because of this, global TMOs are simple to implement and are faster [32] than local TMOs. The downside of such an operation is the loss of details in an image, such as contrast level. Typically, Global TMOs are used for performing operations such as contrast reduction and colour inversion.

2. **Local Operators (Spatially Varying)** — In Local TMO the operator parameter is non-linear, spatially varying for each pixel based on the pixel values of its local neighbourhood [33]. Compared to global TMOs, local TMOs are complex and slower. Since the pixel's spatial position influences the mapping operation, much

more details are preserved in the output image than a global operator. The local TMOs if applied correctly, produces visually pleasing images.

The TMOs besides categorisation based on the type of mapping on pixels of an image, can be classified based on the intent [14] of use into three types:

1. **Visual System Simulator (VSS)** — VSSs are designed to simulate biological property and limitation of a Human Vision System (HVS). A VSS TMO can simulate colour saturation, contrast limitation, limited vision in night and add glare [14]. VSS in-effect try to adjust a real-world scene of high dynamic range to suitable viewing condition for HVS.

2. **SRP Operators** — SRP TMOs are designed to preserve the visual appearance of the original scene as much as possible. Some of the factors that it tries to preserve are contrast, colour gamut, luminance and sharpness [14].

3. **BSQ Operators** — BSQ TMOs are designed to focus on the visual aesthetics rather than accurate scene reproduction [14]. The reproduced scene are made based on subjective preference. This TMO is mostly used for artistic application since the BSQ's preference may change depending upon the project or situation it is used for.

2.3 Bio-Inspired TMO

2.3.1 Introduction

A human eye can perceive about twelve orders of the dynamic range of light, allowing a person to see both during the day with high light level and at night with moonlight at starlight [8]. Similarly, the insect's visual system can adapt up to eight orders of magnitude [34] of the dynamic range of light while preventing saturation and simultaneously enhancing local contrast [35; 36]. Because of their simple neurological system, such biological capability of vision-system can be modelled and implemented as analog systems capable of real-time calculations [37]. The influence of such biological inspiration or bi-mimicry can be seen field of soft-robotics [38], motion detectors [39], obstacle avoidance [37] and many others [36].

Owing to such capability of biological systems, HVS has been used to develop biologically inspired TMO [40; 41]. However, such TMOs are geared towards VSS fit for human eyes and do not take into account the features required for MV. Also, the simplicity of the insect vision system compared to HVS makes them ideal for real-time applications.

The bio-inspired TMO [15] being evaluated in this paper is based on the photo-receptor model of blow-fly [42]. The photo-receptors are the biological equivalent of image sensors, converting the incoming photons into electricity. The mathematical model of the blowfly was initially proposed by [34], further elaborated by [43; 42] upon which the bio-inspired TMO was developed by [15]. The model [34; 43; 42] when used has shown benefits in MV application such as motion detection [44; 45]. Besides this, the model has shown benefits in compression [43] of incoming data.

2.3.2 Bio-inspired TMO algorithm

The bio-inspired TMO consists of four main stages, with an additional pre-processing stage for normalising and post-processing for rescaling. The pipeline for creating a bio-inspired tone-mapped LDR image is shown in Figure 2.2. The shown pipeline is for a single channel image and has to be repeated for the remaining colour channels in a colour image. For the colour images captured in this thesis, the HDR images in RGB colour format was used. All the colour channels were individually tone-mapped, combined and scaled to produce a tone-mapped LDR colour image.

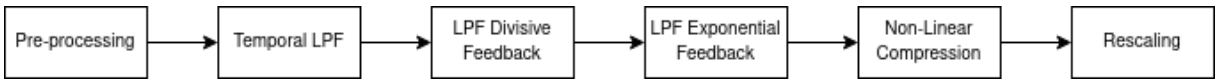


Figure 2.2: Pipeline of bio-inspired TMO

1. **Pre-processing** — In this stage, the pixels in the input image with an illumination value of zero are re-initialised to the smallest non-zero value in the image, and the luminance values are normalised to a range of 0 to 1.

2. **Temporal LPF** — This stage applies an adaptive LPF on the normalised image to remove short-lived, high-frequency values. This stage ensures SNR on dark areas of the image after tone-mapping is not amplified by reducing dark regions' response to high-frequency noises.

3. **LPF Divisive Feedback** — The square root of the input "Temporally LPF'd Image" is used to initialise the Adaptive LPF. This LPF is responsible for filtering out the high frequency components in the image. The input to the LPF is either "Supress Transients" function which implements square root or the output of previous frame as feedback. And finally the "Temporally LPF'd Image" is divided by the LPF'd image which compresses the dynamic range. The main intention of this block is to incorporate short-term adaption so that the TMO can adapt to rapid shifts in light intensity due to change in light source or motion.

4. **LPF Exponential Feedback** — This stage is similar in structure to the previous stage, but it involves adapting the input image to slow change in lighting source. The division of the input "Divisively Filtered Image" with its own exponent causes non-linear re-scaling of the image which further amplifies the high frequency components.

5. **Non-linear Compression** — In this stage, for suppressing pixels with high value Naka-Rushton transform [46] is applied followed by a gamma function to produce tone-mapped LDR image with floating-point representation. The Naka-Ruston transform is applied to all the pixels dividing the input image by sum of itself and a offset.

6. **Rescaling** — The floating-point data of the image are rescaled to a range for 0-255 for storing the image in 8-bit formats.

2.3.3 Evaluation Metrics

The bio-inspired TMO was compared with six other TMOs Reinhard02 [47], Reinhard05 [48], Drago03 [49], Ward94 [50], and Stockham72 [51] using proposed novel metrics such as Flicker, Motion Artefacts and Noise Suppression. These metrics focused on the information content of the images which is essential for any MV application rather than aesthetics of the image or suitability for consumption by human eyes.

The metric flicker accounts for short-lived high-frequency changes in pixel intensities that can cause unbalanced images and inconsistent images. It is crucial to suppress and

not create additional flicker as removing flicker post-production is challenging and tedious [52; 53].

Noise suppression metric gives the measure of image quality after it has been tone-mapped. It is the measure of Signal to Noise Ratio (SNR) in the image. It is desirable to have low noise and the TMO being able to suppress some amount of noise. Noise suppression metric considers the SNR before and after an image is tone-mapped.

Furthermore the motion artefacts metric focuses on increased information in the image for feature detection such as edges and contrast due to motion.

2.3.4 Bio-Inspired TMO's Result and Research Gap

Griffiths [15] found that bio-inspired TMO outperformed other TMOs in noise suppression. Bio-inspired TMO was able to improve signal quality for SNR $\leq$ 20dB and for SNR at 40dB maintain the image quality at 96.5%. In terms of motion artefacts, bio-inspired TMO increased global contrast in an image by 30% and increased edge detection. Lastly, in terms of metric flicker, Reinhard02 outperformed bio-inspired TMO. In short, Bio-inspired TMO was able to enhance certain features such as edge detection, contrast, reduce noise and flicker in images which are vital for the application of MV algorithms such as object detectors.

However, all these evaluations were done for static images in a controlled environment. To fully evaluate the benefits of bio-inspired TMO, it is vital to evaluate it for a sequence of images in dynamic setting for verifying the temporal improvements in flicker, noise suppression and motion artefacts as suggested by Griffiths [15]. Besides, this the bio-inspired TMO has not been implemented for tasks related to MV. Hence, this thesis evaluates the performance of bio-inspired TMO for MV application of object classification on frames of images captured in various lighting condition in dynamic settings that is similar to surveillance and autonomous vehicle.

2.4 Annotation and Image processing techniques for annotation transfer

Different image processing techniques were explored for identifying objects in two set of images for transferring the annotated data. Image registration technique finds the identitical features in two set of images and computes a transformation matrix to align the images, which in turn can be used to transfer boudning box co-ordinate of annotated data. While template matching searches for an input template over the target image computing difference between template and a section of target image to find the least difference for a match.

2.4.1 Annotation

Annotation in CV for images refers to adding metadata for certain parts of an image in a dataset. The metadata in image annotation is primarily a label or class name associated with the object or feature being marked in the image and its location in the image itself. The label is often the name of the object being identified. The location of the object is often marked using a tight-fit rectangular box. While polygon, Cuboids are also used to mark the location of an object.

Annotation is used to make an organised dataset with metadata information that can be used to train, evaluate and test computer vision algorithms. The annotated dataset is also known as ground truth and is used to compare the results of a predicted output. While a pre-trained object detector can be used to annotate a dataset, the result of such a process is often incomplete and unreliable. Hence, annotation is done manually, and since this is the ground truth data, the images have to be labelled correctly and bounding boxes drawn accurately. While this data can be stored in a text file in any format, certain standards have been introduced, such as PASCAL VOC [54], and COCO [55] being the most popular. The ground truth data is stored in a JSON file in COCO standard where the image name, id, tagged objects, and location are stored as arrays of data. In comparison, PASCAL VOC creates individual XML files for each image in the dataset with all associated metadata.

For increasing efficeicy of manually annotation certain tools such as Computer Vision Annotation Tool (CVAT) [56], MATLAB's Ground Truth Labeler, LabelMe [57], LabelImg [58] are available. Besides MATLAB all of the other tools are open-source.

2.4.2 Image Registration

Image registration [59; 60] is the process of transforming multiple images from different sources at a different angle of a common scene to align them for analysis. Image registration is typically used to combine information from multiple sources and compile them into a single helpful image for analysis. Image registration for aligning images applies geometric transformations such as translation, rotation, shearing, scaling. The application of image registration ranges from changing perspective view of an image to Optical Character Recognition (OCR) to medical imaging [59] to military uses [59].

Image registration techniques can be classified into two categories[59; 60] Intensity-based, Feature-based Image Registration. The working process of both these techniques is similar and can be summarised into three steps [61]:

1. Identify features or intensity pattern in source and target image
2. Apply similarity metrics to determine the quality of matching in the source image.
3. Compute transformation matrix to apply an appropriate transformation to achieve alignment of the source image to the target image.

Intensity Based Image Registration

Intensity-based techniques compare the intensity pattern of pixels in the source and target image to compute the required transformation. The identified intensity patterns are compared using similarity metrics [62; 63; 64] such as Sum of Squared Differences, Cross-Correlation, Mutual Information and absolute difference.

Feature Based Image Registration

Feature-based techniques identify distinct features or key-points such as points, corners, lines in source and target image and maps the key-points. Feature detector and descriptor algorithms are employed for identifying features in an image. Some feature detector algorithms and feature matchers are:

1. Scale-invariant feature transform (SIFT) [65] — It is a feature detector used to detect features and identify them in an image. SIFT is invarient to image size, orientation, brightness changes.

2. Speeded Up Robust Features (SURF) [66] — SURF is a feature detector and descriptor inspired by SIFT. SURF has similar advtages offered by SIFT and is several times faster than SIFT.

3. Oriented FAST and Rotated BRIEF (ORB) [67] — ORB is a open source feature detector and descriptor developed by OpenCV by combining Features from Accelerated Segment Test (FAST) keypoint detector and Binary Robust Independent Elementary Features (BRIEF) descriptor. It is found to be invariant to rotation and robust to noise.

4. Brute Force (BF) Matcher [68] — BF Matcher is a simple feature matching technique that takes the description of features generated by feature detectors for two images and matches these features based on distance calculation. The technique has little optimisation and involves comparing the features each feature in first image with every mother feature in second image. The technique has been implemented in OpenCV and as per the documentation for distance measurement when using SIFT, SURF cv2.NORM_L2 is used while for ORB cv2.NORM_HAMMING is used.

Intensity-based techniques have the advantage of being able to achieve sub-pixel accuracy compared to feature-based techniques as it considers all the pixels of an image. However, the advantage of intensity-based methods is limited to cases when the source and target image both have been recorded in similar lighting condition [60]. The feature-based techniques are immune to this and perform better when registering images shot under different lighting conditions [60].

2.4.3 Template Matching

Template matching [69] is the process of searching and finding the location of a template, such as a face, cars, and other smaller objects in a larger image. The smaller object referred to as template is moved across the larger image to calculate the similarity between them comparison methods such as Squared Difference (TM_SQDIFF), Cross-Correlation (TM_CCORR), Correlation Coefficient (TM_CCOEFF) and their normalised versions for template matching [69]. As the template is moved across the image, each patch of the image is processed using the above methods. After the whole image has been searched, the patch in the image that yields the least difference is identified as the patch area that consists of the searched template.

2.5 Object Detection

2.5.1 Introduction

An object detector is a CV algorithm that deals with locating and identifying certain classes of objects in an image or video. In essence, an object detector combines two sub-tasks of object classification and object localisation [70; 71]. The term object classification is used interchangeably with image classification, and its aims to identify the contents or objects of interest in an image and identify which class the objects belong to, such as person, car, dog, cat. Furthermore, the task of object localisation identifies the location of detected instances of classes of objects and marks them by drawing a tight bounding box around each identified instance. Today most of the object detectors are implemented using deep learning [71] but before this hand-crafted CV algorithms such as SIFT [65], Histogram of oriented gradients (HOG) [72] were used.

Feature detector algorithm such as SIFT identifies distinct features, key-points, in an image which are cross-checked with a database of images to identify the required matches [73]. Likewise, HOG is a feature descriptor algorithm that divides an image into smaller squared cells, compute histograms of oriented gradients in each, normalise the image and return descriptors of each cell. The extracted features of each cell are passed to a machine learning algorithm such as Support vector machine (SVM) [72] for object detection. These algorithms have performed well in tasks such as pedestrian detection [74; 72], face detection [73] but greatly suffers for detection of generic objects [75]. The key-points and descriptor from SIFT and HOG are vector data. The number of vectors increases as the number of features to be detected increases, and so does the memory and processing time/power required to process it [72]. One more difficulty with this approach is choosing which features are important for each given task [75]. The traditional hand-craft methods for object detection have eventually been replaced with deep learning.

2.5.2 Deep Learning

Deep learning is a subset of machine learning technique made using artificial neural networks to mimic the structure and function of the human brain. Like the human brain, the neural network of deep learning is composed of numerous neurons, each of these neurons performs a simple task, and through interaction, they make a decision [76; 2; 77]. The term 'deep' refers to hidden layers in the neural network. It is referred as hidden, as it lies between input and output layers of a neural network. Fully-connect layer is one of the common type of hidden layers. In hidden layers, none of the neurons in the same hidden layer are connected but each neuron is connected to neurons on adjacent layers. The

number of hidden layers can range from 2-3 in a traditional neural network to more than 100 in deep networks [2; 77]. The hidden layers as in essence are layers of mathematical functions designed to process input data to produce a output for a intended result.

The main advantage of Deep Learning algorithm is that they learn by example. Instead of using a comprehensive mathematical model used in traditional programming paradigm to perform a task, i.e. descriptive analysis [75], the machine learning algorithms use predictive analysis [75] whereby the error between the actual and predicted outcome is minimised, taking into consideration all possible factors [78]. Machine learning algorithms learn by being fed a large number of training data from which it learns to identify new data [79; 75].

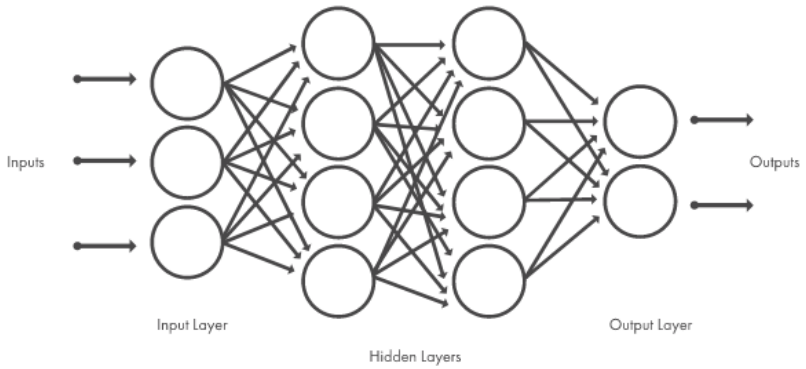


Figure 2.3: A simple neural network with 2 hidden layers [2]

Machine Learning algorithms require a model's input data to be organised or formatted [77]. In terms of an image, relevant features must be selected manually for performing object detection tasks. While on the other hand, the representational learning method [80] is included in deep learning, allowing deep learning methods to automatically ingest and process unstructured data like text and images [77]. It can automate the manual feature extraction process in machine learning. With enough training and data, such deep learning algorithms can automatically learn to detect and classify objects from given raw data automatically [2; 80]. Figure 2.4 gives an overview of the difference in approach between machine and deep learning for performing a task such as object classification.

Two important terminologies in a neural network are weight and bias. These parameters are associated with neurons, and these are learnable parameters. When the neural network is trained to perform a task these parameters are set during the training phase to produce a prediction that has minimum error. For a neuron A that receives input I with weight w and bias b has an output Y as shown in Equation (2.1). The computed value is then fed through an activation function which is a mathematical function that compares the result to a threshold value which decide whether to pass the output value to the next neuron i.e. activate neuron or not.

