EPISTEMIC GUIDANCE OF VISUAL ATTENTION FOR ROBOTIC AGENTS IN DYNAMIC VISUAL SCENES

BY

ARINDAM BHAKTA

A thesis
submitted to the Victoria University of Wellington
in fulfilment of the requirements for the degree of
Doctor of Philosophy in Computer Science

Victoria University of Wellington
2018

Abstract

Humans and many animals can selectively sample important parts of their visual surroundings to carry out their daily activities like foraging or finding prey or mates. Selective attention allows them to efficiently use the limited resources of the brain by deploying sensory apparatus to collect data believed to be pertinent to the organism's current task in hand.

Robots or other computational agents operating in dynamic environments are similarly exposed to a wide variety of stimuli, which they must process with limited sensory and computational resources. Developing computational models of visual attention has long been of interest as such models enable artificial systems to select necessary information from complex and cluttered visual environments, hence reducing the data-processing burden.

Biologically inspired computational saliency models have previously been used in selectively sampling a visual scene, but these have limited capacity to deal with dynamic environments and have no capacity to reason about uncertainty when planning their visual scene sampling strategy. These models typically select contrast in colour, shape or orientation as salient and sample locations of a visual scene in descending order of salience. After each observation, the area around the sampled location is blocked using inhibition of return mechanism to keep it from being revisited.

This thesis generalises the traditional model of saliency by using an adaptive Kalman filter estimator to model an agent's understanding of the world and uses a utility function based approach to describe what the agent cares about in the visual scene. This allows the agents to adopt a

richer set of perceptual strategies than is possible with the classical winner-take-all mechanism of the traditional saliency model. In contrast with the traditional approach, inhibition of return is achieved without implementing an extra mechanism on top of the underlying structure.

This thesis demonstrates the use of five utility functions that are used to encapsulate the perceptual state that is valued by the agent. Each utility function thereby produces a distinct perceptual behaviour that is matched to particular scenarios.

The resulting visual attention distribution of the five proposed utility functions is demonstrated on five real-life videos.

In most of the experiments, pixel intensity has been used as the source of the saliency map. As the proposed approach is independent of the saliency map used, it can be used with other existing more complex saliency map building models. Moreover, the underlying structure of the model is sufficiently general and flexible, hence it can be used as the base of a new range of more sophisticated gaze control systems.

Acknowledgement

I take this opportunity to express my gratitude to my supervisors who supported me throughout the course of my PhD. I am thankful for their aspiring guidance and advice.

Dedication

I dedicate this thesis to my father without whom I would be nothing.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Computer vision is an important element in modern-day engineering applications. Despite recent improvements, the overwhelming amount of incoming data from the camera is a serious hindrance for real-time vision applications. On the other hand, a human vision system can comfortably operate based on only a small portion of the visual input being of high quality. This work explores how human vision like computer models can be applied to dynamic visual scenes to select relevant visual input.

1.1 Scope

Biological vision systems have an incredible ability to attend to task-relevant and important areas within a complicated visual scene. This ability allows an organism to accomplish activities, such as navigation, foraging or detecting possible prey/mates amongst the real world distractions. The directive focus of visual attention to only a selective portion of the available visual information has been metaphorically described as a spotlight illuminating only a small area [140, 152]. This spotlight 'attention' allows the selection of information that is most relevant to the ongoing behaviour [127, 181].

A typical visual environment presents a vast amount of information and it is important for an agent to decide on which part of the available information is to be selected for processing [154]. Selective attention enables the limited processing resources of the brain to be directed to the most task-relevant visual inputs.

Biological vision systems can efficiently choose task relevant visual information that allows the agent to visually navigate through a cluttered environment and perform other vision based task (e.g. pick and place) with ease. For instance, walking through a crowded city street only requires the knowledge of the relative spatial locations, while facial details of pedestrians can largely be ignored [147].

Visual attention is drawn towards salient locations in the visual scene [181]. This, in turn, triggers motor actions which direct the eyes and the head towards the salient visual locations [46, 58]. These swift movements of the eyes, called saccades, allow the focus of attention to move between regions of interest (ROI) within a visual scene. The ROIs are further selectively captured in high-resolution by the fovea and processed by the higher areas of the brain to gain an enhanced understanding of the scene's content.

The anatomy of the eye facilitates selective attention [7]. The visual acuity of the eye varies across the retina. The fovea at the centre of the retina is responsible for sharp central vision. It is surrounded by a larger peripheral area that delivers visual information at lower acuity. The combination of the high and the low acuity regions are responsible for a foveated vision where the amount of detail varies across the image.

In a foveated vision, only a selected portion of the visual scene is observed in high-resolution and the rest of the scene is observed in low-resolution. The centres of the high-resolution area, which is the target location in the visual scene, is called a fixation point.

The role of the non-foveal areas of the biological eye is to produce a low-resolution image of the scene that only encodes information about what is important in the scene. Then the high-resolution fovea is tar-

geted towards those important regions for further inspection. The areas that are not of interest are still processed, but with a reduced spatial resolution [173]. For example, in figure1.1, the red circle amongst the green circles is important and attracts visual attention. In the saliency map literature, those important visual regions are referred to as the salient regions of the scene and the low-resolution image that encodes which location is conspicuous in the scene is called a saliency map.

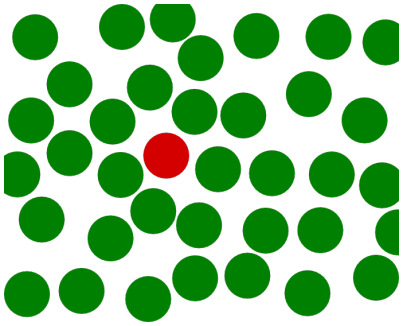


Figure 1.1: A red dot amongst the green dots attract visual attention.

Computer vision suffers from information overload. In a common visual environment, the amount of information to be processed in real-time for a typical image processing task is overwhelming for the available computational resources. Completely processing every scene details to its finest granularity is computationally expensive for an agent.

Given this information overload, a naive computational system cannot select what information to attend to best perform the task in hand. Conversely, biological vision systems accomplish real-world tasks by processing only a small portion of the visual scene. This is a knowledge gap in understanding what information in a visual scene is best to prioritise for further processing. Therefore, how biological systems operate in the real world with limited computational resources of the brain has motivated the saliency map research field [89, 93, 103]. Thus the saliency map research field aims to develop real-time and efficient engineering systems which may be achieved by discovering the secrets of biological attentional

mechanisms.

Over the past three decades, the field of computational saliency and visual attention have attracted a large amount of research interest. There exists a vast amount of literature in visual saliency which reaches from fundamental theories of attention to complicated computational models of human visual attention. The research field has four main directions:

(a) Exploratory research that focuses on discovering and understanding biological vision systems [36, 79, 102, 151, 152, 160, 193]. This class of research work involves experimenting with animals or humans in a controlled setup to study their response with the aim to understand the underlying biological mechanism at work.

(b) Designing computational models that mimic human visual attentional behaviour [20, 75, 107, 109, 122, 126, 149]. The aim of this class of research is to design human-like computational visual attention models.

(c) Research focused on understanding image properties that attract visual attention [144, 148, 155] and statistical properties in a video that attract attention [47].

(d) Designing engineering applications that employ attentional mechanisms [10, 42, 62, 115]. Here the goal is to apply known saliency computation models to practical problems such as visual SLAM, video surveillance and video compression. These models often do not follow any biological motivation but are aimed towards using in engineering applications.

Amongst the above four groups, designing computational models of visual attention have been of special interest as such models allow an artificial system to efficiently select critical information from a complex and cluttered environment [10, 42, 115]. They can also predict visual locations in a given video that are likely to attract human fixation. Applications of

such a system include vision guided vehicles [128], video surveillance [50], content-aware video compression [31, 45, 76, 87, 113], content-aware image manipulation [1, 161], attentional robotics [62], automatic thumbnailing [182], automated image cropping [178] and biological attention inspired simultaneous localization and mapping (SLAM) [65].

There has been an increasing interest to utilize computational human visual attention models in engineering systems to achieve artificial selective attention. This is especially the case for computer vision applications that benefit from selecting the most relevant parts from a visual environment. Therefore, modelling visual attention has been a very active research area over the past years and as a result, extensive research effort has gone into developing computational models, while keeping its engineering implementational aspect in mind.

The intention of modelling visual attention is to design mathematical models that can generate low-resolution images similar to the ones produced by the non-foveal regions of the biological eye. Those low-resolution images would contain salient regions that are enough for an artificial agent to allocate sensory and computational resources to achieve a real-world task.

Contemporary computational models of visual attention find their motivations in two seminal works in psychology: the feature integration theory [184] and the concept that bottom-up and top-down factors work together in human attention [95]. Treisman and Gelade [184] proposed the feature integration theory, which claims that early visual features such as colour, orientation, brightness are processed in parallel, without any conscious effort, and that the perception of an 'object' emerges as a combination of those elementary features at a later stage of visual processing. On the other hand, the seminal work called *The principles of psychology* by William James [95] suggested that two components work in concert on human attention: (i) top-down/endogenous (ii) bottom-up/exogenous. The bottom-up component results from information flowing from the sensors

(i.e. the eyes) to the brain and is responsible for causing involuntary eye movements (e.g. a flashing object catches human attention) [35,91,95,181]. The top-down element accounts for the higher cognitive brain areas directing eye movements as well as modulating the effects of the bottom-up influence (e.g: "I will attend to my favourite soccer player in the playground").

The bottom-up component has gained more research attention due to its promise of providing a deep insight into subconscious feature detectors that produce the saliency map [87,93,108,180,185,197]. These subconscious sensors are believed to have evolved to detect visually contrasting regions of a visual scene [36, 103] Anatomically the retinal ganglion cells (a type of neuron) have a centre-surround structure of complementary colours [67] for detecting colour contrast. For example, some areas have an excitatory centre sensitive to red light enclosed by inhibitory green surrounding cells. The result of this arrangement for colour vision is a final signal that is sent to the brain from these neurones. This causes complementary colours to stand out from their background. An example of this effect is that a red circle would be prominent amongst a large number of green circles (see figure 1.1).

Computational models emulate this subconscious set of salient feature sensors as a set of centre-surround Gaussian filters that are sensitive to contrasts in the visual scene [103]. The excitatory centre versus the inhibitory surround is computationally modelled as the difference of two Gaussian. They are one narrower positive Gaussian and a broader negative Gaussian [69, 192]. This produces a two-dimensional receptive field similar to the biological counterpart.

Koch and Ullman presented the first psychophysically plausible bottom-up computational model [103] using centre-surround Gaussian filters. This model proposed the idea of an estimated saliency map, which represents visual conspicuity of a corresponding scene. This topographical map was called the 'saliency map' in the literature [103] to maintain a conceptual

similarity with the biological systems, albeit, this map is an estimate of the biological counterpart. The saliency map computation approach identifies contrast in visual features such as shape, orientation or colour as conspicuous using centre-surround Gaussian filters. It fuses them with fixed proportion to form the so-called saliency map. The internal state of an agent does not affect the saliency map. Therefore, this map solely depends on the input image. [1]

Finally, saliency maps are exploited by a parallel maximizer (called winner-take-all) to attend to the corresponding visual locations. Once a visual location has been observed, the processing of the visual stimuli at the recent focus of attention is suppressed to avoid the system fixating at the current maximum. This biologically inspired mechanism of blocking previous stimuli is called inhibition of return (IOR) [150]. Blocking the current stimuli encourages the system to explore other parts of the visual scene. Figure 1.2 describes the overall process.

The input block represents an input image to the algorithm. $Feature_1$ to $Feature_n$ represents different features (e.g: colour, object boundaries, intensity) that have been extracted from the input image. The fusion process is represented by an adder-block in the schematic and finally, the output is represented by the saliency map block.

A spatiotemporal saliency map integrates motion features with spatial features. Very often the pixel-wise difference between consecutive video frames is treated as a motion feature. Any swift movement through the visual scene is treated as salient [5, 20, 109, 123, 126]. Referring to the schematic in figure 1.2, one or more of the $Feature_1$ to $Feature_n$ now represents motion features extracted from the input and the input is now a sequence of images (i.e. a dynamic scene).

[1] A differing view computes saliency map by varying relative feature map contributions depending on the current behavioural goals and subjective state of the observer [137, 198].

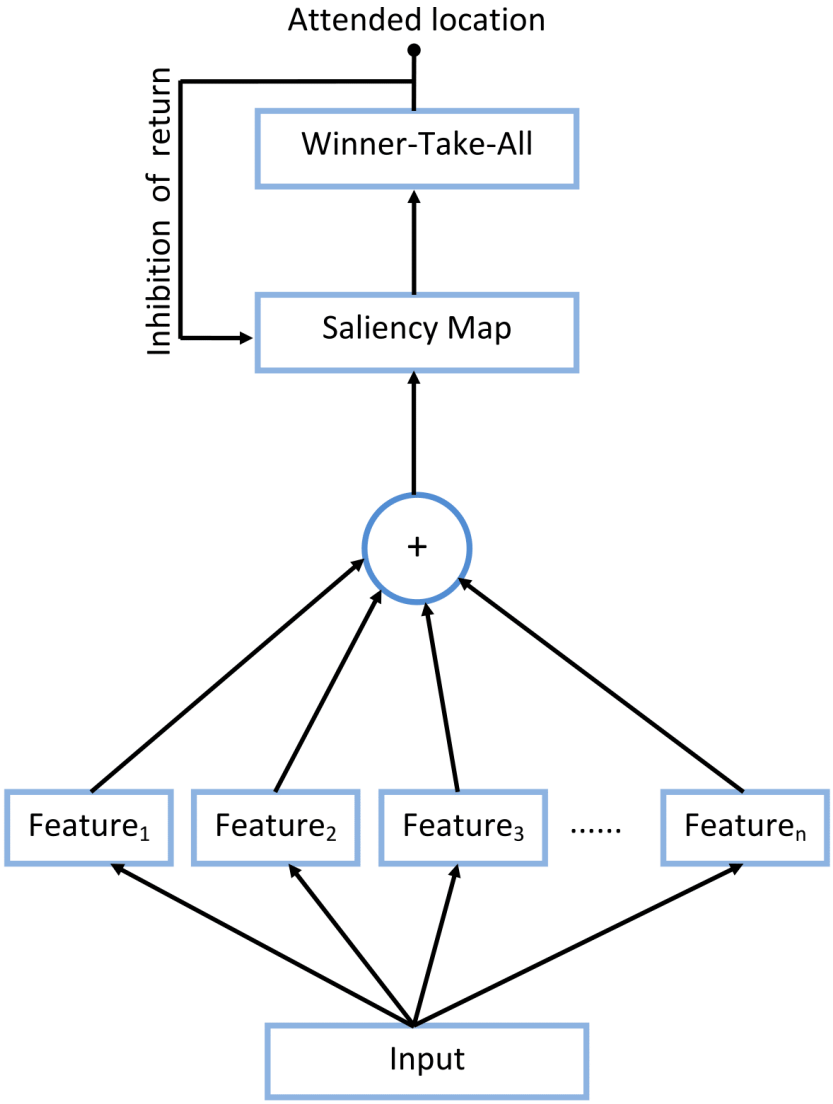


Figure 1.2: A schematic diagram that shows the generic approach of saliency map building algorithms.

Saliency based agents operate in two types of visual scenes: static and dynamic. A static scene involves no temporal change (e.g. looking at a painting) whereas elements in dynamic scenes change over time. This can be either a slow and gradual change or a surprising event occurring in the scene.

The majority of the computational saliency map models were developed for static visual scenes [93,108,141,185]. IOR is a suitable mechanism for static scenario and blocking the recently observed visual location in a static scenario is acceptable. However, dynamic scenes change with time, hence blocking a previously observed part of the scene is inappropriate as visual changes in the blocked region cannot be observed.

1.2 Motivation

Natural scenes are usually composed of several dynamic regions. The movements of visual elements of interest are often embedded in a background which itself could be static or dynamic. In such scenes, the items of interest could be as salient as the background in terms of colour contrast or other static image features. Hence, the key differentiating element in discerning the interesting items is the difference in overall motion associated with them compared to the background.

For example, consider a video of a TV presenter with a static background. The static background itself could have salient colour features. However, an observer would be interested in visiting areas around the face or arms of the presenter. The observer would ignore the static background as it is not informative. In such cases, the visual areas with temporal change are interesting to the observer as they overall present more information. A visual region that does not change often (e.g. the wall behind a news reader) does not need frequent observation but regions that show change need more observations. This strategy enables an agent to distribute limited processing power towards the more informative visual regions. Ideally, an observer would like to revisit the locations of interest frequently to take in information for further processing and ignore the uninteresting static background.

A second example is the movements of an actor amid a dynamic background e.g. one that involves swaying trees, moving water or other ob-

jects such as a crowd. In this case, an observer would be interested in picking up information regarding the actor and would like to ignore the noisy background.

Another example is when an observer is interested in looking for specific targets like human faces amongst a crowd. In this case, the observer would like to look at the visual regions with a human skin like colour features and ignore other regions.

Since such scenes are plentiful in the natural world, successful selection of the interesting regions from the uninteresting background i.e., identifying regions that are spatiotemporally salient, is of strong survival value to an agent.

For all the above examples, the selection and ordering of interesting visual locations are more complicated than just picking out the instantaneous salient locations. Additional knowledge of the overall temporal dynamics of change in saliency along with the knowledge of the instantaneous saliency in the scene would be the key to selecting the proper visual location in such dynamic visual scenes. The additional knowledge of overall temporal dynamics of the scene would guide the agent to generate a visual attention distribution pattern that matches that of the scene.

The agent must be able to use its current and historical knowledge of the temporal dynamics of the scene to reason about where to direct its visual attention. Such an agent would be using epistemic (i.e. based on knowledge) decision making unlike the instantaneous maximiser used in traditional saliency models. The epistemic mechanism would enable the agent to predict the future of a visual target location, as a result, the agent should be able to operate more effectively in a dynamic visual scene.

In contrast, the traditional saliency map approach presents a tight coupling between perception (detecting what is salient in the scene) and action (visual target selection) that is capable of responding to world changes without any complex reasoning and decision-making. Existing literature treats motion features in a similar way to the static features [5,20,109,126].

This approach can detect unexpected or alarming changes in the visual scene but cannot distribute attention to meet the spatiotemporal behaviour of a visual scene.

As the traditional saliency map building approach was intended to work on static visual scenes, it does not incorporate any past knowledge based (epistemic) world models that could help the agent achieve complex reasoning based on past experiences. A purely reactive system design is suboptimal for distributing visual attention in dynamic scenes. In the absence of an extension, the static saliency map framework is incompatible with the necessary temporal behaviour for an agent.

Theoretically, an epistemic visual target selector would be able to select a visual target based on predictions and past experiences. The existing bottom-up saliency models have the valuable ability to detect what is salient in each instant of a visual scene. It is expected that the combination of an epistemic visual target selector operating on top of an existing bottom-up saliency map model can be used to distribute visual attention effectively in dynamic visual scenes. In such case, the feature detectors of the existing bottom-up model will be used by the epistemic visual target selector to distribute visual attention in a dynamic visual scene. This has not been explored before in the saliency map literature.

1.3 Thesis Statement

This thesis argues that a limited resources agent improves its ability to select visual targets in a dynamic scene with the introduction of an epistemic visual target selector driven by the predicted utility of future observations.

To the best of the author's knowledge, this approach is the first attempt to combine an epistemic visual target selector with traditional bottom-up saliency models to distribute visual attention in a dynamic scene. The presented approach is inspired by how biological systems behave but is not intended as a computational analogue of them. Nevertheless, the author

believes that the proposed system is not biologically implausible in its basic structure.

1.4 Research Goals and Objectives

The overall goal of this work is to extend the traditional bottom-up saliency map model to include an epistemic visual target selector for operating in dynamic scenes. This is divided into the following four goals.

1. The first goal is to distribute visual attention in a dynamic visual scene without blocking previously observed locations. The corresponding objective is to use a Bayesian framework that incorporates uncertainty into an agent's knowledge this allows principled ways to reason over the agent's knowledge to calculate the utility of a future observation. Necessary features of this model will include: (i) behaviourally achieving inhibition of previously observed visual locations, (ii) achieving different inhibition time for different parts of the visual scene, (iii) predicting the utility of a future observation by reasoning over an agent's knowledge, (iv) providing a principled framework for learning inhibition time from observations.

2. The second goal is to display a diverse range of useful visual scene sampling behaviours given the same visual scene. The corresponding objective is to design a set of utility functions each of which should be capable of producing a distinct visual scene sampling behaviour. The utility functions' ability to generate different behaviours will be verified by studying the similarity between the attention distribution histogram and the spatiotemporal variance of input videos.

3. The third goal is to learn variance in a dynamic visual scene from observations so that the epistemic target selector proposed in objective 1 can adapt to the scene. The corresponding objective is to use

statistical methods to learn the variance of each pixel from repeated measurements. Multiple statistical algorithms will be explored. Each of them will be applied to videos with known process noise and will be evaluated based on their accuracy of learning.

4. The fourth goal is to devise a mechanism that detects unpredictable visual events that could not be captured by the epistemic target selector. The corresponding objective is to add a statistical distance measure between a new observation and the prediction from the epistemic visual target selector. If this measured distance is beyond a threshold, a surprise is triggered. This surprise detector will be tested for how quickly it can detect surprise compared with the traditional Itti method.

1.5 Major Contributions

The presented model is developed in the context of the traditional bottom-up model to overcome the limitations of inhibition of return based visual scene sampling strategy. This work presents itself as an improved framework for the saliency map research and application. The contributions of this these are the following.

- **This thesis proposes a novel Kalman filter aided epistemic visual target selector.**
  The novel visual target selector encapsulates uncertainty regarding the saliency of a dynamic visual scene into the decision making process of visual target selection. For the first time, uncertainty was introduced in saliency based visual target selection. As a result of considering uncertainty in decision making, an agent behaviourally achieves inhibition of previously observed visual locations. Whereas, existing saliency map computation approaches work along with the add-on inhibition of return (IOR) mechanism. The role of inhibition

of return is to block the previously observed location long enough so that other regions gain visual attention. Blocking any part of a dynamic visual scene is not appropriate as important visual changes may happen in blocked regions of the scene. In contrast, the proposed mechanism will be capable of detecting any visual change faster as it does not block any part of the scene.

Also due to considering Kalman filter aided target selector, different IOR values for different parts of the visual scene can be achieved. This work is the first demonstration of sampling a dynamic visual scene with different inhibition time for different parts of the scene. In contrast, the classical saliency map approach uses same IOR time for all the previously observed regions.

The use of the Kalman filter also provides a principled framework for other algorithms to learn the process noise of the visual scene. Learning process noise enables the Kalman filter to adapt to the visual scene.

- **This thesis proposes a novel surprise detector for the Kalman filter aided target selector.**
  It works along with the Kalman filter based epistemic target selector for detecting events that could not be detected by the epistemic target selector. The proposed surprise detector can detect surprise faster than the traditional Itti model.

- **The thesis contributes two novel algorithms that can learn process noise from observations made with varying measurement noises.**
  To the best of the author's knowledge, the presented work is the first approach towards estimation of process noise from observations taken with varying measurement noises as previous process noise measurement approaches only involved measurements taken with fixed measurement noise. These novel methods can be used to adapt inhibition of return (IOR) time for a visual scene. Whereas, the tradi-

tional bottom-up saliency model needs the designer to manually set the inhibition of return time by trial and error.

This work proposes two methods to learn the IOR time from the observed data. The mathematical derivations, their numerical stability, repeatability and accuracy of estimation were studied.

- **This thesis contributes four novel utility functions for the Kalman filter aided target selector.**
  The four utility functions are aimed to be used with the Kalman filter based target selector for selecting visual targets in a dynamic visual scene. Three of them are based on uncertainty and one is based on uncertainty and the saliency of the scene. Each function produces different visual sampling behaviour. This is the first time in the field of saliency map literature such a range of visual target selection behaviour while operating on the same input was demonstrated. The same framework can be used to develop further utility functions that could generate other desired visual scene sampling behaviours. In contrast, the traditional model can generate the winner-take-all based sampling of the visual scene in descending order of saliency.

- **This thesis improves the classical Itti model by replacing the traditional winner-take-all and inhibition of return mechanisms with the novel Kalman filter aided epistemic target selector.**
  It was shown for the first time that an epistemic target selector based mechanism can predict human fixation better than the traditional bottom-up models on a standard video dataset. This indicates that the proposed approach could readily improve the performance of important applications like video compression, video content extraction etc.

1.6 Organisation of the Thesis

The rest of this thesis is divided into eight chapters. The second chapter, the background, describes the current state of the field. The proposed model chapter details the proposed mechanism and provides the necessary theoretical background. The contribution chapters report the method of experiment and details the experiments along with the results used to demonstrate the proposed system's behaviour. Finally, the conclusion chapter presents overall remarks and scope for future work.

1.7 Chapter Summary

Firstly this chapter presented that the scope of the saliency map research field is to study biological visual target section mechanisms and to apply them to real-life engineering problems. The canonical saliency models are aimed to operate in static scenarios and they perform inadequately in dynamic scenarios. This thesis aims at extending the traditional bottom-up saliency structure for dynamic scenarios. In particular, this thesis presents a knowledge based approach for distribution of visual attention in dynamic scenes.

The following chapter will present the literature review in detail.

Chapter 2

Background

This chapter begins with a discussion on traditional saliency map research and its current status. Then the saliency based engineering applications are discussed. Finally, the limitations of the existing approach while operating in a dynamic scenario are presented.

2.1 A Brief History of Saliency Map Research

The aims of the previous saliency map research were threefold: (i) to gain insight into the visual mechanism of living creatures by psychophysical studies [34,36,43,132,133,190] (ii) apply the learnt mechanisms to develop computational models that can predict eye movements of primates [22, 89, 91, 93, 108, 146, 185] and (iii) to design engineering applications using computational saliency maps [10,36,37,42,115].

The first psychophysically plausible bottom-up computation model for visual attention was proposed by Koch and Ullman in 1985 [103]. A refined version of the same model was later proposed by Itti, Koch and their colleagues, which became the reference point for further development of the saliency map research [93]. The computational model proposed by Itti et al. draws its basic concept from the feature integration theory developed earlier by Treisman and Gelade [184]. This approach combines

multiple low-level image features (e.g. colour, intensity, orientation) to obtain a saliency map. The location of the maximum intensity pixel in the saliency map is identified as the most conspicuous location in the corresponding visual scene. Many extensions of this model have been proposed [107,109,126,185] where the major focus was on adding new image features to better determine the saliency map.

Relatively few attempts have been made to incorporate motion conspicuity into the production of saliency maps. Methods that include motion cues into the saliency map treat motion features in a similar manner to the static image features. A typical approach is to compute a secondary motion-conspicuity map using optical flow or similar ideas, then fuse it with the spatial saliency map.

Examples of supplementing the basic model with motion cues include pixel-wise difference [20], flicker in a visual scene [109, 126] or motion difference computed by centre-surround filters [5]. High saliency is assigned to sections of the input visual scene having such motion. The motion saliency map is then fused with the static-saliency map to obtain the final saliency map.

Such models only detect the predetermined motion cues, despite the much wider variety of motion cues occurring in real-life dynamic scenes.

The prototypical model of bottom-up saliency detection is the Itti model [93]. Due to the generality of the Itti model, it has become the reference point for studying bottom-up computational saliency models and is considered as an important contribution in the saliency map literature. This model takes inspiration from the gradual development of bottom-up saliency computational framework over the past three decades [91, 103, 184]. This development provided the necessary tools to understand the functional behaviour of biological vision systems and utilize them to design computational saliency models. The key contribution of Itti's work is that this model combined multi-scale image features into a single topographical saliency map. This method of computing saliency is inspired by the neu-

ronal architecture of the early primate visual system. This model will be discussed in details in the following section.

2.2 A General Model of Computational Bottom-up Saliency

Itti et al. modelled the combination of visual acuity in the centre, surrounded by blurry peripherals as a difference of two Gaussian distributions; a narrower positive Gaussian and a broader negative Gaussian [93, 103].

Figure 2.1 shows the architecture of the Itti model [93]. The Gaussian based centre-surround filters compute the difference in features to determine contrast in elementary image features. This operation is aimed at detecting locations which locally stand out from their immediate surround. The contrast in features over the entire visual field is presented in multiple feature maps. Within each of the feature maps, locations which significantly differ from their neighbours become highlighted. The features map are computed across multiple spatial scales.

The first set of feature maps compute intensity contrast. They detect dark centres on bright surrounds or bright centres on dark surrounds.

The colour feature maps are also computed by contrasting the centre with its surround for chromatic opponency. Such chromatic opponency is created for the red/green and yellow/blue colour pairs, which is motivated by operations in the human primary visual cortex.

The orientation features are obtained using Gabor filters in a multi-spatial scale pyramidal structure [92, 93]. The orientation feature maps encode local orientation contrast between the centre and surround locations.

For the purpose of generalization, the Itti model allows for adjustment of the number of feature maps computed, the different scales at which

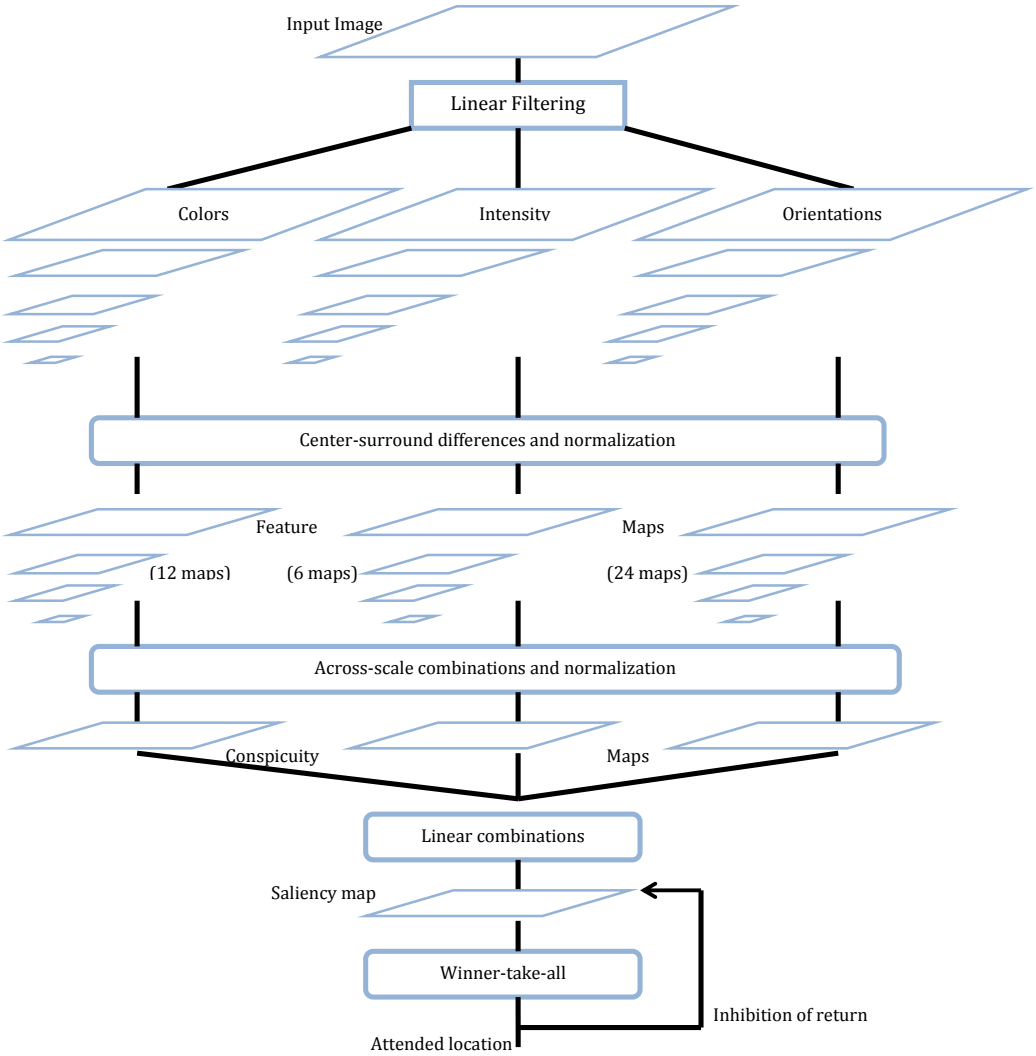


Figure 2.1: The schematic diagram of the Itti model. (this figure has been adapted from L. Itti, C. Koch and Ernst Niebur [93]. Notice that there are three stages involved in generating saliency maps: 1. multi-scale feature maps, 2. three conspicuity maps, 3. the final saliency map.

they are computed, the colour pairs and the type of orientation features. The default model computes 42 feature maps in total: 6 for intensity, 12 for colour and 24 for orientation.

In the next step, feature maps are normalised to a fixed range. This is done in order to eliminate across-modality amplitude differences due to dissimilar feature extraction mechanisms. Normalisation is achieved by globally multiplying the map by the squared difference of the maps global maximum (M) and the average of all its other local maxima ($\tilde{m}$) given by $(M - \tilde{m})^2$.

The normalised feature-maps are then combined into three conspicuity maps which are obtained through cross-scale point-by-point addition. The three conspicuity maps are again normalised to bring individual maps in terms to each other. Finally, all conspicuity maps are linearly combined into a single topographical representation of salience in the visual field called the saliency map [103,141]. The saliency map represents the overall conspicuity of the corresponding visual field.

It is hard to apply Itti like computational models to machine vision tasks because of the difficulty in understanding how simple local features give rise to complex global saliency. Also, the normalisation step used in the model is difficult to understand. Typically, a carefully hand-crafted combination of parameters is used to achieve desired results. Recently Constrained Particle Swarm Optimisation (CPSO) has been employed to determine an optimal weight vector to combine features to obtain a saliency map [172]. Although this approach outperforms existing state-of-the-art methods in different statistical scores (precision-recall, F -measure and area under the curve) it is conceptually unclear due to the lack of a principled approach towards finding optimal feature combinations.

In another approach, saliency map computation has been presented as a regression problem [96]. This approach is based on multi-level image segmentation and uses supervised learning to map a regional feature vector to a saliency score, and finally, fuses the saliency scores across multiple levels to yield the saliency map. As the model uses regional 'backgroundness' as a descriptor, it can not discriminate between the salient regions in the visual scene from their background when the salient regions have

features similar to their background.

A limited set of low-level features can only produce a narrow set of saliency maps. Hence the competence of a model improves by adding more specific features to the existing ones. Consequently, a great deal of effort has gone into developing new and innovative features to produce better performing saliency maps [107, 109, 126, 185]. Although these models perform reasonably when measured against a small collection of ground truth data, it seems that if a model captures too much of the complexity of the world it may become just as cluttered as the real world that it attempts to model.

After obtaining the saliency map it is used to drive visual attention in the scene. A parallel maximizer called Winner-Take-All (WTA) is employed to find the most salient pixel in the saliency map, the location of this pixel corresponds to the most salient location in the visual scene [93, 103]. The WTA always selects the maximally salient location. Hence a system using only WTA in a static scene becomes fixated at the most salient location forever. To encourage further exploration of the visual scene, the visual stimuli from the recent point of focus is inhibited (called Inhibition of return: IOR). This results in WTA selecting the second most salient location after which that stimulus is also inhibited. This process continues until the algorithm explores the complete visual scene.

2.3 Graph-Based Visual Saliency(GBVS)

An alternative model, called the graph-based visual saliency [78], was proposed by Jonathan Harel, Christof Koch and Pietro Perona a decade after the classical Itti model [93]. The authors claim that the model is simple and biologically plausible insofar as it is naturally parallelized [78]. This model introduced an innovative graph-based method to compute visual saliency. First, similar feature maps are extracted from the input images as in the classical Itti model. Then, a fully connected graph is built over each

feature map, and a weight is assigned between the graph nodes. Finally, each graph is treated as Markov chains to build an activation map where nodes which are highly dissimilar to surrounding nodes are assigned high values. The core contribution of this alternative approach is in its second stage of computing the graph based activation map. All activation maps are merged into the final saliency map. Intuitively, this model is mainly based on local context as only locally contrasted features are brought out using the graph based activation map.

According to the published article [78], this model achieves better performance compared with the classical algorithms of Itti & Koch [90, 93]. This model was shown to powerfully predict human fixations on 749 variations of 108 natural images, achieving 98% of the ROC area of a human-based control, whereas the classical Itti model achieves 84%.

This algorithm consists of three steps [78] which will be discussed below.

- **(s1) extraction:** This step is similar to the previously discussed approached of extracting low-level image features like colour, orientation from input images. In this step, feature map or maps (M) are computed from feature vectors that are comprised of single or multiple feature(s) which have been extracted at locations over the entire image plane.

- **(s2) forming activation:** The intent of this step is to form an 'activation map' (or maps depending on the number of features used) using the previously computed feature maps M. Given a feature map $M : [n]^2 \to \mathbb{R}$, the goal is to compute an activation map $A : [n]^2 \to \mathbb{R}$, such that, locations $(i, j) \in [n]^2$ where $M(i, j)$ is somehow unusual in its neighbourhood will correspond to high values of activation A.

  In the context of a mathematical formulation, let $[1 \cdots n] \triangleq \{1, 2, \cdots, n\}$.

  An 'organic' dissimilarity between $M(i, j)$ and $M(p, q)$ was defined

by:

$$d\big((i,j)||(p,q)\big) \triangleq \left|\log \frac{M(i,j)}{M(p,q)}\right| \tag{2.1}$$

Notice the contrast to the previously discussed classical Itti algorithm where an analogous step is accomplished by subtracting feature maps at different scales of centre and surround.

Consider now the fully-connected directed graph G_A, obtained by connecting every node of the lattice M, labelled with two indices $(i,j) \in [n]^2$, with all other $n-1$ nodes. The directed edge from node (i,j) to node (p,q) will be assigned a weight

$$w_1((i,j),(p,q)) \triangleq d((i,j)||(p,q)) \cdot F(i-p, j-q) \tag{2.2}$$

, where

$$F(a,b) \triangleq \exp\left(-\frac{a^2+b^2}{2\sigma_k^2}\right) \tag{2.3}$$

and σ_k^2 is a free parameter of the algorithm.

The weight of the edge from node (i,j) to node (p,q) is dependent on to their dissimilarity and to their closeness in the domain of M. Note that the metric is symmetrical, so the two edges in the opposite direction have exactly the same weight.

A Markov chain on G_A is now defined by normalizing the weights of the outbound edges of each node to 1, and drawing an equivalence between nodes & states, and between edge weights & transition probabilities. The equilibrium distribution of this chain that reflects the fraction of time a random walker would spend at each node/state if he were to walk forever, accumulates weight at nodes that have high dissimilarity with their surrounding nodes. The result is an activation measure which is derived from the pairwise contrast.

The authors call this approach 'organic' because, biologically, individual nodes (neurons) exist in a connected and retinotopically organized, network called the visual cortex. These nodes communicate with each other (in a formation called synaptic ring) in a way which gives rise to emergent behaviour, including fast decisions about which areas of a scene require additional processing, i.e the detection of visual saliency in a scene.

Similarly, the proposed approach exposes connected regions (in terms of F) of dissimilarity (via w), in a way which can in principle be computed in a parallel fashion. The computations can be carried out independently at each node in a synchronous manner, at every time step. In which case, each node simply sums incoming mass, then passes along measured partitions of this mass to its neighbours according to outbound edge weights. The same process happening at all nodes simultaneously gives rise to an equilibrium distribution of mass.

- **(s3) normalization and combination:** The aim of the 'normalization' step of the algorithm is critical. The goal of this step is to concentrate mass on activation maps. If the mass is not concentrated on individual activation maps prior to additive combination, then the resulting master saliency map may be too nearly uniform and hence uninformative. Although this step may seem trivial, it is on some level the very core of any saliency algorithm. The result of this step is concentrating activation into a few key locations.

  Armed with the mass-concentration definition, the authors proposed another Markovian algorithm as follows: This time, the authors begin with an activation map $A : [n]^2 \to \mathbb{R}$ which they wish to normalize. A graph G_N is constructed with n^2 nodes labelled with indices from $[n]^2$. For each node (i, j) and every node (p, q) (including (i, j)) to which it is connected, the authors introduced an edge from (i, j)

to (p, q) with weight:

$$w_2((i,j),(p,q)) \triangleq A(p,q) \cdot F(i-p, j-q) \tag{2.4}$$

Again, normalizing the weights of the outbound edges of each node to unity and treating the resulting graph as a Markov chain computes the equilibrium distribution over the nodes. it is expected that mass will flow preferentially to those nodes with high activation. This algorithm is a mass concentration algorithm by construction, and also one which can be computed in parallel.

Although this algorithm is claimed to be better than the classical Itti , the serious drawback of this model is that its performance was not evaluated on many input images. The study involved testing the model on only 108 images and nine modified version of those images. Modifications were made to change the luminance contrast either up or down in selected circular regions in the original images. In total 749 unique modifications of the original 108 images were used to derive the model's performance. Clearly, this is a small test data-set in contrast to the classical Itti model which was 'extensively' tested with unique set of images [90, 93].

2.4 Engineering Applications of Saliency Models:

This section is aimed to present a general overview of saliency based applications. Domain specific implementation details will not be presented.

There are numerous applications of saliency maps and they can occur in different engineering contexts. Applications of visual attention commonly rely on bottom-up models [121, 127, 169, 170]. For some applications, the saliency maps are the final goal, while for others, saliency maps are only an intermediary step and act as an information-filter for the next

step. The following subsections will discuss some of the popular engineering applications of saliency maps.

2.4.1 Applications Based on Abnormality Detection.

These applications directly take advantage of the detection of abnormal areas in the visual signal. Surveillance, surprising events or defect detection are examples of applications of this category.

Detecting abnormal motion has been in crowded scenes is an important research topic. Authors in [17] proposed a model which detect 'irregular' event from videos given a past dataset of 'regular' videos. The model presented in [17] can also be applied to static images to find generic defects. Also, saliency models were proposed to spot unusual audio [39,125] like a gunshot in a rail-station ambience.

In a fruit grading application presented in [125], saliency models were used for defect detection. Saliency models were used for detecting defects in semiconductor manufacturing [9], metallic surfaces [21] and wafers [134].

Saliency models were also applied to optimize graphical representation of abnormal regions in computer graphics [120].

It is important for video hosting sites to measure the quality of the large amounts of uploaded multimedia for the purpose of video ranking and expected number of views [168]. The ability to predict human fixation is a key element in predicting video quality and it depends on the fact that the sensitivity of the HVS to motion and texture differs significantly between areas of the stimuli attended to and those in peripheral vision [173]. This knowledge can be used to measure perceptual quality of videos [40, 167, 196]. Attention and saliency in videos have recently begun to be considered as a way for video quality assessment [41, 53, 116, 116, 168, 194]. Seshadrinathan and Bovik provide a recent survey [168] of Video Quality Assessment (VQA) approaches.

2.4.2 Applications Based on Normality Detection

The focus of the second category of applications is based on the locations having the lowest saliency scores (i.e. areas of homogeneous, repetitive, usual nature). Those areas correspond to repeating and less informative regions, which might be easily compressed. The main application domain is signal compression.

Unlike the classical compression methods, which distribute coding resources evenly [57, 70, 101, 195], attention-based methods encode visually salient regions with high priority, while treating less interesting regions with low priority [117]. The aim of these methods is to maintain the same perceived quality before and after compression.

In [87], a saliency map was used to smooth videos, which led to higher spatial correlation, therefore a reduced bit-rate of the overall encoded video. An extension of [87], uses a similar neurobiological model of visual attention to generate a saliency map [113] which is then used to guide the bit allocation for encoding. Using the bit allocation model of [113], a scheme for attention video compression has been suggested by [77]. This method is based on visual saliency propagation using motion vectors, to save computational time. Recently, an attention-based efficient image compression patent [207] has been accepted.

The authors in [182] used the Itti algorithm to compute the saliency map [91], that serves as a basis to automatically draws a rectangular cropping window. The Self-Adaptive Image Cropping for Small Displays [32] is based on the Itti and Koch bottom-up attention algorithm but also includes top-down considerations like face detection or skin colour. The authors in [106] presented a saliency distribution based automatic thumbnails creation algorithm. Another algorithm proposed in [206] adaptively partitions an image according to gradient information and saliency. Grundmann et. al. [73] showed methods for video retargeting using motion saliency.

The authors in [158] propose an improved video retargeting by remov-

ing 2D seam manifolds from 3D space-time volumes by replacing dynamic programming method with graph cuts optimization to find the optimal seams. An improved version of this model (reference: [158]) is proposed by [74]. In [71], the authors describe a saliency map which takes the context into account and proposes to apply it to seam carving.

Summarization of images or videos is a term which is similar to retargeting where the purpose is to provide a relevant summary of a video. In [205] the authors used saliency base video summarization to provide a mashup of several videos into a unique video containing the important sequences of all the concatenated videos.

Computational models of attention were also used to address the problem of video skimming [119,121].

2.4.3 Applications Based on Attentive Robotics

The third application category is related to detecting the salient parts of the signal and further processing them. Application domains such as robotics highly benefit from this category of applications. There are three areas where robots can take advantage of saliency models: (i) image registration, landmarks extraction and salient scene feature detection, (ii) object detection and recognition, (iii) robots action guidance.

Image Registration, Landmarks Extraction and Salient Scene Feature Detection: An important requirement of a mobile robot is to know its location. For this aim, the robot can use salient features extraction to find landmarks and register images taken at different times to build a model of the environment [170]. The general process of real-time building of an internal map of the scene is called Simultaneous Localization and Mapping (SLAM) [49,52,187]. Saliency models can help the extraction of more stable landmarks from images which can be more robustly used for SLAM [65].

Salient landmarks that are detected with a visual attention system have a high uniqueness and it has been shown that the repeatability of salient

image regions is significantly higher than for other standard region detectors [61].

Several studies have used salient landmarks for robot localization [65, 139, 145]. More recently, Siagian and Itti presented an approach for scene classification and global localization based on salient landmarks [170].

Bottom-up attention model has been successfully used in content-based image retrieval [127], scene classification [169].

Object Detection and Recognition: Object detection and recognition are important tasks for mobile robots that are especially required when a robot is supposed to manipulate objects or interact with humans [118]. Information about saliency based proto-objects [191] or areas of objectness [3] can help detect objects. Rudinac et al. [159] have presented a saliency based approach in a robotic context for learning and recognising objects.

Further, filtering of features based on saliency map would help object recognition. Papers like [204] and [8] use saliency based feature extraction technique to drastically decrease the number of key points needed to perform object recognition. New trend proposes methods that apply classifiers to regions of interest suggested by a saliency based object detection algorithm [83], in contrast to the classical way of applying classifiers to the entire scene. In another two approaches [138] and [63], the discriminant object features (features that are not in the surrounding) are learned as a set of weights for bottom-up attention models. The approach presented in [179] uses the relative positions of salient points (called cliques) for image recognition.

In [64], the authors have generated object candidates with a method that combines saliency and segmentation. The approach was extended in [82] to image sequences, in which candidate regions were tracked over time to generate sequence-level candidates. Potapova et al. [153] proposed a method that finds objects based on a symmetry-based saliency method that operates on RGB-D data. Martn Garca et al. [68] integrate colour and

depth data to obtain complementary object candidates.

Some groups have used attentive object detection to support object manipulation on robots or robot arms. One of the earliest works on this topic was presented by Bollmann and his colleagues [19] where a Pioneer1 robot used the saliency based neural active vision system to play dominoes. Tsotsos and his colleagues are working on a smart wheelchair based on saliency models to support disabled children [157, 186].

Rasolzadeh et al. [156] use bottom-up and top-down attention to control a KUKA arm for detecting, recognizing, and grasping objects on a table. In [15] and [97] the focuses of attention were used as seeds for 3D segmentation of objects from stereo data.

Guiding Robot Action: In a robotics context, some groups have integrated attentive salient region detection on human-like two resolution camera systems. To simulate the different resolutions of the human eye, several groups use two cameras: one wide-angle camera for peripheral vision and one narrow-angle camera for foveal vision. For example, Gould et al. [72] and Meger et al. [130] determine regions of interest with visual attention in a peripheral vision system, focus on these regions with a foveal vision system, and investigate these high-resolution images along with an object recognition method.

Clark and Ferrier [33] described how to steer a binocular robotic head with visual attention and perform simple experiments to fixate and track the most salient region in artificial scenes composed of geometric shapes. Bollmann et al. [19] have used the neural active vision system (NAVIS) to steer the pan-tilt unit of a domino-playing Pioneer1 robot. Vijayakumar et al. [188] presented an attention system which is used to guide the gaze of a humanoid robot with two eyes. In this work, the authors represented each eye by a wide-angle camera for peripheral vision and a narrow-angle camera for foveal vision [188]. Schillaci et al. equipped a humanoid Nao robot with an attention mechanism based on optical flow and face detec-

tion [166].

Approaches to endow robots with a gesture detecting capability were proposed by Heidemann et al. [80] and Schauerte et al. [163, 164]. An interesting survey on attention based interactive robots can be found in [56].

A robot that learns visual scene exploration by imitating human gaze shifts is presented by Belardinelli [14]. Nagai et al. developed an action learning model based on spatial and temporal continuity of bottom-up features [136].

In addition to the above mentioned applications, 3D saliency is a very promising future research direction [30,110]. Here the main idea is to compute the saliency score of each viewpoint of a 3D model. The best viewpoint is the one where the saliency score is maximum [183]. Marketing is one of the targets of this research topic.

Also, the saliency model of Itti et al. has recently been employed to improve the prediction of packet loss effects [116]. As a side note, bottom-up saliency algorithm has also been implemented on GPU [201].

2.5 Evaluation of Saliency Map Models

Computational models are usually evaluated using synthetic or natural images [76, 108, 123, 149]. The outputs of each model are compared with human performance subjected to the same inputs in terms of how closely the models' outputs mimic human behaviour. There are many notable datasets that are used by researchers; a comprehensive list of datasets can be found at the MIT saliency benchmark [27].

There are three major metrics [24, 105, 112] commonly used in the evaluation of visual attention distribution. They are listed as below:

- **Area under ROC curve (AUC):** In this metric, the saliency map is treated as a binary classifier of fixations at various threshold values. A ROC curve is swept out by measuring the true and false positive

rates under each binary classifier. Finally, the area under the true vs false curve is computed as the measure.

- **Information Gain (IG):** This metric measures the average information gain of a saliency map above a baseline.

  The information gain metric calculates the difference in saliency–measured in bits–at the human fixated locations between two saliency maps. Given a binary map of ground truth fixations Q^B, a saliency map U, and another baseline saliency map V, information gain is computed as:

$$IG(U,V) = \frac{1}{N}\sum_{i}^{N} Q_i^B \Big[\log_2(\epsilon + U_i) - \log_2(\epsilon + V_i) \Big] \tag{2.5}$$

  where i indexes over pixels, N is the total number of fixated pixels, ϵ is a small regularization constant, and information gain is measured in bits per fixation. When the baseline is considered to be the traditional Itti model, this metric will produce the gain in information by the proposed model compared to the Itti model.

- **Kullback-Leibler divergence (KL):** The KL metric is similar to the information gain but this metric evaluates the loss of information between a computed saliency map and an ideal saliency map. The worse saliency computation method would show more loss.

  The KL metric takes a saliency map U and a ground truth fixation map Q^B as inputs, and evaluates the loss of information when U is used to approximate V as below:

$$KL(U,V) = \sum_{i}^{N} V_i \log\left(\epsilon + \frac{V_i}{\epsilon + U_i}\right) \tag{2.6}$$

  where ϵ is a regularization constant.

2.6 Drawbacks of the Traditional Approach When Applied to a Dynamic Visual Scene

Traditional inhibition of return (IOR) systems lack any comprehensive method to define how much area to impede in inhibition of return [93, 103, 188]. The Matlab® implementation of the Itti et. al. model, called the saliency toolbox, provides five predefined choices for IOR area selection [191]. One of those choices is a circular area with fixed diameter and the other four are computed from intermediate results of the saliency map computation (e.g. from the feature map or the conspicuity map etc.). Due to complicated interdependency, it is difficult to understand the dynamics of interactions between the computed saliency map and the IOR area computed from the feature map. Therefore, suitable IOR area is often determined by trial and error [93].

There are several different implementations of IOR. Huelse et al. [85] proposed an approach where a list of all the earlier visited locations was kept as blocked target locations to achieve IOR. This method requires an increasingly large amount of memory dedicated for the purpose.

Other approaches preferred to enhance the importance of target objects in the scene by adding a slowly decaying Gaussian shaped *habituation function* [25]. This function initially increases the saliency of the centre of the field of view and slowly decays the saliency value of the central objects until a new off-centre object gains the attention [162]. Although this mechanism seems useful, it is based on detecting objects in early stages of a computer vision task, which is not robust as object detection algorithms are sensitive to ambient light, camera orientation and are often slow and computationally demanding. In addition, the idea of manipulating the saliency map can prove to be problematic if a change in the visual scene such as the appearance of a new object does not match the update rate of the saliency map (decaying of the saliency value of the central objects).

When applied to a video, the conventional saliency model along with the winner-take-all (WTA) mechanism selects the most salient location in a video frame and keeps staring at that location until it is blocked by IOR or another more salient region appears in the video [93, 103, 188]. Figure 2.2 shows how a WTA based visual target selection strategy selects the most salient location throughout multiple frames of a video. The WTA based approach cannot observe any other locations even with marginally lesser saliency until the most salient location is blocked by IOR.

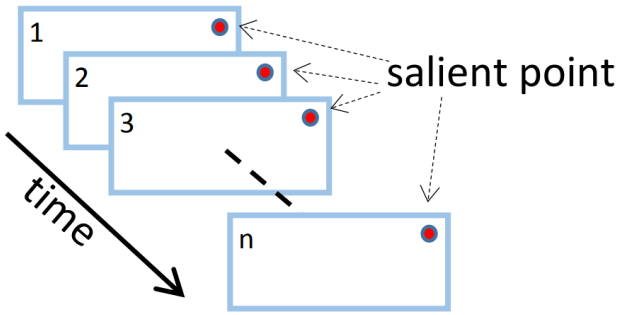


Figure 2.2: A schematic diagram depicting how a winner-take-all based visual target selector operates on a video. Notice that the agent looks at the same salient location (shown as a red colour dot) throughout multiple frames.

In practice, a dynamic visual scene can be thought of constituent visual regions that have distinctly different temporal behaviours. For example, the backdrop behind a solo newsreader does not change during a story. An irregular arch-shaped visual region that does not change with time can be imagined around the news reader's head. A human observer can look at the backdrop once or twice and can realise that it is an inactive region and is not a rich source of information. Hence this region does not require frequent attention. Conversely, the visual region associated with the newsreader requires comparatively more attention. In this fashion, ranking of visual regions based on their temporal behaviour allows efficient visual attention distribution in a dynamic scene. This would require

different inhibition of return times for different parts of the scene.

Existing systems assign equal inhibition of return time (IOR time) to all the regions in a scene. Consequently, equal visual attention is distributed to demanding and inactive visual regions although different visual regions need a different amount of visual attention. Further, the IOR blocking time is chosen to be long enough for the system to sample every salient location.

The shape of the visual area to inhibit is hard-coded in these approaches. Therefore, these systems inhibit a predefined amount of the visual scene area although the size and the shape of the visual stimuli just attended may vary.

The traditional system can not learn the IOR time needed for visual regions as it does not include any framework for learning mechanisms. An ideal agent should learn the temporal properties of the dynamic visual scene to assign region specific inhibition of return times that match the temporal modality of the scene.

Also, blocking an area in a visual dynamic scene is not appropriate as changes in that location cannot be observed by the agent.

2.7 Chapter Summary

This chapter presented a brief history and purpose of the saliency map research. Then it presented the two major models of saliency based visual attention distribution architecture. After that the saliency based engineering applications were presented. Finally, the drawbacks of the traditional approach were discussed.

Although bottom-up saliency is believed to be based on image features, there is an ongoing debate about whether biological attention is focused on objects in the scene or on elementary features (e.g. brightness, colour etc.) [93, 191, 203]. Researchers have mainly concentrated on the neural mechanism of the attentional selection, however, evidence exists for both object-based and features-based attentional selection [13, 16, 84]. The com-

putational advantage of a feature-based model is that it allows attention to be implemented without involving any sophisticated object detection algorithms. Hence this work will focus on features-based attentional selection.

The following chapter will present our conception of visual target selection approach with necessary schematics.

Chapter 3

Proposed model

The last chapter presented how the traditional saliency models work and their drawbacks. This chapter presents the proposed model. However, a short recapitulation of the last chapter is first presented in the following paragraphs.