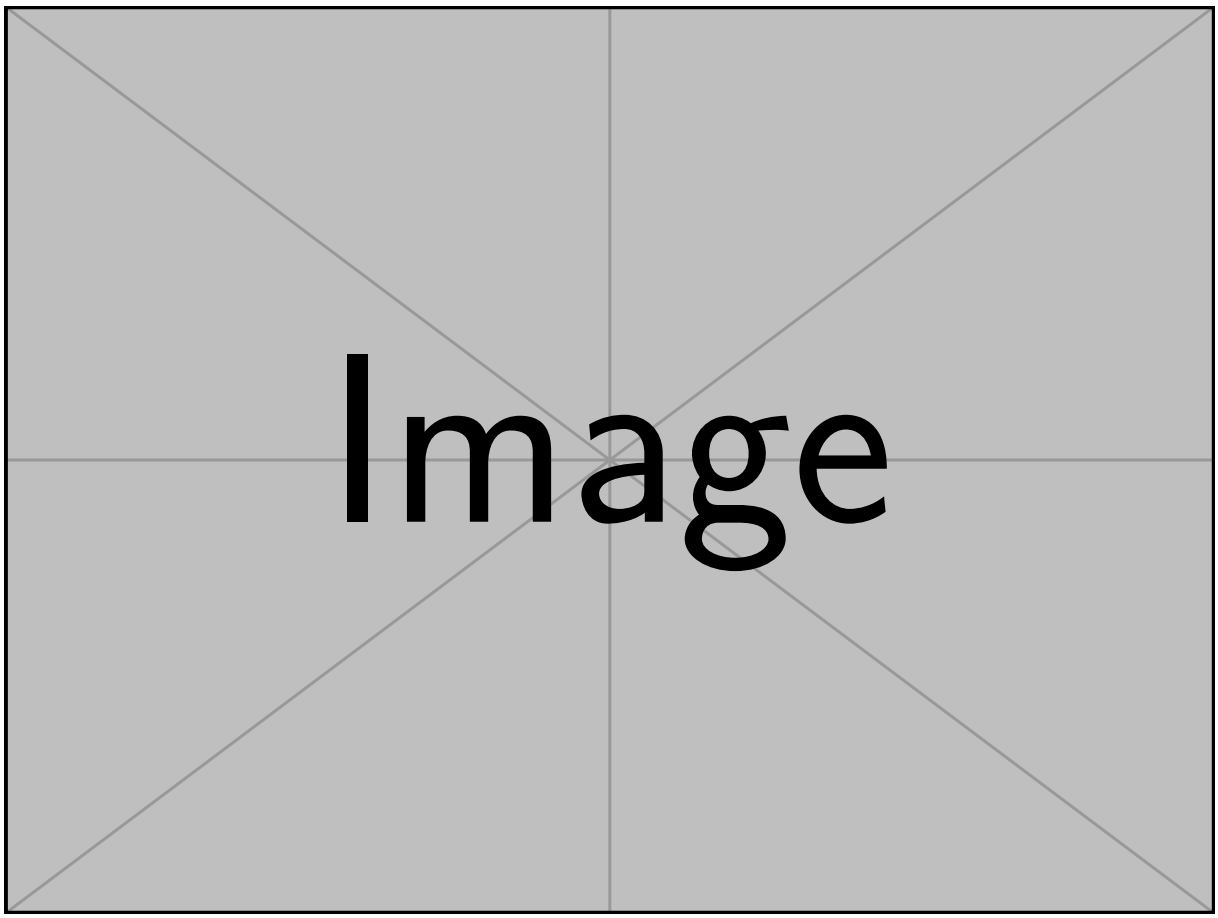


Figure 2.4: Comparative approach for vehicle classification in Machine Learning and Deep Learning [2]

$$Y = \sum (w * I) + b \tag{2.1}$$

As shown in Figure 2.3, deep neural networks consist of two visible layers, input and output. The input layer performs the task of loading data for processing, and the output layer performs the final prediction. The hidden layers are responsible for performing various functions as required to produce the desired result. The behaviour of the hidden layers are set during the training phase of the neural network. As data is read in input and passes through the layers towards the output, hidden layers perform some computation based on weight and bias associated with neurons on each layer. This propagation of data through the network is known as forward propagation [80]. While during the training phase, the error of the network is minimised by adjusting weights and bias during which prorogation is from output to input is known as back propagation [80]. With the implementation of forward and back propagation, a neural network can make predictions and correct any errors over time. The accuracy hence increases over time gradually.

2.5.3 Convolution Neural Network(CNN)

CNN is one of the most popular architecture for the implementation of deep learning. The development of CNN in recent years has pushed the ability of object detectors to new limits [81]. The visual cortex of animals [82; 83] inspired CNN. CNN automatically and adaptively learns to identify features and patterns from data with grid patterns such as images [3]. CNN is typically composed of three building blocks [3; 75; 84]:

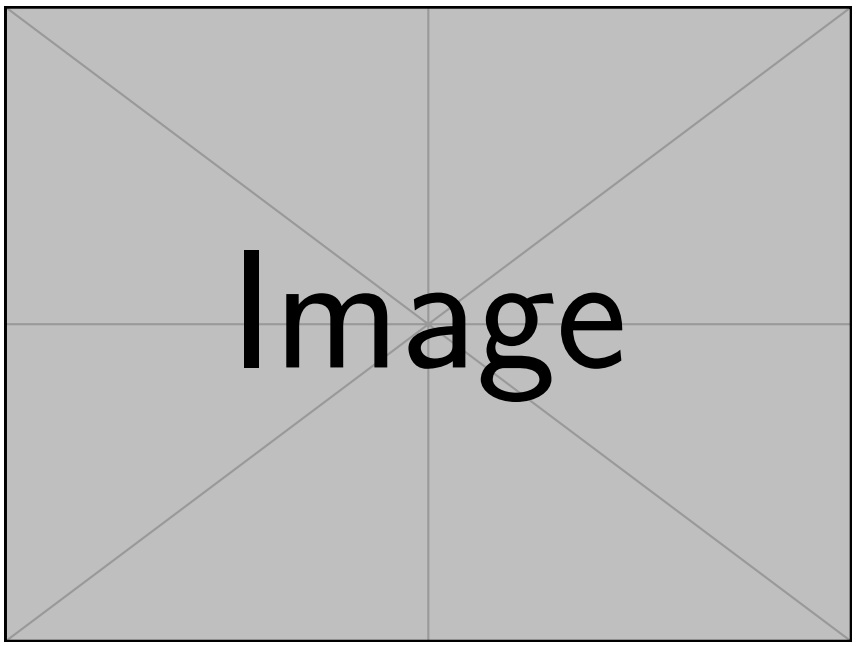


Figure 2.5: An overview of Convolution Neural Network (CNN) architecture[3] showing forward propagation for prediction and back propagation for training

1. **Convolution Layer**
   This layer extracts features from an input array of data, i.e. images using Convolution and Activation Function.

   - **Convolution** — A small matrix, typically 3x3 called a kernel, is applied across the input image called a tensor. Convolution is an element-wise product between overlapping kernel and input tensor elements as the kernel is moved across the input tensor, is calculated and summed. The summed value is stored in a corresponding position in the output tensor called feature map. The values in the kernel matrix called weights are assigned during the training phase.
   - **Activation Function** — The feature map generated through convolution is passed through a non-linear activation function such as sigmoid or Hyperbolic tangent function (TanH) or Rectified linear unit (ReLU). Depending upon the requirement or task at hand, one of these activation functions are selected [85].

2. **Pooling Layer/Region of Interest (RoI) pooling**
   The pooling layer down-samples the size of the feature map generated in the convolutional layer while still preserving the extracted features. This helps to reduce the memory consumed by feature-map, thereby decreasing the computational power and time required for processing the data for both prediction and training. For pooling, typically, a 2x2 kernel size is used.

   There are type pooling methods available max pooling and global average pooling. In max pooling, from a patch in feature map window of size 4x4 maximum value is extracted. Similarly, average pooling instead of maximum value returns the average of values.

3. **Fully Connected Layer**
   A fully connected layer flattens the feature matrix into a 1-D vector. This feature vector is passed through the fully connected layer to an output layer that computes the probability of occurrence for a list of classes through a dense network. This output is then passed through a function such as ReLU or softmax, which maps them to a vector whose sum is equal to one.

2.5.4 CNN Based Object Detection Models

One of the first CNN based object detectors was Overfeat [86]. Using a multi-scale sliding window approach Overfeat was able to perform image classification, localisation and detection. This was quickly followed by Region Based Convolution Neural Network (R-CNN) detectors [87], You Only Look Once (YOLO) [88] and others. All of these CNN based Object detection models can be classified into two categories [89]:

1. **Two-Stage Detectors** A two-stage detector in the first stage identifies regions of interest and generates region or object proposals using methods such as Region Proposal Network (RPN), or selective search [90]. In the second stage, the proposals are passed through to identify objects of interest, classification, and bounding box drawn over them, regression. Two-stage detectors are highly accurate but slower. Object detectors R-CNN [87], Fast R-CNN [91], Faster R-CNN [5], Mask R-CNN [92], Feature Pyramid Network (FPN) [93], Region based Fully Convolutional Network (R-FCN) [94] fall under this cateogry.

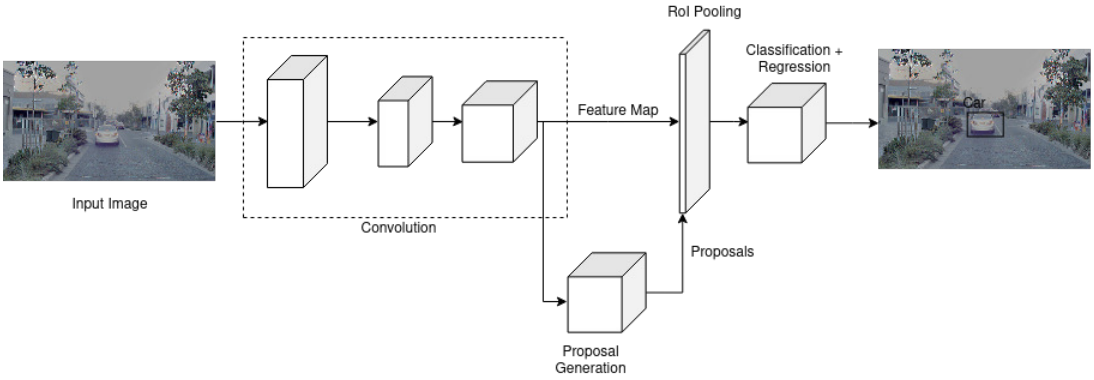


Figure 2.6: Basic architecture of two-stage detectors

Fast R-CNN

Fast R-CNN [91] was released in 2015 a year later after the release of R-CNN [87]. R-CNN was slower because each region proposal made for an image was passed through the CNN without any means for sharing the computation. This made training and detection using R-CNN slower.

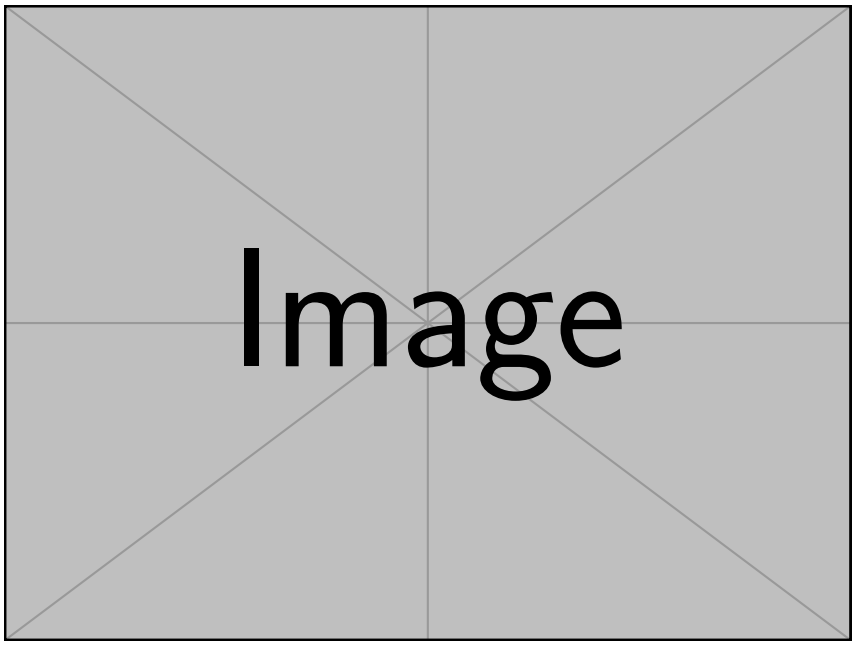


Figure 2.7: Basic Architecture of R-CNN [4]

In Fast R-CNN, an image is passed through the CNN to generate Regions of Interest or feature maps. Fast-RCNN uses selective search to generate the proposals. This proposal is passed through an RoI Pooling layer to resize the proposed regions to be of the same size. These regions are then passed through a Fully Connected Layer which classifies these regions and draws bounding boxes on them.

The RoI pooling layer in Fast R-CNN was an improvement to extract a fixed-sized feature map from a region proposal of different sizes as this no longer required

wrapping regions. One of the significant improvement compared to R-CNN was reduction in training time by 88.61%[89].

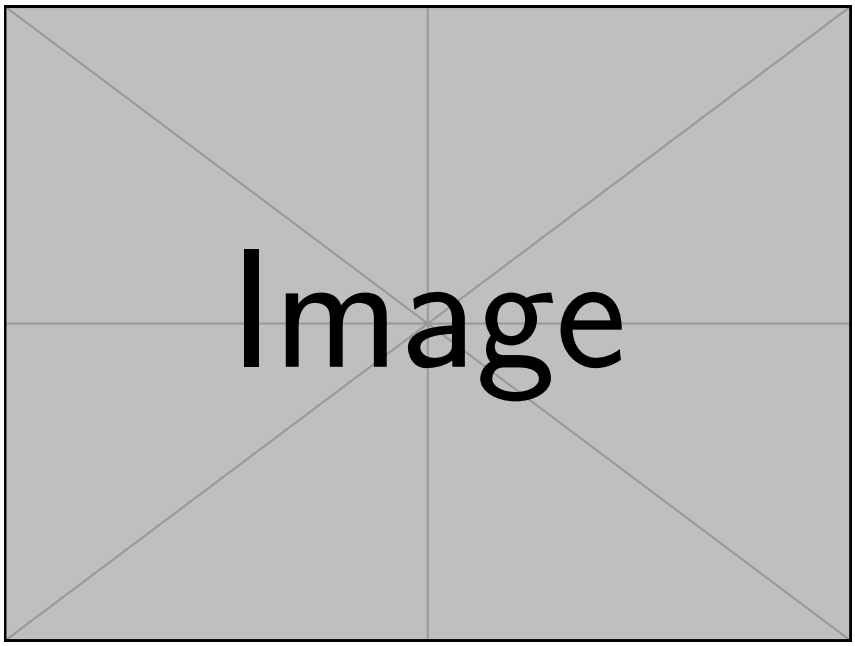


Figure 2.8: Basic Architecture of Fast-RCNN [4]

Faster R-CNN

Faster R-CNN [5] released few moths after Fast R-CNN [91] introduced RPN to replace selective search. Selective search in Fast R-CNN is used for generating RoI proposal was slow and required run-time equal to the detection network [89]. The proposed RPN is a fully convolution network that is faster and increased the efficiency of the proposal generation. The addition of RPN has introduced limited downtime in the network is nearly cost-free [5]. In reference to the basic architecture of faster R-CNN in Figure 2.9, the workings of Faster R-CNN can be summarised in the following steps [4; 89; 5]:

(a) The input image is passed through the Convolution Network, which extracts features from the image and generates a feature map or RoI.

(b) RPN is applied on the feature map to generate object proposals, each with an objectness score and are sent to RoI pooling layer.

(c) RoI pooling layer applies max or average pooling to down-sample all the proposals.

(d) The proposals are then passed down to a fully connected layer where objects are classified, and a bounding box is drawn over them.

[5; 89] have shown that Faster R-CNN has increased speed and accuracy in object detection compared to Fast R-CNN. Experiments [89] showed on PASCAL VOC 2007 dataset [54], Faster R-CNN scored Mean average precision (mAP) of 69.9%

compared to 66.9% score of Fast R-CNN. Similarly, the run-time of Faster R-CNN was 198ms, nearly ten times faster than Fast R-CNN, which was 1830ms.

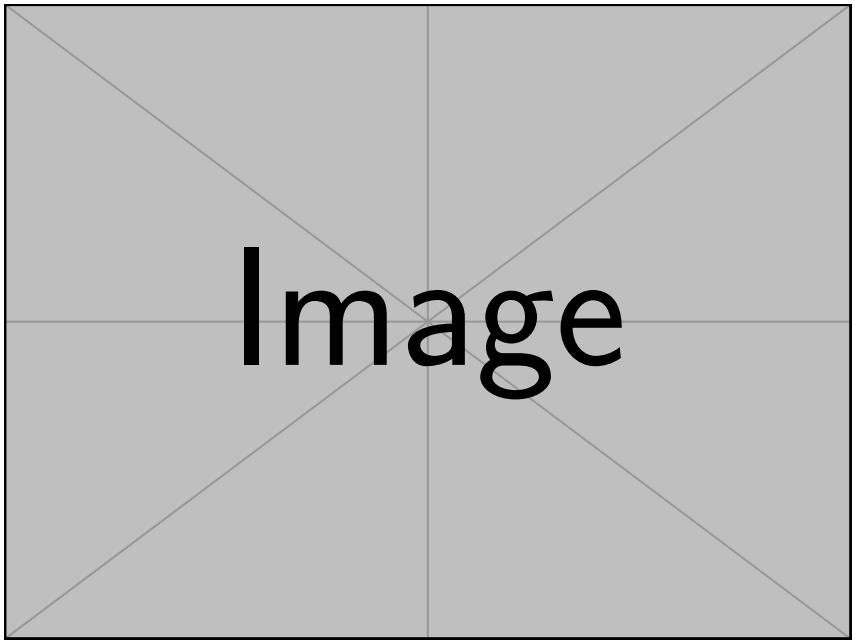


Figure 2.9: Architecture of Faster R-CNN [5]

2. **One-Stage Detectors**

   One-stage detectors perform classification and regression in one step. These detectors are faster than two-stage detectors but have lower accuracy rates. YOLO [88], SSD [6], YOLOv2 [95] and YOLOv3 [96] are some one stage object detectors.

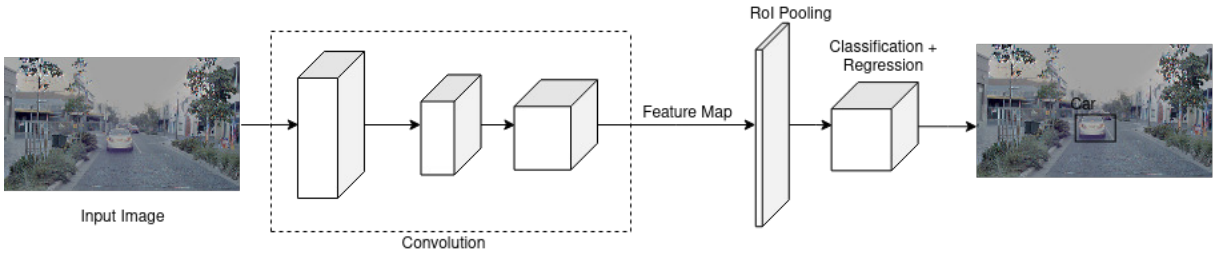


Figure 2.10: Basic architecture of one-stage detectors

SSD

Liu et al. [6] proposed SSD in 2016 as a one-stage detector. The single-shot means the task of classification and localisation is done in a single step, Multibox [97] is a bounding box regression technique, and Detector as the network performs classification of the detected objects. SSD predicts multiple objects of different categories using a collection of fixed-sized bounding boxes of different scales and confidence score at each location in the generated feature map followed by a non-maximum suppression.

The network layer of SSD is based upon VGG-16 architecture with the fully connected layers removed and is known as the base network. Multiple auxiliary layers are added to the network to produce detections with the following features:

- Multi-Scale Detection — The convolutional layers after the base network are progressively decreasing in size. This allows multiple predictions to be made on a feature map at multiple scales.
- Convolutional predictions — SSD, instead of a dedicated region proposal network, uses a small convolution filter to produce a set of predictions. These predictions are made after the feature map is extracted using a 3 x 3 convolution filter.
- Default boxes and aspect ratios — Each feature map cell is associated with a set of default fixed-sized bounding boxes. In each feature map cell, the offset relative to the default box bounding box is predicted along with confidence scores. These boxes are applied to several feature maps of different resolution to efficiently decide the best possible bounding box size for a prediction.

[89] have shown SSD for an input image size of 512 x 512 achieved an mAP of 81.6% on PASCAL VOC 2007 test dataset and 80.0% on PASCAL VOC 2012 test dataset compared to Faster R-CNN's 78.8% and 75.9%..

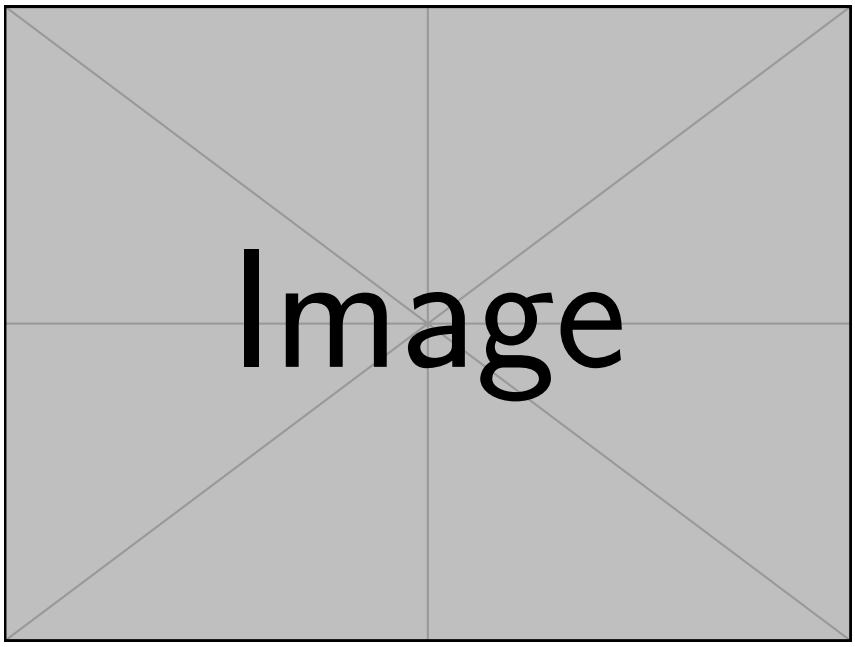


Figure 2.11: Basic architecture of Singe Shot MultiBox Detector (SSD) [6]

2.5.5 Choosing Object Detector for Evaluation

While comparisons made by Jiao et al. [89] states SSD to be more precise and faster than other detectors such as Fast R-CNN and Faster R-CNN when tested under PASCAL

VOC dataset. COCO dataset consists of objects that are harder for detections Jiao et al. [89]. When evaluated on COCO 2017 train dataset, Faster R-CNN with a ResNet backbone outperformed both SSD detectors using ResNet and VGG-16 backboneJiao et al. [89]. While single-stage detectors are faster than two-stage detector, the accuracy of both the detector types are similar, and if speed were not a factor for discussion Faster R-CNN would outperform the single-stage detector SSD by a small margin Huang et al. [7]; Hui [98]. Huang et al. [7] analysed the effect of image size on detectors and found higher resolution images lead to better mAP result for smaller objects and larger objects for most detector models. However, SSD models were found to be doing well on larger object not smaller. Furthermore, it was found that by decreasing the number of proposals in Faster R-CNN from 300 to 50, the speed could be increased 3x while suffering a decrease in accuracy by the only 4%. For this thesis determining the improvement in the performance of object detector when using a dataset processed by bio-inspired TMO is the main goal and real-time processing is not. While the benefit of faster processing of dataset is available with SSD models, SSD was found to not take advantage of higher resolution image for detection of smaller objects and hence Faster R-CNN was selected as the object detector for the evaluation.

2.6 Evaluation Metrics

2.6.1 Introduction

For an object detector, its performance is evaluated by comparing the predictions made with the actual truth. While accuracy may come to mind for evaluation, it is not a good metric for evaluation when working with datasets that have class-imbalanced data. For such evaluation of class-imbalanced dataset, precision and recall are good evaluation metrics. Precision gives a measure of all the positive predictions made how many are correct, while recall tells of all the things that are true how many of them were correctly predicted and identified. Based on these two, PASCAL VOC [54], Open Images [99], COCO [55], Imagenet [100] are some of the metrics used for evaluation of object detectors. Before introducing these metrics, it is essential to clarify basic concepts related to these metrics.

2.6.2 Basic Concepts

- **Intersection of Union (IoU)**
  Consider an image has a class of object over which bounding box is draw and represented by G for Ground Truth, and an object detector predicts the presence of an object with a bounding box represented by P for Prediction. Then, IoU is given by the following Equation (2.2) and this is illustrated in Figure 2.12. IoU gives the difference between ground truth and predictions in values that are less than 1.

$$IoU = \frac{area\,(G \cap P)}{area\,(G \cup P)} = \frac{area\ of\ intersection}{area\ of\ union} \tag{2.2}$$

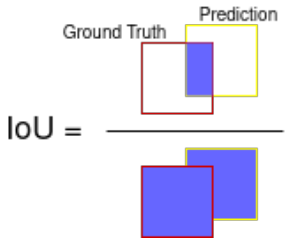


Figure 2.12: Illustration of IoU

- **True positive (TP), True negative (TN), False positive (FP), False negative (FN)**
  When predictions are made besides the bounding box a score or confidence level is assigned to every predictions, indicating the confidence score that detector thinks it the said class of object.

For a prediction to be considered True positive (TP) all of the following conditions must be satisfied:

1. The predicted class must match the ground truth
2. The confidence score of the prediction must be above a threshold score
3. IoU must be greater than a threshold (0.5 in PASCAL VOC)

If the prediction violates the first or third condition, the prediction is False positive (FP). This indicates that a false prediction has been made.

If the ground truth has not been detected, then such undetected items are counted as False negative (FN). True negative (TN) is not considered during the evaluation.

- **Precision (Pr)**
  Precision is the representation of accuracy of positive detentions.

$$Pr = \frac{\sum TP}{\sum TP + \sum FP} = \frac{\sum TP}{all\ detections} \tag{2.3}$$

- **Recall (Rc)**
  Recall is the ratio of positive detection that are correctly detected.

$$Rc = \frac{\sum TP}{\sum TP + \sum FN} = \frac{\sum TP}{all\ ground\ truths} \tag{2.4}$$

- **Precision x Recall**
  When the threshold of confidence level is increases, the number of FP will decrease which increases the precision level but the recall decreases. To account for this limitation where increasing precision decreases recall and vice-versa, a number of values for precision and recall have to be taken with different thresholds.

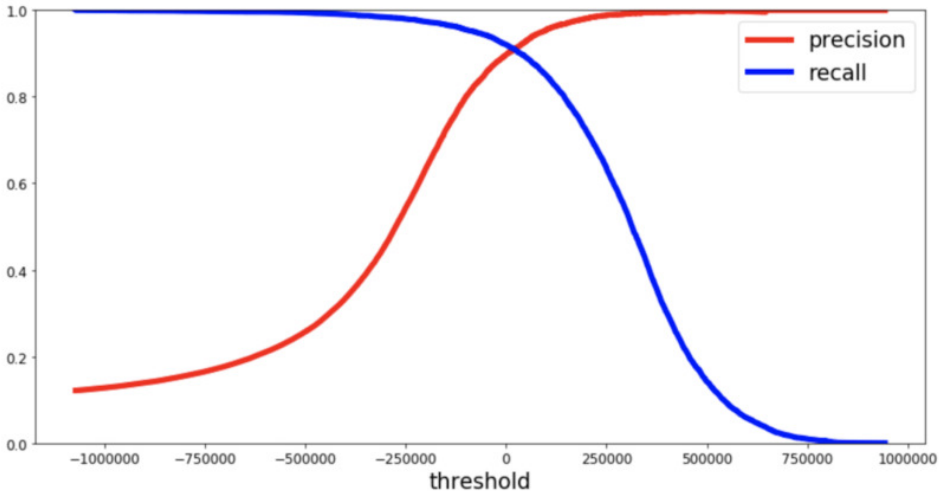


Figure 2.13: Precision x Recall curve

- **Average precision (AP)**
  AP [54] is calculated by summing up Area under curve (AUC) of the precision-recall curve. AP in essence is the numerical averaged representation of precision x recall curve for comparing performance of detectors.

  AP is the area under the Pr x Rc curve using K recall values,

$$AP = \sum_{k=0}^{K} (R_r(k) - R_r(k+1))\, Pr_{interp}(R_r(k)) \tag{2.5}$$

  Here, the value of $Pr_{interp}(R)$ is the maximum precision value with recall $R_c(k)$ where $R_c(k) \geq R$

$$Pr_{interp}(R) = \max_{Rc(k) \geq R} Pr(k) \tag{2.6}$$

- **Mean average precision (mAP)**
  mAP [54] is the mean of all the APs of all the classes (C).

$$mAP = \frac{1}{C} \sum_{i=1}^{C} AP_i \tag{2.7}$$

- **Average Recall (AR)**
  Similar to AP, AR [55] is a metric used to measure assertiveness of object detectors for a given class. Unlike AP confidence score of estimated detections are not taken in to account for calculating AR. This metric taken into account all recall values

within IoU threshold limit of [0.5,1]. AR can be calculated by using the following formula:

$$AR = 2 * \int_{0.5}^{1} Rc_{IoU(o)} do \tag{2.8}$$

Where o is IoU, $Rc_{IoU(o)}$ is a function that retreives recall Rc value for given IoU o from a recall-IoU curve.

- **Mean Average Recall (mAR)**
  mAR [55] is defined as mean AR across all classes C.

$$mAR = \frac{1}{C} \sum_{i=1}^{C} AR_i \tag{2.9}$$

2.6.3 PASCAL VOC Metrics

Pascal VOC metric [54] for benchmarking the performance of Object Detectors uses the following metrics:

1. AP with IoU threshold of 0.5
   AP under this metric is calculated for each class individually by using the Pr x Rc curve to determine the AUC using Equation (2.5). The IoU threshold for this metric is 0.5.

2. mAP with IoU threshold of 0.5
   Using the AP calculated with IoU threshold of 0.5 for all individual classes, these values are summed and averaged using Equation (2.7). While AP is used to represent precision score for individual classes, mAP is used for representing and comparing the performances level of the object detectors and is the primary metric for comparison in PASCAL VOC.

2.6.4 COCO Metrics

In the COCO metric [55], the mAP and mAR values are mentioned as AP and AR, respectively. Unlike PASCAL VOC, the individual classes are not taken into account

1. *AP@0.5 and AP@0.75*
   The metric AP@0.5 is similar to mAP from PASCAL VOC, which also has an IoU threshold of 0.5. AP@0.75 is also similar and differs only in regards to the IoU threshold of 0.75. AP@0.75 is a strict metric compared to AP@0.5.

2. *AP@[0.5:0.05:0.95]*
   This metric expands upon the previous AP@0.5 and AP@0.75 metric by computing multiple AP from 0.5 to 0.95 at an incremental step of 0.05 and taking an average of the computed APs. This is the primary metric used for comparison in the COCO metric.

3. AP_s, AP_m, AP_l
   These metrics also refereed as AP across scales produces AP for smaller, medium and large sized objects based by applying AP@[0.5:0.05:0.95] and taking its average. The area under consideration is the ground truth bounding box area.

   (a) AP_s — This metric represents AP for smaller objects in ground truth whose area $< 32^2$ pixels.

   (b) AP_m — This metric evalulates medium sized objects whose $32^2 <$ area $< 96^2$ pixels.

   (c) AP_l — This metric evalulates large sized objects whose area $> 96^2$ pixels.

4. AR_1, AR_{10}, AR_{100}
   These metrics are evaluated by limiting the number of detections per image. Here, AR is calculated for a fixed number of detections per image and is averaged over all classes and IoUs for thresholds [0.5:0.05:0.95].

   (a) AR_1 — This metric is calculated considering one detection per image.

   (b) AR_{10} — It considers 10 detections per image.

   (c) AR_{100} — It considers 100 detections per image

5. AR_s, AR_m, AR_l
   These metrics are a variant of AP_s, AP_m, AP_l but for recall values for different sized objects in ground truth.

2.6.5 Choosing Evaluation Metric for Evaluation

For the evaluation for better results the larger dataset was used and to save some time, annotations (ground truth) data had to been transferred from one dataset to the other when possible. During the transfer because of varying distance of an object with respect of each camera and difference in resolution between datasets, there will be some errors resulting in bounding boxes being places at slightly shifted location than intended. The process used for bounding box transfer have been explained in detail in Section 4.2.

The AP score given by COCO metric is based on varying the IoU threshold value from 0.5 to 0.95. While the actual intention of COCO metric for varying the threshold

was to account for bias in evaluation because of predictions with lower IoU and higher IoU values. But in this case, this intention might lead to classifying most TP data as FP because of imperfect bounding box transfer. Hence, COCO metric while used a fixed IoU threshold of 0.5 was selected as the metric for evaluation of the results.

Chapter 3

Literature Review

The inability of the traditional Low Dynamic Range (LDR) imaging system to capture details of a scene exposed to a wide range of dynamic range of light has been identified as the limiting factor hindering the implementation of Machine Vision (MV) [101] in fields of surveillance [102; 103], autonomous navigation [104], agriculture [105] and automation [101]. High Dynamic Range (HDR) imaging can account for the wide range of change in illumination because of its use of floating-point to represent a dynamic range of light instead of 8 bpp of LDR. Several studies and experiments have been carried out to prove the benefit of HDR and LDR in changing illumination conditions. Some of these experiments have been discussed below.

An experiment by Chermak and Aouf [104] had compared the performance of HDR imaging sensor (Aptina MT9M024) with a HD digital camera on the basis number of feature detection and matching under illumination condition that extends from indoor to outdoor, with direct sun exposure, dark and low light conditions. For feature detection Scale-invariant feature transform (SIFT) [65], Harris corners [106], Good Features To Track (GFTT) [107], SURF [66] and FAST [108] were used and for feature matching SIFT and SURF. Result of the experiment [104] showed HDR imaging sensor in extreme lighting condition was able to produce 2.45 to 29.35 times more matches.

Chermak and Aouf [104] was more focused on using HDR image for comparison with LDR image for improvement in feature detection but did not use any tone-mapped images. In comparison, an experiment by Přibyl et al. [109] focused on improving feature detection in HDR datasets made using Global TMO and Local TMO. For the LDR image dataset, this experiment had a different dataset: a filtered data using Wallis filter [110]. Wallis filter was used to pre-process the LDR images as it had shown improvement in feature detection in some experiments [111; 40]. The dataset for [109] was captured in an indoor setting on 2D images that were purpose-built consisting of dark, bright areas. Also, 3D images were captured of multiple purpose-built scenes from various angles and distance.

For the lighting condition, more variations were made to include extreme lighting changes for the dataset. Compared to [104]'s dataset, the lighting condition is more complicated to capture the most extreme lighting condition. The feature detector used were Harris corner [106], Features from Accelerated Segment Test (FAST) [108], Shi-Tomasi or Good Features To Track (GFTT) [107], Fast Hessian or Speeded Up Robust Features (SURF) [66]. Compared to Chermak and Aouf [104] which used the ratio of feature matched as the metric for comparison, Přibyl et al. [109] made comparison based on the repeatability rate (RR) of FP detectors. RR is the standard metric for Feature point detectors and is the ratio of the number of feature points detected in an image to the number of feature points in the reference image. The results showed TMO using local operator improved RR by 19% for 2D scenes while TMO using global operator improved RR by 15% for 3D scene. The result of this experiment are highly dependent on the lighting gradient of the scene to draw conclusion on which operator performed better. However, the results do show use of HDR imaging has helped to increase the performance of FP detector.

The results from experiments by Chermak and Aouf [104] and Přibyl et al. [109] both point to an improvement in feature detection and matching with the use of HDR imaging. However, comparison by Chermak and Aouf [104] was on HDR images with LDR, and by Přibyl et al. [109] were on local tone-mapper, global tone-mapper and LDR datasets. Still, a single experiment comparing HDR, tone-mapped images, and LDR would shed more light. Such experiment was done by Rana et al. [30]. The goal of the experiment was the same as the two experiments. The dataset for [30] was captured indoors in two setups, Project Room and Light Room. The lighting setup for the Project room was done using indoor light at different intensity, while the lighting condition for Light Room dataset was influenced by natural light. Both the dataset had frames exposed to a wide dynamic range of light. While the experiment by Přibyl et al. [109] had only two tone-mapped datasets (one local and one global), this experiment had a total of nine tone-mapped datasets (two global and seven local). The TMO used were Drago [49], Ward [50], Ashikhmin [112], Chui [113], Mantiuk [13], Fattal [114], Pattnaik [115], Reinhard [47], and Schlick [116]. While three HDR datasets were made: HDR-lin using linear luminance values, HDR-Log a log encoded HDR image, HDR-PU a perpetual uniform encoded HDR image. Like, [109] repeatability rate (RR) was used as the evaluation metric. The result of the evaluation was HDR, and Tone-mapped images performed significantly well compared to LDR images. Among the HDR formats, HDR-PU encoded HDR performed better than HDR-Lin and HDR-Log. The performance of tone-mapped images was, in most cases, on par with HDR or lower. The reduced performance was partially because of loss udring 8-bit quantisation of HDR image to LDR.

While the methodology used was different, all of these experiments [104; 109; 30] show that HDR imaging improves feature detections and tone-mapped LDR images were just as functional as HDR images for feature detection and matching. Griffiths [15] proposed

a novel bio-inspired TMO for HDR images to improve the immunity of images for MV applications. In his thesis, evaluation was made using a different set of metrics such as Noise Suppression, Motion Artefacts and Flicker reduction. For comparisons six other TMOs Reinhard02 [47], Reinhard05 [48], Drago03 [49], Ward94 [50], and Stockham72 [51] were used. Evaluation results showed that Bio-inspired TMO was able to enhance certain features such as edge detection, contrast, reduce noise and flicker in images which are vital for the application of MV algorithms such as object detectors. This further adds to the evidence that HDR images and even the tone-mapped LDR images can reduce the negative effect of changing lighting condition while still preserving details that can be used for improved feature/object detection for various applications.