To reduce the computational burden of processing the vast amount of visual information coming from a visual scene, vision enabled engineering applications can benefit from a biologically motivated two-stage data processing approach. In the first stage, a computationally inexpensive low-resolution map called the saliency map is computed. This map topographically represents the worthiness for further detailed processing of visual information from the corresponding region in the visual scene. Then in the second stage, selective sections of the visual scene that are chosen in descending order of their saliency, are captured by a high-resolution camera for further detailed processing.

Overall, the traditional saliency map building approach works by computing the saliency of a visual scene by combining a set of image features (e.g. colour, orientation etc.). Then the visual locations are attended in decreasing order of saliency using a parallel maximiser called the Winner-Take-All (WTA). WTA causes the system to fixate at the most salient location. Therefore, the traditional models need an additional mechanism

called the Inhibition Of Return (IOR) which impedes previously observed locations to achieve sampling of the visual scene beyond its most salient location. As a result, IOR prevents WTA from fixating and without the IOR the saliency computation models will fixate to the most salient location forever.

Traditional models cannot operate without IOR due to the problem of fixation. On the other hand, the traditional IOR mechanisms are not suitable for dynamic scenes as it blocks previously observed visual locations where important changes may occur, that require immediate attention. In addition, the shape of the IOR area (the shape of the visual area to impede) and IOR time (how long the area is impeded) must be set manually.

In addition, the manually specified IOR time is the same for every region of the visual scene. However, many dynamic visual scenes require different IOR times for different regions, due to variability between different regions of the scene.

The traditional approach along with the IOR fails to achieve sensible behaviour when applied to a dynamic visual scene as a result of the above reasons. No extension on the traditional bottom-up saliency model has been explored to add necessary characteristics needed to operate in a dynamic visual scene.

The next section begins by proposing a hypothetical solution to the problem of applying IOR. In particular, it presents how to build a model of the saliency in a visual world with the inclusion of uncertainty in the model. The inclusion of uncertainty in the decision making in turn allows behavioural generation of IOR without impeding any part of the scene. Also, inferring of the IOR time is formalised due to the use of the framework.

3.1 The Proposed Approach

It is conjectured that a visual target selector that incorporates uncertainty

regarding visual saliency could achieve improved visual target section in a dynamic visual scene. The uncertainty will indicate the agent's level of confidence that the saliency actually lies within the range defined by the uncertainty interval. It allows the agent to assess belief reliability for the purposes of comparison between possible target locations. Hence the uncertainty becomes a measure of the degree of belief about the measured saliency in the visual field.

Every pixel in the saliency map is usually considered as an individual visual target. The agent would have 1200 visual target locations if the saliency map is a 30×40 image. This agent associates individual levels of uncertainty with all the possible visual target locations in the saliency map.

Visual saliency in a dynamic scene is time dependent. The saliency map of a dynamic scene changes as the visual scene itself changes over time. Immediately after an observation, an agent's uncertainty associated with that visual location is low. As time passes, this uncertainty should increase as the underlying visual scene changes. The increase in uncertainty over time indicates the agent's growing lack of confidence in the saliency at the visual location.

When two visual regions are considered for a future visual target, the location with higher uncertainty should gain the visual attention and the recently observed low uncertainty region remains unobserved until its uncertainty increases. Thus, an agent's internal state of uncertainty about the saliency of the external world would act as a restriction to re-observation. If the quality of the internal knowledge is high that region needs no further observation and if the quality is low, (i.e. the agent is uncertain about a visual region) it should be re-observed. In this way, the IOR would become a behavioural outcome of the visual scene sampling process.

Inhibiting known parts of the visual scene from re-observation should flow as a natural outcome of considering uncertainty in the agent's internal representation. Hence the agent should be able to behaviourally

achieve IOR without blocking any part of the dynamic visual scene. Also as an added outcome, the visual area to inhibit should not need any prior assumption as all the low uncertainty internal states will not be observed. Thus, the quality of the agent's knowledge about the world should decide the shape of IOR area.

An epistemic target selector can operate based on its knowledge, hence should be able to overcome the problems faced by the traditional saliency models in a dynamic scene. It can project the utility of a future observation based on its current internal state. This ability is required to select the visual target in a dynamic scene. Figure 3.1 shows a schematic diagram of an epistemic target selector operating on a traditional bottom-up saliency map. Notice that the IOR mechanism has not been added as IOR is expected to be achieved behaviourally.

3.2 Theoretical Background on Sources of Uncertainty

There are two sources of uncertainty.

- **Modelling error:** The results of any computational saliency model are accurate only to a certain degree. Any computational model bears imperfections due to the designer's lack of knowledge about the real world process, unintended or accidental design errors, and random adversarial effects on the real world processes which are not under the agent's control.

- **Measurement noise:** All measurements are inherently noisy in practice due to unavoidable limitations in the measurement process (e.g. sensor thermal and shot noise), as well as disturbances such as fluctuations in operating conditions like temperature, pressure etc. Measurement noise results in variation in repeated measurements of the same quantity.

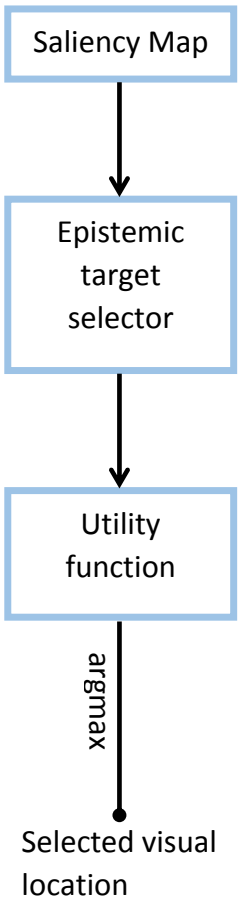


Figure 3.1: A schematic diagram of an agent operating with an epistemic target selector on the saliency map instead of the winner-take-all mechanism. Notice that no inhibition of return mechanism has been added as it is expected to be achieved as a behavioural outcome.

An agent operating on the visual saliency with an epistemic visual target selector experiences the above mentioned sources of noise. This is reflected in the uncertainty associated with its internal belief states of the world. To keep itself updated about the external world, the agent is expected to reduce this internal uncertainty. To do so it takes measurements which are executed as sampling the high saliency regions of the visual

scene. The internal uncertainty reduces as the new measurements are included in the agent's knowledge.

The following section describes how probability distributions embody the visual saliency of a region along with the associated uncertainty regarding the saliency.

3.3 Relevant Probability Theory

Consider an agent that relies on internal representations of the saliency of a dynamic visual scene in order to select visual targets. The internal representations, called the belief states, are the agent's impression of the true state of the world. The internal belief states can be conveniently modelled by probability density functions, which represent quality assessments of the belief state. They are represented as probability distributions on the real world states. The mean of the probability distribution indicates the estimated value of the state (perhaps after updates from multiple observations over time). The dispersion around the probabilistic mean indicates the uncertainty in the belief state.

For example, given a pixel in the saliency map, the mean of the internal belief state is the agent's expectation of saliency at that visual location and the dispersion around the mean of the belief state reflects the agent's level of confidence in that expected value.

In a probabilistic modelling scenario, the true states (denoted $\boldsymbol{x}$) of the world are represented by internal belief states[1]. Each belief state is a probability distribution of the agent's belief about the world. The mean of the belief state distributions is arranged in a vector $\hat{\boldsymbol{x}}$, where i^{th} element of $\hat{\boldsymbol{x}}$ is the estimate of the saliency at the corresponding location in the visual scene.

An agent uses an internal model in order to encode its belief about how the world changes over time. It can use this model to predict a future state

[1] An alternative term used to describe the internal belief is the state estimates.

of the world from the past observations.

The internal model is never error free and also, the world states cannot be measured perfectly due to random noise that affects the measurements. Hence the internal beliefs are approximations of the real-life process and are reliable only within a margin of error.

The reliability of the internal states is quantified by the variance in the estimation of the belief states. The variance associated with each belief state is arranged in a covariance matrix $\boldsymbol{P}$. The diagonal elements of $\boldsymbol{P}$ represent the variance of a belief state and the off-diagonal elements represent the covariance ($\mathrm{Cov}[X, Y] = \boldsymbol{E}[XY] - \boldsymbol{E}[X]\boldsymbol{E}[Y]$) between two belief states.

An agent predicts a future state by extrapolating from last known belief state of the world. The accuracy of an agent's prediction (i.e. the uncertainty associated with the prediction) depends on the extent of extrapolation. Hence the uncertainty of a predicted belief state grows larger with the extent of prediction. However, new measurement reduces uncertainty.

Figure 3.2 illustrates the behaviour of uncertainty in a representative system with and without evidence being obtained from observation. It is a schematic diagram of the mean of a probabilistic belief state and the uncertainty associated with it. In this example, the agent has an internal model that the state is not undergoing any temporal change, but is subject to random perturbation of some known statistical character. The horizontal axis represents time and the vertical axis represents the mean (this value is the saliency of a visual location). The black dashed line shows the expected value (mean of the belief probability distribution) of the state. The underlying process was observed with negligible measurement noise at $t = 0$ and it was re-observed at $t = 17$. No observations were made between these two time instances. The area in blue shows the confidence range in the estimate (variance of the belief probability distribution) up to one standard deviation. This area grows wider with time, indicating the increase in variance over time as the agent loses its confidence in the belief

state.

At the 17^{th} time instant, the real world state was re-observed. The uncertainty level is reduced at that time instant, as an effect of the new measurement, The uncertainty grows again afterwards, as the state remains unobserved again.

The blue coloured error bars give a visual comparison between the uncertainty at different times. Notice the increase in the height of the error bars from left to the right until the new observation, indicating higher uncertainty in internal belief.

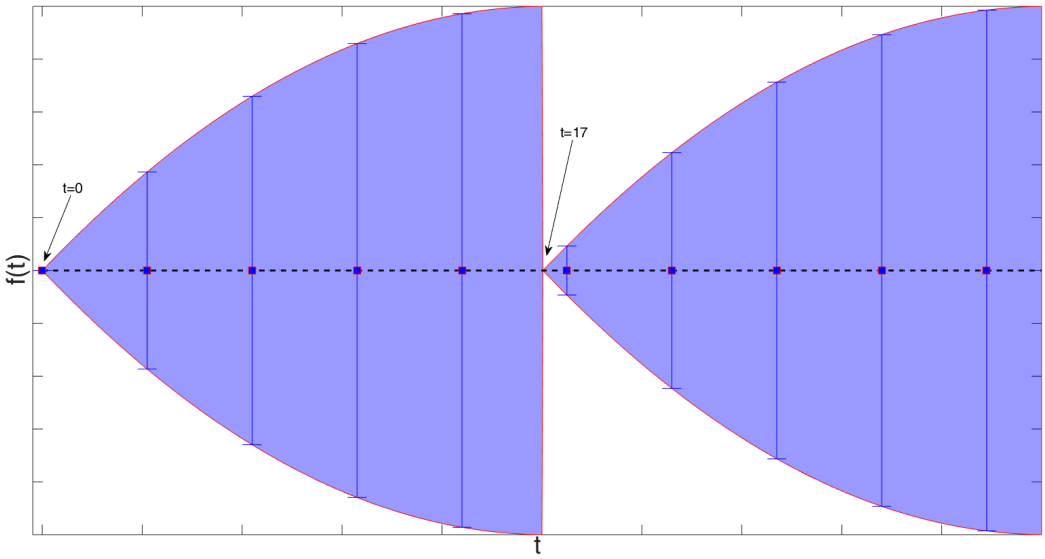


Figure 3.2: A schematic diagram of uncertainty related to a state growing with time. The horizontal axis is time and the vertical axis plots the value of the unobserved state denoted $f(t)$. The area covered in blue shows the level of uncertainty up to one standard deviation. Note that the uncertainty grows larger with time, unless an observation takes place, as at $t = 17$.

The previous discussion shows that to successfully operate as the target selector on a dynamic saliency map, the epistemic target selector must know how to (i) update time dependent internal uncertainty (ii) merge

new noisy measurements with existing internal states (iii) update internal uncertainty in light of the new measurements.

The Kalman filter algorithm is a statistical learning algorithm that can maintain anticipatory belief states in time and can update those beliefs in light of new measurements [99]. The Kalman filter is an appropriate algorithm to combine the prediction of a belief state with a new measurement using a weighted average between the two. The result is a posterior belief state that has a better-estimated uncertainty than either the predicted or the measured state alone. This process is repeated in every time step when a new measurement is available. In the case of a linear system perturbed by Gaussian process and measurement noise, the Kalman filter provides the optimal algorithm. This algorithm is suitable to be used in the epistemic target selector and will be discussed in the next section.

3.4 Kalman Filter

There are two functional processes of the Kalman filter: *a*) predicting the future state estimate of the real world using a mathematical model and *b*) updating the current state estimate in the light of new observations [99]. As the Kalman filter is recursive, future state estimates are calculated from the previous time step's estimates and the current measurement.

The saliency of a location in a visual scene can be thought of as an autonomous linear time invariant system that can be modelled in discrete time as

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{w}_t \tag{3.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is a vector that represents the true state, $\boldsymbol{A}$ is the state transition matrix, which is applied to the previous state ($\boldsymbol{x}_t$) to obtain the next state ($\boldsymbol{x}_{t+1}$) and $\boldsymbol{w}_t$ is the process noise, which is assumed to be drawn from a zero-mean, multivariate normal distribution with covariance $\boldsymbol{Q}$, it is given by $\boldsymbol{w}_t \sim \mathcal{N}(0, \boldsymbol{Q})$. The mathematical equation that describes a zero mean Gaussian is shown in footnote 2. The process noise represents

the unpredictability of the process being modelled and modelling limitations of $\boldsymbol{A}$.

The Kalman filter framework is general enough to model a variety of real-world sequential processes. For example, it could model temporal behaviours like constant velocity, exponentially decaying or sinusoidal variation. The $\boldsymbol{A}$ matrix in the Kalman filter equations epitomise the temporal relationship. Each element in the $\boldsymbol{A}$ matrix is the relationship between a past state and the future belief state. This describes how the mean propagates in time.

In this work, the state variables represent saliency in corresponding visual locations of a dynamic visual scene. Each pixel of the saliency map is modelled as an individual internal state.

At any given time t an observation $\boldsymbol{y}$ is formed using the measurement equation given by

$$\boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}_t \tag{3.2}$$

where $\boldsymbol{C}$ is the observation matrix which maps the true state to the measurement domain. The measurement noise $\boldsymbol{v}_t$ is additive white Gaussian noise[2] with zero mean and known variance $\boldsymbol{R}$, which arises from noise in the sensor used to obtain the measurement. In this case, where the state variables are directly measured, $\boldsymbol{C}$ becomes the identity matrix and the measurement equation becomes $\boldsymbol{y}_t = \boldsymbol{x}_t + \boldsymbol{v}_t$, where $\boldsymbol{I}$ is the identity matrix. The one-step-ahead prediction of the belief states and the associated covariance at any time step t can be calculated using [100]:

$$\hat{\boldsymbol{x}}_{t+1|t} = \boldsymbol{A}\hat{\boldsymbol{x}}_{t|t} \tag{3.3a}$$

$$\boldsymbol{P}_{t+1|t} = \boldsymbol{A}\boldsymbol{P}_{t|t}\boldsymbol{A}^{\mathsf{T}} + \boldsymbol{Q} \tag{3.3b}$$

[2]Here $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ means a random variable v follows the normal distribution that is completely defined by its mean ($\boldsymbol{\mu}$) and variance ($\boldsymbol{\Sigma}$) and is analytically described as

$$p(\boldsymbol{v}) = \frac{1}{(2\pi)\,|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{g}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{g}-\boldsymbol{\mu})}$$

where $\boldsymbol{g}$ is a random variable

where the prediction $\hat{\boldsymbol{x}}_{t+1|t}$ includes measurements only up to time instant t. Notice that the amount of increase in uncertainty in prior uncertainty estimate $\boldsymbol{P}_{t+1|t}$ is determined by the process noise matrix $\boldsymbol{Q}$.

The state update equations are based on the Kalman gain $\boldsymbol{K}$. They are given as:

$$\boldsymbol{K}_{t+1} = \boldsymbol{P}_{t+1|t}\boldsymbol{C}^{\mathsf{T}}\left(\boldsymbol{C}\boldsymbol{P}_{t+1|t}\boldsymbol{C}^{\mathsf{T}} + \boldsymbol{R}\right)^{-1} \tag{3.4a}$$

$$\hat{\boldsymbol{x}}_{t+1|t+1} = \hat{\boldsymbol{x}}_{t+1|t} + \boldsymbol{K}_{t+1}\left(\boldsymbol{y}_{t+1} - \boldsymbol{C}\hat{\boldsymbol{x}}_{t+1|t}\right) \tag{3.4b}$$

$$\boldsymbol{P}_{t+1|t+1} = \left(\boldsymbol{I} - \boldsymbol{K}_{t+1}\boldsymbol{C}\right)\boldsymbol{P}_{t+1|t} \tag{3.4c}$$

where $\hat{\boldsymbol{x}}_{t+1|t+1}$ is the updated belief state that includes the measurement up to time instant $t+1$.

The above mentioned statistical procedures are based on the assumption that the value of states will exhibit a Gaussian distribution. It is important to note at this stage that in real life the observed state variables might not be Gaussian. The deviation from normality might not be an issue for state variables with small variance. Whereas the sensitivity of the proposed method to potential deviations from normality are two-fold.

- The state estimation and the algorithm used (i.e. the Kalman Filter) will become suboptimal.
- As a result of suboptimal state estimation, there will be an increase in the residuals from the Kalman filter model. This will result in an increased amount of surprise.

Once the state estimates and the uncertainties associated with them are calculated, the state uncertainties can be compared with each other using decision theory to find the visual location that offers the maximum reduction in uncertainty. This process will be discussed in the next section.

3.4.1 Bayesian Decision Theory

Bayesian decision theory is used to trade off between various decisions based on a utility function that accompanies each candidate decision. It is

used to rank one decision over another. Ideally, the utility function monotonically increases with the desirability of outcome [104] of a decision.

This refers to the set of assumptions related to ranking alternative target locations based on the degree of utility they provide. These assumptions decide the agent's behaviour and different utility functions can be used to achieve different real-life behaviours.

Unlike the Winner-take-all in the traditional saliency map approach, the presented Kalman filter based mechanism makes a decision based on the mean and the variance of the distribution. There are numerous ways the mean and the variance information can be combined to form a utility function. Since the normal distribution is completely defined by its first two moments, a utility function is dependent only on the mean and the variance of the state estimates. The utility function is maximised to obtain the optimal choice.

Operating in the real-world requires a range of system behaviours. The presented mechanism of making a decision based on a utility function can be used to generate different system behaviours with appropriate choices of the utility function. One example utility function is to reduce the average internal uncertainty of all the states. This would generate a system behaviour that selects visual locations that offer the maximum reduction of average internal uncertainty.

Now, referring back to the figure 3.1, the epistemic target selector is a Kalman filter and the utility function is based on the mean and the variance of the state estimates of the Kalman filter.

3.5 Hypothesis

A Kalman filter aided epistemic visual target selector improves a traditional saliency model's (i) visual attention distribution (ii) surprise detection and (iii) salient target detection abilities.

3.6 Bottom-up Saliency with an Epistemic Target Selector Model

The overall motive of this work is to design a probabilistic framework for visual target selection in a dynamic scene. The framework is aimed to model the dynamic saliency using probabilistic internal belief states and select visual targets by maximising a utility function, that operates on the internal states of an agent.

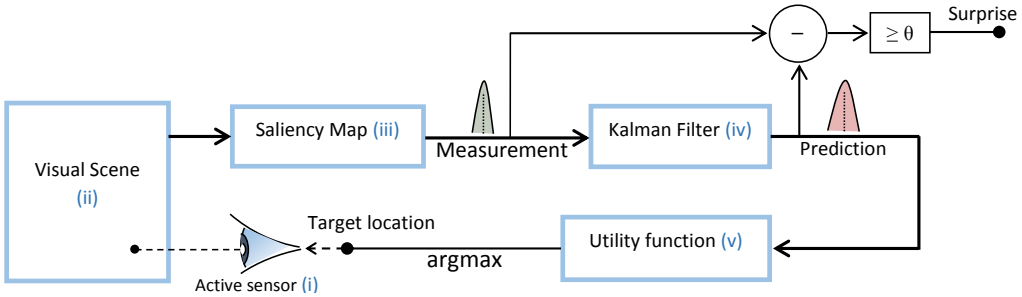


Figure 3.3: Schematic diagram of the proposed method. The black arrows represent the flow of visual information in the model. The visual scene is at the bottom level of the signal processing hierarchy and the information flow starts here. A low-resolution saliency map is produced at every time step from the visual scene. Then the visual information flows from the saliency map towards the Kalman filter based internal model. The new saliency states of the real world update the internal belief states of the agent at every time step. The internal states are used by the utility function to calculate the utility of all possible future observations. Finally, the target location is selected by choosing the visual location that offers maximum utility.

Figure 3.3 shows the schematic diagram of the proposed method. This is a complete system where individual components in isolation will not function as desired. It consists of six conceptual units, namely:

(i) active sensor

(ii) visual scene,

(iii) saliency map,

(iv) Kalman filter, and

(v) utility function

(vi) a surprise detector.

Each of these units is described in the paragraphs below.

(i) **Active sensor:** An active sensor is one that is capable of choosing its vantage point. It accepts positional commands that set its vantage point within the visual scene to captures the visual scene from that viewpoint. The decision about where to direct the sensor comes from a utility based decision making algorithm inside the agent. Human eyes are an example of active sensors, as the eye muscles allow the brain to orient the eyes towards a target. The line diagram of a human eye in the schematic diagram 3.3 represents the active sensor.

(ii) **Visual scene:** The leftmost block in the schematic diagram in figure 3.3 represents the external dynamic visual scene. This scene is an input to the bottom-up saliency model and the active sensor.

(iii) **Saliency map:** The bottom-up saliency model takes input from the visual scene to produce a saliency map. Feature detectors extract desired features from the input images. Image features could be as simple as colour, brightness or as complex as detecting the presence of a shape. The feature detection algorithm needs to compute features that the agent is interested in for a particular task. These features are fused together to form the saliency map. The intensity of each pixel in the saliency map topographically represents the saliency of the corresponding visual region. The intensity of the pixels in the saliency map is in the range of 0 (not salient) to 255 (most salient).

(iv) **Kalman filter:** The third block from the left in the upper-row in figure 3.3 shows the Kalman filter. The Kalman Filter is made up of the following subcomponents.

World model: The world model postulates a real-world process by making causality assumptions on two successive events of the observed process. The causality assumption is a mapping (a mathematical function) from a history at a time instant t into the future at time $(t + 1)$.

This causality relationship is assumed to be time invariant, which means that how the saliency changed to this time instant (t) from the past time instant $(t - 1)$ is the same as how it will change from t to $(t + 1)$. An agent can utilise this causal relationship to predict unobserved future states of the dynamic saliency.

This assumption is aimed to study scenarios that do not change rapidly in time. The gradual changes in the scene are captured by the noise parameter $\boldsymbol{w}_t$ in the equation 3.1. This model is not suited for a scenario where the visual scene abruptly changes. For example, this model is not suited for a video containing a random sequence of images, for example, an image of a cat followed by an image of a baby then an image of a forest and so on. Also, this model is inappropriate for a video that includes scene cuts, though extension to such a scenario would be straightforward.

As a Kalman filter is a stochastic modelling approach, it is capable of incorporating the random adversarial effects in a temporal relationship. This is incorporated in the process noise in the Kalman filter. The process noise sums up modelling limitations, adversarial effects on the real-world process from unknown sources as a set of additive noise parameters.

There are two ways to chose process noise in a Kalman filter: (i) these parameters can be inferred from the noisy observations (ii) or they

can be obtained from theoretical knowledge of the underlying process that is being modelled. In practical applications, the process noise is usually inferred from observations as it is not always possible to have good theoretical knowledge about the underlying process [2, 4, 12, 26, 29, 44, 48, 51, 55, 66, 81, 111, 131, 135, 142, 143, 171, 174, 175, 200].

The temporal causality relationship is encoded as a matrix in a Kalman filter. This work mathematically denotes it as matrix $\boldsymbol{A}$ in equation 3.1. To adopt a Kalman filter to this work, the elements in this matrix represent the relationship between the internal state of the agent at two consecutive time steps.

Data fusion algorithm: This algorithm is also a part of the Kalman filter equation. It updates the internal belief states in the light of new measurements. Once the current measurement is observed, the internal estimates are updated using a weighted average of the internal state of the agent and the new measurement. If the new measurements are accurate then more weight is given to them. If the new measurements are less accurate than the current belief more weight is given to the prediction obtained from the causal relationship.

One common sensor arrangement is to have two types of sensors observe the visual scene at each time step in the presented model. The low acuity sensor and the high acuity sensor. The data from the low acuity sensor is used to build the saliency map and presents low-resolution observations of the visual scene. Whereas, the high acuity data is used to further analyse the saliency regions of the scene. The agent must supervise the intake of both these qualities of data. The data fusion algorithm must somehow rationally combine these two different qualities of data while updating the internal states.

(v) **Utility function:** The utility function incorporates a hypothesised relationship between a potential action of looking at a target location

in the visual scene and the pay-off in the outcome of the consequential observation [104]. Every target location in the visual scene is assigned a utility. Hence an agent can make the decision of where to attend to in the visual scene by maximising the utility values.

At every time step t the utility function calculates the payoff of a future observation at $(t+1)^{\text{th}}$. This is possible due to the prediction ability of the epistemic target selector. At every time step, the prediction equation of the Kalman filter is used to anticipate the state estimates one time step ahead (i.e. at $t+1$). The mean and the variance of the anticipated internal states are used to predict the utility of future observations.

In the previous discussion in the chapter, it was assumed that the utility function was simply the variance of the estimated salience. That is, it was assumed that attention was allotted to the location with the highest variance. However, this can be generalised to many possible utility functions that depend on both the estimated state and its variance (and possibly other variables).

The utility is calculated for all the possible target locations. That is, the utility is calculated for the possibility of centring an observation on every pixel in the saliency map. This allows the agent to compare all possible future visual target locations at each time instant. A visual location is selected if it offers the highest utility for the next observation.

In this way, the agent gets a reward by taking the action of looking at a visual target. The reward is defined by the utility function such as reduction in uncertainty.

This action-reward relationship represents the agent's intention by ranking or setting the preference for one outcome over another. It is a set of assumptions related to ordering the set of all possible visual locations, based on the degree of utility they provide. For example,

an agent whose intention is to spot circular objects in a visual scene would prefer round shaped objects over other shapes.

The action-reward relationship can be defined in multiple ways according to the requirements of the task at hand. For example, finding a specific shape or certain colour may have a high utility in a particular visual search task. In another case, reducing the inertial uncertainty of the agent could be the intent. These two intentions of the agent would require different utility functions.

The presented framework allows straightforward selection and design of such utility functions. This is because the choice of the utility function is independent of the other functional units of the proposed observation mechanism. This allows the agent to display a range of behaviours according to the need.

(vi) **A surprise detector:** The limitation of the Kalman filter based model is that it cannot cope up with sudden changes in the scene. The purpose of the surprise detector as an additional block is to detect sudden changes in the visual scene that could not be captured by the Kalman filter. This functional unit is shown on top of the Kalman filter block in figure 3.3. This block takes input from the new measurements and the Kalman filter predictions. Then the difference between the two is computed using a suitable metric. A surprise is triggered if the difference between the prediction and the new measurement is beyond a pre-set threshold (possibly a large threshold). Therefore a surprise is defined as a large deviation from the prediction.

An Active Measurement Cycle: This is not a functional unit but describes a full cycle of information flow starting from the bottom layer, the external visual scene, and ending at assimilation of new measurements into the internal belief states. The complete measurement cycle consists of

four steps. At the first step, the internal model is used to predict future belief states. In the second step, the utility function is used to decide which visual location offers the maximum payoff. This location becomes the next visual target in the dynamic visual scene. New positioning commands are sent to the active sensor. In the third step, the active sensor orients itself and collects new measurements of the world. Finally, new measurements update the world model by providing new information which is incorporated into the belief states. This cycle then repeats in the next time step.

3.7 Foveation Profile:

All real image-like sensors show spatial variation in their properties that must be accommodated within the Kalman filter structure. Some sensors, such as the human eye, show spatially varying resolution, which means that the central regions are more informative than the periphery. Others exhibit spatial variation in noise, spatial variation in sensitivity or faulty sensor elements distributed across their sensing area. Each of these possibilities generates a different pattern of trustworthiness of the data generated across the sensing area.