While in terms of application of HDR imaging and TMOs for real-world scenarios Pinho et al. [105] performed as an experiment on datasets produced using Reinhard [47], Drago [49] and a camera's (Canon EOS 5D Mark III) embedded tone mapper to analyse improvement in agriculture particularly fruit identification and counting. Contrary to previous experiments, the feature detected for this experiment was the colour detection to identify fruits in an image covering the whole tree. The results for this was an improvement by about 30% compared to LDR images for fruit detection. Moreover, contrary to previous experiments, the results of the tone-mapped images from various TMOs were comparable with no significant difference, but the author did mention a decrease in performance because of ghosting. Likewise, further experiments were done by Wang et al. [117] for real-time vehicle signal light recognition using an HDR camera using AlexNet, a deep neural network. However, this setup was not solely relying on the camera but also on data from a LIDAR sensor. This experiment saw an improvement in detection to be 97.5% when using an HDR camera compared to LDR.

Summarising the above discussed experiments and their findings, HDR imaging can reduce the challenge faced by LDR images in extreme lighting condition. The use of HDR imaging had shown improvement in feature detection, object detection and preservation of details compared to LDR even when tone-mapped LDR images were used. While the TMO's used in the experiment discussed varied from experiment to experiment and their performance, the bio-inspired TMO shows potential as the purpose was built for Machine Vision applications.

Chapter 4

Methodology

4.1 Dataset

For evaluating the performance of Bio-inspired TMO [15], multiple footages were recorded in two different setups, Stationary and Moving, using three different cameras, MONO, LRES and COL, on two different lighting conditions; Normal and Low Light. The camera setup referred to as Stationary (STA) is when the camera is mounted on the side of a building and remains stationary throughout the recording. While camera setup Moving (MOV) refer to footage captured using cameras mounted on top of a car and moving through the streets of Adelaide CBD.

The details of the HDR cameras used for capturing the dataset are on Table 4.1. Datasets recorded using a monochrome camera are referred as MONO. LRES refers to images recorded by colour camera using row-based multi-exposure, which yielded HDR image of 12-bit depth but with a lower resolution of 960 x 538. Lastly, the datasets recorded by the colour camera with an end resolution of 1920 x 1080 are refereed as COL. The dataset was recorded at a frame rate of 50 frames per second (fps). In terms of HDR capture method used, MONO and COL cameras used a single exposure method a 12-bit CMOS image sensor.

The recorded footage from each of the cameras were used to construct two additional sub-datasets named PRC and RAW. PRC refer to tone-mapped processed images using bio-inspired TMO, and RAW refers to LDR images. The LDR images were created by passing the 12-bit HDR images through a gamma correction process with a gamma value of 2.0 followed by a histogram equalisation process. The details of cameras used for capturing the footage are on Table 4.1.

Table 4.1: Details of cameras used to capture the dataset

Camera ID	*Manufacturer*	*Model*	*Camera Resolution*	*Depth*	*bpp*	*Image Resolution*	*HDR Capture Method*
MONO	IDS	UI-3060CP-M-GL R2	1936 x 1216	Monochrome	12	1920 x 1080	Single exposure
LRES	IDS	UI-3360P-C-HQ R2	2044 x 1088	Colour	12	960 x 538	Row based multi-exposure
COL	IDS	UI-3060CP-C-HQ R2	1936 x 1216	Colour	12	1920 x 1080	Singe exposure

4.1.1 Categorisation of dataset

The main categorisation of the datasets was done based on the camera setup: Moving and Stationary. In each of these categories, the dataset was sub-categorised based on the time of recording. The footage recorded before sunset was labelled as Normal Light (NL) and after sunset as Low Light (LL). Furthermore, in each sub-category, the recordings are grouped based on the type of camera (MONO, LRES, COL) used. Finally, each of the grouped recordings consisted of a pair of image sets: PRC and RAW. The image set that has been processed using bio-inspired TMO was referred to that PRC and the other as RAW. Based on this hierarchy of grouping of the image sets, the datasets were named accordingly to identify them. For example, a dataset named $STA\_NL\_MONO\_RAW$ refers to a stationary dataset recorded in normal lighting condition using a monochrome camera (MONO), and the dataset has not been processed using bio-inspired TMO hence RAW. The complete categorisation of datasets, the name of each dataset and the number of frames in each of them can be seen in Table 4.2.

Stationary: Normal and Low Light

The datasets relating to the Stationary category was recorded on 2019/06/23 from the second floor of University of South Australia (UniSA)'s Yungondi building in Adealide CBD facing the North Terrace road. MONO, LRES and COL cameras were used to record two footage before (NL) and after sunset (LL). The sunset time was at Australian Central Standard Time (ACST) 17:52 on 2019/06/23. The footage captured on this setup is similar to a Surveillance system where camera is static and the scene is dynamic changing frame by frame

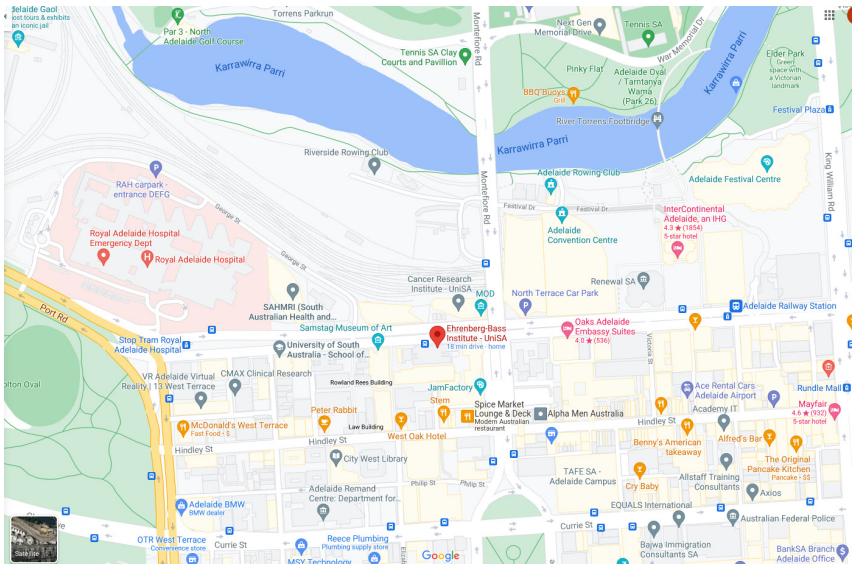


Figure 4.1: Datasets belonging to camera position: Stationary were recorded from UniSA's Yungondi building facing the North Terrace road in Adelaide CBD

(a) STA_NL_MONO_RAW

(b) STA_NL_MONO_PRC

(c) STA_NL_LRES_RAW

(d) STA_NL_LRES_PRC

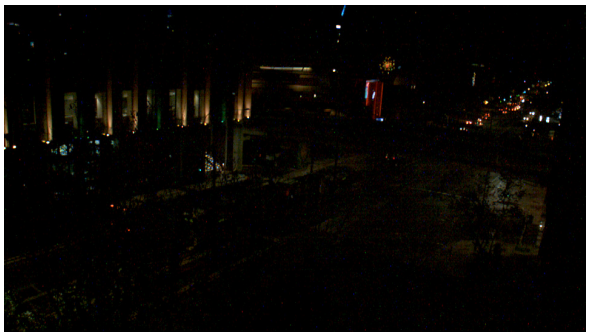
(e) STA_NL_COL_RAW

(f) STA_NL_COL_PRC

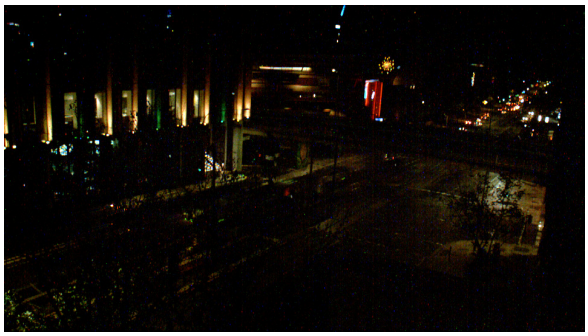
Figure 4.2: Frames (a-f) from stationary datasets under normal lighting condition, RAW on left and PRC on right for MONO, LRES and COL cameras

(a) STA_LL_MONO_RAW

(b) STA_LL_MONO_PRC

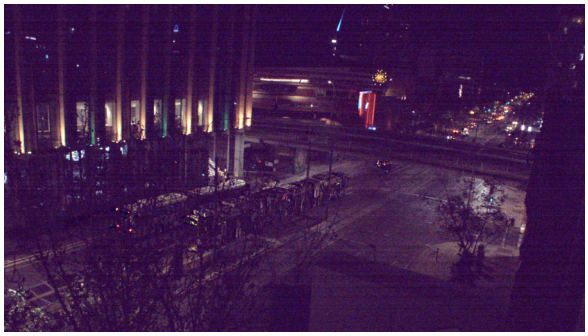


(c) STA_LL_LRES_RAW



(d) STA_LL_LRES_PRC

(e) STA_LL_COL_RAW



(f) STA_LL_COL_PRC

Figure 4.3: Frames (a-f) from stationary datasets under low lighting condition, RAW on left and PRC on right for MONO, LRES and COL cameras

Moving: Normal and Low Light

The datasets relating to the Moving category were recorded on 2019/06/26 with cameras mounted on a rig mounted on top of a car's roof. As shown in Figure 4.4, the route followed by the car is a loop around the Adelaide CBD's Hindley Street. All the footages in this category are roughly 15 minutes which at a rate of 50 fps amounts to 45000 frames. Because of limited time, only a subset of around 15000 frames was used in the thesis for evaluation. Like the stationary category, the footages were recorded before and after sunset. The time of sunset was ACST 17:53 on 2019/06/26. The footage captured on this setup is similar to a autonomous self-driving car system where both camera and the scene are in motion.

The transfer of annotations between datasets only worked for stationary datasets but

failed for moving datasets. It would have been exhaustively time consuming to have to re-annotate the datasets that were recorded at the same time from different cameras i.e. MONO, COL and LRES seperately. So, to save time LRES dataset has been omitted for Moving category. The details for annotation transfer are in Section 4.2.

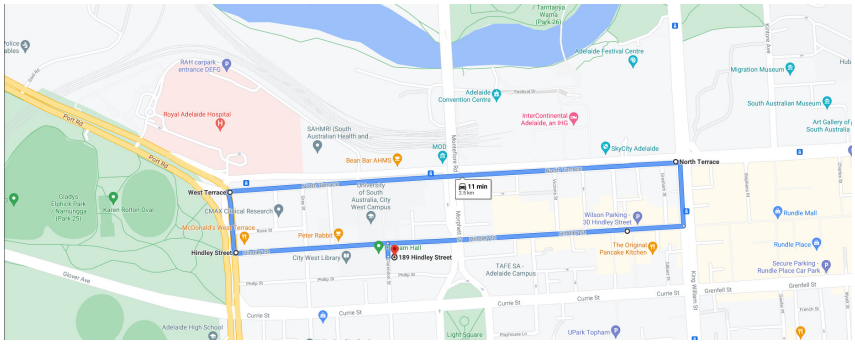


Figure 4.4: Datasets belonging to camera position: Moving were recorded on the route starting from 189 Hindley Street then through Hindley Street, King William Street, North Terrace, West Terrace and finishing back at the starting location.

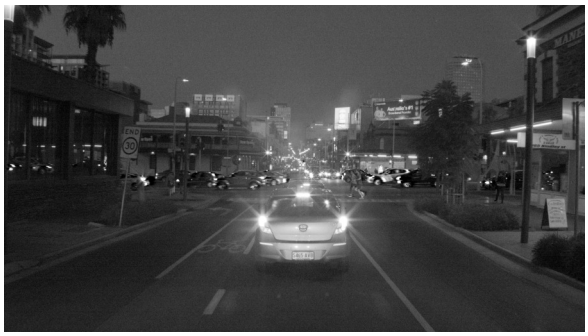
(a) MOV_NL_MONO_RAW

(b) MOV_NL_MONO_PRC

(c) MOV_NL_COL_RAW

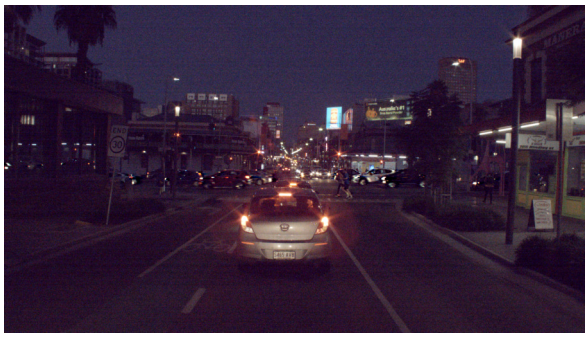
(d) MOV_NL_COL_PRC

(e) MOV_LL_MONO_RAW



(f) MOV_LL_MONO_PRC

(g) MOV_LL_COL_RAW



(h) MOV_LL_COL_PRC

Figure 4.5: Frames from moving datasets with normal light (a-d) and low light condition (e-h), RAW on left and PRC on right for MONO and COL cameras

Table 4.2: Details of categorisation of datasets, dataset naming with number of frames in each dataset

Camera Position	Time of Day	Camera	PRC/RAW	Dataset name	Number of frames
Stationary	2019/06/23; 16:46:05	CAM1	RAW	***STA_NL_MONO_RAW***	15087
			PRC	***STA_NL_MONO_PRC***	15087
		CAM3	RAW	***STA_NL_LRES_RAW***	15087
			PRC	***STA_NL_LRES_PRC***	15087
		CAM6	RAW	***STA_NL_COL_RAW***	15087
			PRC	***STA_NL_COL_PRC***	15087
	2019/06/23; 17:59:39	CAM1	RAW	***STA_LL_MONO_RAW***	15996
			PRC	***STA_LL_MONO_PRC***	15996
		CAM3	RAW	***STA_LL_LRES_RAW***	15996
			PRC	***STA_LL_LRES_PRC***	15996
		CAM6	RAW	***STA_LL_COL_RAW***	15996
			PRC	***STA_LL_COL_PRC***	15996
Moving	2019/06/26; 16:33:36	CAM1	RAW	***MOV_NL_MONO_RAW***	5000
			PRC	***MOV_NL_MONO_PRC***	5000
		CAM6	RAW	***MOV_NL_COL_RAW_#1***	5000
			PRC	***MOV_NL_COL_PRC_#1***	5000
			RAW	***MOV_NL_COL_RAW_#2***	5000
			PRC	***MOV_NL_COL_PRC_#2***	5000
			RAW	***MOV_NL_COL_RAW_#3***	6000
			PRC	***MOV_NL_COL_PRC_#3***	6000
			RAW	***MOV_NL_COL_RAW***	16000
			PRC	***MOV_NL_COL_PRC***	16000
	2019/06/26; 17:34:01	CAM1	RAW	***MOV_LL_MONO_RAW***	14215
			PRC	***MOV_LL_MONO_PRC***	14215
		CAM1	RAW	***MOV_LL_MONO_RAW_#1***	5000
			PRC	***MOV_LL_MONO_PRC_#1***	5000
			RAW	***MOV_LL_MONO_RAW_#2***	5000
			PRC	***MOV_LL_MONO_PRC_#2***	5000
			RAW	***MOV_LL_MONO_RAW_#3***	4215
			PRC	***MOV_LL_MONO_PRC_#3***	4215
		CAM6	RAW	***MOV_LL_COL_RAW***	5000
			PRC	***MOV_LL_COL_PRC***	5000

4.1.2 Datasets Alignment

While recording the datasets, the three cameras were manually turned on, resulting in different start times. This resulted in datasets with unaligned sequences of frames. Unaligned datasets would create problems later in the thesis. For aligning these datasets from different cameras to have a common starting point, all the datasets were manually inspected to find a common starting frame. The process was subjective, and several landmarks in the images were used to identify the offset.

The file names of image frames in datasets have been chronologically named in ascending order, and this was useful in sorting and aligning the images in sequential order. This naming pattern was exploited for renaming the image frames for aligning the datasets. The datasets recorded on 2019/06/23 at 16:46:05 from MONO and LRES were aligned,

but these were ahead of COL by 16 frames. To correct this, the first frame of datasets belonging to COL, i.e. the 0^{th} frame, was renamed as the 16^{th}, 1^{st} as 17^{th} and so on for the rest of the frames in the datasets.

The datasets recorded on 2019/06/23 at 17:59:39 from MONO and LRES were aligned, but COL was ahead by 18 frames. To fix this, similar to the previous step the 18^{th} frame was renamed at be the 0^{th} frame, 19^{th} as 1^{st} and so on for the rest of the images in the datasets.

In the dataset recorded on 2019/06/26, all the frames for MONO and LRES were recorded simultaneously and required no further processing for alignment.

4.2 Annotation

For ground truth annotation an open-source tool, CVAT [56] was used. The datasets were examined beforehand to identify the objects in them. The STA and MOV datasets contain footage similar to setup of surveillance and autonomus self-driving applications. So, the objects most relevant for these applications were identified and used as labels for annotating the ground truth. The labels used in the ground truth annotation for the thesis are car, bus, truck, van, bicycle, motorcycle, scooter, person, traffic_light, and train (tram). These labels were initialised in CVAT and used to create the datasets as seen in table 4.2. The annotated labels on various datasets can be seen in Figures 5.2, 5.4, 5.6 and 5.8 as the blue bounding boxes.

Manually annotating all the datasets would be a rather time-consuming and slow process. Datasets under STA_NL, STA_LL, MOV_NL and MOV_LL were recorded simultaneously, so it would be possible to transfer annotation from one of the dataset to the others within that group.

Annotation Transfer

While the initial intention was to annotate all the dataset manually, this soon proved to be a time-consuming process. Since a group of datasets were captured simultaneously and since the cameras were mounted horizontally on a rig, it would be possible by adding offsets to the coordinates of bounding boxes from a reference frame to transfer the ground truth annotations to other datasets recorded at the same time. So, initially, only one of the dataset among the group recorded at the same time was annotated with the intent of transferring the ground truth annotation to the remaining datasets once the annotation task was completed. The underlined datasets in Table 4.2 are the reference datasets

from which the ground truth annotations were transferred to other datasets recorded simultaneously.

For the pair of datasets captured using the same camera, i.e. for RAW and PRC the annotation transfer was very straight forward. The process involved renaming the file names in the ground truth annotation data to the target dataset. The COCO annotation format [55] has all the ground truth annotations in a single JSON file is the annotation format used for the thesis for transfer of ground truth annotations. Using this format for transferring annotation between RAW and PRC involved changing part of the file name for all the frames in annotation data from RAW to PRC or vice-versa.

Below are the process/techniques explored for transferring the ground truth annotations between datasets from different cameras:

1. **Measure disparity**

   This approach is simple and straight forward where the reference image was divided into multiple quadrants. Then the difference in pixels for aligning the reference and target frame on each of these quadrants was measured to calculate the offset required to align the two images, and finally, the calculated offset value was used to transfer the ground truth bounding-box from the reference frame to the target frame.

   The number of quadrants depended upon the requirement at hand. For transferring annotations from reference dataset $STA\_MONO\_NL\_PRC$ to $STA\_COL\_NL\_PRC$ and $STA\_MONO\_LL\_PRC$ to $STA\_COL\_LL\_PRC$ four quadrants were required. The offset for alignment in the above datasets for the four quadrants was measured as seen on Figure 4.6 along with the area of each quadrant.

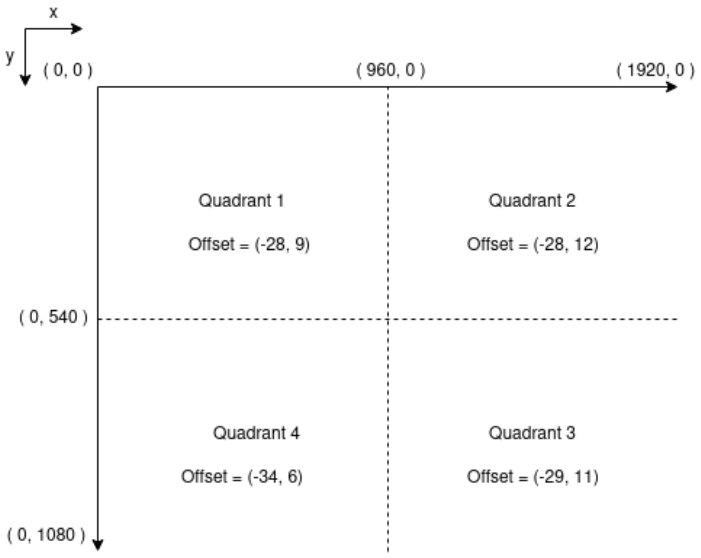


Figure 4.6: Image divided into four quadrants with offset values to transfer bounding box co-ordinates from $STA\_MONO\_NL\_PRC$, $STA\_MONO\_LL\_PRC$ to $STA\_COL\_NL\_PRC$

, $STA\_COL\_LL\_PRC$ respectively

However, for transferring the annotations from $STA\_MONO\_NL\_PRC$ to $STA\_LRES\_NL\_PRC$ and $STA\_MONOL\_LL\_PRC$ to $STA\_LRES\_LL\_PRC$ thirty-nine quadrants were required and the offset used for each of thirty-nine quadrants are shown in Figure 4.7.

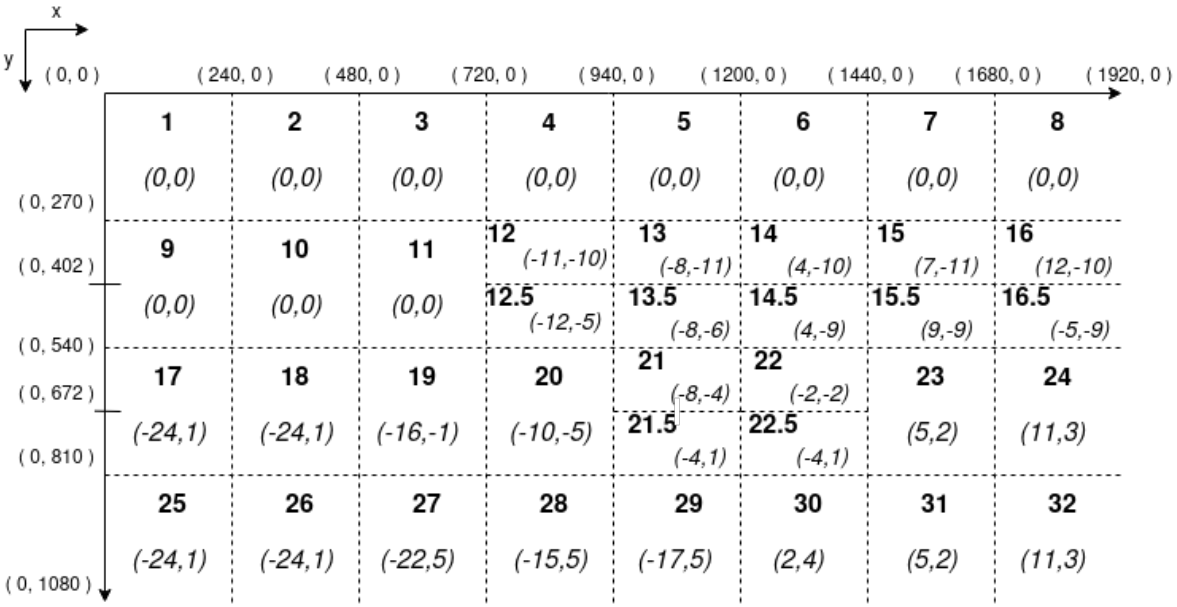


Figure 4.7: Image divided into 39 quadrants with offset values to transfer bounding box co-ordinates from STA_MONO_NL_PRC, STA_MONO_LL_PRC to STA_LRES_NL_PRC

, $STA\_LRES\_LL\_PRC$ respectively

The offset values shown in Figures 4.6 and 4.7 corrects the positional disparity an object has because of difference in the position or angle the scene was captured from MONO, LRES and COL cameras. The correction in MONO to COL datasets was relatively easier than MONO to LRES where the positional disparity is very high and required more quadrants to account for proper transfer of annotation bounding boxes.

This approach successfully transferred annotations within stationary datasets as all the objects in the scenes captured by cameras were at a relatively fixed distance at difference areas as the cameras were far away from the scene. Because of this, for each quadrant a single offset value was sufficient as the distance of that quadrant or section of image varies by only a small margin through out the whole dataset. However, in the case of MOV datasets where objects in the scene are at relatively closer distance to the cameras, the disparity between objects in images between different cameras was very high which could not be taken in account using this method and other methods had to be explored.

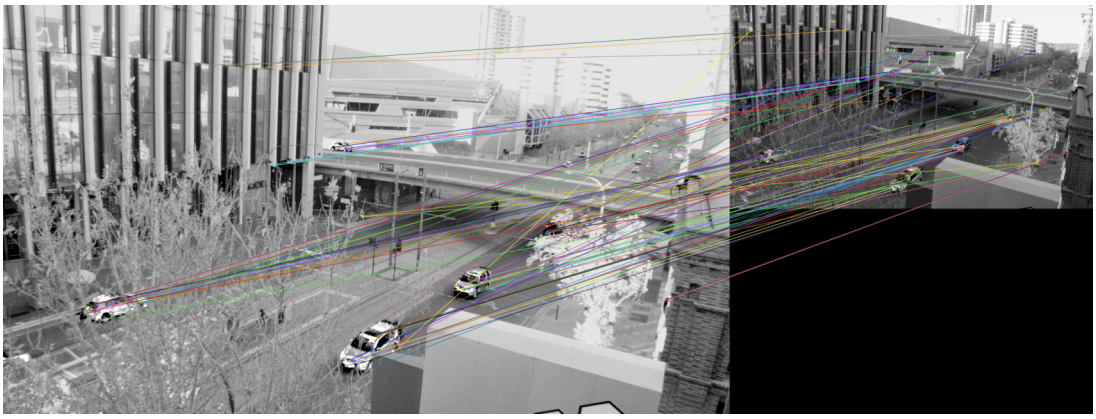
2. **Image Registration**

For the ground truth annotation transfer using Image registration, a feature detector method ORB [67] was used to identify key-points such as corners in the reference frame and the target frame. The identified key-points were the reference, and target frames were matched using a Feature Descriptor Matcher "BRUTEFORCE" in OpenCV. The matches were sorted based on their score given by the Matcher, and the ones with low scores were removed. The remaining matches were used to compute for the homography matrix. This matrix was then used to transform the reference frame to geometrically align with the target frame.

However, based on the previous attempt "Measure disparity" and its findings in Figures 4.6 and 4.7 the offset was mostly linear translation in x and y axes with not rotation. Moreover, the linear translation varied based upon the distance of the object from the camera. So, having a single matrix perform the transformation would not always as there are multiple objects at varying distances in each frame. With this realisation, this process was abandoned as the initial attempt to align a frame from the MONO dataset to the LRES dataset failed.

(a) Reference image from MONO dataset

(b) Target image from LRES dataset



(c) Keypoints identified, highlighted and linked between (a) and (b)

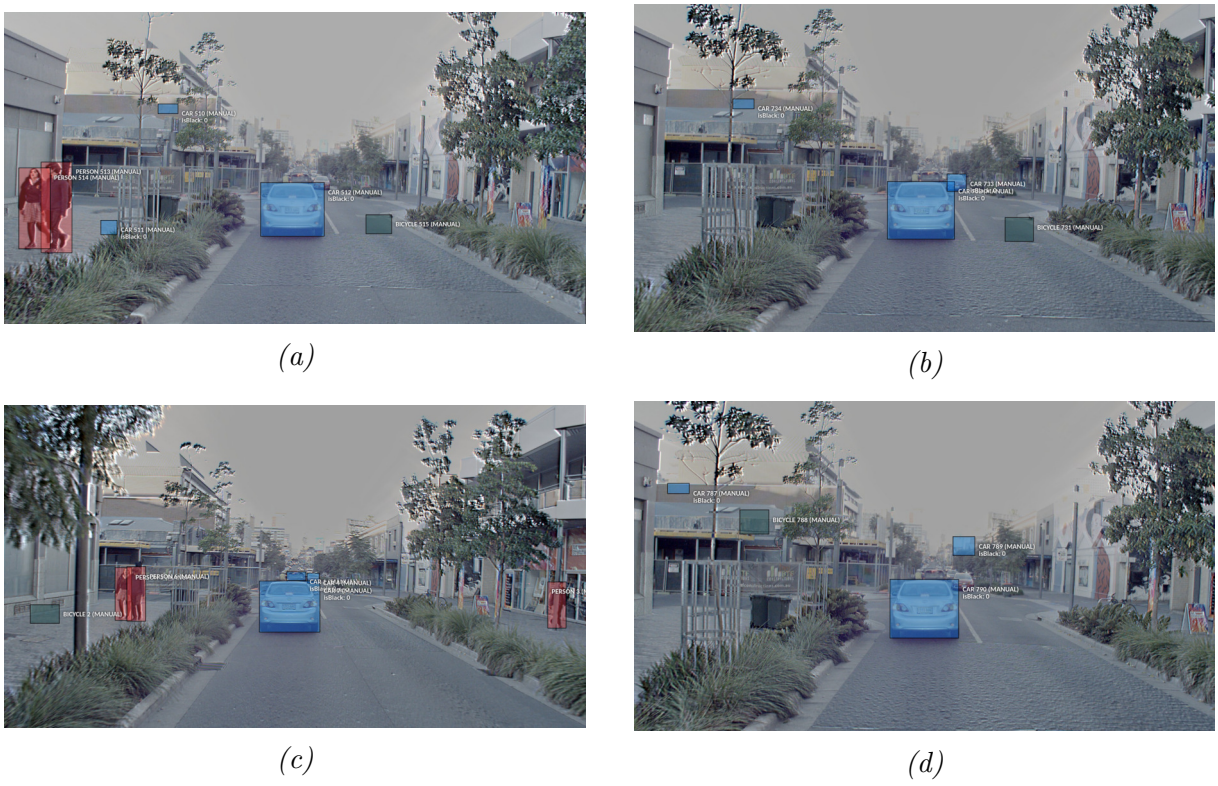
(d) Reference image (a) being transformed to align to target image (b) based on keypoints few of them highlighted in (c)

Figure 4.8: Results of image registration method

3. **Template Matching**

   Template matching was used to search for objects in the target frame using ground truth data in the reference frame. The ground truth data contained the location of each labelled object in the reference frames. Using the coordinate location of objects from ground truth annotations, these were extracted as templates and was

used to search and locate similar templates in the target frames. OpenCV [69] has multiple methods for comparing the template in the target image and for this case *TM_SQDIFF* and *TM_SQDIFF_NORMED* offered the best results. The template matching method was only partially able to correctly identify and locate objects from the reference frame in target frames. The results can be seen in Figure 4.9 where the method could not correctly identify some of the frames. Manually moving the incorrect placement of the bounding box in each frame would be more time-consuming than manually re-annotating the dataset.



(a) *(b)* *(c)* *(d)*

Figure 4.9: Result of template mathcing to transfer reference dataset's annotation to target frames (a-d)

Since none of the methods explored was able to transfer the annotations from the reference datasets of the MOV category fully, the datasets *MOV_NL_MONO_PRC*, *MOV_LL_MONO_RAW*, *MOV_LL_COL_PRC* and *MOV_NL_COL_RAW* were labelled manually. 5000 frames were annotated in each dataset. However, since the datasets *MOV_NL_COL_PRC*, *MOV_NL_COL_PRC* had 16000 and *MOV_LL_MONO_PRC*, *MOV_LL_MONO_PRC* had 14215 labelled images, an unbiased comparison cannot be made with a smaller dataset of 5000 frames.

So, to make the comparison, the larger datasets were divided into three sub-groups, each with 5000 frames in the first two groups and the third group with remaining frames. The new datasets were given the same name followed by a number for identification.

4.3 Object Detection

For object detection Detectron2 [118] framework was used. To generalise the results of the detections and save time for training and fine-tuning object detector, a pre-trained Faster R-CNN model was used. The id model was downloaded from Model Zoo of detectron2 [119]. The model was trained and tested on COCO dataset. The object detector task was run on an Intel i7-8500U CPU with 8GB RAM running Ubuntu 20.04.02, running only on CPU. The average time required for running Faster R-CNN on each frame was approximately 3 seconds.

The chosen model supported all the classes of objects in annotated datasets except van, and scooter.

4.4 Evaluation: PASCAL VOC

After the predictions were made for all the datasets, the predictions and the respective ground truth data were evaluated to determine whether the predicted results matched the annotated ground truth data. For this, each annotated ground truth was compared with its predicted result, the IoU of their bounding boxes and the labels compared to determine if the results matched. Each positive match was marked as TP and negative match marked as FP, and if no predictions were made for a given ground truth, they were marked as FN. These data were compiled for the entire dataset and used to generate Precision x Recall curve from which AP and mAP were calculated using an open-source tool called "Object-Detection-Metrics" [56]. The use of AP and mAP for comparing object detectors performance is a PASCAL VOC metrics [54]. The AP and mAP data was generated for all the classes of object in ground truth. The generated results are presented in the Tables 5.1 and 5.2 in Chapter 5.

Chapter 5

Results

The results in Tables 5.1 and 5.2 were computed for all the datasets in Table 4.2 using PASCAL VOC metrics [54] to evaluating the performance of bio-inspired TMO [15] for Machine vision application of Object Classification and Localisation. The comparison between the unprocessed datasets (RAW) with bio-inspired TMO pre-processed dataset (PRC) were made in Normal (NL) and Low Light (LL) conditions. The comparison are made on datasets recorded using Monochrome (MONO), Colour (COL) and Low-resolution Colour (LRES) cameras in camera setup in Stationary (STA) Section 5.1 and Moving (MOV) Section 5.2. All the comparisons are made based on the mAP scores of the respective dataset.

Table 5.1: Results using PASCAL VOC metric for Stationary Datasets [1]

Dataset	Statistics [2]	mAP	car	bus	truck	bicycle	motorcycle	person	traffic_light	train
STA_NL_MONO_RAW	mAP / AP [%]	9.139	32.795	13.725	0	-	0	17.451	0	-
	Detection Rate		34671 / 100686	172 / 1244	0 / 110	-	0 / 68	3596 / 18235	0 / 45264	-
STA_NL_MONO_PRC	mAP / AP [%]	15.094	51.718	10.41	0	-	13.64	29.887	0	-
	Detection Rate		56479 / 100686	130 / 1244	0 / 110	-	17 / 68	6663 / 18235	2 / 45264	-
STA_NL_LRES_RAW	mAP / AP [%]	4.889	14.983	9.244	0	-	0	9.993	0.0002	-
	Detection Rate		17689 / 100686	115 / 1244	0 / 110	-	0 / 68	2549 / 18235	0 / 45264	-
STA_NL_LRES_PRC	mAP / AP [%]	8.099	26.431	12.127	0	-	3.455	14.677	0.002	-
	Detection Rate		35331 / 100686	151 / 1244	0 / 110	-	17 / 68	4279 / 18235	8 / 45264	-
STA_NL_COL_RAW	mAP / AP [%]	11.847	41.533	15.506	0.005	-	0.21	25.674	0	-
	Detection Rate		47854 / 100686	193 / 1244	2 / 110	-	1 / 68	5642 / 18235	0 / 45264	-
STA_NL_COL_PRC	mAP / AP [%]	15.234	55.666	12.531	0.006	-	1.402	37.034	0.000005	-
	Detection Rate		63674 / 100686	156 / 1244	3 / 110	-	12 / 68	8264 / 18235	1 / 45264	-
STA_LL_MONO_RAW	mAP / AP [%]	13.84	34.743	0.406	0.022	-	-	36.041	-	11.825
	Detection Rate		17126 / 46439	21 / 468	2 / 153	-	-	2076 / 4856	-	409 / 3072
STA_LL_MONO_PRC	mAP / AP [%]	16.453	43.104	1.126	0.167	-	-	38.014	-	16.308
	Detection Rate		21546 / 46439	41 / 468	11 / 153	-	-	2223 / 4856	-	562 / 3072
STA_LL_LRES_RAW	mAP / AP [%]	1.459	8.744	0	0	-	-	0.008	-	0
	Detection Rate		5332 / 46439	0 / 468	0 / 153	-	-	2 / 4856	-	0 / 3072
STA_LL_LRES_PRC	mAP / AP [%]	1.93	9.876	0	0	-	-	1.703	-	0
	Detection Rate		6301 / 46439	0 / 468	0 / 153	-	-	119 / 4856	-	0 / 3072
STA_LL_COL_RAW	mAP / AP [%]	10.888	35.301	0.063	0.005	-	-	29.959	-	0
	Detection Rate		18022 / 46439	10 / 468	1 / 153	-	-	1790 / 4856	-	0 / 3072
STA_LL_COL_PRC	mAP / AP [%]	9.977	36.023	0.585	0.094	-	-	23.157	-	0
	Detection Rate		19963 / 46439	37 / 468	9 / 153	-	-	1531 / 4856	-	0 / 3072
STA_NL_MONO_RAW	mAP / AP [%]	9.139	32.795	13.725	0	-	0	17.451	0	-
	Detection Rate		34671 / 100686	172 / 1244	0 / 110	-	0 / 68	3596 / 18235	0 / 45264	-
STA_NL_MONO_PRC	mAP / AP [%]	15.094	51.718	10.41	0	-	13.64	29.887	0	-
	Detection Rate		56479 / 100686	130 / 1244	0 / 110	-	17 / 68	6663 / 18235	2 / 45264	-
STA_NL_LRES_RAW	mAP / AP [%]	4.889	14.983	9.244	0	-	0	9.993	0.0002	-
	Detection Rate		17689 / 100686	115 / 1244	0 / 110	-	0 / 68	2549 / 18235	0 / 45264	-
STA_NL_LRES_PRC	mAP / AP [%]	8.099	26.431	12.127	0	-	3.455	14.677	0.002	-
	Detection Rate		35331 / 100686	151 / 1244	0 / 110	-	17 / 68	4279 / 18235	8 / 45264	-
STA_NL_COL_RAW	mAP / AP [%]	11.847	41.533	15.506	0.005	-	0.21	25.674	0	-
	Detection Rate		47854 / 100686	193 / 1244	2 / 110	-	1 / 68	5642 / 18235	0 / 45264	-
STA_NL_COL_PRC	mAP / AP [%]	15.234	55.666	12.531	0.006	-	1.402	37.034	0.000005	-
	Detection Rate		63674 / 100686	156 / 1244	3 / 110	-	12 / 68	8264 / 18235	1 / 45264	-
STA_LL_MONO_RAW	mAP / AP [%]	13.84	34.743	0.406	0.022	-	-	36.041	-	11.825
	Detection Rate		17126 / 46439	21 / 468	2 / 153	-	-	2076 / 4856	-	409 / 3072
STA_LL_MONO_PRC	mAP / AP [%]	16.453	43.104	1.126	0.167	-	-	38.014	-	16.308
	Detection Rate		21546 / 46439	41 / 468	11 / 153	-	-	2223 / 4856	-	562 / 3072
STA_LL_LRES_RAW	mAP / AP [%]	1.459	8.744	0	0	-	-	0.008	-	0
	Detection Rate		5332 / 46439	0 / 468	0 / 153	-	-	2 / 4856	-	0 / 3072
STA_LL_LRES_PRC	mAP / AP [%]	1.93	9.876	0	0	-	-	1.703	-	0
	Detection Rate		6301 / 46439	0 / 468	0 / 153	-	-	119 / 4856	-	0 / 3072
STA_LL_COL_RAW	mAP / AP [%]	10.888	35.301	0.063	0.005	-	-	29.959	-	0
	Detection Rate		18022 / 46439	10 / 468	1 / 153	-	-	1790 / 4856	-	0 / 3072
STA_LL_COL_PRC	mAP / AP [%]	9.977	36.023	0.585	0.094	-	-	23.157	-	0
	Detection Rate		19963 / 46439	37 / 468	9 / 153	-	-	1531 / 4856	-	0 / 3072