This work deals with different sensor properties, such as sensor noise, by using a spatially varying measurement noise called the foveation profile. It is implemented by choosing different values for the measurement noise matrix $\boldsymbol{R}$ in the Kalman filter. The foveation profile therefore allows modelling a wide range of sensor types without making changes to the underlying Kalman filter implementation. The shape of the foveation profile can be square (e.g. for a computer camera), a concave shaped (e.g. human eyes have a concave foveation profile) or any other arbitrary shape. It is a property of the measurement qualities provided by the sensors. Notice that the foveation profile could be used to selectively isolate one faulty pixel by raising the corresponding measurement noise parameter high.

At every time instant an agent receives two qualities of visual inputs: (i) low-resolution images from peripheral vision for computing saliency and (ii) high-resolution to inspect salient locations in details. Visual samples from peripheral regions with decreased spatial resolution produce an average over a larger area of the scene. In these measurements, the individual constituent measurements of the average will considerably differ from the mean. Multiple measurements taken using the peripheral vision would show higher variance compared with measurements taken using high-resolution camera. Therefore, the two qualities of input visual data should not be equally trusted for the purpose of inclusion into the agent's belief states.

The Kalman filter aided data fusion algorithm which integrates new measurements with existing beliefs encounters both the low and high-resolution data. It must somehow incorporate these two different resolutions of data into the agent's belief state. It should trust the high-resolution data more than the low-resolution data.

In contrast to the usual implementation of the Kalman filter, where every measurement is trusted equally, the data fusion algorithm becomes somewhat difficult in this case due to the variability in data resolution. In this case, the two measurements with different measurement noises must be trusted differently. The high acuity high-resolution data must be trusted more than the low-resolution data. Therefore, the measurement noise considered in the matrix $\boldsymbol{R}$ in the Kalman filter equation 3.4a must reflect the quality of the incoming data.

A simplifying assumption of modelling the phenomena of varying visual acuity with a spatially varying measurement noise profile is proposed. Here the measurement noise for the Kalman filter is decided based on the source of the data. Measurements coming from the low-resolution source is assumed to produce high measurement noise and the high-resolution camera is believed to produce low measurement noise. It is important to quantify the quality of measurement noise so that an agent knows how

accurate its measurements are and can incorporate that into its algorithm.

The Kalman filter algorithm would trust the low measurement noise data more than the high measurement noise data. This means that the internal state transition model is trusted more for the states that are updated with the low-resolution data.

Usually, the low-resolution and the high-resolution cameras are separate entities. At every time instant, the low-resolution camera observes the entire scene. The low-resolution image is only used to produce the saliency map whereas the high-resolution camera is used to capture the salient sub-sections of the visual scene. The high-resolution images, carry detailed information, hence they are used for further understanding and inspection of the visual scene.

For the practicalities of implementation, this thesis would assume one sensor with a varying measurement noise. This assumption does not affect the decision making and visual target selection of the agent, as the agent's decision making is not dependent on the number of sensors.

Inspired by the foveation characteristics of the human eye, a so-called 'foveation profile' is added to our model, which captures the particular characteristics of high-resolution central and low-resolution peripheral image. This work uses a simple sensor model that exhibits the characteristics of incorporating high and low-resolution data in the belief states. This profile allows us to express our relative confidence in the different locations within the sensed area. This foveation profile can be visualised as a central trough in measurement noise, which gradually increases with distance from the centre of attention.

Figure 3.4 shows a foveation profile. Notice that the measurement noise increases with distance from the fixation point at pixel no 50. The agent observes the entire scene at each instant with increasing noise away from the location of fixation.

The internal state of the agent is updated in light of the new measurements at each time step. If the new measurement is accurate then more

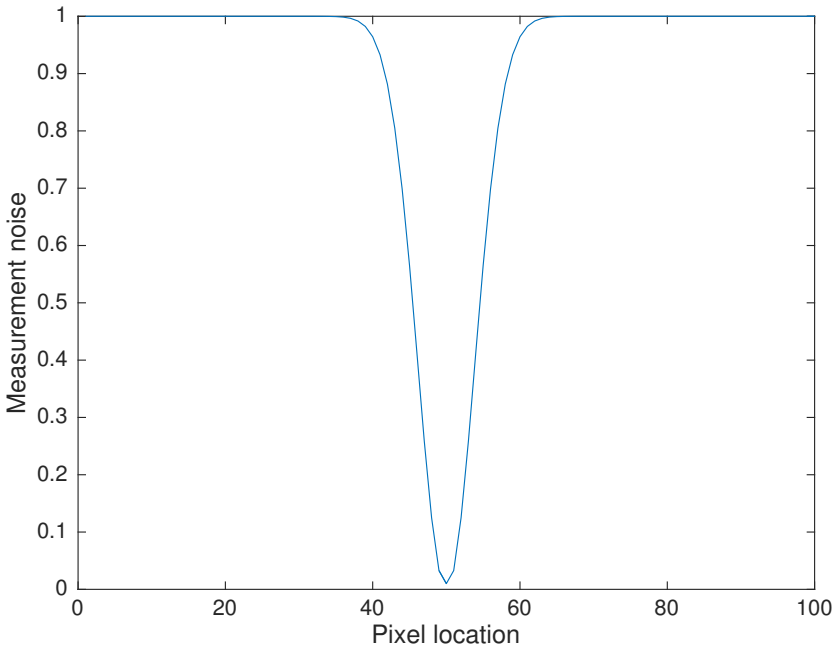


Figure 3.4: An example foveation profile. The vertical axis shows measurement noise and the horizontal axis shows pixel location. Notice the increase in measurement noise over distance from the fixation point.

weight is given to it.

The measurement noise is assumed to be additive white Gaussian noise, whose variance increases with distance from the centre point of fixation. The variation in measurement noise is modelled by choosing different values for the elements in the measurement noise matrix, the $\boldsymbol{R}$ matrix of the Kalman filter equations (equation: 3.4a).

It is assumed that the measurements can be directly mapped to the internal belief states. Therefore, the Kalman filter can combine measurements with the internal beliefs directly. This corresponds to the $\boldsymbol{C}$ matrix, which defines the mapping between measurement and internal state, equal to the identity.

The implementation of a foveation profile in the proposed system is straightforward as the consideration of the measurement noise in the Kalman

filter equation is in the form of a matrix. Individual elements of the measurement noise matrix can be adjusted to achieve a desired foveation profile shape.

The magnitude of measurement noise and its spatial distribution are properties of the low and high-resolution sensor combination. Considering separate measurement noise for the low and the high-resolution cameras allows studying the effect of foveation profile on the behavioural outcome of the agent. Also, under the proposed framework, it is straightforward to test different foveation profiles to study the behavioural outcome of the agent. This facilitates the engineering demand for testing an algorithm with different sensors.

3.8 Adapting the Kalman Filter to Model Saliency

The Kalman filter is implemented by a set of equations (discussed in details in the section 3.4) that allows it to *a*) predict based on the internal model and *b*) update the internal model in light of new measurements.

In this work, the Kalman filter is modelling the saliency of visual locations in a dynamic visual scene. The saliency of corresponding visual locations is mapped into the saliency map by a bottom-up saliency computation algorithm, that is computed from the input image at every time instant. The features used to compute the saliency map could be simple features like the pixel intensity or complex features like the shape of an object.

In this thesis, when required, Itti's bottom-up saliency model will be used as it is the most commonly used bottom-up saliency model [93]. Also, it is straightforward to implement the Itti model due to the availability of its source code [191]. However, the proposed method is not limited to using the Itti model and in the conclusion section of the thesis in chapter 9, the generality of this approach will be discussed.

The internal belief states represent the agent's belief about the external world. The belief state of an agent at time t is all of the information the agent has remembered from the previous times. It encapsulates all of the information about its history that the agent can use for current and future to command the active sensor. At any given time t, an agent has access only to its belief state and its measurements.

For the purpose of this thesis, the internal belief states will be used to represent the bottom-up saliency in a visual scene. Each pixel in the saliency map will be modelled by a separate belief state.

In most real scenarios, it is expected that the value of one pixel would depend on its neighbours. In such cases, the target locations in a saliency map could be grouped into super pixels based on some attribute of the scene and each group could be assigned one belief state. However, computing correlations between all the possible pixel-pairs in a visual scene to achieve such grouping would be computationally very expensive. Therefore, all the states are assumed to be independent, which results in reduction of computational burden and simplified implementation of the Kalman filter based model.

The drawback of this assumption is that the agent will not be able to make an inference about other neighbouring locations based on an observation of a location. For example, an agent does not need to observe every pixel on a wall, instead observing one portion of the wall gives an agent information about other portions. This means that the agent will not be able to exploit the natural spatial regularities of a visual scene. However, a correlated visual scene can be reduced down to smaller number of uncorrelated pixels [11, 165, 202]. The aim of this work is to demonstrate the proposed system's behaviour on such a set of uncorrelated pixels. Hence the assumption of independence of pixels made here will not hinder the system's operation in visual scenes with correlated pixels.

In the case where one belief state is considered to be dependent on other belief states, the computation becomes complicated. Under this sce-

nario, one might need to compute the correlation between one pixel with all the pixels in a neighbourhood. One might want to start with dividing the entire visual scene in equal parts. This would reduce the number of state variables, but is still suboptimal as multiple of those equally dividing regions might belong to one object in the scene. Hence there is no straightforward way to compute which pixel belongs to what neighbourhood and this is an interesting problem for future studies.

Choosing a total number of belief states to be equal to the number of pixels in the saliency map translates to the number of dimensions of the state vector $\boldsymbol{x}$ being equal to the total number of pixels in the saliency map (n). Another view of $\boldsymbol{x(t)} \in \mathbb{R}^n \times \mathbb{Z}_+$ would be to see the time dependent saliency map as a set of n individual time-series.

All pixel intensities (a 2D snapshot image of the saliency of the dynamic visual scene) at any given time are collective samples from the sequence of time-series data points arranged in space. Each belief state in the Kalman filter represents the agent's internal belief of a corresponding state of the visual scene and the $\boldsymbol{A}$ matrix encodes how the time series unfolds in time.

The $\boldsymbol{A}$ matrix is assumed to be an identity matrix. This means that the agent believes that there is no change in saliency over time. This assumption is to study scenarios that do not change rapidly in time. The gradual changes in the scene are captured by the noise parameter of the Kalman filter as presented in the equation 3.1. This model is therefore not suited for scenarios where the visual scene contents change abruptly, such as for videos including scene cuts.

3.9 Utility Based Decision Making

Gaussian belief states are completely represented by the first two moments: *a*) mean and *b*) variance. Hence these two statistical moments can be used to define a utility function according to the desired behaviour.

One elementary desired behaviour of an agent is to take measurements to reduce the average internal uncertainty over all the belief states. This would result in an attention distribution that it would keep the agent updated about the world. The agent's purpose is to reduce internal uncertainty (hence keep up to date knowledge of the world) by taking measurements with a given combination low and high acuity sensors. At each instant, the agent is interested in placing the high acuity camera at the location that promises maximum reduction of overall internal uncertainty.

In general, a utility driven system should maximise a utility function of the form

$$u_{avg} = \frac{1}{n} \sum_{i=1}^{n} u_i \tag{3.5}$$

where the u_i is the utility measure of the individual visual target regions, u_{avg} is the average utility of considering one action and n is the total number of observed pixels. This equation computes the average utility of looking at a pixel and its the surrounding n pixels.

A visual scene with n target regions has n such average utility measures as there is one for each potential future fixation point. The average utility measures can be arranged into a vector $\boldsymbol{u} \in \mathbb{R}^n$. Each element in this vector is the average utility of fixating on the corresponding region.

An action a is the act of looking at a target visual location in the visual scene with a given foveation profile. In this context looking at a region can be thought of as placing the profile-centre of the foveation profile on a target region.

Hence, finding the target visual location for the next time step can be expressed as a maximisation problem shown below:

$$a^\star = \arg\max_a f(a) \tag{3.6}$$

where a is the anticipated future action of choosing a point of fixation, $f(a)$ is the outcome of that action and $a^\star$ is the optimal action that provides maximum utility. The utility based reasoning block in the schematic 3.3 shows this functionality.

The visual region that is observed with the lowest measurement noise results in the maximum reduction in the corresponding uncertainty associated with its belief state and the nearby regions are observed with progressively higher measurement noise.

At each time step the Kalman filter prediction equations are used to estimate the future state of the world and then the agent simulates placing the centre of the foveation profile at each possible fixation points. With each placement considered, the mean and the variance of state estimates are calculated. A given scene of n target locations, would have n means and n uncertainties. Potential future actions are compared against each other based on a specified utility function, and the observation location having the highest utility is selected.

The effects of mean, variance and choice of $\boldsymbol{Q}$ on the system behaviour are investigated in experiments chapter 4. The methods of experiment (chapter: 4) for this thesis work has been designed to study the effect of these parameters on the system behaviour and are discussed in the next chapter.

3.10 Model Parameters Inferring:

Given the probabilistic framework for observing the saliency map at each time step, an additional functional block that infers parameters of process noise for the internal model from the observations can be designed. The purpose of inferring is to adjust process noise to adapt to the temporal characteristics of the visual scene. Two types of inferring algorithms are envisaged at this stage:

(i) Batch inferring- where model parameters are updated after a pre-set number of new measurements have been collected. This approach infers the noise parameters from a *set* of measurements.

(ii) Online inferring- where noise parameters are inferred with every it-

eration of the algorithm (i.e. the parameters are updated with every new measurement).

The mechanism in effect would allow the agent to infer inhibition time for the visual scene. Different inhibition time can be applied to different visual target regions as the noise parameters are inferred separately.

3.11 Chapter Summary

This chapter presented the proposed model and how the Kalman filter equations fit in with the proposed working principle. The philosophical understanding of the utility functions was also discussed. It was presented that given this proposed framework, suitable measurement noise parameters (i.e. foveation profile) and utility functions can be chosen easily. The foveation profile plays an important role in system behaviour and it will be a parameter of investigation in the design of experiments.

Chapter 4

Inhibition of return as an emergent behaviour

A visual attention distribution mechanism based on saliency-maps looks at a visual scene in descending order of saliency. A parallel maximizer called the Winner Take All (WTA) finds the most salient location and samples that location. Observed locations are then blocked to prevent the agent from fixating at the most salient location. The mechanism of blocking previously observed locations is called inhibition of return (IOR) and is analogous to the biological mechanism of the same name. IOR does not arise naturally from the WTA formalism but must be added to the basic WTA architecture. As discussed in the background chapter 2, previous approaches to the generation of IOR have relied on maintaining a list of past observed locations which are vetoed as possible saccade targets for some predetermined time. In contrast, this chapter argues that the proposed system achieves inhibition of return as an emergent behaviour and as such requires no addition to the underlying framework to show the desired IOR behaviour.

The effect of process noise and foveation profile on the system's behaviour of visual attention distribution is presented. This chapter is divided into three sections: The first section shows that an agent that oper-

ates based on internal uncertainty regarding the visual locations will avoid looking at known visual regions, hence inhibits returning to past observed locations. The second section studies the effect of the choice of a foveation profile on the system behaviour and finally, the last section demonstrates the effect of process noise on the system behaviour.

4.1 Experimental Method

The proposed mechanism is implemented using the MATLAB® programming language on an Arch Linux platform.

In the initial phase of the experiments, a one dimensional saliency map is used for simplicity and clarity of presentation. This simplification does not hurt the purpose of the presentation as the dimensionality of the input saliency map does not affect the model's decision making process. That is, our agent acts on a $1D$ world rather than $2D$ images, albeit the proposed method can scale to any dimensionality given sufficient computational resources.

To create an agent who's visual attention distribution matches the distribution of spatio-temporal variation in saliency of its environment. The aim here is to demonstrate that an agent's visual attention distribution can match the spatio-temporal variation in saliency of its environment. Also, it is demonstrated that inhibition of return is a behavioural outcome of the proposed method using mathematical analysis. Then heat-maps of inhibition time over the visual space and frequency histograms of visual attention distribution is used to demonstrate the effect of process noise on decision making. Further, the effect of foveation profile on the agent's inhibition of return behaviour is examined. Although the last three objectives are presented sequentially, they are the simultaneous outcomes of the experiment.

To simplify initial experiments and mitigate complications in real data that could complicate the interpretation of results, we decided to use syn-

thetic saliency maps rather than obtaining saliency maps from input images. This allows bypassing the implementation of low-level feature extraction and de-noising algorithms without compromising on the prime focus of the work. The first results chapter 4 uses a simple $1D$ synthetic saliency map. The next section describes the method used to generate the synthetic saliency map.

4.1.1 Generation of Synthetic Saliency Map

We designed a method for creating synthetic saliency maps with known number and width of salient peaks to test the proposed method. We as-

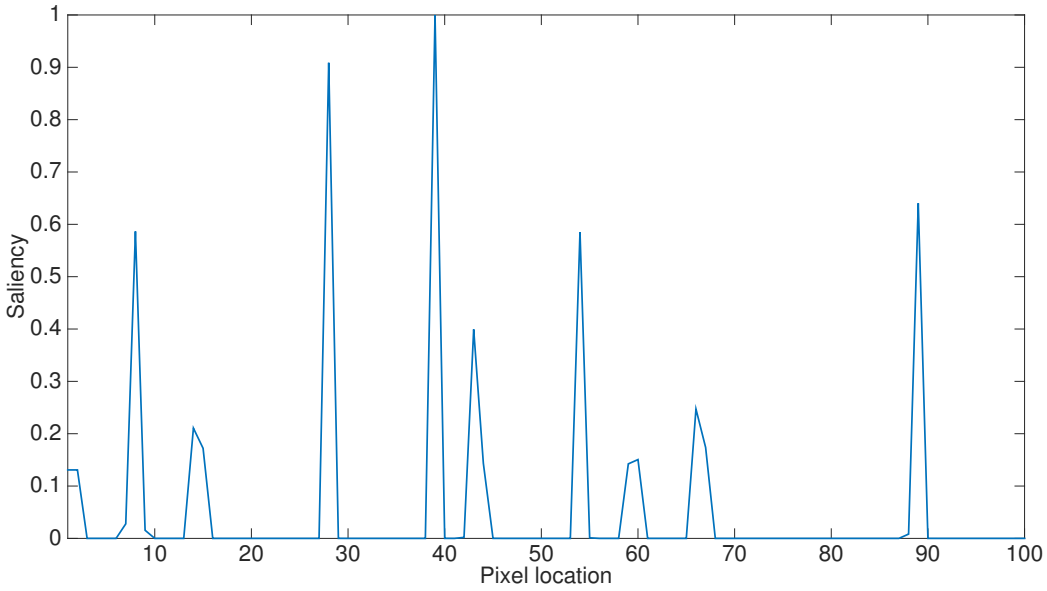


Figure 4.1: The saliency map: There are 10 distinct peaks which stretch from 1 to 100 in a horizontal scale (the visual scene). The height of the maximum peak is 1. The distortion in shape is due to sampling.

sumed 10 one-dimensional Gaussian bumps of fixed size in a 100 pixel wide visual scene (see figure 6.2). The following equation was used to

generate such curves:

$$f(x_b) = \frac{1}{\sigma_b\sqrt{2\pi}} e^{-(x_b-\mu)^2/2\sigma_b^2} \tag{4.1}$$

μ describes the location of the peak in the above equation and σ_b determines how wide the curve is around the peak.

As all probability distributions integrate to 1, a narrow Gaussian has a higher peak at the mean than a wide Gaussian. We make use of this property by randomly sampling 10 times to obtain different values of σ (one σ for each peak) from a uniform distribution with predefined boundaries. This generates different heights for each peak. Similarly, the locations of those peaks were chosen by randomly sampling from another uniform distribution, which spans the same limits as the visual scene. Finally, all the Gaussian distributions were normalised to the range 0–1 to obtain the saliency map. We restrict ourselves to reporting one instance of such multi-modal saliency map throughout all the experiments, but this method has been tested on many similar problem instances with equivalent results. Figure 6.2 shows an example of saliency map used in our experiments. Notice the peak saliency value is 1 and there are 10 distinct peaks.

The foveation profile is a property of the sensor in use. In the absence of a specific hardware platform, a set of plausible foveation profiles were generated using a simple exponentially decaying function.

4.1.2 Generation of Foveation Profile

The foveation profile is the distribution of measurement noise over space under the observable region of the agent's field of view. The choice of $\boldsymbol{R}$ varies in every iteration of the algorithm execution based upon the chosen fixation point. A plausible foveation profile should have minimum measurement noise at the point of fixation and smoothly increasing measurement noise towards the periphery. A variety of functions could be

used to model this behaviour, an exponentially increasing function was chosen for its simplicity in choosing in the rate of increase with distance. This function is given as:

$$f(d, \rho) = 1 - \exp^{-\frac{d^2}{\rho^2}} \tag{4.2}$$

where d is the distance from the profile-centre and the foveation profile parameter ρ^2 determines the rate of increase in measurement noise with distance of the foveation profile (or the width of the foveation profile). The exponential term in this equation is analogous to a standard normal distribution equation, where ρ^2 in this equation plays a similar role to the variance. A smaller value of ρ^2 results in sharp rise in the measurement noise and a large value results in a slowly rising measurement noise.

We illustrate the system's behaviour with three representative foveation profiles having varying widths. Figure 4.2 shows each of the profiles, termed: *narrow* ($\rho^2 = 0.001$), *medium* ($\rho^2 = 10$) and *wide* ($\rho^2 = 100$) respectively. A wider foveation profile results in larger number of pixels being sampled with low measurement noise in a single observation. Notice that the maximum value of measurement noise is 1. To avoid numerical problems arising from using zero uncertainty at the profile centre, the minimum value was set to 0.0001. This is a realistic modification as all real sensors have some noise associated with every measurement. Note that the foveation profile is characteristic of a particular sensor. A designer is not free to choose the profile used but must select one that is matched to the actual sensor used in a practical system.

We are interested in understanding the overall system behaviour, which is independent of the specific choices of minimum or maximum value of the measurement noise. Rather it depends on the distribution of the measurement noise, which is determined by the choice of the foveation profile (narrow, medium or wide).

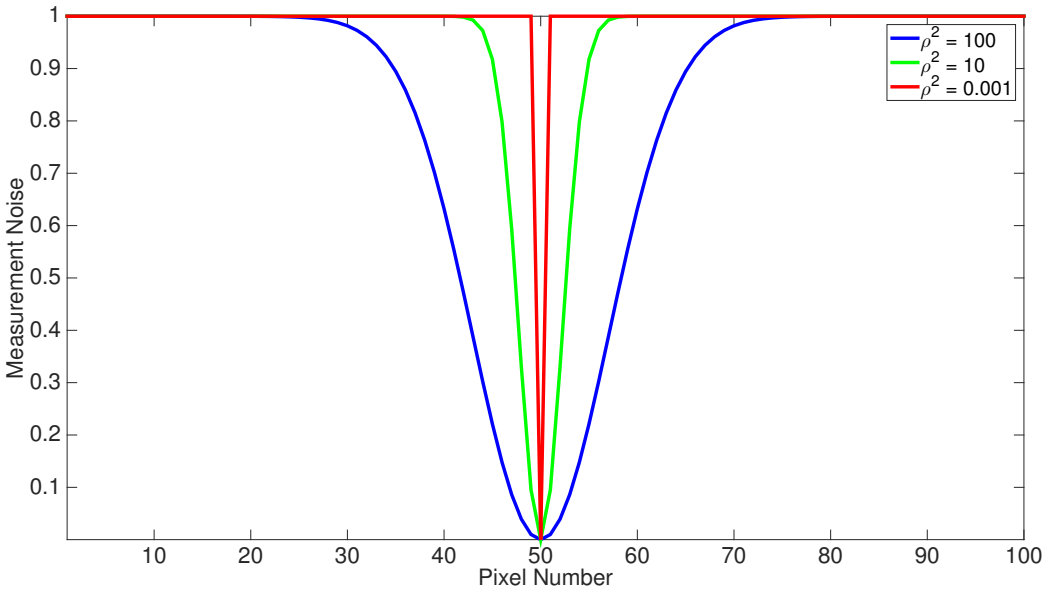


Figure 4.2: The fixation point for each profile is the middle of scene. *Narrow, medium* and *wide* foveation profiles are shown in red, green and blue respectively.

4.2 Preventing Fixation Due to Consideration of Uncertainty

The three Kalman filter equations that are at play are: The prediction of state covariance $\boldsymbol{P}$ is given by

$$\boldsymbol{P}_{t+1|t} = \boldsymbol{A}\boldsymbol{P}_{t|t}\boldsymbol{A}^{\mathsf{T}} + \boldsymbol{Q} \tag{4.3a}$$

where $\boldsymbol{P}_{t|t}$ is the current uncertainty. The Kalman gain $\boldsymbol{K}$ is given as

$$\boldsymbol{K}_{t+1} = \boldsymbol{P}_{t+1|t}\boldsymbol{C}^{\mathsf{T}}\left(\boldsymbol{C}\boldsymbol{P}_{t+1|t}\boldsymbol{C}^{\mathsf{T}} + \boldsymbol{R}\right)^{-1} \tag{4.3b}$$

The updated estimate of state covariance after including the newest measurement is given by:

$$\boldsymbol{P}_{t+1|t+1} = \left(\boldsymbol{I} - \boldsymbol{K}_{t+1}\boldsymbol{C}\right)\boldsymbol{P}_{t+1|t} \tag{4.3c}$$

where $\boldsymbol{P}_{t+1|t}$ is the predicted uncertainty and t denotes time in the three equations above. These three equations are the guiding principles for how the internal uncertainty of a belief state changes over time.

In the presented model, the sensor observes the entire scene at each instant with increasing noise away from the profile centre. This means that the measurements are taken with varying measurement noise. Also, $\boldsymbol{C}$ matrix is assumed to be an identity matrix. Therefore, $\boldsymbol{C}$ can be dropped from equations 4.3 to obtain the following:

$$\boldsymbol{K}_{t+1} = \boldsymbol{P}_{t+1|t} \left(\boldsymbol{P}_{t+1|t} + \boldsymbol{R} \right)^{-1} \tag{4.4}$$

This equation can be substituted into (4.3c) to obtain:

$$\boldsymbol{P}_{t+1|t+1} = \left(\boldsymbol{I} - \boldsymbol{P}_{t+1|t} \left(\boldsymbol{P}_{t+1|t} + \boldsymbol{R} \right)^{-1} \right) \boldsymbol{P}_{t+1|t} \tag{4.5}$$

where $\boldsymbol{P}_{t+1|t}$ and $\boldsymbol{P}_{t+1|t+1}$ are the pre-observation and post-observation uncertainties. For an intuitive understanding of equation 4.5 let us consider one such belief state, along with the assumption of independent internal belief states (i.e. all the matrices in the equation are diagonal). Equation 4.5 can then be presented as

$$P_{t+1|t+1} = \left(1 - \frac{P_{t+1|t}}{\left(P_{t+1|t} + R \right)} \right) P_{t+1|t} \tag{4.6}$$

where $P_{(\bullet)}$ is an element of the state covariance matrix $\boldsymbol{P}$ and R is similarly an element of the noise matrix $\boldsymbol{R}$.

The interest is in understanding how the measurement noise affects the updated state covariance. Notice that equation 4.6 shows that measurement noise variance ($\boldsymbol{R}$) determines post observation uncertainty. Hence it governs the improvement after observation for a given prior uncertainty ($\boldsymbol{P}_{t+1|t}$) and posterior uncertainty ($\boldsymbol{P}_{t+1|t+1}$). Further the process noise ($\boldsymbol{Q}$) decides the prior uncertainty at the next time step ($\boldsymbol{P}_{t+2|t+1}$). Hence the current state of uncertainty at any given time is jointly influenced by the process noise $\boldsymbol{Q}$ and the measurement noise variance $\boldsymbol{R}$. Section 4.3 and

section 4.4 in this chapter investigate the effect of these two parameters separately.

Equation 4.6 shows that the updated state covariance reaches zero when R is zero. A zero state-covariance means no error in the belief states, which is possible in theory only in the case of a perfect measurement. However as the measurement noise covariance is not zero in practice, the updated covariance is always non-zero in practice. Also, the updated state covariance is equal to the predicted covariance when the measurement noise covariance is high because the measurement had no effect on the belief state due to its unreliability. In between these two extremes, the value of $P_{t+1|t+1}$ monotonically increases with the increase of measurement noise covariance R.

Figure 4.3 shows the relation between updated state covariance ($P_{t+1|t+1}$) and the noise variance R. The figure shows that error covariance matrix and the measurement noise follows a positive monotonically increasing relation and the posterior variance is asymptotic to the prior variance at high measurement noise covariances (refer to equations 4.3).