[1] Results of class van and scooter are omitted as the detector used does not support detection of these classes

[2] mAP / AP values are in percentage, Detection rate = True Positive / Ground Truth

'-' values in mAP / AP and Detection Rate means the object was not annotated in ground truth data

Table 5.2: Results using PASCAL VOC metric for Moving Datasets [2]

Dataset	Statistics [2]	mAP	car	bus	truck	bicycle	motorcycle	person	traffic_light	train
MOV_NL_MONO_RAW	mAP / AP	61.547	85.096	-	70.021	33.645	-	57.428	-	-
	Detection Rate		9765 / 11134	-	149 / 190	231 / 667	-	3005 / 4883	-	-
MOV_NL_MONO_PRC	mAP / AP	60.461	79.683	-	65.716	37.491	-	58.956	-	-
	Detection Rate		9198 / 11134	-	160 / 190	270 / 667	-	3229 / 4883	-	-
MOV_NL_COL_RAW_#1	mAP / AP	75.685	83.881	-	56.811	88.303	-	73.745	-	-
	Detection Rate		8448 / 9454	-	145 / 190	115 / 127	-	2786 / 3592	-	-
MOV_NL_COL_PRC_#1	mAP / AP	74.365	80.593	-	67.248	77.819	-	71.801	-	-
	Detection Rate		8098 / 9454	-	177 / 190	103 / 127	-	2794 / 3592	-	-
MOV_NL_COL_RAW_#2	mAP / AP	56.138	79.669	81.16	-	95.891	-	80.106	-	
	Detection Rate		19005 / 19512	904 / 1076	-	3223 / 3339	-	6063 / 7000	-	-
MOV_NL_COL_PRC_#2	mAP / AP	50.743	79.575	51.289	-	95.169	-	78.422	-	-
	Detection Rate		18717 / 19512	607 / 1076	-	3204 / 3339	-	5951 / 7000	-	-
MOV_NL_COL_RAW_#3	mAP / AP	40.144	66.159	78.973	56.05	63.48	41.825	57.587	69.222	-
	Detection Rate		16369 / 19107	123 / 139	1208 / 1656	3378 / 4250	210 / 222	17112 / 19580	4603 / 5220	-
MOV_NL_COL_PRC_#3	mAP / AP	35.941	64.473	16.024	48.232	52.965	31.689	55.18	54.91	-
	Detection Rate		16031 / 19107	63 / 139	1177 / 1656	2869 / 4250	209 / 222	16897 / 19580	4257 / 5220	-
MOV_NL_COL_RAW	mAP / AP	50.018	75.007	81.04	46.105	80.066	40.443	67.85	59.652	-
	Detection Rate		44174 / 48620	1027 / 1215	1208 / 1656	6773 / 7773	210 / 222	26018 / 30229	4603 / 5220	-
MOV_NL_COL_PRC	mAP / AP	40.274	74.017	45.707	35.045	71.171	26.848	71.171	43.875	-
	Detection Rate		43224 / 48620	670 / 1215	1177 / 1656	6233 / 7773	209 / 222	25699 / 30229	4257 / 5220	-
MOV_LL_MONO_RAW	mAP / AP	21.916	75.961	0.12	9.022	42.086	2.906	45.234	-	-
	Detection Rate		44886 / 55660	9 / 151	674 / 4544	571 / 1241	53 / 492	34562 / 71387	-	-
MOV_LL_MONO_PRC	mAP / AP	28.056	73.83	6.893	9.626	57.816	18.023	58.258	-	-
	Detection Rate		44432 / 55660	84 / 151	634 / 4544	811 / 1241	236 / 492	47404 / 71387	-	-
MOV_LL_MONO_RAW_#1	mAP / AP	56.169	74.689	-	50.191	60.985	-	38.812	-	-
	Detection Rate		9631 / 12142	-	122 / 238	28 / 44	-	2434 / 5751	-	-
MOV_LL_MONO_PRC_#1	mAP / AP	51.017	69.104	-	59.167	29.146	-	46.653	-	-
	Detection Rate		9098 / 12142	-	147 / 238	28 / 44	-	3108 / 5751	-	-
MOV_LL_MONO_RAW_#2	mAP / AP	21.481	76.59	-	2.488	23.046	0.392	45.389	-	-
	Detection Rate		21163 / 25829	9 / 151	29 / 140	61 / 201	2 / 68	7169 / 14309	-	-
MOV_LL_MONO_PRC_#2	mAP / AP	39.528	74.197	39.694	1.221	54.843	56.27	50.468	-	-
	Detection Rate		20987 / 25829	84 / 151	21 / 140	136 / 201	59 / 68	8703 / 14309	-	-
MOV_LL_MONO_RAW_#3	mAP / AP	24.05	68.376	-	6.571	43.555	3.528	46.318	-	-
	Detection Rate		12839 / 16192	-	523 / 4166	308 / 500	51 / 424	24450 / 49747	-	-
MOV_LL_MONO_PRC_#3	mAP / AP	30.873	70.999	-	8.413	57.574	15.926	63.201	-	-
	Detection Rate		13058 / 16192	-	466 / 4166	385 / 500	177 / 424	34751 / 49747	-	-
MOV_LL_COL_RAW	mAP / AP	37.934	80.596	-	22.83	10.929	-	37.381	-	-
	Detection Rate		10339 / 11895	-	120 / 498	15 / 116	-	1905 / 3677	-	-
MOV_LL_COL_PRC	mAP / AP	37.196	67.619	-	29.93	12.919	-	38.316	-	-
	Detection Rate		8754 / 11895	-	157 / 498	29 / 116	-	1921 / 3677	-	-

[1]Results of class van and scooter are omitted as the detector used does not support detection of these classes

[2]mAP / AP values are in percentage, Detection rate = True Positive / Ground Truth

'-' values in mAP / AP and Detection Rate means the object was not annotated in ground truth data

5.1 Analysis of results of Stationary Datasets

The following comparisons are made between datasets captured using MONO, LRES and COL cameras with camera setup Stationary. For this, the mAP score of detection by object detector, Faster R-CNN, is lower because of the relatively smaller size of the objects [7]. The object detector, Faster-RCNN's chosen model, is a general model and has not be specially trained to detect smaller objects and hence its detection scores are lower.

5.1.1 Stationary: Normal Lighting Condition

For normal lighting conditions, the pre-processed datsets i.e. PRC, helped increase the mAP score of object detector by 65.66% for LRES, 65.16% for MONO and 28.6% for COL camera datasets. Even though datasets using the LRES camera experienced the most significant boost in performance, the increased performance score was nearly half the score in MONO and LRES. COL camera datasets perform better than MONO by a small margin of 0.14% for PRC dataset, which was originally 2.708%.

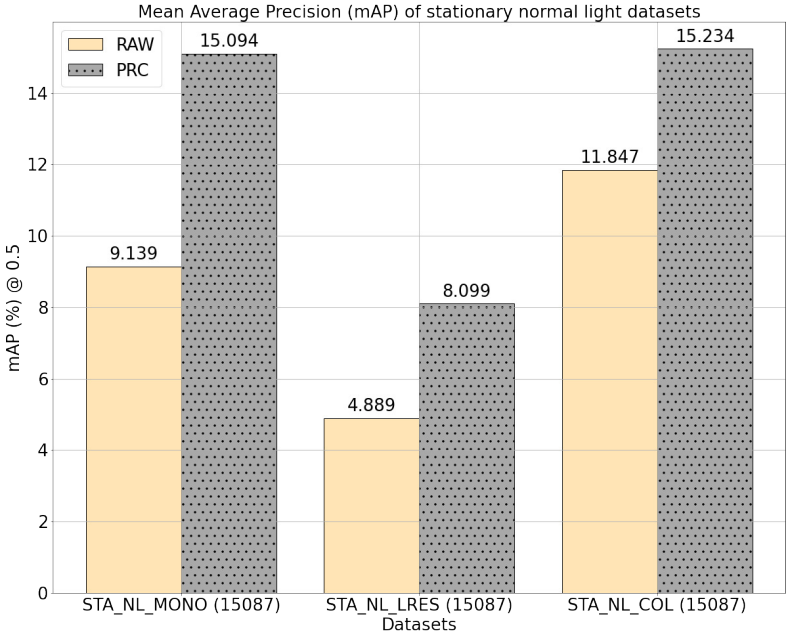
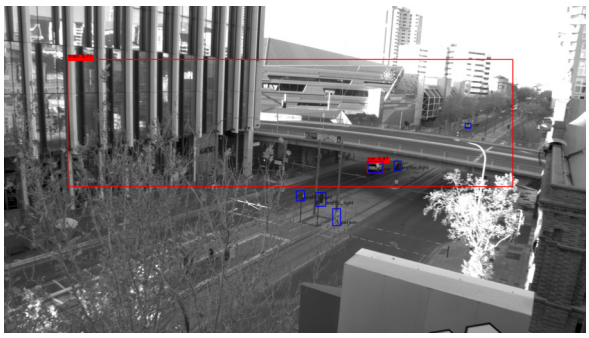
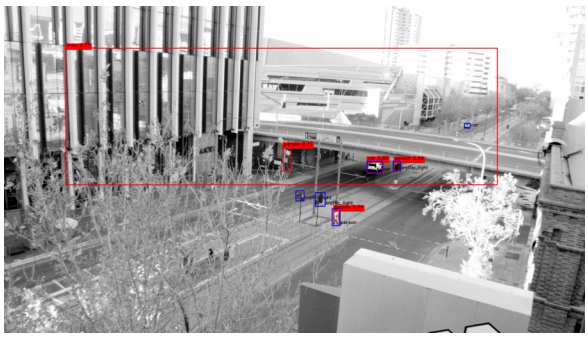


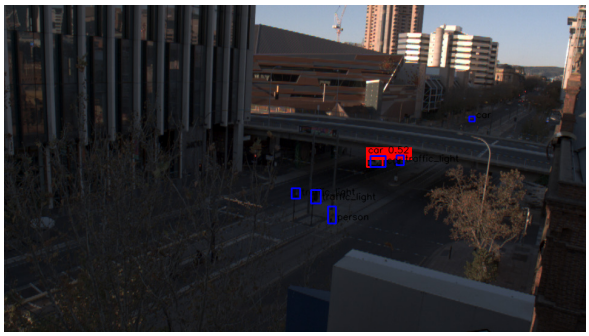
*Figure 5.1: mAP of RAW and PRC in stationary normal light **(STA_NL)** datasets. mAP of COL dataset is higher followed by MONO and LRES.*



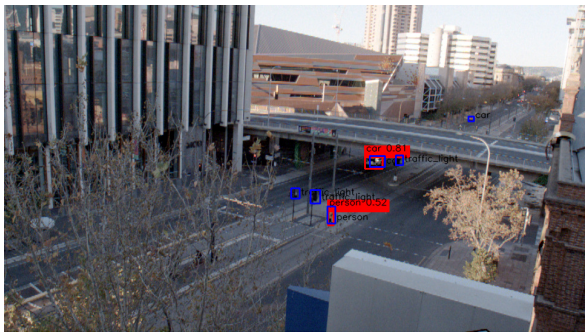
(a) STA_NL_MONO_RAW



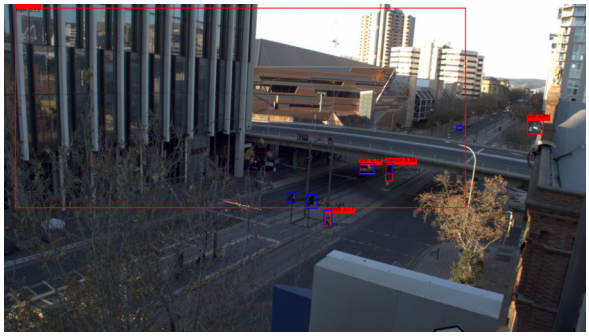
(b) STA_NL_MONO_PRC



(c) STA_NL_LRES_RAW



(d) STA_NL_LRES_PRC



(e) STA_NL_COL_RAW

(f) STA_NL_COL_PRC

Figure 5.2: (a-f) frames with bounding box for ground truth (blue) and object detector prediction (red) with confidence score for Stationary normal light ***(STA_NL)*** *datasets.*

5.1.2 Stationary: Low Lighting Condition

PRC dataset of COL camera, which had mAP boosted by 28.6% in normal lighting condition, had its mAP score significantly decreased by 8.4%. Whereas, like before LRES camera datasets had the most significant increment of 32.28% but same as before LRES dataset had the least mAP score. Lastly, the confidence score of MONO was moderately boosted by about 18% making it the most dataset in low light condition. The mAP score of the PRC dataset of MONO was about 8.5x and 1.6x greater than LRES and COL, respectively.

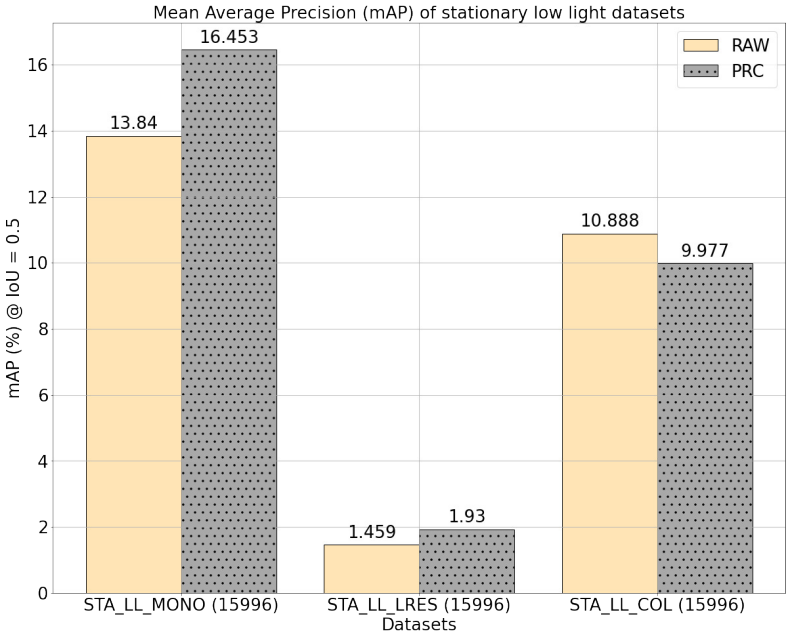
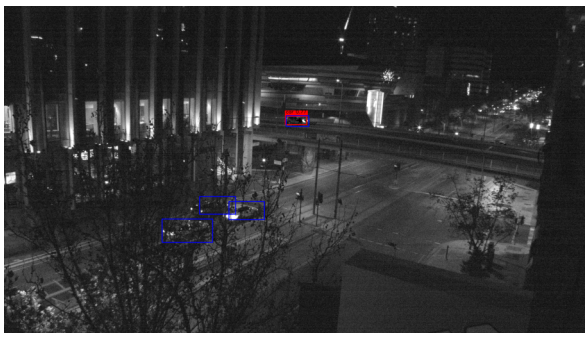
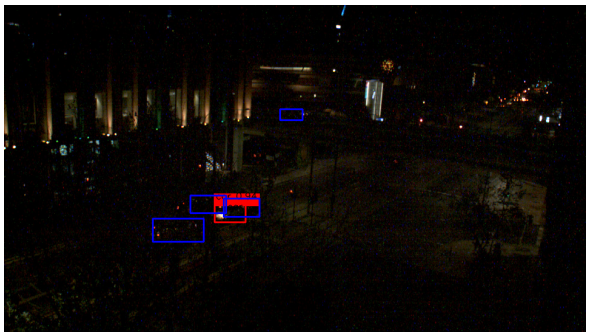


Figure 5.3: mAP of RAW and PRC in stationary normal light ***(STA_LL)*** *dataset*

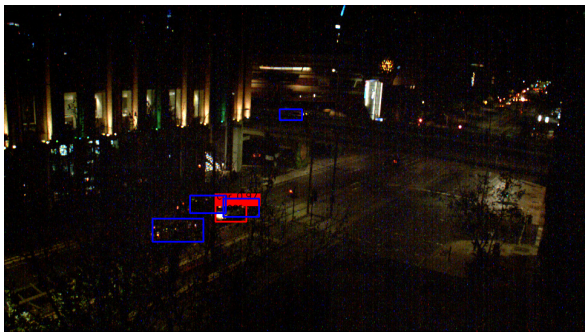
(a) STA_LL_MONO_RAW



(b) STA_LL_MONO_PRC



(c) dataset STA_LL_LRES_RAW



(d) STA_LL_LRES_PRC

(e) STA_LL_COL_RAW

(f) STA_LL_COL_PRC

Figure 5.4: (a-f) frames with bounding box for ground truth (blue) and object detector prediction (red) with confidence score for Stationary low light ***(STA_LL)*** *datasets. The number of detections are lower because of smaller object size. Object detectors have trouble detecting smaller objects [7].*

5.2 Analysis of results of Moving Datasets

The following comparisons are made between datasets captured using MONO and COL cameras with camera setup: Moving. In the datasets below, the larger dataset MOV_NL_COL, and MOV_LL_MONO each consists of 16,000 and 14,215 frames, respectively divided into three datasets with the same name followed by #1, #2, #3 to identify their sequence. This was done to make an unbiased comparison with the other datasets MOV_NL_MONO and MOV_LL_COL.