A visual region that is observed with a higher measurement noise would have higher uncertainty in the associated belief state compared to one that is observed with a lower measurement noise. As an example let us consider two visual regions with equal internal uncertainty that are observed with two different measurement noise levels R_1 and R_2, where $R_1 > R_2$. After incorporating the observation into the internal belief states, region 1 will have greater internal uncertainty than region 2. This property is made use of to achieve inhibition of return as a behavioural outcome of the proposed algorithm.

A rational agent behaviour is to take measurements to reduce the internal uncertainty of the belief state that has the highest level of uncertainty amongst all the regions in the scene. It can be achieved by treating the internal uncertainty as the utility, the agent would look at the visual location with the largest uncertainty at every time step. A visual scene with n

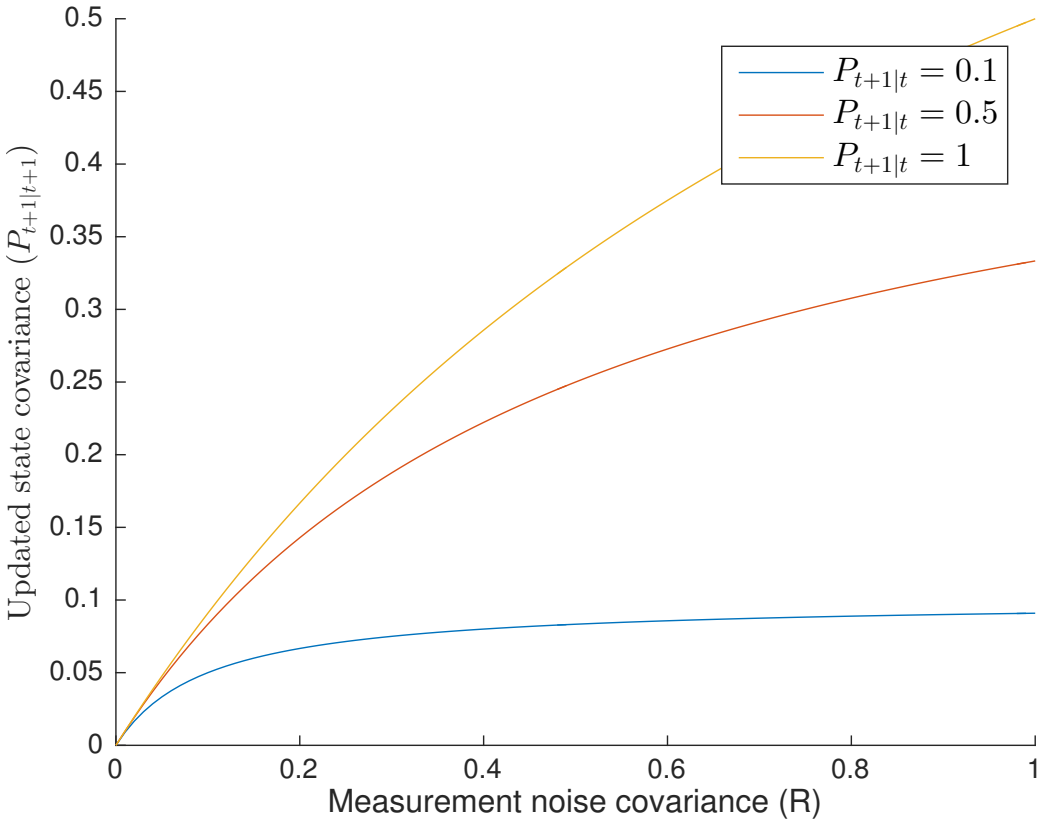


Figure 4.3: Plot of posterior error variance versus the measurement noise variance. The three lines in the plot depict the relation between measurement noise and posterior variance for three different prior variances.

regions of interest has n internal belief state uncertainties, hence it has n utility values, one for each potential future fixation point.

All the utility measures are arranged into a vector $\boldsymbol{u} \in \mathbb{R}^n$. Each element in this vector is the internal uncertainty of a corresponding visual region. Such a system should maximize a utility of the form

$$\boldsymbol{u} = \{P^1_{t+1|t}, P^2_{t+1|t}, P^3_{t+1|t}, \cdots, P^n_{t+1|t}\} \tag{4.7}$$

where $P^n_{t+1|t}$ is the projected uncertainty in the internal beliefs of a corresponding visual region.

The maximum in $\boldsymbol{u}$ gives the location of the visual scene that has the

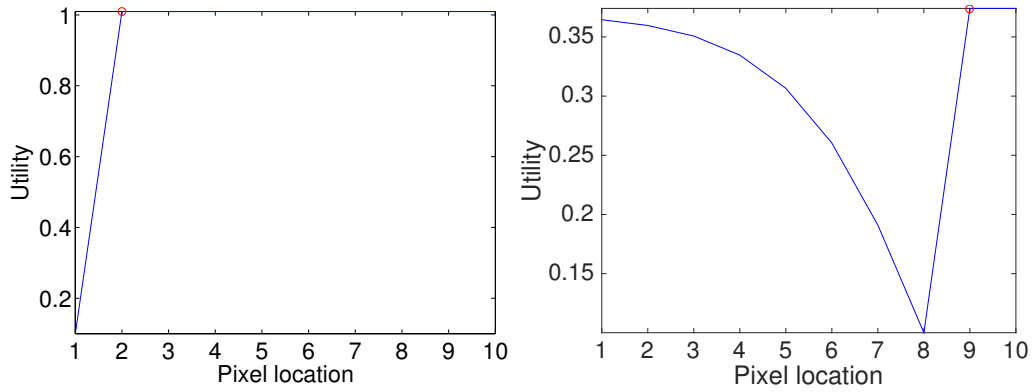
highest internal uncertainty and is to be observed in the next time step. Observing a region is placing the profile-centre of the foveation profile on the target. A visual region that is observed with the lowest measurement noise (profile-centre of the foveation profile) results in a maximum reduction in the corresponding uncertainty of its belief state. In the next time step, the region that has just been observed will have the lowest internal uncertainty amongst all regions. As the agent is interested in observing visual locations that it is uncertain about, it will ignore the region that it has just observed. Hence it will effectively inhibit looking at the visual location that it observed in the last time instant. This behaviourally generates inhibition of return.

As an initial simplified experiment to test behaviourally generated IOR, a simple one-dimensional world with ten pixels was set up. In this simple model, each pixel is an individual visual location of the scene. Those ten pixels were modelled as ten independent states in a Kalman filter. The Kalman filter tracks the ten states over time and makes a decision about where to place the centre of the foveation profile at every time step based on the internal uncertainty. This process was run for 5 seconds in total. During this process, the internal uncertainty reduces after an observation. Therefore the utility offered by that location becomes lower. During the experiment, the utility offered by each pixel was noted at every time step.

Figure 4.4 plots the utility $\boldsymbol{u}$ of a pixel versus the respective pixel number. The horizontal axis represents pixel number and the vertical axis represents the utility of observing that location at a given time instant. The blue line shows the utility and the red circle indicates the peak in the utility. Notice that the utility offered by pixel 1 is the lowest at $t = 1$ and it remains low at the 8^{th} time step. Therefore the visual location was inhibited from being re-observed.

Notice that the pixel location that has just been observed offers the lowest utility and the first of the unobserved pixels offers the highest utility. Also, note that the utility of pixel number 1 remains low almost until the

9^{th} time step. Consequently, that pixel remains inhibited from observation, an example of achieving behavioural inhibition of return.



(a) Utility plot at time step 1 (b) Utility plot at time step 8

Figure 4.4: Plots of the utility of pixels versus the pixel number at time instant 1 (left panel) and 8 (right panel). Projected internal uncertainty $P_{t+1|t}$ of the belief states were adopted as the utility. The blue line shows the utility and the red circle indicates its peak.

4.3 Effect of Foveation Profile Width on Utility

The distribution of the measurement noise over space (the foveation profile) determines the area that is observed with low measurement noise. With a very narrow foveation profile ($\rho^2 = 0.001$) any observation results in a high reduction in the uncertainty associated with the point of focus. This profile does not reduce the uncertainty of any neighbouring pixels. In contrast, a wide foveation profile reduces the uncertainty associated with a wider visual area.

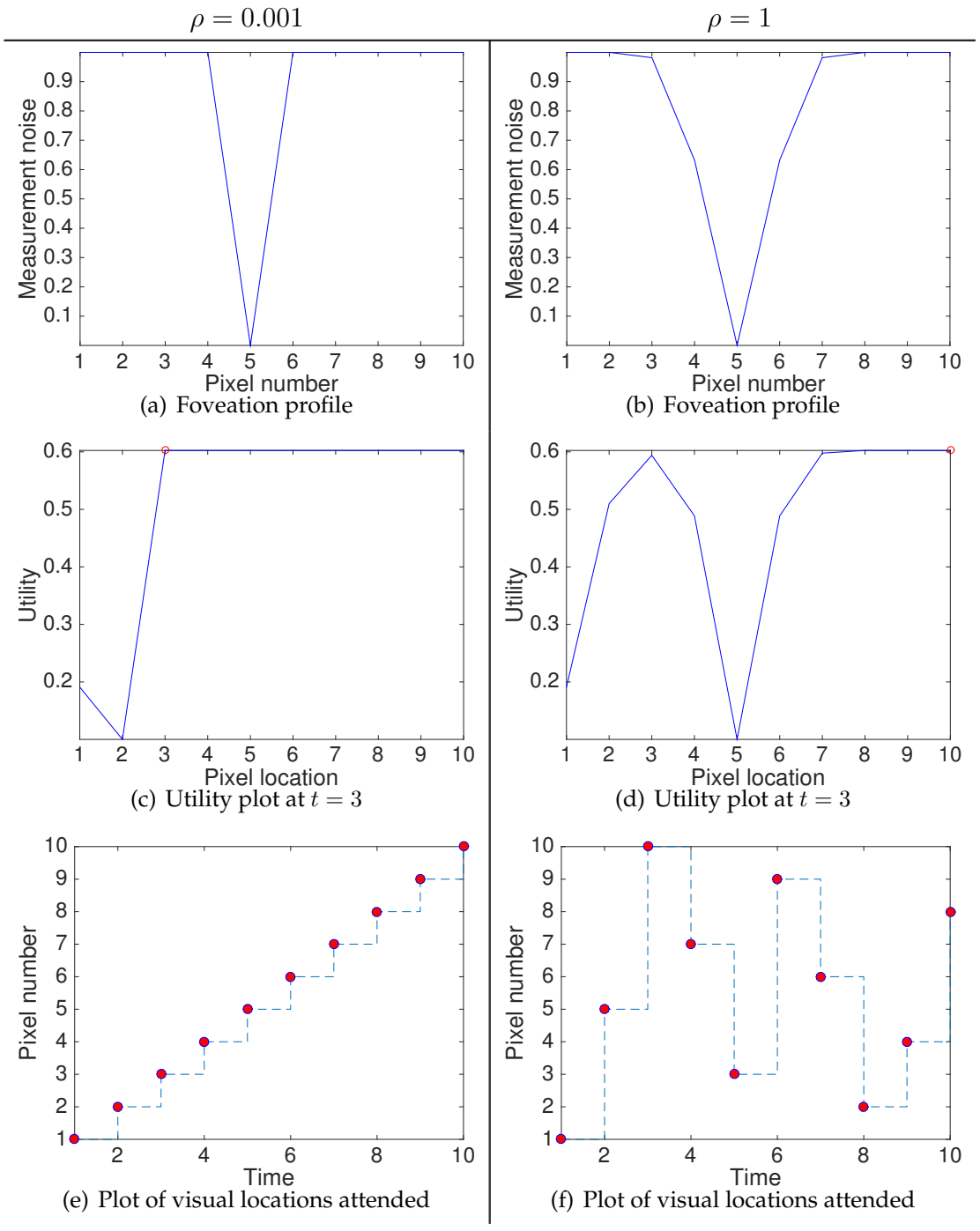
To study the effect of foveation profile on utility, the uncertainty based visual scene sampling approach was applied to a simple 1D visual scene. The system was run separately with a very narrow ($\rho = 0.001$) and a wide

($\rho = 1$) foveation profile. The history of the visual locations attended and the utility at every time step of the system run was noted.

Figure 4.5 shows the plot of the two foveation profiles and the corresponding utility functions and the visual locations attended. The left column shows the narrow foveation profile and its outputs. The right column shows the wide foveation profile and its resulting outputs.

Notice that the observed pixel has low uncertainty in the utility plot (panel: 4.5(c)). Subsequently, the agent observes every pixel in the visual scene in sequence. Figure 4.5(e) depicts that the agent chronologically attended the visual locations.

When a wide foveation profile ($\rho^2 = 1$) was used, one observation results in a reduction in uncertainty associated with a few neighbouring pixels. This consequently reduces the uncertainty associated with a larger set of neighbouring pixels. This can be noticed as a reduction in uncertainty of broader span of pixels (panel: 4.5(d)). Reduction of uncertainty over wider area behaviourally inhibits that area from re-observation, which produces jumps between two successive fixation locations (figure: 4.5(f)). As a broader area of the visual scene gets observed at every time-step, more parts of the visual scene are observed with low measurement noise. Therefore the wider foveation profile results in a lower average internal uncertainty (average internal uncertainty after 10 time steps is: 0.18) compared to the narrow foveation profile (0.2).



(a) Foveation profile (b) Foveation profile
(c) Utility plot at $t = 3$ (d) Utility plot at $t = 3$
(e) Plot of visual locations attended (f) Plot of visual locations attended

Figure 4.5: Each column shows one foveation profile (left column-narrow foveation profile, right column-wider foveation profile) and its resulting system behaviour. The top left and right panels show the foveation profiles, the middle two panels show the pixel utility and the bottom panels show the visual locations attended over time.

4.4 Different IOR Timing From Process-noise

Equation 4.3a shows that the rate of growth of internal uncertainty of a belief state arises directly from the values of process noise in the $\boldsymbol{Q}$ matrix. The $\boldsymbol{Q}$ matrix is a temporal property of the visual scene being observed. The aim of an intelligent algorithm is to use its sensor in a given visual scene to optimise the sensor action according to the attention requirements of the scene. Higher values of $\boldsymbol{Q}$ result in uncertainty growing more quickly, and hence a previously observed location offers higher utility compared to other regions. This forces the agent to re-observe that location and determines how often any given region is re-observed.

To illustrate the point a process noise matrix that varies across the visual scene was constructed. This process noise matrix has two elements that are higher than the other elements in the matrix.

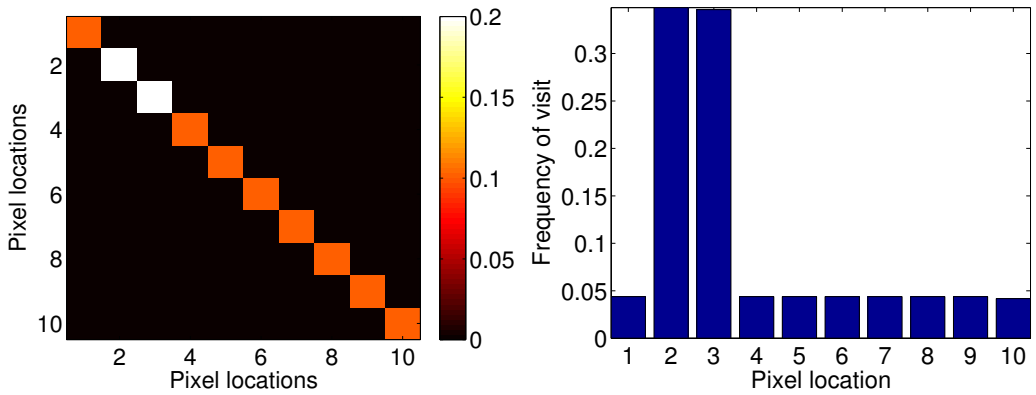
The same ten pixel visual scene discussed in the last section is used again for visual attention distribution. This time the non-uniform process noise matrix with two elements with a higher value than the others were used to distribute visual attention. History of the visual locations attended was noted. Finally, normalised histograms of attention distribution are computed.

Figure 4.6 shows a graphical representation of the process noise matrix and the resulting distribution of visual attention as histograms. The values of each element of the process noise are colour coded. White colour shows high value and block shows zero.

The proposed uncertainty based attention distribution method was run for 500 time steps with each of the process noise matrices and the histogram (normalised to have a unit area under the curve) of the visual attention distribution was plotted to show the attention distribution over visual locations. Note that locations with higher process noise in the $\boldsymbol{Q}$ matrix attract more visual attention in the corresponding visual scene (left and right panels of the top row respectively). Pixel numbers 2 and 3 took

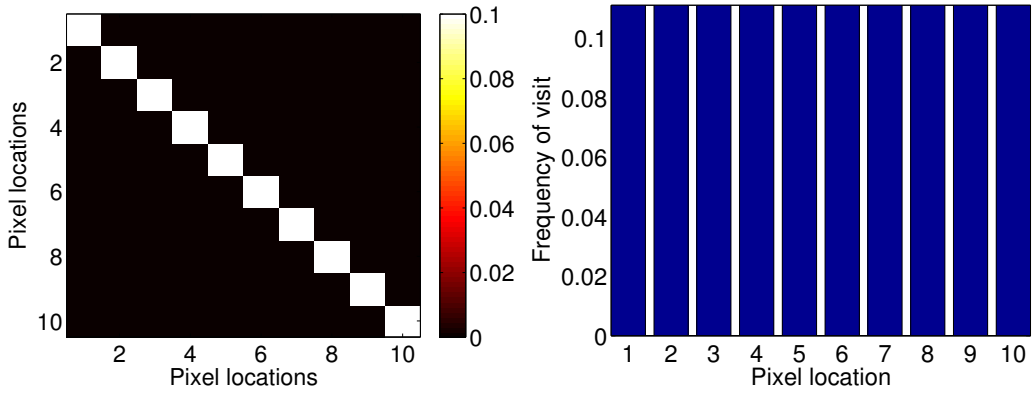
higher process noise values in the $\boldsymbol{Q}$ matrix (top-left panel) which resulted in the peak in attention distribution in the related histogram (top-right panel). The even $\boldsymbol{Q}$ matrix (bottom-left panel) resulted in a uniformly distributed histogram (bottom-right panel). This suggests that a process noise matrix that matches the attention requirements of a dynamic visual scene will result in an efficient visual attention distribution.

Notice that an uneven $\boldsymbol{Q}$ matrix results in an attention distribution that is biased towards highly dynamic visual locations, i.e. the visual locations that have a higher value of process noise in the corresponding element in the $\boldsymbol{Q}$ matrix. In contrast, an even process noise matrix results in an even visual attention distribution.



(a) Heatmap plot of an uneven process noise matrix

(b) Histogram of visual attention distribution



(c) Heatmap plot of an even process noise matrix

(d) Histogram of visual attention distribution

Figure 4.6: Plots of two different process noise matrices with their resulting histogram of visual attention distribution. The goal of the experiment is to observe if our algorithm favours visual regions with high process noise over visual regions with lower process noise in distributing visual attention.

The proposed Kalman filter based system was run for 5 seconds on a visual scene comprised of 10 pixels. The inhibition of return time was calculated by counting the number of iterations between two direct observa-

tions of the same pixel. A narrow foveation profile was chosen ($\rho = 0.001$) for this experiment, where a direct observation means observing a given pixel with the lowest observation noise.

Figure 4.7 shows the inhibition of return (IOR) time distribution over the visual scene. This figure depicts the actual inhibition of return time whereas the earlier figure presented the histogram of attention distribution. The blue line shows the IOR time distribution achieved by the proposed method and the red line shows IOR distribution by traditional WTA based systems. As the traditional system allocates the same IOR time for all regions, the red line shows constant value for all the regions. The dip in the blue line shows unevenness of IOR time distribution by the proposed method. The figure shows that the proposed system achieves dissimilar IOR time distribution whereas the IOR time distribution of the traditional system is equal everywhere in the visual scene.

Uneven distribution in IOR time was realised by choosing unequal process noise ($\boldsymbol{Q}$) matrix, where the two middle pixels (pixel number 2 and 3) were set to have higher process noise. The IOR time for the traditional system came from a theoretical understanding of that method, where the same IOR time is applied to each visual location. An IOR time of 10 was chosen for the traditional system as there are 10 pixels in the scene. This way each pixel in the scene gets a chance to be observed.

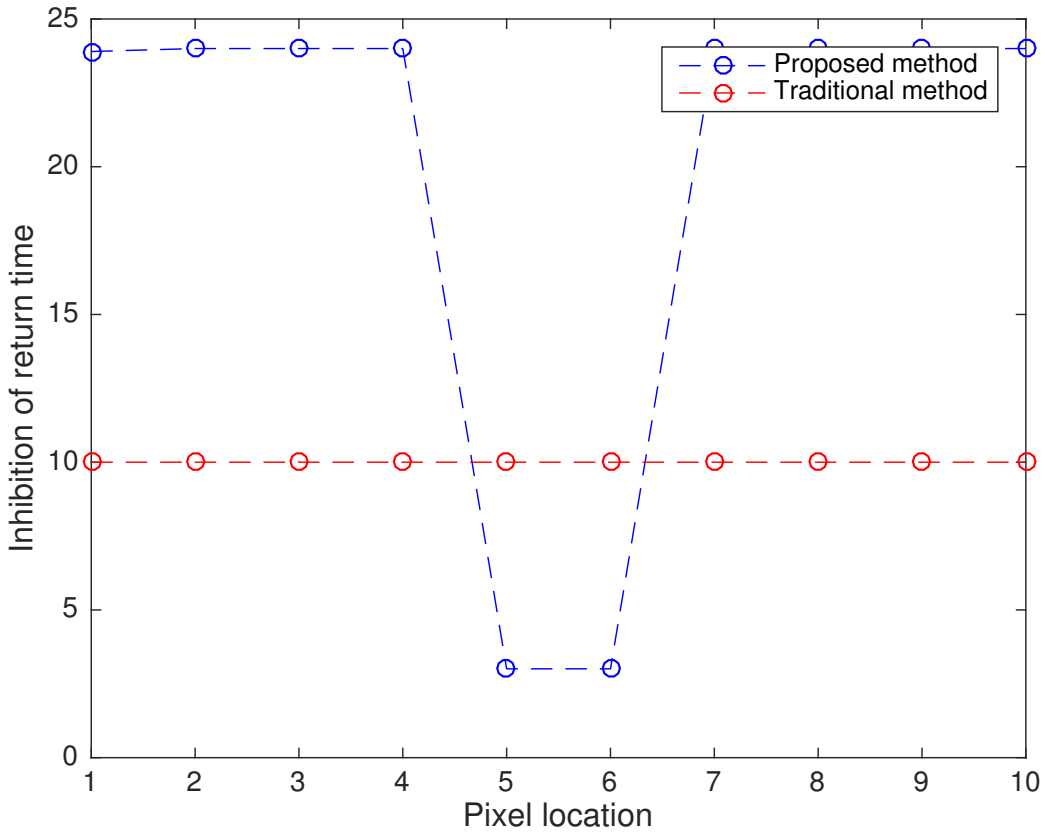


Figure 4.7: Plot of inhibition of return time of all the pixels in the visual scene (10 pixels in the visual scene). The blue line shows the inhibition of return time for the proposed method and the red line shows the inhibition of return time distribution of the traditional system.

4.5 Chapter Discussion

This chapter presented two important contributions of the thesis as below.

1. This chapter presented the novel Kalman filter aided epistemic visual target selector. It encapsulates a layer of uncertainty regarding the saliency of the dynamic visual scene. As a result of considering

uncertainty in decision making, an agent behaviourally achieved inhibition of return without blocking previously observed visual locations.

2. Different inhibition time for different parts of the scene was achieved due to the use of Kalman filter based visual target selection. To the best of the author's knowledge, this was the first demonstration of different inhibition time in a dynamic scene. As a result of different inhibition time, the agent updates its internal state more frequently about the regions that are changing quickly with time and the regions that are not changing are observed less. This was not possible using the traditional IOR based method.

The next four chapters stem from this chapter as below.

1. Knowing the proper values for the process noise variance is key to distributing visual attention. In this chapter, the process noise matrix was obtained beforehand from the video dataset. In practice a dynamic visual scene is observed frame by frame, hence the process noise needs to be learnt during observation. Chapter 5 examines methods to learn process noise from observations.

2. The use of the Kalman filter aided novel visual target selector further enabled varied utility functions to be used for visual target selection. A number of utility function will be discussed in chapter 6 and in chapter 7 of this thesis.

3. Due to the behavioural outcome of inhibition of previously observed locations, the novel target selector should be able to detect sudden changes in the visual scene quicker than the IOR based approaches. This will be studied in the chapter 8.

Chapter 5

Learning Process Noise from Noisy Observations

This chapter proposes two novel approaches for learning the statistical variance of visual regions from observations having varying measurement noise. The first approach estimates the statistical variance using a maximum likelihood method (MLE) while the second approach uses a low pass filter (LPF) for estimation. The MLE approach operates on a batch of observations where each datum in the batch is measured with different measurement noise. On the other hand, the LPF based variance learning is an online method that trusts accurate measurements more than the noisy measurements and recursively builds up an estimate of the process noise over time. This two process noise learning methods help the Kalman filter based visual target selector discussed in chapter 3 adapt to a new dynamic visual scene.

As the existing process noise estimation approaches are designed to operate with a fixed measurement noise, they are not suitable for the purpose of estimating variance in a changing measurement noise scenario. To the best of the author's knowledge, the presented work is the first approach towards estimation of process noise from observations taken with varying measurement noise.

In this chapter, a short background on process noise learning from observations is discussed. The mathematical derivation of the proposed algorithms along with an intuitive discussion of the algorithms are then presented. Finally, the accuracy, repeatability and numerical stability of the algorithms are examined.

5.1 Background

Estimation of process noise is a well studied topic in the context of closed loop process control and the Kalman filter algorithm [2, 4, 12, 26, 29, 44, 48, 51, 55, 66, 81, 111, 131, 135, 142, 143, 171, 174, 175, 200].

These studies can be put into two thematic groups based on their method of estimation:

1. Bayesian statistics based approach [4, 12, 111, 174, 175]:
   This approach formulates the covariance estimation problem as maximising a likelihood function of choice [12]. The likelihood function is devised to inversely correlate with the difference between a predicted state and a measured state (also called the innovation). Elements of the covariance matrix are obtained by maximizing the likelihood function i.e by finding matrix elements that minimise the prediction measurement difference. Analytical minimization often does not work with complex covariance functions, hence gradient based numerical optimisation schemes are used.

2. Deterministic approach [131, 135]:
   The deterministic approach uses simple statistical measures like first and second order moments to estimate the process noise variance. For example, the covariance matching technique [135] presents an algorithm that estimates the process noise covariance ($\boldsymbol{Q}$) at every sampling instant. Here estimates of $\boldsymbol{Q}$ are constructed using the sample covariance of the state prediction error. This approach can be

used either on all the past data at every time instant or over a smaller batch of past data points using a moving time window. There is no recursive version of this algorithm.

A popular related technique was proposed by Raman K. Mehra [131]. This technique estimates the elements of the $\boldsymbol{Q}$ matrix by making use of the sample autocorrelation. This method is slow and computationally expensive as the algorithm involves computing autocorrelation over past innovations.

The same studies can also be grouped based on the quantity of data processed at each time instant:

1. Batch processing [12, 135]:
   This approach finds the best suited process noise values given a collection of data points.

2. Recursive approach [29, 51, 55, 111, 174, 175]:
   This approach learns and updates the $\boldsymbol{Q}$ matrix at every measurement instant.

Notice that there are overlaps amongst the groups presented above. For example [135] comes under batch processing and correlation based approach while [12] sits in the intersection between batch processing and Bayesian statistics based approach. Similar combinations can be observed in a recursive and Bayesian statistics based method [174] and in a recursive-correlation based approach [55].

Kenneth A. Myers [135, 177] proposed a simple approach that computes measurement noise from an innovation sequence (also known as the measurement residual) of the Kalman filter. This method also estimates the process noise from the difference between the true state and a prior belief state (also known as the forcing residual). The forcing residual cannot be computed directly as the true states are unknown, hence it is approximated using the posterior mean of the Kalman belief state [177].

The covariance based techniques [135, 177] assume constant measurement noise and all the forcing residuals are equally trustworthy. Whereas the measurement noise for observing any given visual location of the proposed system changes between observations and this reflects as variations in the measurements. Hence all the forcing residuals are not equally trustworthy. Therefore covariance based techniques are not suited to operate in such circumstances and will produce erroneous results.