The datasets were captured on a moving car that started recording from a parking lot with proper lighting condition. So the initial frames of around 1,600 for dataset MOV_LL

were captured in an artificial lighting condition. While this does not impact the MOV_NL datasets, MOV_LL datasets performance was negatively affected.

Compared to datasets in stationary, the mAP score or detection ratio is significantly higher because of the larger object size as the distance between the camera and nearby object is relatively minor compared to stationary.

5.2.1 Moving: Normal Lighting Condition

In normal lighting condition, contrary to previous results, the detections decreased. Dataset MOV_NL_MONO and MOV_NL_COL_#1, which refer to the same scenes captured using MONO and COL camera respectively, saw its performance decreased by 1.76% on MONO and 1.67% on COL for the pre-processed datasets. Overall, the COL camera's RAW datasets offered better detection score in all the normal lighting datasets.

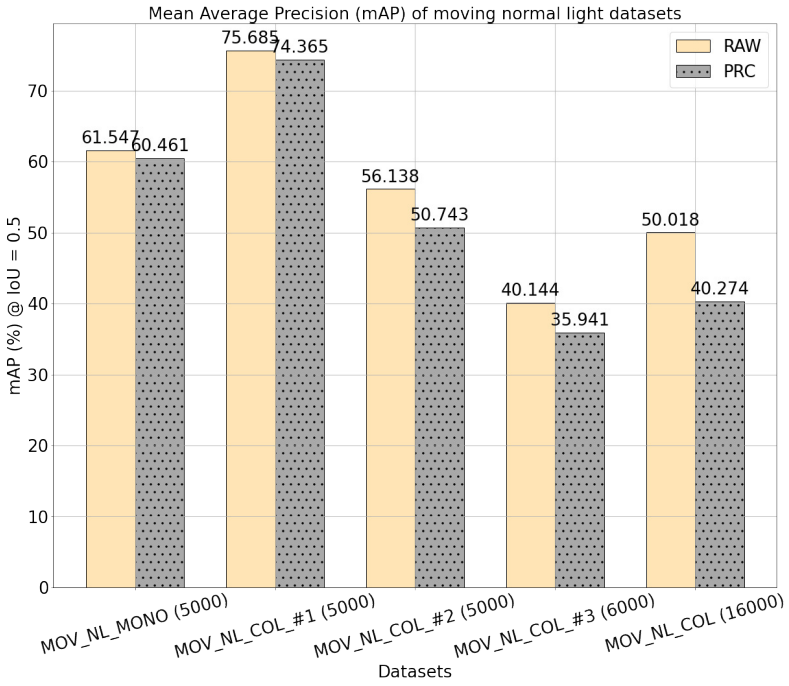
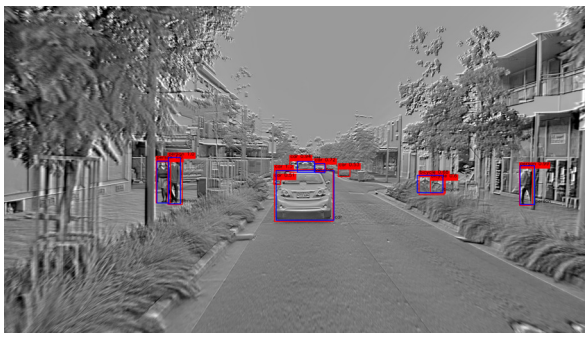


Figure 5.5: mAP of RAW and PRC in stationary normal light ***(MOV_NL)*** *dataset*

(a) MOV_NL_MONO_RAW



(b) MOV_NL_MONO_PRC

(c) MOV_NL_COL_RAW

(d) MOV_NL_COL_PRC

Figure 5.6: (a-d) frames with bounding box for ground truth (blue) and object detector prediction (red) with confidence score for Moving normal light ***(MOV_NL)*** *datasets. Since, the objects are nearer to the camera, the size of the objects are larger which leads to higher mAP scores compared to stationary dataset.*

5.2.2 Moving: MOV_LL

As mentioned earlier, about 1,600 frames in MOV_NL_MONO_#1 dataset were captured in a parking building with proper lighting condition. Because of this, the performance of Faster R-CNN on PRC for dataset MOV_NL_MONO_#1 decreased by about 9%. Similarly, the performance on MOV_LL_COL dataset decreased by almost 2%.

In the latter frames of MOV_MONO_LL, the mAP increased by about 84.01% for dataset #2, 28.37% for dataset #3 and 28.01% for the overall combined dataset.

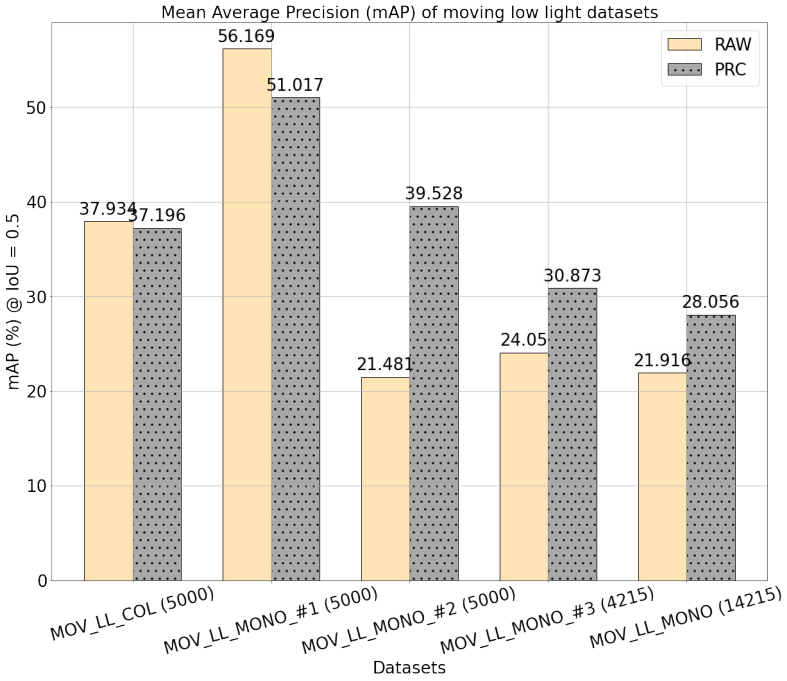
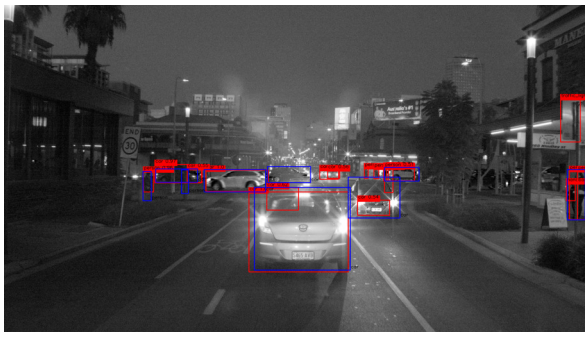
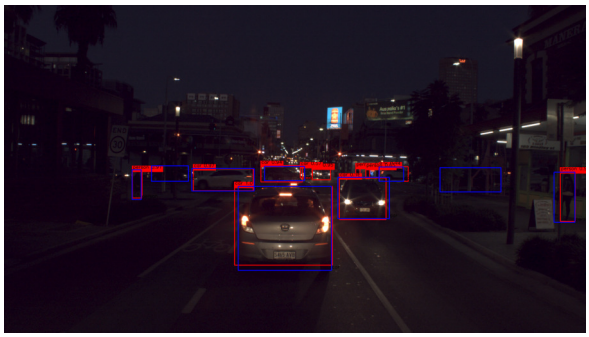


Figure 5.7: mAP of RAW and PRC in stationary normal light ***(MOV_LL)*** *dataset*

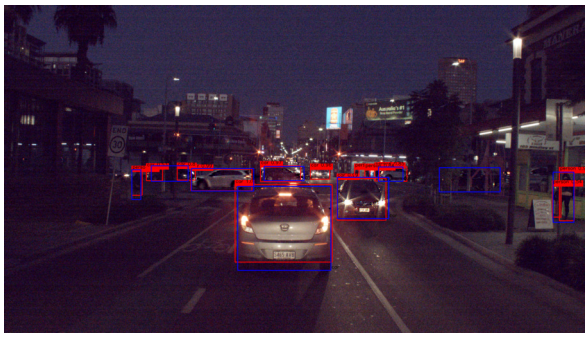
(a) MOV_LL_MONO_RAW



(b) MOV_LL_MONO_PRC



(c) MOV_LL_COL_RAW



(d) MOV_LL_COL_PRC

Figure 5.8: (a-f) frames with bounding box for ground truth (blue) and object detector prediction (red) with confidence score for Moving low light ***(MOV_LL)*** *datasets. Since, the objects are nearer to the camera, the size of the objects are larger which leads to higher mAP scores compared to stationary dataset.*

5.3 Result Summary

Based on evaluation metrics of PASCAL VOC [54]: Mean average precision (mAP), all the datasets on Table 4.2 were evaluated and compared to determine the improvement of object detector of Machine Vision applications of Object Classification, Localisation. These datasets were exposed to natural and artificial lighting condition before and after sunset using MONO, COL and LRES HDR cameras. Also, the camera setup for the stationary position was analogous to the Surveillance system, and Moving was to Autonomous Navigation.

Regarding Object detector's performance on camera setup stationary, surveillance, for normal lighting condition on the images pre-processed (PRC) by bio-inspired TMO, COL camera produced a better result than MONO cameras with a marginal difference of 0.14%. However, in the low light condition, the MONO camera performance was 1.6x the COL camera. The LRES camera's performance even though it experienced the highest gain when using the PRC dataset, the overall prediction score was always significantly lower compared to scores of MONO and LRES in both cases. For stationary setup, the pre-processing with bio-inspired TMO increased performance for all MONO, COL and LRES camera's datasets in normal lighting condition. However, the COL camera's performance dropped by a margin of 0.911% when using the PRC dataset in low light condition.

For camera setup moving, i.e. autonomous navigation, the PRC datasets decreased the overall performance of both MONO and COL. However, the RAW dataset of COL still outperformed the PRC and RAW dataset of MONO. For low light datasets, artificial lighting in a parking building at the starting 1600 frames negatively impacted a significant part of the initial dataset leading to decreased performance of MONO camera on PRC dataset compared to the RAW dataset. However, as the building was exited, the performance gap between pre-processed and unprocessed dataset changed in favour of the PRC dataset. Overall, for moving, i.e. autonomous navigation setup, COL cameras perform better in normal lighting conditions, but the performance was better for the low light condition when MONO cameras were used.

The evaluation results of the bio-inspired TMO has increased the confidence score of object detector for tasks of object classification and localisation. For surveillance and autonomous navigation application, the MONO camera is the ideal choice in low light condition when no significant artificial lighting was observed.

While in normal lighting conditions, the performance increased only in surveillance setup where objects are far away from the camera. The enhancement done in pre-processed datasets increased the contrast and features that helped the object detector to identify more objects than it would have using unprocessed dataset. For autonomous navigation

setup, the relatively larger object size and the proper illumination condition increased the performance level of the unprocessed dataset. The pre-processing for this scenario yielded no significant benefit and visually seemed to have blurred the images, leading to decreased performance of object detector.

Table 5.3: Overall summary of comparision results. Based on mAP score of PRC and RAW dataset for normal lighting (NL) and low lighting (LL) condition, the one with the higher score is marked with a tickmark symbol

Camera Position	Camera	PRC/RAW	NL	LL
Stationary	***MONO***	RAW		
		PRC	✓	✓
	LRES	RAW		
		PRC	✓	✓
	COL	RAW		✓
		PRC	✓	
Moving	***MONO***	RAW	✓	
		PRC		✓
	COL	RAW	✓	✓
		PRC		

Chapter 6

Discussion

The thesis evaluated the performance of Machine Vision application on HDR images that have been tone-mapped using bio-inspired TMO [15] on dataset exposed to different lighting conditions and camera setup. The machine vision application on which the evaluation was done was an Object Detector, Faster-RCNN [5] using PASCAL VOC metrics [54]. The comparison of results in Chapter 5 points to increased Object Detector performance when using pre-processed images with bio-inspired TMO in all low-light conditions. These benefits were observed for Monochrome cameras, and the benefit was not observed for Colour camera. The results of this evaluation are in line with results from evaluations performed by Rana et al. [30], Přibyl et al. [109] and Chermak and Aouf [104] where HDR images, even the tone-mapped LDR images, were found to have more features for detection and matching than normal LDR images. While this thesis's evaluation is not on feature detection and matching, the bio-inspired TMO was designed to increase the detection of edges and enhance contrast under its evaluation metric of Motion Artefacts. The feature detectors used in [30], [109] and [104] detected and matched most distinct features such as corners, points and others which primarily lie on the edge of an object. By enhancing these edges of objects, feature detection can be increased.

In an experiment by Pinho et al. [105], the use of tone-mapped HDR images did improve detection of fruits, but the experiment found no difference between the various TMOs used, which where Reinhard [47], Drago [49] and the camera's inbuilt TMO itself. However, Rana et al. [30] pointed Drago [49] as the best TMO for improving feature detection. Moreover, Griffiths [15] when comparing his novel bio-inspired TMO using proposed metric of Noise Suppression, Flicker and Motion artefacts which focused on information content and objective use for Machine Vision applications, found bio-inspired TMO performing better than Reinhard [47] in Noise Suppression and Motion Artefact but not reduction of image flicker. However, for Drago [49] bio-inspired TMO outperformed on all three metrics. While the datasets used, methodology and evaluation metrics differ in each of these experiments to make a direct comparison difficult, evidence from this

thesis and Griffiths [15] show bio-inspired TMO compared to its LDR images (RAW) show improved detection of features such as edges and increased image contrast which led to increased performance of Object Detectors.

Furthermore, the novel bio-inspired TMO is meant as a general-purpose TMO for Machine Vision application. The datasets captured in the Stationary setup (STA) is analogous to a surveillance system where the main tasks are object classification, localisation, and tracking. For optimal performance, it is expected for such surveillance system to be immune to changing lighting condition. A paper by [120] proposed a hybrid TMO with properties of local and global TMO for object detection and tracking, but it lacks implementation to make any comparison. The bio-inspired TMO, which is also a hybrid TMO used in this thesis, has improved object detection for surveillance application.

Moreover, the dataset with camera setup Moving is analogous with the setup of cameras for autonomous driving using HDR cameras. In this regard, a similar experiment by [117] for vehicle and tracking had used deep learning model of AlexNet fused with LIDAR data. The common thing between [117] is the use of deep learning models for object detection on dataset exposed to extreme lighting conditions. Furthermore, the results further add to the evidence of improved performance because of the use of HDR imaging.

Lastly, a direct comparison of bio-inspired TMO on other application is still lacking. However, this thesis and other papers discussed here point to HDR imaging, even when tone-mapped, has shown to improve feature detection and matching, contributing to increased detection rate for Machine Vision applications such as Object classification, localisation in surveillance and autonomous driving applications.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

The Bio-inspired TMO was implemented on dynamic sequence of images with camera setups similar to surveillance (stationary) and autonomous navigation (moving). For higher-order MV task of object detection, i.e., object classification and localisation to validate improvements in object detector performance when using bio-inspired TMO as the pre-processing algorithm for converting HDR images to the 8-bit standard format. The evaluation for each camera setup consisted of datasets captured using MONO, COL and LRES HDR cameras for normal and low illumination conditions. The MONO and COL cameras captured HDR images using a single exposure method, while LRES used a row-based multi-exposure method to capture HDR images of equal depth and reduced resolution.

The datasets were annotated and ran through a Faster-RCNN object detector model to generate predictions that was evaluated and compared using the PASCAL VOC metric of mAP in %. COL cameras are known to perform better in normal illumination, whereas MONO in low illumination. The thesis results are in line with this fact, where datasets captured using MONO were found to perform better in low light condition compared to COL and LRES. The pre-processed datasets further increased the performance of the object detector in low light condition, but in normal lighting conditions, the performance of the object detector decreased. Besides this, artificial lighting in a parking building also decreased performance on the pre-processed dataset captured using MONO. Likewise, for datasets captured using COL camera, the object detector performed exceptionally well in un-processed datasets, which was further enhanced when using pre-processed datasets for normal illumination condition. However, in low light conditions where the un-processed dataset performed better than pre-processed datasets. Lastly, LRES datasets, when pre-processed, had its prediction score increased the most for surveillance datasets under

normal and low light. However, its performance compared to MONO and COL was sub-par, almost a half in normal and a fifth in low light condition.

During the evaluation, annotated classes of objects van and scooter were not supported for detection, and these were detected as car and motorcycle in the pre-trained model, which led to false detections. Besides this, the object detector had problems detecting objects of smaller size in surveillance-related datasets. These problems led to a decrease in mAP, but it was consistent over the compared datasets used for evaluation and avoided any bias in the final results of the thesis.

The evaluation study has extended and verified the implementation of bio-inspired TMO on dynamic settings such as surveillance and autonomous navigation. For low light conditions, the increased performance of the object detector can be attributed to enhanced noise suppression, contrast, and edge detection, but these benefits were not observed in images under normal lighting conditions. Additionally, the results obtained by MONO camera were more consistent compared to COL and LRES cameras.

7.2 Future Work

7.2.1 Compare Performance on One-stage Detectors and real-time implementation

While in terms of accuracy two-stage detectors are better, in terms of speed one-stage detectors are faster. The accuracy of one-stage detectors like SSD, YOLO are getting faster with new developments. In future a comparative analysis between performance improvement for two-stage and one-stage detectors should be performed. Besides this, for implementation of bio-inspired TMO it is essential for testing it in real-time applications on which one-stage detectors are better than two-stage detector.