Bo Feng et al. [55] presented a recursive covariance estimation approach that works in conjunction with a Kalman filter. The prime focus of this approach is to compute the covariance between observations. The problem in this thesis is the assumption that the individual pixels are independent and there is no correlation between pixels. This simplifies the process noise computation to computing just the variance. Hence the approach proposed by [55] is not readily applicable for the purpose of the presented work.

The work by Bo Feng et al. [55] also assumes constant process noise matrix throughout the entire run of the Kalman filter. Whereas the temporal variance of visual locations can change over time. Hence this method fails to meet the requirements of dealing with changing process noise.

Another assumption of this method is constant measurement noise,which does not meet the requirements of the presented work.

The correlation based technique presented in [131] estimates the process noise making use of the autocorrelation amongst current and past innovations. This process is computationally expensive and memory intensive. Also, the key component of this method is to find how statistically independent one innovation is from the rest of the past sequence (a measure of whiteness of the innovation sequence). Measuring 'whiteness' of an innovation sequence involves computing the expectation between the current sample and past samples, ideally all the past samples. This process requires a large amount of memory.

Although Mehra [131] proposed a block-wise recursive scheme for on-

line update of process and measurement noise, it proves to be restrictive for systems where the number of unknowns in process noise matrix ($\boldsymbol{Q}$) is larger than $n \times r$, where n is the number of states and r is the dimension of the measurement vector. Also, the overall algorithm is computationally demanding due to involving matrix inverses involved with the algorithm and cannot satisfy real-time performance requirements.

Summarising existing approaches reported above, it is necessary to develop a new algorithm for the problem of estimation of process noise ($\boldsymbol{Q}$) that can deal with changing measurement noise while requiring low demand for computational resources.

The rest of this chapter is divided into three more sections. First, a maximum likelihood based approach for variance estimation from noisy observations is presented, then a simpler recursive algorithm is presented along with their respective results.

5.2 Learning Process Noise From Noisy Measurements

Any given region of the visual scene is observed with different measurement noise. Measurements taken with low measurement noise are more trustworthy than measurements taken with high measurement noise. This is because, from a statistical point of view, the less-noisy measurements are drawn from a narrow distribution whereas noisier measurements are drawn from a wide distribution and are therefore more likely to be inaccurate.

The standard variance estimation method is an average of the squared difference of all the data points from the mean. This given as below:

$$Var(X) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2 \tag{5.1}$$

where μ is the mean, X is the random variable and x_i is a datum in the

dataset.

A simple averaging of variance estimation over multiple observations divides a sum of variance by the total number of observations. Stated another way, each squared difference is given an equal weight. As a simple expectation based variance computation method gives equal weights to all observations, it would always overestimate the process noise due to the influence of high measurement noise in the observations. Hence a mechanism is needed to smooth out these instantaneous variances due to change in measurement noise. This method should be able to compute the underlying process noise variance from observations with varying measurement noise.

The aim of the process noise learning method is twofold:

- to reduce the effect of measurement noise from the observed variables and
- to determine the process noise from the observations.

With an assumption that the process noise is quasi-static[1], a maximum likelihood based method and a recursive exponentially weighted average based method for process noise estimation are presented in this chapter.

5.2.1 Maximum Likelihood Estimation of Process noise

This approach aims to solve a maximum likelihood estimation (MLE) of the variance in the observed data. The observations are noisy and the measurement noise changes between observations. The proposed approach is a batch learning approach. It is assumed that the mean and the variance of the real world state is constant within a reasonably small window of time $t = 1$ to $t = N$, which is the length of the batch. The aim is to make the best estimation of the variance in the real world process.

[1] quasi-static process changes very slowly and can be treated as stationary

The mathematical derivation:

- y_t is the observation at the t^{th} time instant.
- ϵ_t is the sample from the noise distribution at t^{th} time instant.
- $\sigma^2_{n,t}$ is the measurement noise, which is a function of time.
- μ_n is the mean of the measurement noise distribution. Its value is zero. i.e. $\mu_n = 0$.
- μ_x is the mean of the true state of the world.
- σ^2_x is the variance of the true state of the world.
- x_i is the true state of the world. i.e. x_i is a sample from the distribution defined by the mean μ_x and variance σ^2_x.

The joint probability distribution of one observation is given as:

$$P(y_t, \mu_x, \sigma^2_x, \mu_n, \sigma^2_{n,t}) \tag{5.2}$$

The joint probability distribution over all the observations ($y_t, t = 1 \cdots N$) is Y. Mathematically it can be written as below:

$$P(Y) = P(y_{t=1\cdots N}, \mu_x, \sigma^2_x, \mu_n, \sigma^2_{nt=1\cdots N}) \tag{5.3}$$

The observation depends on the values of $\mu_x, \sigma^2_x, \mu_n, \sigma^2_{n,t=1\cdots N}$. Let us define a vector $\boldsymbol{\theta}$ that contains all the parent variables that the observation y_t depends on:

$$\boldsymbol{\theta} = \{\mu_x, \sigma^2_x, \mu_n, \sigma^2_{n,t=1\cdots N}\} \tag{5.4}$$

Now let us assume a uniform prior over $\boldsymbol{\theta}$

$$P(\theta) = P(\mu_x)P(\sigma^2_x,)P(\mu_n)P(\sigma^2_{n,t}) \tag{5.5}$$

where every term in the prior probability is independent of the other.

Now the conditional probability distribution can be written as:

$$P(Y|\boldsymbol{\theta}) = P(y_{t=1\cdots N}|\mu_x, \sigma_x^2, \mu_n, \sigma_{n,t=1\cdots N}^2) \tag{5.6}$$

The aim is to find parameters θ that are plausible under the prior and would make all the $y_t\{t = 1 \cdots N\}$ likely. This is defined by the posterior function M

$$M = P(\boldsymbol{\theta}|Y) \tag{5.7}$$

Using Bayes' theorem the following is obtained

$$\begin{aligned} M &= P(\boldsymbol{\theta}|Y) \\ &= \frac{P(Y|\boldsymbol{\theta})P(\boldsymbol{\theta})}{P(Y)} \end{aligned} \tag{5.8}$$

where $P(Y)$ is a normalization factor that does not depend on Θ Hence the following can be written:

$$P(\boldsymbol{\theta}|Y) \propto P(Y|\boldsymbol{\theta})P(\boldsymbol{\theta}) \tag{5.9}$$

where $P(\boldsymbol{\theta})$ is the prior belief. Under the assumption of uniform prior over θ, the posterior distribution is equivalent to the likelihood. Now substituting 5.6 into 5.9 the following is obtained:

$$\begin{aligned} M &\propto P(Y|\boldsymbol{\theta})P(\boldsymbol{\theta}) \\ &\propto P(y_{t=1\cdots N}|\mu_x, \sigma_x^2, \mu_n, \sigma_{n,t=1\cdots N}^2)P(\boldsymbol{\theta}) \end{aligned} \tag{5.10}$$

Equation 5.10 can be expanded as below:

$$P(y_{t=1\cdots N}|\mu_x, \sigma_x^2, \mu_n, \sigma_{n,t=1\cdots N}^2)P(\boldsymbol{\theta}) = \prod_{t=1}^{N} P\Big[(y_t|\mu_x, \sigma_x^2, \mu_n, \sigma_{n,t}^2)\Big] P(\mu_x, \sigma_x^2, \mu_n, \sigma_{n,t}^2) \tag{5.11}$$

where every term in the prior probability is independent of the other. Hence the above equation can be rewritten as the following likelihood function below:

$$M = \prod_{t=1}^{N} P\Big[(y_t|\mu_x, \sigma_x^2, \mu_n, \sigma_{n,t}^2)\Big] P(\mu_x)P(\sigma_x^2,)P(\mu_n)P(\sigma_{n,t}^2) \tag{5.12}$$

The log-likelihood function would be

$$\begin{aligned}\ln(M) &= \ln\Big[\prod_{t=1}^{N} P\Big[(y_t|\mu_x,\sigma_x^2,\mu_n,\sigma_{n,t}^2)\Big]P(\mu_x)P(\sigma_x^2)P(\mu_n)P(\sigma_{n,t}^2)\Big] \\ &= \sum_{t=1}^{N}\ln\Big(P\Big[(y_t|\mu_x,\sigma_x^2,\mu_n,\sigma_{n,t}^2)\Big]\Big) + \cdots \\ &\quad \ln(P(\mu_x)) + \ln(P(\sigma_x^2)) + \ln(P(\mu_n)) + \ln(P(\sigma_{n,t}^2))\end{aligned} \tag{5.13}$$

Now from the assumptions it is known that μ_x, σ_x^2, μ_n are constants. Hence $\ln(P(\mu_x)) = \ln(P(\sigma_x^2)) = \ln(P(\mu_n)) = 0$. This is because constants are delta distributed at the value of the constant, i.e. $P(\mu_x) = P(\sigma_x^2) = P(\mu_n) = 1$. Hence 5.13 can be rewritten (putting the values of zero in proper places) as below:

$$\begin{aligned}\ln(M) &= \sum_{t=1}^{N}\ln\Big(P\Big[(y_t|\mu_x,\sigma_x^2,\mu_n,\sigma_{n,t}^2)\Big]\Big) + \ln(P(\sigma_{n,t}^2)) \\ &= \sum_{t=1}^{N}\ln(P\Big[(y_t|\boldsymbol{\theta})\Big]) + \ln(P(\sigma_{n,t}^2))\end{aligned} \tag{5.14}$$

From basic assumptions it is known that the measurement noise is additive Gaussian noise, hence $P(y_t|\boldsymbol{\theta})$ is Gaussian distributed as below:

$$\begin{aligned}P(y_t|\boldsymbol{\theta}) &\sim \mathcal{N}(\mu_x, \sigma_{n,t}^2 + \sigma_x^2) \\ &= \frac{1}{\sqrt{2\pi(\sigma_n^2(t)+\sigma_x^2)}}\exp\left(-\frac{(y_t-\mu_x)^2}{2(\sigma_n^2(t)+\sigma_x^2)}\right)\end{aligned} \tag{5.15}$$

Equation 5.15 is plugged into the log-likelihood function to obtain:

$$\begin{aligned}\ln(M) &= \ln(P(\sigma_{n,t}^2)) + \sum_{t=1}^{N}\Big[\ln\Big(\frac{1}{\sqrt{(\sigma_n^2(t)+\sigma_x^2)}\sqrt{2\pi}}\Big) + \ln\Big(\exp^{-\frac{(y_t-\mu_x)^2}{2(\sigma_n^2(t)+\sigma_x^2)}}\Big)\Big] \\ &= \ln(P(\sigma_{n,t}^2)) + \sum_{t=1}^{N}\Big[-\ln\sqrt{2\pi(\sigma_n^2(t)+\sigma_x^2)}\Big] + \sum_{t=1}^{N}\Big[-\frac{(y_t-\mu_x)^2}{2(\sigma_n^2(t)+\sigma_x^2)}\Big]\end{aligned} \tag{5.16}$$

As the possible values that $\sigma_{n,t}^2$ can take are all equally likely, a uniform prior over $\sigma_{n,t}^2$ can be safely assumed. The maximum and the minimum values of the measurement noise variance is known beforehand. Let them be σ_n^{max} and σ_n^{min}. A uniform distribution between these two limits would be as below:

$$P(\sigma_{n,t}^2) = \frac{1}{\sigma_n^{max} - \sigma_n^{min}} \tag{5.17}$$

Our purpose is to find how the log-likelihood changes with σ_x^2. Notice that $P(\sigma_{n,t}^2)$ does not change with respect to change in σ_x^2, so it will be dropped from the log-likelihood equation and the following is obtained:

$$\frac{\partial \ln(M)}{\partial \sigma_x^2} = \frac{\partial}{\partial \sigma_x^2}\left(\sum_{t=1}^{N}\Big[-\ln\sqrt{2\pi(\sigma_n^2(t)+\sigma_x^2)}\Big] + \sum_{t=1}^{N}\Big[-\frac{(y_t-\mu_x)^2}{2(\sigma_n^2(t)+\sigma_x^2)}\Big]\right) \tag{5.18}$$

The first term on the right hand side of the above equation gives:

$$\frac{\partial \sum_{t=1}^{N}\Big[-\ln\sqrt{2\pi(\sigma_n^2(t)+\sigma_x^2)}\Big]}{\partial \sigma_x^2} = -\sum_{t=1}^{N}\frac{\frac{1}{2}}{(\sigma_n^2(t)+\sigma_x^2)} \tag{5.19}$$

and the second term gives:

$$\frac{\partial \sum_{t=1}^{N}\Big[-\frac{(y_t-\mu_x)^2}{2(\sigma_n^2(t)+\sigma_x^2)}\Big]}{\partial \sigma_x^2} = \sum_{t=1}^{N}\frac{\frac{1}{2}(y_t-\mu_x)^2}{(\sigma_n^2(t)+\sigma_x^2)^2} \tag{5.20}$$

the final answer of the partial is and it is set to zero to find the minima:

$$\sum_{t=1}^{N}\Big[\frac{(y_t-\mu_x)^2}{(\sigma_n^2(t)+\sigma_x^2)^2} - \frac{1}{(\sigma_n^2(t)+\sigma_x^2)}\Big] = 0 \tag{5.21}$$

Observe that when $\sigma_{n,t}^2$ is set to 0 the above equation reduces to the standard definition of variance.

The log-likelihood function proposed in equation 5.21 can be numerically solved to estimate process noise variance. It is important to gain insight into how this function depends on changes in the variance estimations. A simple way to visualise the function's behaviour is to evaluate and plot the function value at different values of estimated variance.

Figure 5.1 shows a plot of the log-likelihood function presented in equation 5.21 versus estimated variance. For the purpose of this figure, the true variance was chosen to be 5 and a set of 500 samples were chosen as the batch size. The samples were drawn from normal distributions with zero mean and varying variances. The variance of the measurement noise normal distribution changes as the measurement noise is varying between measurements. A vertical red dashed line shows the true value of variance and a horizontal red dashed line shows zero on the vertical axis.

The two red dashed lines and the likelihood function meet at the true value of variance (5). A green circle marks the crossover. Note that the value of the likelihood function reaches zero when the value of the estimated variance is the true variance. Afterwards, with further increase in estimated variance, the likelihood values become negative.

Zero value of the likelihood function indicates that the function has reached its local maximum. This indicates that the root of the function provides maximum likelihood estimate of the process noise. Hence it is the best estimate of the variance given the set of data. A numerical root finding algorithm such as an inverse quadratic interpolation based method can be used to solve this equation for zero.

5.2.2 Recursive Learning of Process Noise

For a measurement, it is supposed that the total variance is

$$\xi = \boldsymbol{w}_t + \boldsymbol{v}_t \tag{5.22}$$

where w is a process noise sample drawn from a normal distribution given by $\boldsymbol{w}_t \sim \mathcal{N}(0, \boldsymbol{Q})$. and v_t is the measurement noise drawn from a normal distribution given by $\boldsymbol{v}_t \sim \mathcal{N}(0, \boldsymbol{R}_t)$. It is assumed that the measurement noise and the process noise are uncorrelated [100].

It can be seen that estimating the process noise variance is simple if the

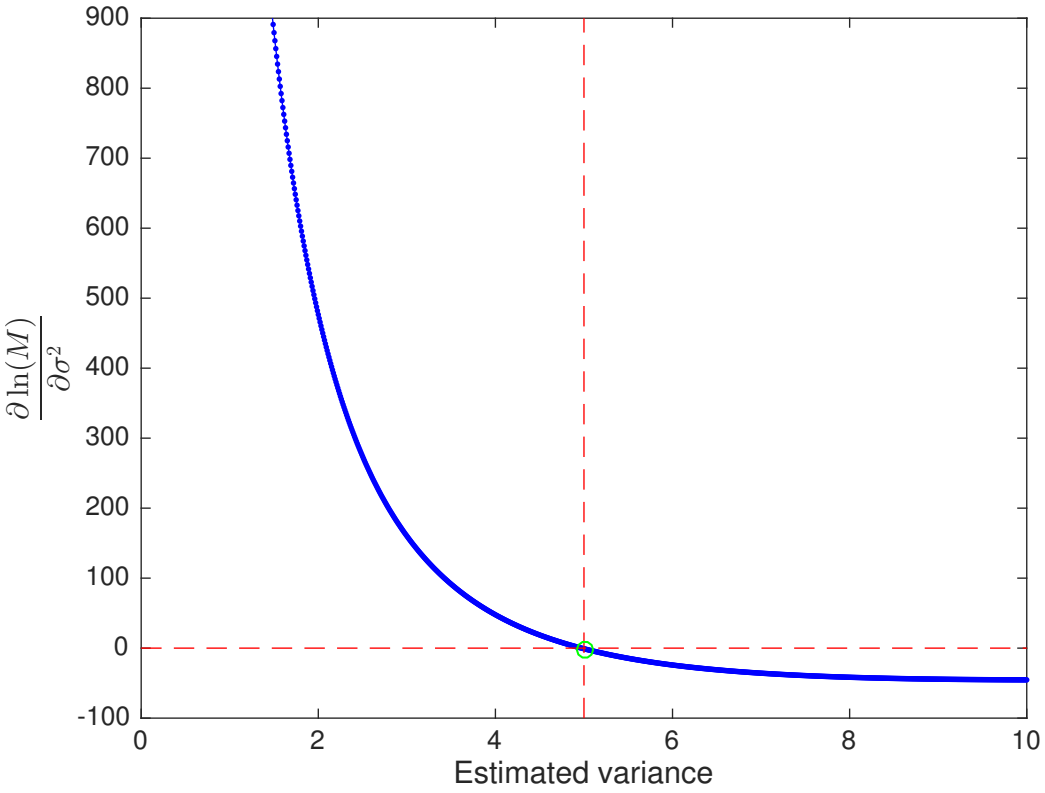


Figure 5.1: Plot of log-likelihood function versus estimated process noise variance. The blue line in the plot depicts the relation between log-likelihood function and estimated variance. The horizontal and vertical dashed lines show the zero and the true variance respectively. Those three line cross at the true variance which is marked with a green circle.

measurement noise is constant for all the measurements.

$$var(w) = var(\xi) - R \tag{5.23}$$

In the case where measurement noise variance changes between observations this cannot be directly applied.

As different observations are made with different measurement noise variance all the observations cannot be trusted equally. An observation with high measurement noise is expected to be further away from the true

value. If this noisy observation is given equal weight to another accurate observation, it will contribute equally like to other observations in a straightforward variance calculation shown in equation 5.23. As an effect, the noisy observation will increase the estimate of variance. Therefore all the observations cannot contribute equally to the variance estimation.

One simple but effective way to modify the contribution of a datum (observation) into the variance computation is to reduce the contribution of noisy observations in variance computation. This reduces the effect of varying measurement noise on observations. Then an averaging mechanism can be employed to compute the final variance. Notice the difference between the flat average based variance estimator and the weighted average is that the noisy observations are given less weight.

As a simple averaging method equally weights all observations it is less adaptive to changes. The weight of each observation in the case of simple average varies in inversely proportional to the number of observations. Which means that for a large number of observations any new data will have a very small contribution towards the estimated mean and the estimator will effectively ignore new changes in the trend of the observed data.

The exponentially weighted moving average (EWMA) improves on simple average estimation. The equal weights are replaced by an exponentially weighted moving average (EWMA). In this method, more recent observations have greater weight when calculating the variance. The weight coefficients of exponential averaging decreases as the temporal index grows into the past. The most recent measurement has the greatest weight in the result and the influence of the previous observations is reduced exponentially as they become older.

In this section, an inverse variance weighted exponential moving average based method is presented. This method is able to estimate process noise under the influence of changing measurement noise. This approach aims to estimate the variance in pixel intensity online.

This approach mitigates the effect of measurement noise variance by inverse-variance weighting the observations by the measurement noise. As noisy observations have higher measurement noise variance, the inverse-variance weighting will be small and this will reduce the contribution of the noisy observation towards the variance estimation. Finally, an EWMA is employed to compute the average variance in the observations.

The mathematical derivation:

The traditional EWMA is given by the following equation

$$\widetilde{s}(t) = \frac{\sum_{k=0}^{t} \gamma^k f(t-k)}{\sum_{k=0}^{t} \gamma^k} \tag{5.24}$$

where $\widetilde{s}(t)$ is the EWMA estimate of variance, t represents time and k is a dummy variable. The above equation (equation 5.24) can be written as a recursive form as below [18]:

$$\widetilde{s}(t) = (1-\gamma) f(t) + \gamma \widetilde{s}(t-1) \tag{5.25}$$

where $\widetilde{s}_{t-1}$ is the estimate at previous time step and $0 < \gamma < 1$.

As the denominator of 5.24 $\sum_{k=0}^{t} \gamma^k$ is a sum of a geometric progression, it approaches $\frac{1}{1-\gamma}$ as time t approaches infinity. This allows equation 5.24 to be written in a compact recursive form shown in equation 5.25.

Notice that there is no inverse variance weighting in the EWMA formulation 5.24. This work improves upon the EWMA by adding an inverse variance weighting so that the proposed method can be used in a varying measurement noise scenario. The proposed method is given by the equation

$$\widetilde{q}_t = \frac{\sum_{k=0}^{t} \gamma^k \lambda(t-k) f(t-k)}{\sum_{k=0}^{t} \gamma^k \lambda(t-k)} \tag{5.26}$$

where $\widetilde{q}_t$ is the estimate of variance, k is a dummy variable, t represents time, $\lambda(t)$ is proportional to measurement precision and $f(t)$ is the squared deviation of the observation from the mean. The sum in the denominator acts as a normalizer.

Notice that in comparison to the traditional EWMA (equation: 5.24) an extra term $\lambda(t-k)$ which changes inversely to the measurement noise has been introduced in equation 5.26.

The proposed filter in 5.26 cannot be written in a compact form similar to equation 5.25 as the values of $\lambda(t-k)$ in the denominator $\sum_{k=0}^{t}\gamma^k\lambda(t-k)$ are unknown in advance.

A recursive-like version of the proposed method can be achieved by writing the denominator and the numerator of the proposed filter individually in a recursive form.

Expansion of the numerator term in equation 5.26 yields

$$\begin{aligned}\widetilde{N}_t &= \gamma^0\lambda(t)f(t) + \sum_{k=1}^{t}\gamma^k\lambda(t-k)f(t-k)\\ &= \lambda(t)f(t) + \gamma\lambda(t-1)f(t-1) + \gamma^2\lambda(t-2)f(t-2) + \\ &\quad \gamma^3\lambda(t-3)f(t-3) + \cdots\\ &= \lambda(t)f(t) + \gamma\big[\lambda(t-1)f(t-1) + \\ &\gamma\lambda(t-2)f(t-2) + \gamma^2\lambda(t-2)f(t-2) + \cdots\big]\end{aligned} \tag{5.27}$$

Now, the numerator of equation 5.26 up to the time instant $t-1$ would be

$$\begin{aligned}\widetilde{N}_{t-1} &= \sum_{k=0}^{t-1}\gamma^k\lambda(t-1-k)f(t-1-k)\\ &= \lambda(t-1)f(t-1) + \gamma\lambda(t-2)f(t-2) + \gamma^2\lambda(t-2)f(t-2) + \cdots\end{aligned} \tag{5.28}$$

From equation 5.28 and equation 5.28 a recursive version of the numerator is obtained which is shown below.

$$\widetilde{N}_t = \lambda(t)f(t) + \gamma\widetilde{N}_{t-1} \tag{5.29}$$

The denominator of equation 5.26 can be written in a recursive format similar to equation 5.29. The recursive form is shown below:

$$\widetilde{D}_t = \lambda(t) + \gamma\widetilde{D}_{t-1} \tag{5.30}$$

Finally the proposed filter can be written as a ratio of the numerator and denominator as shown below

$$\begin{aligned} \widetilde{q}_t &= \frac{\widetilde{N}_t}{\widetilde{D}_t} \\ &= \frac{\lambda(t)f(t) + \gamma\widetilde{N}_{t-1}}{\lambda(t) + \gamma\widetilde{D}_{t-1}} \end{aligned} \tag{5.31}$$

Although equations 5.26 and 5.31 are mathematically equivalent, implementing a recursive version of an equation reduces memory requirements.

As the current output of an EWMA filter depends on both the past outputs and the present input, it is an infinite impulse response filter. How much the filter output depends on past outputs is controlled by the constant parameter γ. γ and the response-speed τ of a filter are related as below:

$$\gamma = \exp(-\frac{\Delta t}{\tau}) \tag{5.32}$$

where Δt is the sampling interval and assumed to be 1. The above equation is rewritten to solve for τ as below

$$\tau = -\frac{1}{\log(\gamma)} \tag{5.33}$$

A high value of γ results in a high value of τ which means that the filter trusts its past and gives less weight to new measurements. Hence a filter with $\gamma = 0.5$ would result in comparatively noisier output than a filter with $\gamma = 0.99$.

Individual measurements are made with different measurement noise in the presented problem. It is desired that a filter in this situation trusts the low noise measurements more than the noisy measurements. That can be interpreted as a filter whose response-speed changes with the measurement noise. The filter should have a high response-speed while incorporating measurements with high noise and a very low response-speed for measurements with low measurement noise. This will effectively ignore the noisy measurements and the filter output will follow past outputs,

whereas a filter with a constant time-constant (τ) will produce comparatively more noisy behaviour.

The time-constant τ_p of the proposed filter depends on both $\lambda(t)$ and γ as given below

$$\tau_p = -\frac{1}{\log(\gamma\lambda(t))} \tag{5.34}$$

where $0.001 \leq \lambda(t) \leq 1$. 0.001 is the minimum measurement noise of a foveation profile and the relationship between λ and measurement noise is given as

$$\lambda = \frac{r_{min}}{r(t)} \tag{5.35}$$

where r is the measurement noise variance and the r_{min} is the minimum measurement noise variance.

It can be observed that $\lambda = 1$, when $r(t) = r_{min}$. Hence from equation 5.26, it can be noted that the response-speed of the proposed method is the same as for EWMA at r_{min}.

Both variance estimation methods (MLE and recursive) require an estimation of the mean to compute the variance. For the recursive process noise estimator, a recursive exponentially weighted moving average was used. An inverse variance weighted mean as shown below was used for the MLE.

$$M_w = \frac{\sum_{i=1}^{b} y(i)/r(i)}{\sum_{i=1}^{b} 1/r(i)} \tag{5.36}$$

where M_w is the weighted mean, $r(i)$ is measurement noise, $y(i)$ is the measurement and b is the size of a batch. The initial choice of a recursive mean is always chosen as half of the maximum pixel intensity (0.5).

Accuracy

Since neither variance nor standard deviation can take on a negative value, the support of the probability distribution describing either is not $[-\infty, \infty]$,

thus the normal distribution cannot be the distribution of a variance or a standard deviation. The correct PDF must have a support on the closed interval of $[0, \infty]$. It can be shown that if the original population of data is normally distributed, then the expression

$$\frac{(n-1)s^2}{\sigma^2} \tag{5.37}$$

where s is a point estimate of the process noise, n is the total number of samples has a chi-squared distribution with $n-1$ degrees of freedom given by

$$\tilde{\chi}^2 = \frac{1}{d}\sum_{k=1}^{n}\frac{(O_k - E_k)^2}{E_k} \tag{5.38}$$

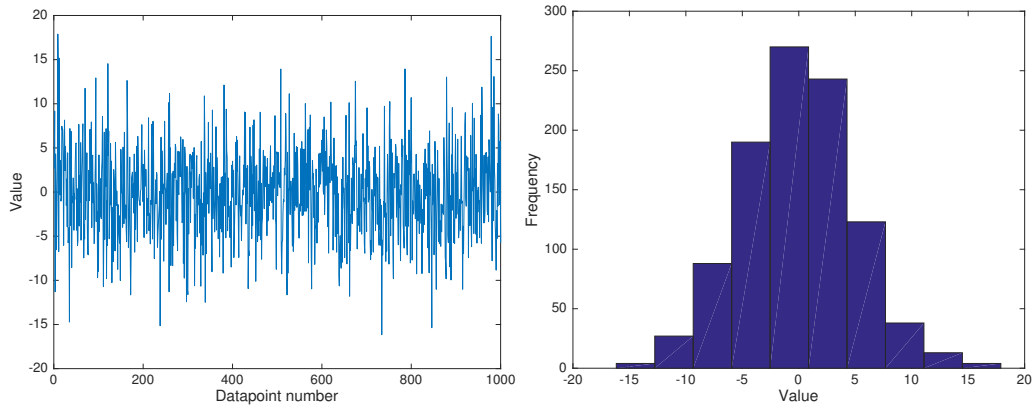
The chi-squared distribution of the quantity $\frac{(n-1)s^2}{\sigma^2}$ allows a confidence interval to be constructed for the estimated variance.

5.3 Results

To evaluate the proposed algorithms, a synthetic dataset of known mean and variance was generated. This dataset is a collection of multiple samples drawn from a Gaussian distribution with a known mean and variance. Both the MLE and recursive variance methods were tested on this data and their performance and error statistics were plotted.

Figure 5.2 shows a plot of the dataset and its histogram. This dataset was drawn from a zero mean Gaussian with standard deviation of five. The left pane of the figure shows the plot of the data points, where the horizontal axis is the data-point and the vertical axis is the value of that data point. The right pane of the figure shows the histogram of this dataset. Notice that as the dataset comes from a zero mean Gaussian, the peak of the histogram is at zero.

Which pixel of a visual scene will be observed with what measurement noise variance cannot be predicted beforehand as the position of the



(a) (b)

Figure 5.2: The dataset used for testing the MLE algorithm and its histogram. The left panel shows a plot of the dataset. The right panel shows the histogram of the dataset. It can be noticed from the histogram and the dataset plot that the mean of the dataset is zero.

foveation profile in a visual scene is a function of the specific visual scene and other conditions such as the internal state of uncertainty. Hence measurement noise variances for the purpose of evaluating the variance estimation algorithms were generated by choosing randomly from a uniform distribution of measurement noise. The distribution had the same maximum and the minimum of the foveation profile discussed in the proposed model chapter 3. The selection process is random with uniform probability for any of the values. As the sampling was done with a uniform probability of selection of any value and without replacement, which results in an unbiased choice for measurement noise variances.

5.3.1 Maximum Likelihood Estimator

This subsection presents the results of the maximum likelihood estimator. The MLE estimator was tested for its estimation error, the effect of dataset size etc. using the same dataset used to test the recursive estimator.

Figure 5.3 shows a plot between absolute percentage error and number of data-points. Each data-point in the figure represents separate experiments. Separate datasets, each drawn from the same Gaussian distribution with zero mean and known variance, with each having an increasing size from the previous one were generated to be used as the dataset for the experiment.

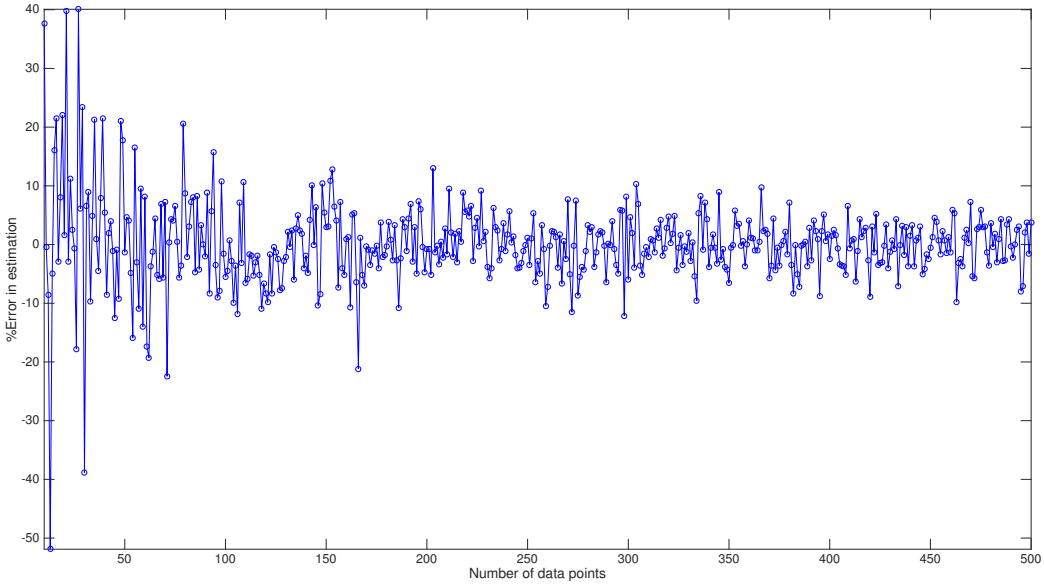


Figure 5.3: Plot of the error statistics of the MLE estimator. The horizontal axis shows the total number of data-points used in the estimation and the vertical axis shows the error value of estimated variance. The plot in blue shows the error in estimated variance. It can be clearly observed that the estimation is better with more data.

Figure 5.4 shows a plot between estimated variance and the number of data-points for an MLE estimator.

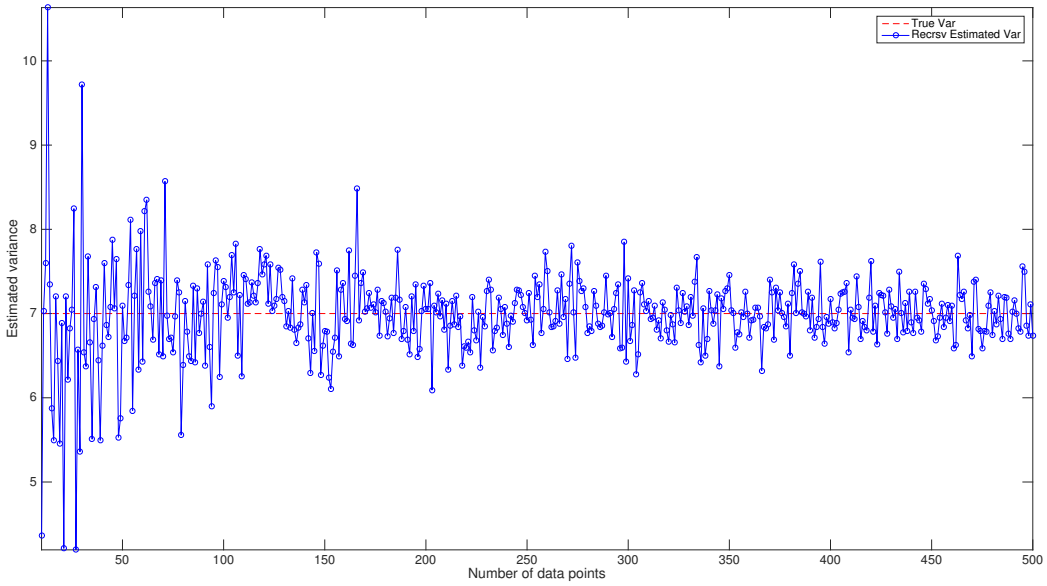


Figure 5.4: Plot of estimated variance and true variance using the Maximum likelihood estimator. The horizontal axis shows the total number of data-points used in the estimation and the vertical axis shows the value of estimated variance. The plot in blue shows the estimated variance and the red dashed line shows the true variance. It can be clearly observed that the estimation is better with more data.

Figure 5.5 presents a comparison between the forcing residual based method proposed by Mayers [135, 177] and the proposed maximum likelihood estimator (MLE) based method. Note that the proposed MLE based method performs better than the forcing residual based method.

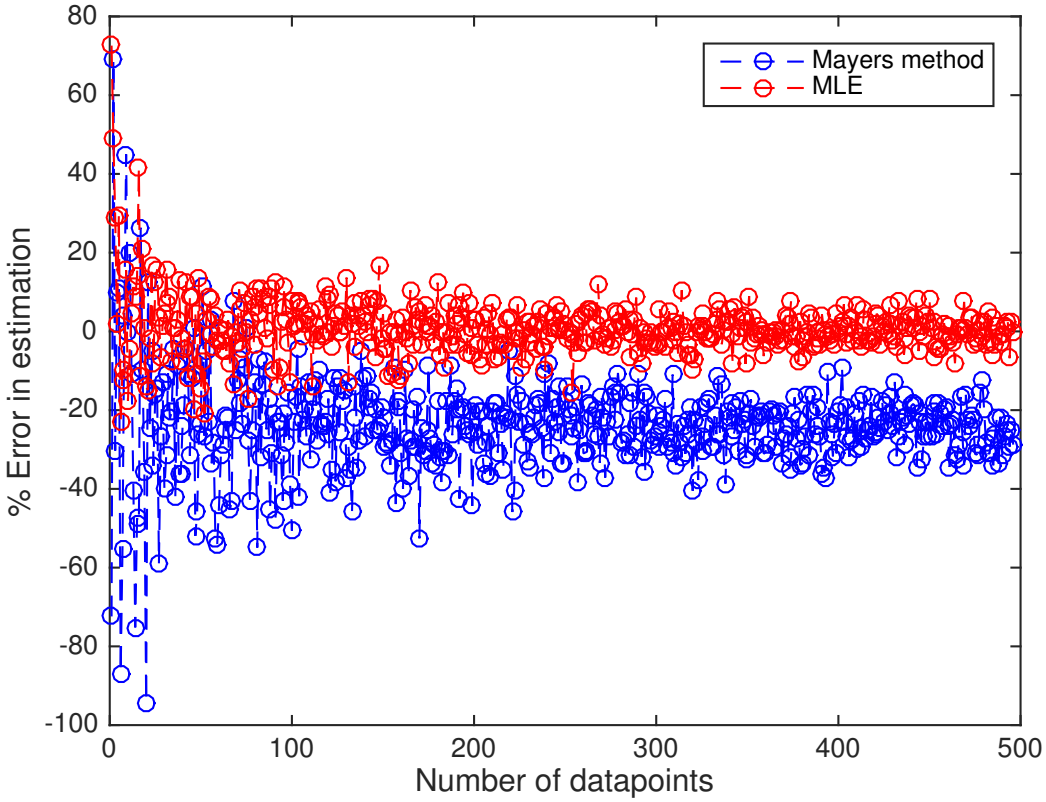


Figure 5.5: Plot of the results of the two estimation methods. The blue line shows the percentage error in estimation of the method proposed by Mayers [135, 177] and the red line shows the percentage error for the proposed maximum likelihood based method. Note that the proposed maximum likelihood based method performs better.

5.3.2 Recursive Estimator

Equation 5.34 was used on linearly spaced measurement noise ranging to find the corresponding response-speeds. The time-constant values were plotted against the measurement noise values. Figure 5.6 shows a plot of measurement noise variance versus the response-speed of the exponentially weighted moving average (EWMA) and the proposed filter. The

horizontal axis shows measurement noise variance and the vertical axis shows response-speed for the proposed filter and EWMA. Minimum measurement noise variance (r_{min}) used for the plot is 0.001 and the maximum measurement noise is 1. Notice that, only at low measurement noise variance, the time-constant (τ) of the proposed filter is equal to that of the EWMA (high response-speed). Elsewhere the time-constant of the proposed filter is low. The response-speed at higher measurement noise changes with the change in measurement noise but the change is too small to be noticed with the vertical scaling of figure 5.6.

It can be observed from equation 5.34 and equation 5.35 that the upper limit of the response-speed of the proposed filter is determined by γ, as when measurement noise variance r is minimum then λ is 1. Whereas the lower limit is decided by the minimum measurement noise variance r_{min} as $\lambda = r_{min}$ when $r(t) = 1$. Also, note that when $\lambda = 1$ the response-speed of the proposed estimator is the same as the response-speed of EWMA. These attributes of the proposed estimator are true for values other than the values chosen at present. Hence the property of the proposed filter of having a high response-speed for measurements with low measurement noise and having a low response-speed for measurements with high measurement noise is not specific to the presented foveation profile and can be used along with other foveation profiles.

Ideally, the proposed filter should behave the same as the EWMA when the measurement noise is minimum. But it is suspected that due to the lack of a compact recursive form of the proposed filter (equation 5.31), the response-speed of the filter may not be as expected, particularly during a transient. As the numerator and the denominator of the equation 5.31 reach steady state values it should start acting as expected. Hence if the proposed filter is run with the minimum measurement noise at every time instant (i.e. $r(t)$ is replaced with r_{min} for all the measurements) on same measurements, its output would not match the EWMA output for an initial period of time, but as the recursive numerator and denominator settle

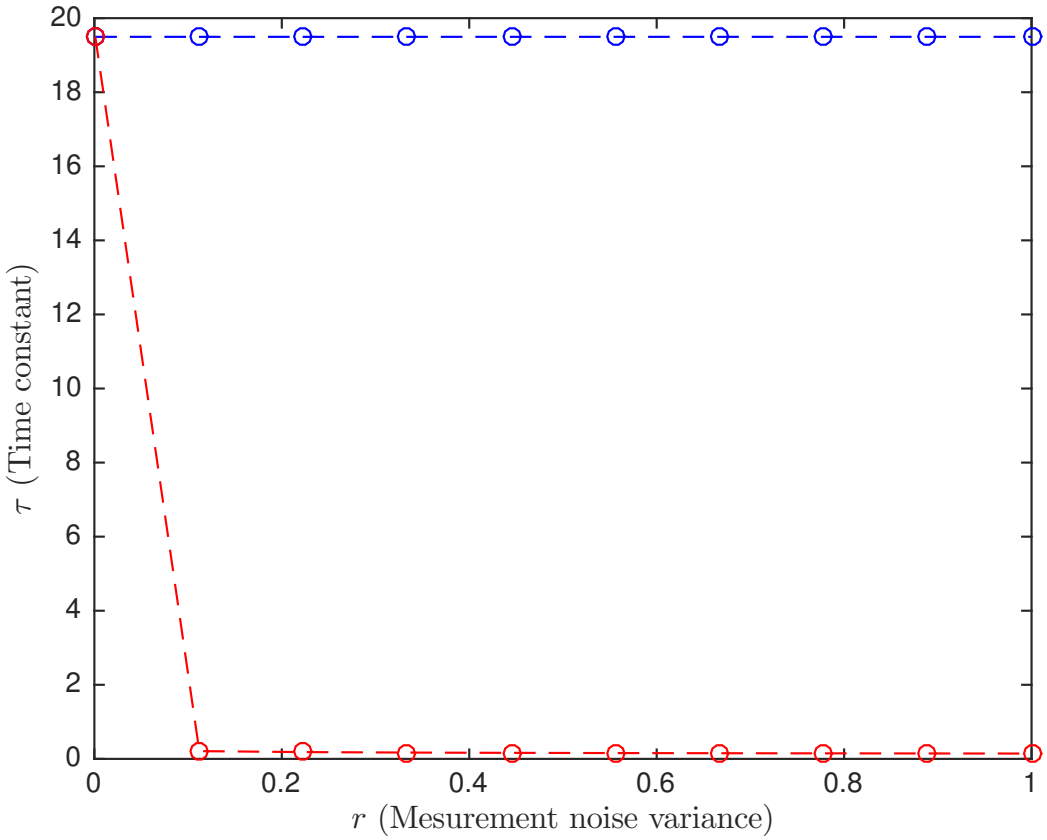


Figure 5.6: Plot of filter response-speed τ versus measurement noise variance. The horizontal axis is measurement noise and the vertical axis is the filter response-speed. The blue line shows the filter response-speed for the exponentially weighted moving average and the red line shows the response-speed of the proposed filter. It can be observed that the filter response-speed of the proposed filter is high at low noise (at $r = r_{min}$) and is low at high measurement noise variance, which makes the proposed filter less susceptible to instant variations in measurement.

down it should start matching the EWMA output.

The proposed filter along with EWMA was run for 500 time steps and the filter outputs were recorded. The minimum measurement noise ($r_{min} = 0.001$) was used for all the measurements.

Figure 5.7 plots the outputs of the proposed filter and EWMA. Notice that the output of the proposed filter differs from that of EWMA for about 100 time steps and matches the EWMA afterwards. The specific trajectories produced by the filters are not fixed and vary from trial to trial.

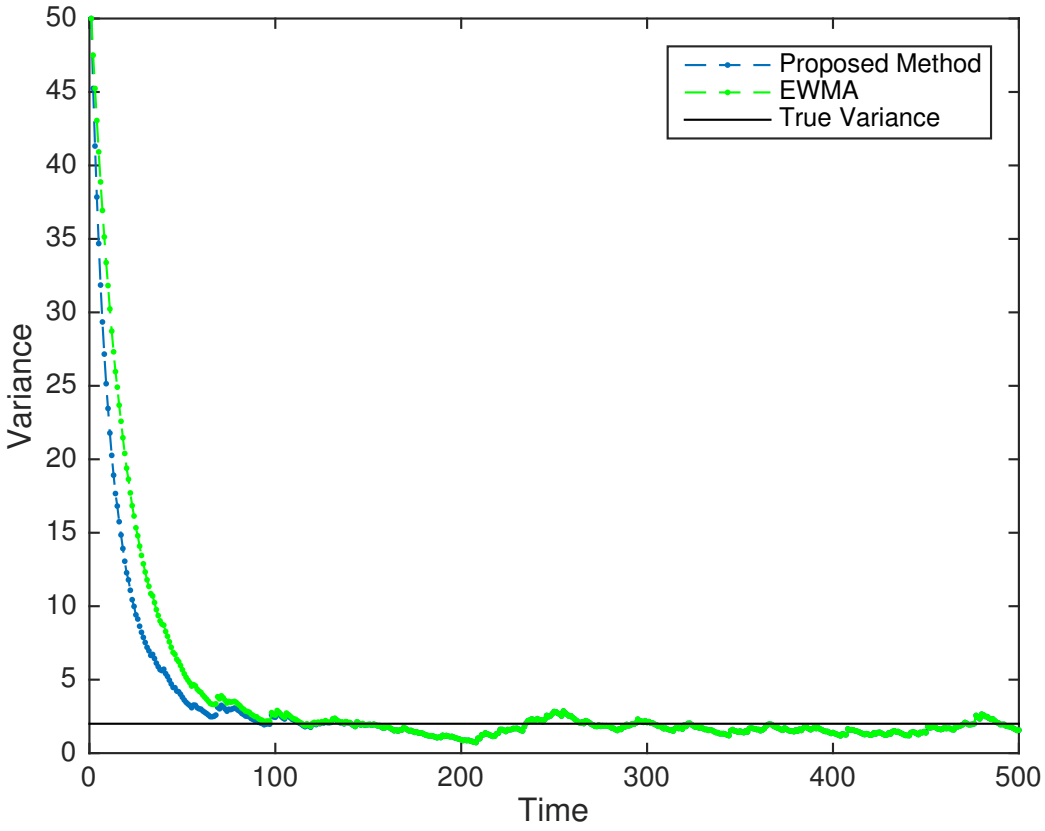


Figure 5.7: Plot of outputs of the proposed estimator and EWMA. The horizontal axis shows time and the vertical axis shows estimated variance. The blue line shows the output of the proposed estimator and green shows the EWMA's output. The black line shows the true variance (4). Notice that the proposed estimator's output starts matching EWMA's output after the first 100 time steps.

As the response-speed of the proposed method is high only for measurements with low noise and is low elsewhere it is expected that the fil-

ter output will ignore noisy measurements. In contrast, the EWMA will incorporate all the measurements whether it is made with low measurement noise or high measurement noise. Hence noisy measurements will influence the EWMA's output to fluctuate, but the output of the proposed method will stay comparatively stable.

To examine the behaviour of the proposed filter, it was run along with EWMA for 500 time steps and the filter outputs were recorded. To capture how the proposed estimator deals with accurate and noisy measurements, the measurement noise was chosen to be the highest (1) for all measurements except for one measurement at $t = 120$. The proposed estimator's output was expected to be more stable compared to the EWMA output. This should capture the estimator's tendency to stick to old values when the measurement noise is high.

Figure 5.8 shows a plot comparing the outputs of the proposed estimator's and the EWMA output along with the measurement noise variances. As the measurement noise variance and the estimated variance are of different scale, figure 5.8 plots two different y-axes one on the left and the other on the right side. Hence the two variances were plotted with two different vertical axes. The axis on the left side of the plot shows estimated variances, whereas the axis on the right side of the plot shows measurement noise.

The horizontal axis shows time. The transient phase of the proposed estimator's output, the first 100 time instances, is not plotted. Hence the horizontal scale of the plot starts for 101. The black horizontal line shows the true variance of the entire dataset. Notice that the output of the proposed method is more stable compared to the EWMA.

Figure 5.9 shows the overall performance of the proposed method compared to the traditional EWMA. The horizontal axis shows time and the vertical axis shows estimated variance. The black horizontal line is the true variance. It can be noted that the EWMA based estimation fluctuates more than the proposed method. Measurement noise variance was chosen

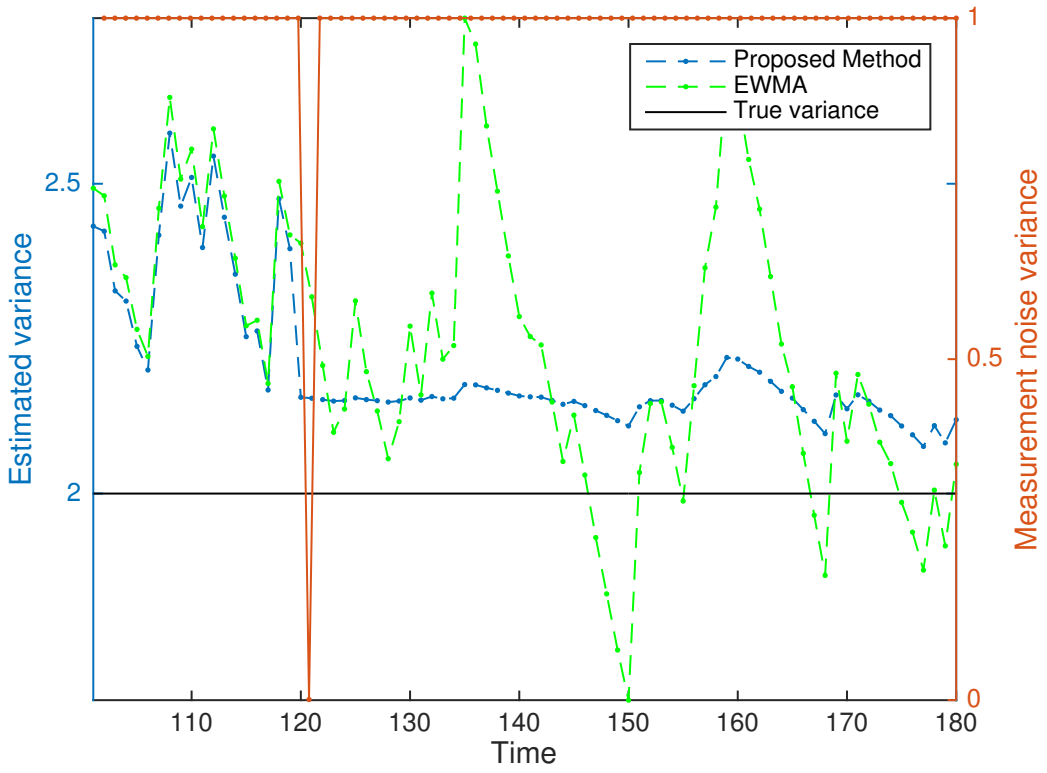


Figure 5.8: Plot of the output of the recursive variance estimator. The vertical axis on left shows estimated variance (blue colour) and the vertical axis on the right (red colour) shows the measurement noise variance. The horizontal axis shows time for both the vertical axes. Notice that the output of the EWMA is noisier than the proposed method.

randomly from a set of measurement noise values with 1 as the maximum and 0.001 as the minimum measurement noise. This selection process has already been discussed at the beginning of this section.

The ability of the proposed filter to effectively ignore the noisy measurement and maintain its previous estimate is important in the context of visual sampling based on internal uncertainty. Short time fluctuation in variance estimation is not a desired quality of the estimator as that would undesirably change the visual sampling distribution between two

constitutive frames of a video. As shown in figure 5.9, with the proposed method visual sampling distribution remains relatively stable for a longer period of time, until an accurate measurement changes the estimation.

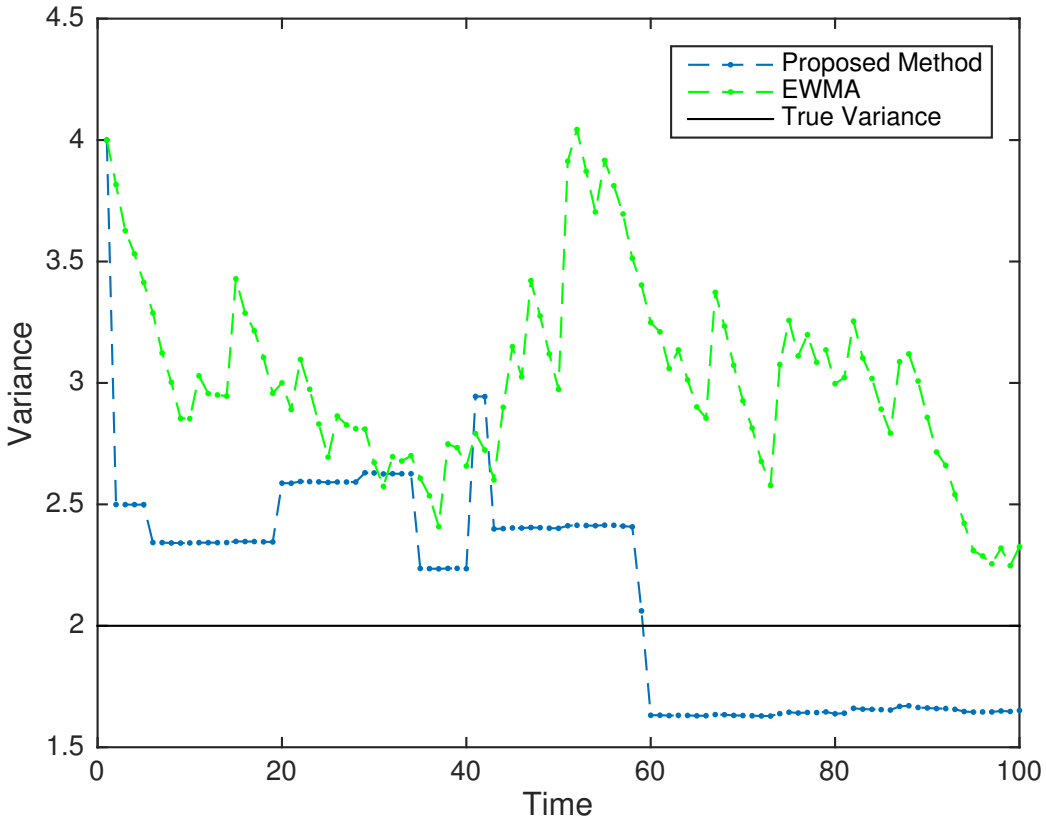


Figure 5.9: Plot of estimation of variance using the proposed method (blue line), EWMA (green line) and true variance (black line). The horizontal axis shows time and the vertical axis shows the estimated variance. Notice that the output of EWMA fluctuates more hence the proposed method is more trustworthy.

Figure 5.10 shows the percentage errors in the estimation of variance using the recursive variance estimation method versus the total number of data-points. It is observed that the percentage error goes down as the total number of data points used in estimation is increased.

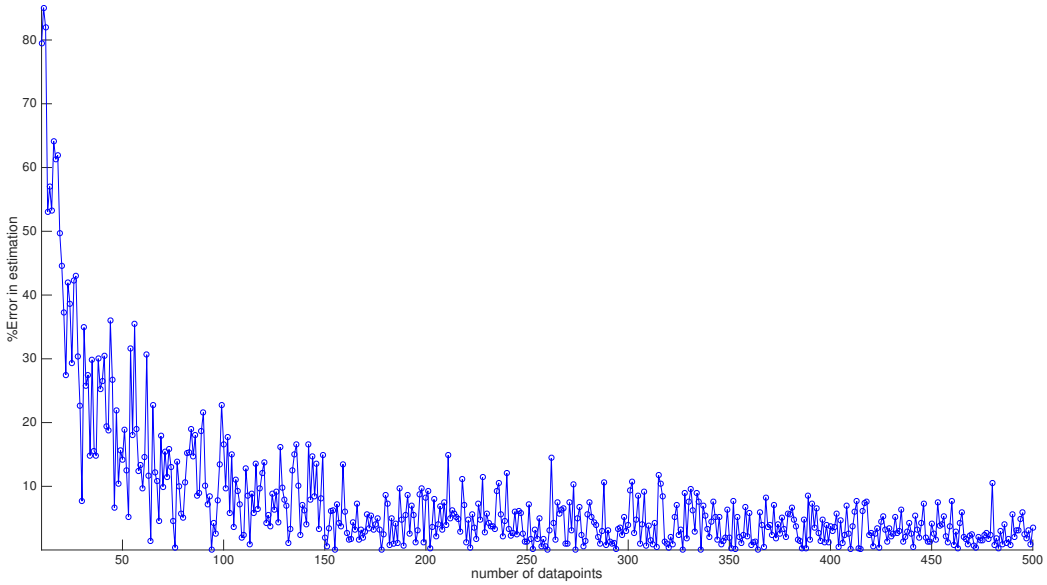


Figure 5.10: Plot of the error statistics of the recursive estimator. The horizontal axis shows the total number of data-points used in the estimation and the vertical axis shows the error in the estimation of variance.