7.2.2 Custom-trained detector

The Faster-RCNN model used was a pre-trained model to detect general objects from people, cars to boats, handbags, clock which are not essential depending upon the task at hand. Having a custom-trained detector may increase the performance level of object detector on bio-inspired pre-processed images.

Bibliography

[1] Getty Museum,, "The Brig."

[2] Mathworks, "Deep learning," https://au.mathworks.com, 2020, accessed on: 2021-04-24. [Online]. Available: https://au.mathworks.com/discovery/deep-learning.html

[3] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[4] P. Sharma, "A step-by-step introduction to the basic object detection algorithms," https://www.analyticsvidhya.com, Oct. 2018, Accessed on: 24-04-2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2018/10/a-step-by-step-introduction-to-the-basic-object-detection-algorithms-part-1/

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *arXiv preprint arXiv:1506.01497*, 2015.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[7] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7310–7311.

[8] J. Y. Suk, M. Schiler, and K. Kensek, "Development of new daylight glare analysis methodology using absolute glare factor and relative glare factor," *Energy and Buildings*, vol. 64, pp. 113–122, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378778813002569

[9] S. Mann and R. Picard, "Beingundigital'with digital cameras," *MIT Media Lab Perceptual*, vol. 1, p. 2, 1994.

[10] F. Banterle, K. Debattista, A. Artusi, S. Pattanaik, K. Myszkowski, P. Ledda, and A. Chalmers, "High dynamic range imaging and low dynamic range expansion for generating hdr content," in *Computer graphics forum*, vol. 28, no. 8. Wiley Online Library, 2009, pp. 2343–2367.

[11] G. Ward and M. Simmons, "Jpeg-hdr: A backwards-compatible, high dynamic range extension to jpeg," in *ACM SIGGRAPH 2006 Courses*, 2006, pp. 3–es.

[12] T. Dobashi, A. Tashiro, M. Iwahashi, and H. Kiya, "A fixed-point implementation of tone mapping operation for hdr images expressed in floating-point format," *APSIPA Transactions on Signal and Information Processing*, vol. 3, 2014.

[13] G. Eilertsen, R. Mantiuk, and J. Unger, "A comparative review of tone-mapping algorithms for high dynamic range video," *Computer Graphics Forum*, vol. 36, pp. 565–592, 05 2017.

[14] R. K. Mantiuk, K. Myszkowski, and H. Seidel, "High dynamic range imaging," 2015.

[15] D. Griffiths, "Biologically inspired high dynamic range imaging for use in machine vision," Ph.D. Thesis, 2017.

[16] J. J. McCann, *The art and science of HDR imaging*, ser. Wiley-IS&T series in imaging science and technology. Chichester, West Sussex, U.K. ; Hoboken, N.J.: Wiley, 2012.

[17] C. Wyckoff, "Silver halide photographic film having increased exposure-response characteristics," U.S. patentus US3 450 536A, 1969. [Online]. Available: https://patents.google.com/patent/US3450536

[18] Federation of American Scientists,, "The Militarily Critical Technologies List," FFAS.org, p. 12, June. 1998, accessed: 20th April 2020.

[19] J. Fernandez-Berni, R. Carmona-Galan, and A. Rodriguez-Vazquez, "Single-exposure hdr technique based on tunable balance between local and global adaptation," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no. 5, pp. 488–492, 2016.

[20] G. Sicard, H. Abbas, H. Amhaz, H. Zimouche, R. Rolland, and D. Alleysson, "A cmos hdr imager with an analog local adaptation," in *Int. Image Sensor Workshop (IISW'13)*, 2013, pp. 1–4.

[21] R. Shapley and C. Enroth-Cugell, "Visual adaptation and retinal gain controls," *Progress in retinal research*, vol. 3, pp. 263–346, 1984.

[22] M. Schanz, C. Nitta, A. Bussmann, B. Hosticka, and R. Wertheimer, "A high-dynamic-range cmos image sensor for automotive applications," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 7, pp. 932–938, 2000.

[23] G. Eilertsen, J. Kronander, G. Denes, R. K. Mantiuk, and J. Unger, "Hdr image reconstruction from a single exposure using deep cnns," *ACM Trans. Graph.*, vol. 36, no. 6, Nov. 2017. [Online]. Available: https://doi-org.ezproxy.flinders.edu.au/10.1145/3130800.3130816

[24] S. Cho, H. S. Hong, H. Han, and Y. Choi, "Alternating line high dynamic range imaging," in *2011 17th International Conference on Digital Signal Processing (DSP)*, 2011, pp. 1–6.

[25] Y. Bandoh, G. Qiu, M. Okuda, S. Daly, T. Aach, and O. C. Au, "Recent advances in high dynamic range imaging technology," in *2010 IEEE International Conference on Image Processing*, 2010, pp. 3125–3128.

[26] G. Ward, "Fast, robust image registration for compositing high dynamic range photographs from hand-held exposures," *Journal of graphics tools*, vol. 8, no. 2, pp. 17–30, 2003.

[27] S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High dynamic range video," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3, pp. 319–325, 2003.

[28] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[29] Y. Huo and Q. Peng, "Evaluation of hdr tone mapped image and single exposure image," in *2011 International Conference on Computational Problem-Solving (ICCP)*, 2011, pp. 48–50.

[30] A. Rana, G. Valenzise, and F. Dufaux, "Evaluation of feature detection in HDR based imaging under changes in illumination conditions," in *2015 IEEE International Symposium on Multimedia (ISM)*. IEEE, 2015, pp. 289–294.

[31] M. Nilsson, "Smqt-based tone mapping operators for high dynamic range images." in *VISAPP (1)*, 2013, pp. 61–68.

[32] G. Qiu, J. Guan, J. Duan, and M. Chen, "Tone mapping for hdr image using optimization a new closed form solution," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 1. IEEE, 2006, pp. 996–999.

[33] F. Drago, W. L. Martens, K. Myszkowski, and H.-P. Seidel, "Perceptual evaluation of tone mapping operators," in *ACM SIGGRAPH 2003 Sketches & Applications*, 2003, pp. 1–1.

[34] J. H. Van Hateren and H. P. Snippe, "Information theoretical evaluation of parametric models of gain control in blowfly photoreceptor cells," *Vision research*, vol. 41, no. 14, pp. 1851–1865, 2001.

[35] E.-L. Mah, R. S. Brinkworth, and D. O'Carroll, "Bio-inspired analog circuitry model of insect photoreceptor cells," in *BioMEMS and Nanotechnology II*, vol. 6036. International Society for Optics and Photonics, 2006, p. 603613.

[36] J. Davis, S. Barrett, C. Wright, and M. Wilcox, "A bio-inspired apposition compound eye machine vision sensor system," *Bioinspiration & biomimetics*, vol. 4, no. 4, p. 046002, 2009.

[37] N. Franceschini, J.-M. Pichon, and C. Blanes, "From insect vision to robot vision," *Philosophical Transactions of The Royal Society Of London. Series B: Biological Sciences*, vol. 337, no. 1281, pp. 283–294, 1992.

[38] S. Kim, C. Laschi, and B. Trimmer, "Soft robotics: a bioinspired evolution in robotics," *Trends in biotechnology*, vol. 31, no. 5, pp. 287–294, 2013.

[39] B. Hassenstein and W. Reichardt, "Systemtheoretische analyse der zeit-, reihenfolgen-und vorzeichenauswertung bei der bewegungsperzeption des rüsselkäfers chlorophanus," *Zeitschrift für Naturforschung B*, vol. 11, no. 9-10, pp. 513–524, 1956.

[40] A. Benoit, D. Alleysson, J. Herault, and P. Le Callet, "Spatio-temporal tone mapping operator based on a retina model," in *International Workshop on Computational Color Imaging*. Springer, 2009, pp. 12–22.

[41] A. Benoit, A. Caplier, B. Durette, and J. Hérault, "Using human visual system modeling for bio-inspired low level image processing," *Computer vision and Image understanding*, vol. 114, no. 7, pp. 758–773, 2010.

[42] R. S. A. Brinkworth and D. C. O'Carroll, "Robust models for optic flow coding in natural scenes inspired by insect biology," *PLoS Comput Biol*, vol. 5, no. 11, p. e1000555, 2009. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2766641/pdf/pcbi.1000555.pdf

[43] R. S. A. Brinkworth, E.-L. Mah, and D. C. O'Carroll, "Bio-inspired pixel-wise adaptive imaging," vol. 6414. International Society for Optics and Photonics, Conference Proceedings, p. 641416.

[44] R. S. Brinkworth, P. A. Shoemaker, and D. C. O'Carroll, "Characterization of a neuromorphic motion detection chip based on insect visual system," in *2009 International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*. IEEE, 2009, pp. 289–294.

[45] K. Haltis, L. Andersson, M. Sorell, and R. Brinkworth, "Surveillance applications of biologically-inspired smart cameras," in *International Conference on Forensics in Telecommunications, Information, and Multimedia*. Springer, 2009, pp. 65–76.

[46] K. Naka and W. A. Rushton, "S-potentials from colour units in the retina of fish (cyprinidae)," *The Journal of physiology*, vol. 185, no. 3, pp. 536–555, 1966.

[47] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 2002, pp. 267–276.

[48] E. Reinhard and K. Devlin, "Dynamic range reduction inspired by photoreceptor physiology," *IEEE transactions on visualization and computer graphics*, vol. 11, no. 1, pp. 13–24, 2005.

[49] F. Drago, K. Myszkowski, T. Annen, and N. Chiba, "Adaptive logarithmic mapping for displaying high contrast scenes," in *Computer graphics forum*, vol. 22, no. 3. Wiley Online Library, 2003, pp. 419–426.

[50] G. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 3, no. 4, pp. 291–306, Oct 1997.

[51] T. G. Stockham, "Image processing in the context of a visual model," *Proceedings of the IEEE*, vol. 60, no. 7, pp. 828–842, 1972.

[52] V. Naranjo and A. Albiol, "Flicker reduction in old films," in *Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*, vol. 2. IEEE, 2000, pp. 657–659.

[53] P. M. Van Roosmalen, R. L. Lagendijk, and J. Biemond, "Correction of intensity flicker in old film sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 7, pp. 1013–1019, 1999.

[54] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[55] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[56] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/3/279

[57] K. Wada, "labelme: Image Polygonal Annotation with Python," https://github.com/wkentaro/labelme, 2016.

[58] Tzutalin, "LabelImg," https://github.com/tzutalin/labelImg, 2015.

[59] L. G. Brown, "A survey of image registration techniques," *ACM computing surveys (CSUR)*, vol. 24, no. 4, pp. 325–376, 1992.

[60] OpenCV, "Image registration," https://docs.opencv.org, Accessed on: 06/05/2021. [Online]. Available: https://docs.opencv.org/master/db/d61/group__reg.html

[61] M. Abdel-Basset, A. E. Fakhry, I. El-Henawy, T. Qiu, and A. K. Sangaiah, "Feature and intensity based medical image registration using particle swarm optimization," *Journal of medical systems*, vol. 41, no. 12, pp. 1–15, 2017.

[62] D. L. Hill, P. G. Batchelor, M. Holden, and D. J. Hawkes, "Medical image registration," *Physics in medicine & biology*, vol. 46, no. 3, p. R1, 2001.

[63] M. V. Wyawahare, P. M. Patil, H. K. Abhyankar *et al.*, "Image registration techniques: an overview," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 2, no. 3, pp. 11–28, 2009.

[64] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and vision computing*, vol. 21, no. 11, pp. 977–1000, 2003.

[65] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[66] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*. Springer, 2006, pp. 404–417.

[67] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[68] OpenCV, "Feature matching," https://docs.opencv.org, Accessed on: 06/05/2021. [Online]. Available: https://docs.opencv.org/master/dc/dc3/tutorial_py_matcher.html

[69] ——, "Template matching," https://docs.opencv.org, Accessed on: 5th May 2021. [Online]. Available: https://docs.opencv.org/master/d4/dc6/tutorial_py_template_matching.html

[70] H. Harzallah, F. Jurie, and C. Schmid, "Combining efficient object localization and image classification," in *2009 IEEE 12th international conference on computer vision*. IEEE, 2009, pp. 237–244.

[71] F. Sultana, A. Sufian, and P. Dutta, "A review of object detection models based on convolutional neural network," *Intelligent Computing: Image Processing Based Applications*, pp. 1–16, 2020.

[72] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005, pp. 886–893.

[73] C. Geng and X. Jiang, "Face recognition using sift features," in *2009 16th IEEE international conference on image processing (ICIP)*. IEEE, 2009, pp. 3313–3316.

[74] T. Kobayashi, A. Hidaka, and T. Kurita, "Selection of histograms of oriented gradients features for pedestrian detection," in *International conference on neural information processing*. Springer, 2007, pp. 598–607.

[75] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh, "Deep learning vs. traditional computer vision," in *Science and Information Conference*. Springer, 2019, pp. 128–144.

[76] N. O'Mahony, T. Murphy, K. Panduru, D. Riordan, and J. Walsh, "Adaptive process control and sensor fusion for process analytical technology," in *2016 27th Irish Signals and Systems Conference (ISSC)*. IEEE, 2016, pp. 1–6.

[77] I. C. Education, "Deep learning," https://www.ibm.com, May 2020, accessed on: 2021-04-24. [Online]. Available: https://www.ibm.com/cloud/learn/deep-learning

[78] G. Bonaccorso, *Machine Learning Algorithms: Popular algorithms for data science and machine learning*. Packt Publishing Ltd, 2018.

[79] N. O'Mahony, T. Murphy, K. Panduru, D. Riordan, and J. Walsh, "Real-time monitoring of powder blend composition using near infrared spectroscopy," in *2017 Eleventh International Conference on Sensing Technology (ICST)*. IEEE, 2017, pp. 1–6.

[80] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[81] P. Koehn, "Combining genetic algorithms and neural networks: The encoding problem," 1994.

[82] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of physiology*, vol. 195, no. 1, pp. 215–243, 1968.

[83] K. Fukushima, "A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, pp. 193–202, 1980.

[84] S. Saha, "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way," https://towardsdatascience.com, Dec. 2018, Accessed on: 25-04-2021. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[85] S. Hayou, A. Doucet, and J. Rousseau, "On the selection of initialization and activation function for deep neural networks," *arXiv preprint arXiv:1805.08266*, 2018.

[86] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[87] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[88] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[89] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128 837–128 868, 2019.

[90] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[91] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[92] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[93] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[94] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," *arXiv preprint arXiv:1605.06409*, 2016.

[95] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[96] ——, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[97] C. Szegedy, S. Reed, D. Erhan, D. Anguelov, and S. Ioffe, "Scalable, high-quality object detection," *arXiv preprint arXiv:1412.1441*, 2014.

[98] J. Hui, "Object detection: speed and accuracy comparison (faster r-cnn, r-fcn, ssd, fpn, retinanet and yolov3)," https://medium.com, Mar. 2018, Accessed on: 7 May 2021. [Online]. Available: https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae35

[99] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov *et al.*, "The open images dataset v4," *International Journal of Computer Vision*, pp. 1–26, 2020.

[100] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[101] C. A. Rosen, "Machine vision and robotics: Industrial requirements," in *Computer vision and sensor-based robots*. Springer, 1979, pp. 3–22.

[102] Y. Horita, S. Kawai, T. Furukane, and K. Shibata, "Efficient distinction of road surface conditions using surveillance camera images in night time," in *2012 19th IEEE International Conference on Image Processing*, 2012, pp. 485–488.

[103] S. Mangiat and J. Gibson, "Inexpensive high dynamic range video for large scale security and surveillance," in *2011 - MILCOM 2011 Military Communications Conference*, Nov 2011, pp. 1772–1777.

[104] L. Chermak and N. Aouf, "Enhanced feature detection and matching under extreme illumination conditions with a hdr imaging sensor," in *2012 IEEE 11th International Conference on Cybernetic Intelligent Systems (CIS)*. IEEE, 2012, pp. 64–69.

[105] T. M. Pinho, J. P. Coelho, J. Oliveira, and J. Boaventura-Cunha, "Comparative Analysis between LDR and HDR Images for Automatic Fruit Recognition and Counting," *Journal of Sensors*, vol. 2017, 2017. [Online]. Available: http://downloads.hindawi.com/journals/js/2017/7321950.pdf

[106] C. G. Harris, M. Stephens *et al.*, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.

[107] J. Shi *et al.*, "Good features to track," in *1994 Proceedings of IEEE conference on computer vision and pattern recognition*. IEEE, 1994, pp. 593–600.

[108] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 1, pp. 105–119, 2008.

[109] B. Přibyl, A. Chalmers, and P. Zemčík, "Feature point detection under extreme lighting conditions," in *Proceedings of the 28th Spring Conference on Computer Graphics*, 2012, pp. 143–150.

[110] K. F. Wallis, "Seasonal adjustment and relations between variables," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 18–31, 1974.

[111] F. Remondino and L. Zhang, "Surface reconstruction algorithms for detailed close-range object modeling," vol. 36, no. 3, p. O_10.

[112] P. Debevec and S. Gibson, "A tone mapping algorithm for high contrast images," in *13th eurographics workshop on rendering: Pisa, Italy. Citeseer*. Citeseer, 2002.

[113] K. Chiu, M. Herf, P. Shirley, S. Swamy, C. Wang, K. Zimmerman *et al.*, "Spatially nonuniform scaling functions for high contrast images," in *Graphics Interface*. Canadian Information Processing Society, 1993, pp. 245–245.

[114] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 2002, pp. 249–256.

[115] S. Pattanaik and H. Yee, "Adaptive gain control for high dynamic range image display," in *Proceedings of the 18th spring conference on Computer graphics*, 2002, pp. 83–87.

[116] C. Schlick, "An adaptive sampling technique for multidimensional integration by ray-tracing," in *Photorealistic rendering in computer graphics*. Springer, 1994, pp. 21–29.

[117] J. Wang, L. Zhou, Z. Song, and M. Yuan, "Real-time vehicle signal lights recognition with hdr camera," in *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Conference Proceedings, pp. 355–358. [Online]. Available: https://ieeexplore.ieee.org/document/7917112/

[118] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He, "Detectron," https://github.com/, 2018. [Online]. Available: https://github.com/facebookresearch/detectron

[119] ——, "Detectron model zoo," https://github.com/. [Online]. Available: https://github.com/facebookresearch/Detectron/blob/master/MODEL_ZOO.md

[120] A. Boschetti, N. Adami, R. Leonardi, and M. Okuda, "An optimal video-surveillance approach for hdr videos tone mapping," in *2011 19th European Signal Processing Conference*, 2011, pp. 274–277.