Figure 5.11 shows a plot between the number of data points used versus the value of the estimated noise for the recursive variance estimation method. A red dashed line at the value of true variance is plotted as a reference. Note that the estimated variance is closer to the true variance as the number of data-points increases.

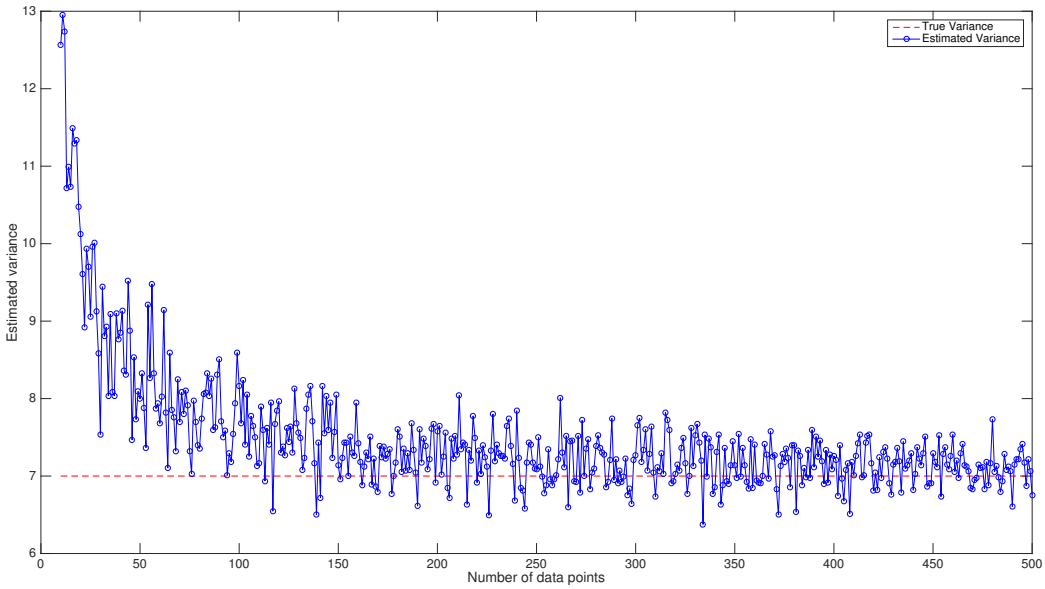


Figure 5.11: Plot of estimated variance and true variance. The horizontal axis shows the total number of data-points used in the estimation and the vertical axis shows the value of estimated variance. The plot in blue shows the estimated variance and the red dashed line shows the true variance. It can be clearly observed that the estimation is better with more data.

5.4 Application to video

It is important to assess the performance of the proposed variance estimators on real life video data rather than the synthetic data drawn from normal distributions. The changes in the pixel intensity of a natural video are exact pixel intensity. Not necessarily always Gaussian, but it is expected that pixels with high and low temporal variances can be modelled with a simple additive Gaussian noise model.

Also, the pixel intensity ranges from 0 to 1 whereas a Gaussian distribution used to model the belief state in a Kalman filter can extend beyond this range. Hence the pixel intensity predicted by the Kalman filter will

not be always accurate. This lack of accuracy is not crucial as the use of a Kalman filter in this work is not aimed at tracking the exact pixel intensity. Instead, the aim is to use the filter's ability to combine new observations with existing beliefs.

The Kalman filter would track the pixel intensity better with a constraint on the mean values of the belief states but that would introduce complication into the filter implementation. The data fusion and belief state uncertainty update feature of a Kalman filter is independent of the mean of the belief state. Therefore the pixel intensity of the videos was directly used without any modification for updating belief states.

5.4.1 Description of Natural Video Content

Videos of natural scenes contain a wide variety of motion patterns of varying complexities. An example of a simple motion would involve an object, for example, a ball moving through the visual scene. Whereas a random group of people walking across a four-way crossing is an example of complex motion as it lacks any specific pattern.

A small subsection of the video which is on the trajectory of the path of the moving ball would show changes in its average intensity over that small area. It would show average background intensity when the ball does not overlap with the subsection and would show the intensity of the ball as it passes through the section. Hence there would be a clear change in pixel intensity.

Events such as changes in camera angle or the appearance of a new object in the scene affect pixel intensity of the video. These events can be grouped into two distinct sets of 'causes';

- Global causes: These are causes that affect the visual scene globally. A change in a global cause, for example, the lighting in a room affects all the pixels in the scene globally. A similar effect can be observed for camera pan, camera tilt, use of a coloured filter, or more generally

change in camera parameters or a scene change.

- Local cause: These causes affect a visual scene locally. For example, an appearance of a new object in a specific part of a scene affects the intensity of that part only. Stochasticity of motion in a region of the scene, for example, someone waving hands, a leaf shaking from wind have local effects.

It can be posited that local causes are more informative of events happening in the scene. Usually, the information content of the video is conveyed by local motions and there are more local motions in a video than global causes. For instance, during a news presentation, the gesticulation of the news anchor carries information and it is there for longer than any global change in the scene. The effect of global causes are surprising for a short while but are not informative afterwards. For example, if the lights go off during a television interview, it triggers surprise at that moment but loses importance after a while.

The local temporal change in pixel intensity is a time-series whose temporal characteristics depend on the underlying set of motions in the video. For example, the temporal characteristics of a subsection of a video showing people walking are different from a ball moving across a uniform scene. As there is no definite pattern of motion in the people walking video the pixel intensity will change abruptly based on many parameters like the colour of the dress worn by an individual, his walking speed, the trajectory of his motion etc. Hence the average intensity of the patch on the video will fluctuate abruptly. Whereas a similar patch on the ball moving across the video shows a more bistable type change. During the presence of the ball, the intensity of the pixel changes to a different state and it stays there until the ball leaves the region and the pixel intensity falls back to the background intensity. Conversely, a wall or a backdrop has very different temporal characteristics. As it is unaffected by any local motion the associated pixel intensity does not show any change.

A location where pixel intensity does not change is predictable in the temporal sense. One past observation can be used to forecast future values accurately. This location can be observed a few times and the measured pixel intensity can be carried forward in time using a predictive model. On the contrary, a location where the pixel intensity changes abruptly need regular observations to keep oneself updated about its state.

In a broad sense pixels from natural videos can be considered to have three groups of temporal predictability. Each group corresponds to a distinct type of local motion in the scene.

- Highly predictable group: Elements in this group correspond to sections of a video that can be predicted with high accuracy. For example, regions associated with a wall, table or chair in a room.

- Unpredictable group: This corresponds to video regions with no distinct temporal patterns. Past observations are not useful in predicting the future state of the pixel. An example would be the random quiver of a tree leaf in wind.

- The medium predictable group are in between predictable and unpredictable groups. When there is a change in pixel intensity it holds the changed state for longer than an unpredictable region. In contrast, unpredictable and predictable regions show frequent and no changes respectively. These changes could be regular or irregular in time, but usually natural videos will observe irregular changes in the pixel intensity. During the steady period, there is always a small quantity of noise on top of the steady value. For example, a shot of turbulent water in a whirlpool would have its natural motion which is in between predictable and unpredictable.

The predictability of image points carries information about local stochasticity of motion hence it can be used as a measure of attentional demand of a visual region. This measurement of attentional demand can be further

used to drive visual attention in proportion to the demand in a dynamic scene.

An average measure of pixel intensity over a wide area of a video includes changes over multiple smaller regions of the video. In that big area, each region may include individual and disconnected changes. Changes in those smaller regions are indistinguishable from the average change in the big area.

For initial experiments, an intensity change over a single pixel was chosen for modelling as it is the most elementary building block of a video. Figure 5.12 shows a plot of the change in pixel intensity over time of a pixel in a video.

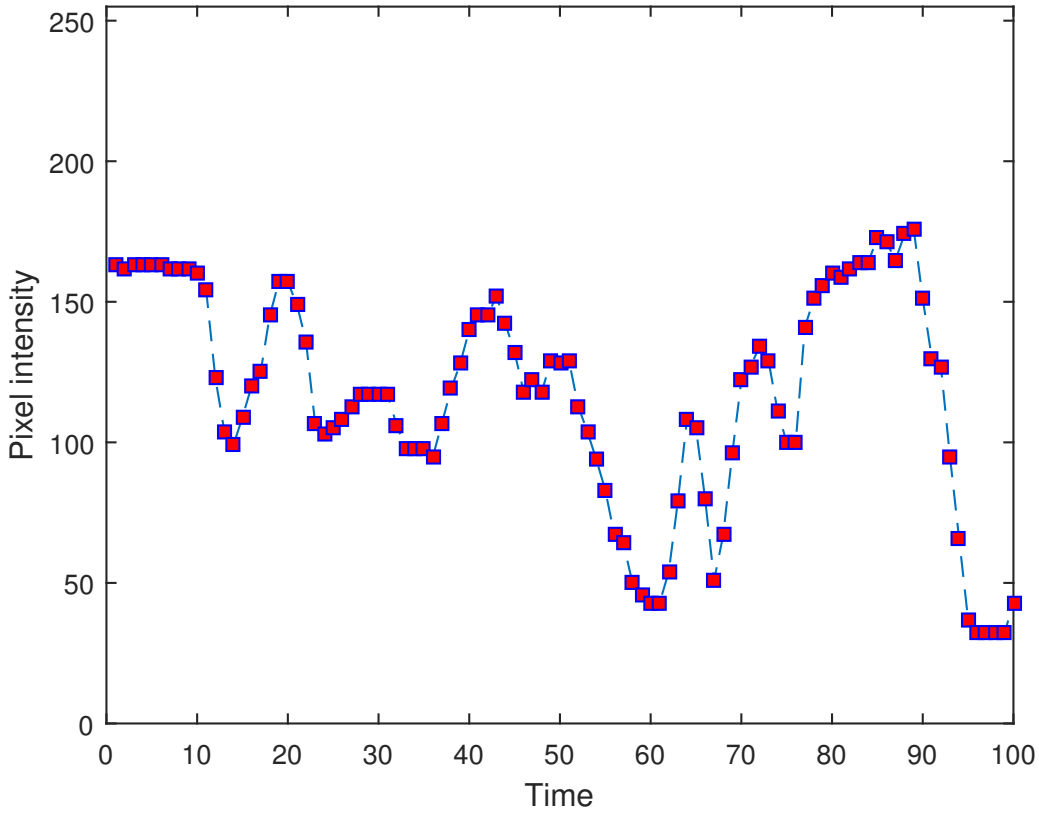


Figure 5.12: Plot of pixel intensities for first 100 time frames versus time from a video. The horizontal axis shows time. Each square in the vertical axis denotes pixel intensity at a specific time instant.

The mathematical relation between two consecutive pixel intensities allows an unobserved state of the future to be predicted based on the knowledge of observed past state. A prediction is never completely accurate due to random adversarial effects on the future state and the limitation of the mathematical relation that describes the relationship between two consecutive states in time. Hence the accuracy of prediction lies within a certain degree of uncertainty.

For a given model that describes the relationship between two states in time, the process noise matrix ($\boldsymbol{Q}$) of Kalman filter equations quantifies how accurate its prediction is. Higher process noise indicates unpre-

dictability by the model and low process noise indicates that the future can be predicted with high accuracy. Also if a pixel's intensity in a real world is affected by unpredictable effects like random motion in the visual scene, the future state is unpredictable and the process noise matrix captures that as high process noise in the corresponding element of the $\boldsymbol{Q}$ matrix. Hence the process noise matrix defines how predictable an observed real-world state is.

A real life agent needs to make accurate and reliable estimates of process noise from observations. Although the proposed estimators were evaluated with synthetic data, it is important to also evaluate their performance on real life video data. The rest of this chapter evaluates the performance of the proposed process noise estimators (maximum likelihood and recursive estimator) with real life videos. It is intended to qualitatively assess the performance of the MLE and the recursive algorithm in following pixels with low and high predictability and how the algorithms cope with unexpected large changes in pixel intensity.

5.4.2 Choice of Videos

Conversation scenes are a typical example in which low, medium and high predictable regions can easily be found. Usual conversations happening in a natural scene will include a background, often containing walls, furniture and perhaps buildings as well as the subjects engaged in conversation, random movements of tree leaves, random movements of people not in focus etc. as a part of the scene. Pixels from the building, furniture and walls have predictable intensity, whereas pixels that are part of the random crowd in the background are unpredictable. The area surrounding the people's head, torso, hands would show medium predictability as it can be expected that people would move their body during a conversation and stay in the new location for some time before returning back to original position.

The Coutrot Database 1 [38] provides visual materials consisting of 15 one-shot conversation scenes extracted from French cinema. Each video features two to four conversation partners embedded in a natural environment. Videos last from 12 to 30 seconds (mean = 19.6 sec; SD = 4.9 sec), have a spatial resolution of 720×576 pixels and a frame rate of 25 frames per second. All the conversations take place in complex but natural scenes (cafe, streets, corridor, office, etc.) involving different moving objects (glasses, spoons, cigarettes, papers, hands, back of a head, face, body part etc.) in separate videos. Faces occupied most of the area in each scene.

To investigate the performance of the proposed process noise estimators, five random videos from the Coutrot Database 1 [38] was chosen. The estimation process involves measurements taken with varying measurement noise as mentioned in the earlier section on results using synthetic data. Each of those videos has low, medium and high predictable visual regions and are generally free from any scene cut or other major global effects (e.g. change in camera angle, pan, zoom etc.). One exception is 'Faces-46' which includes a small camera shake in the beginning of the video.

Pixels that are low, medium and unpredictable were chosen manually by trial and error method by looking at the time-series waveform for a variety of pixels. Also, the standard deviation is a measure of predictability of an outcome hence is high for unpredictable regions and low for predictable regions. Hence the standard deviation of a chosen pixel over time was used to make the selection of the pixel's predictability.

Pixels intensities in the original videos range in value from 0 to 255 (due to 8-bit quantization of sensor output) which was normalised to a standard monochrome image whose pixel intensities range from 0 to 1. In the normalised video, 0 represents black, 1 is the brightest value possible representing white and other intensities in between represent different shades of the grey-scale.

'Faces-clip 46' is a 21 second long video clip of a group of people having a conversation in a restaurant. Only two people of the group are visible and hold centre stage throughout the clip. The background is mostly the wall of the restaurant and partly the couch on which the actors are seated. Any pixel in this region is a predictable time-series (SD=0.0085724). The actor on the right side gesticulates excitedly during the conversation and the area around his hand is an unpredictable region (SD=0.17555). The area surrounding the actors' heads are of medium predictability (SD=0.1354) as the actors move their heads back and forth during the conversation which causes short term (but stable during the term) changes in pixel intensity.

It is important to find out how well the presented model can describe the changes in pixel intensity over time. To understand that the temporal behaviour of selected pixels was plotted. These graphs show how the intensity changes over time.

Figure 5.13 shows one frame from the 'Faces-clip 46' video and a plot of pixel intensity over time for a particular pixel from the video. The left column shows the first frame of the video, the red square that overlaps the frame indicates the pixel choice. The right column shows the change in pixel intensity over time for the entire length of the video clip.

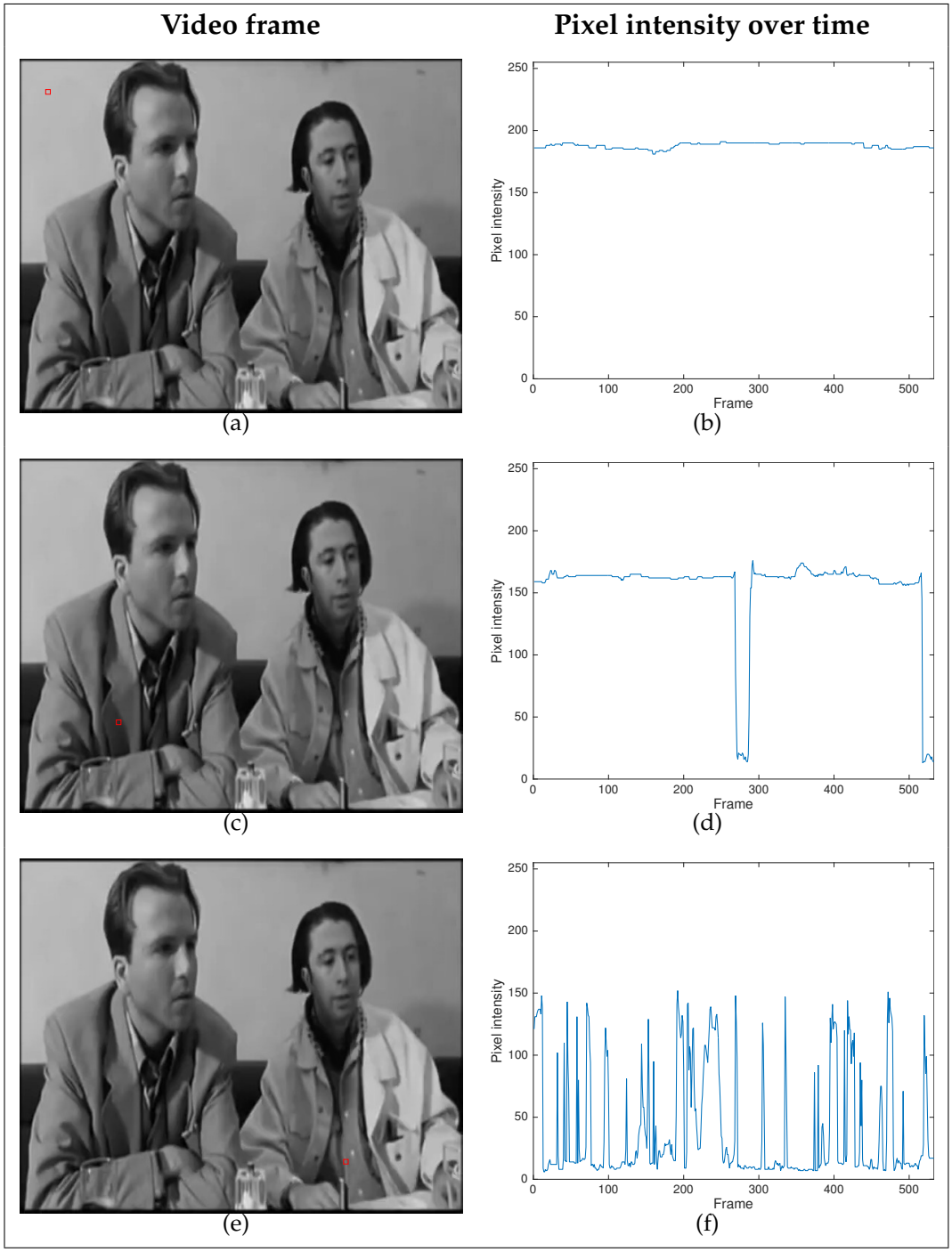


Figure 5.13: The first frame of the faces-46 video and the plot of pixel intensity of a chosen pixel (shown as a red square) over time.

Notice the clear difference in temporal behaviour of the three pixel choices. On the top right panel of 5.13 is the predictable pixel and its intensity does not change rapidly over time. The middle panel on the right has medium predictability. The pixel value changes but remains stable at the new value for a short amount of time before it goes back to the background pixel intensity. In this specific example, the actor on the left holds his drink close to his chest at around 13 sec into the video play. This is reflected as the dip in the pixel intensity near the 300^{th} frame in the time-series plot. Finally, the pixel near the right actor's hand shows unpredictable fluctuation in pixel intensity as the actor moves his hand unpredictably throughout the video (as an act of gesticulation).

An important guide to the persistence of pixel intensity in a time-series is given by the series of sample autocorrelation coefficients. They measure the statistical correlation between observations at different times. The set of autocorrelation coefficients (acf) arranged as a function of lag in time is called the sample autocorrelation function.

The autocorrelation function is a statistical description of how reliably an observation at a current time instant can be used to predict a future state at a certain time-steps ahead. Hence, for the same time lag, the autocorrelation function would have high values for predictable regions and values for unpredictable regions of a visual scene.

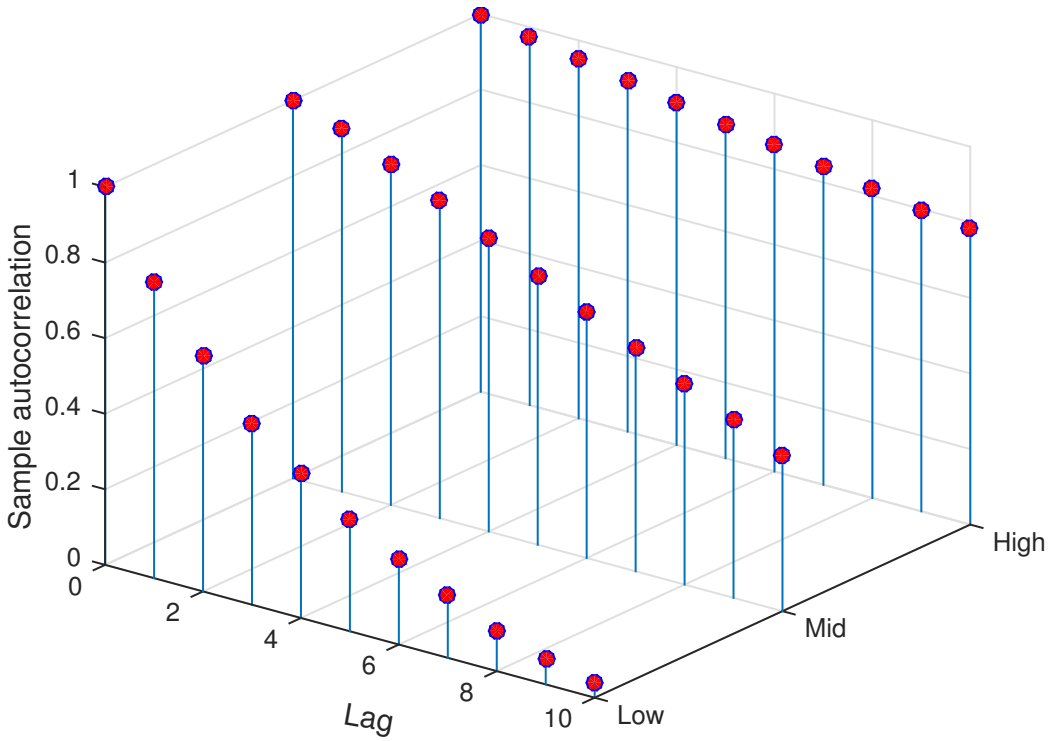


Figure 5.14: Plot of autocorrelation coefficients versus time lag for the locations in video Faces-46. The three lines in the plot depict the relation between measurement noise and posterior variance for three different prior variances. The figure shows that error covariance matrix and the measurement noise follows a positive monotonically increasing relation and the posterior variance is asymptotic to the prior variance at high measurement noise covariances (refer to equations 4.3).

Figure 5.14 shows a 3-dimensional plot of sample autocorrelation versus time lag, called a Correlogram, for a low, medium, and highly predictable pixels. The x-axis shows time lag, the y-axis shows the type of pixel (low, medium or high predictability) and the z-axis shows coefficients of autocorrelation. At zero time lag,t the autocorrelation coefficient is always 1, as is shows the correlation of a data point with itself. The three Correlograms are plotted in the same graph, separated in the y-direction,

for ease of visual comparison.

Notice that the autocorrelation coefficients for the low predictability pixel quickly reaches a value close to zero ($acf = 0.038965$) after 10 time-lags. The autocorrelation coefficient for the predictable pixel remains almost close to 1 ($acf = 0.78272$) at 10 time lag and the same for the medium predictable pixel is ($acf = 0.4137$).

Figure 5.14 is an estimate of predictability of the three different types of pixels. It can be observed that a past observation can be used to forecast the future states of a predictable pixel whereas a past observation does not forecast the future state of an unpredictable region.

5.4.3 Type of Noise for the Chosen Videos

It is important to determine if the additive white Gaussian model holds true for the videos. To test that the temporal mean and variances of multiple time series of a few chosen pixels are computed. The mean and the variance computation considered pixel intensities over the length of the entire video.

Table 5.1 shows three different values of the mean for each high, medium and low predictability with their corresponding standard deviations. Each pixel in any group was chosen manually from the 'Faces-clip 46' video by visually observing the time-series pattern.

The leftmost column on table 5.1 shows the predictability of the pixel. The centre and the right column shows the mean and the standard deviation measures of three pixels belonging to each of the high, medium and low predictability groups.

Notice in table 5.1 that pixels with different means of similar predictability show standard deviations in close range. In each of the predictability groups, the measured mean changes from almost 0.1 to 0.7 but the standard deviations within a group do not scale with the mean. Therefore the average amount of disturbance, measured by the standard deviation,

is not a function of the average pixel intensity. Hence it is reasonable to model the disturbance (noise) as additive in nature.

Pixel type	Mean	Standard deviation
High predictability	0.321	0.009
	0.411	0.009
	0.734	0.008
Mid predictability	0.0789	0.139
	0.361	0.138
	0.736	0.138
Low predictability	0.133	0.186
	0.364	0.188
	0.665	0.178

Table 5.1: Table of pixel intensity mean and standard deviation for three pixels belonging to each of the high, medium and low predictability groups in the 'Faces-clip 46' video.

5.4.4 State Modelling

The use of multiple Gaussian models to describe dynamic scenes at the pixel level, specifically methods involving a mixture of Gaussian distributions, have become popular in the recent years [60, 176, 199].

N. Friedman and S. Russell [60] proposed a model that uses a mixture of three Gaussian components to model visual properties of each pixel.

C. Stauffer and W. E. L. Grimson [176] proposed an adaptive method that models each pixel as a mixture of Gaussian distributions with a variable number of Gaussian components.

C.R. Wren and colleagues proposed a method for tracking human motions in a relatively static background [199]. This method model each pixel with a multi-modal normal distribution. Each dimension of the multi-modal normal distribution models one feature. For example, this approach

will model a quick and random motion of an object in front of a contrasting static background with a covariance that is very elongated in the luminance direction, but narrow in the chrominance direction.

The strength of a mixture of Gaussian models approach is that it can converge to any arbitrary distribution provided there are a sufficient number of components. Multiple states per pixel require a large number of components or states to be tracked by an estimator like the Kalman filter. For instance, a 720×576 video with 3 states per pixel would roughly need more than one million states to be maintained by an agent. This is computationally very expensive.

To address such a computationally challenging situation this work models each pixel with only one state, the pixel intensity. It is very likely that a single Gaussian model is inadequate to accurately model the temporal changes of pixel intensity over time.

A multi-modal representation of the world is not a necessity for the presented work as the main aim of this thesis is to study the behaviour of utility function based sampling. Although a multi-modal model will be more informed and will produce more accurate results, the essential behaviour produced by the utility function does not change. Hence the behaviour can reasonably be demonstrated with a single Gaussian per state model without getting into challenges of computational complexity.

5.4.5 Experimental Setup

The proposed maximum likelihood estimator (MLE) and the recursive estimator were run on videos from the Coutrot video database 1 [38]. Five conversation clips (Faces-46,Faces-51,Faces-53,Faces-55,Faces-57) that have multiple low, medium and high predictability visual regions were chosen from the database for experiments. Each of the five videos contains two actors engaged in conversation in settings like in a cafe(Faces-46), balcony(Faces-51), an airport(Faces-55), public bar(Faces-57), and inside

a room(Faces-53). All five videos show a non-synthetic natural setting, i.e. there is no computer generated graphics in any video.

The area around actors' body parts that move in space, form a medium predictability zone. For example area around the head, torso etc. Elements of the setting which contain the actors mostly show predictable temporal behaviour. For instance the wall in the background (Faces-46,Faces-55,Faces-53), furniture(Faces-51), buildings in the background(Faces-51), windows(Faces-57) etc. A variety of objects and the area around them in the scene show unpredictable behaviour. Examples are someone using his hand to gesticulate (Faces-46, Faces-53), random people walking in the background(Faces-55), smoke coming off a cigarette (Faces-51), moving torso while talking (Faces-55).

The MLE estimator requires the input observations to be grouped in a batch. In contrast, the recursive estimator updates its estimation at every time step.

The change in pixel intensity of a video is one time-series and it needs to be divided into batches for MLE estimator. For the purpose of evaluation of the MLE estimator, all the data-points in a batch were chosen from parts of the video with similar variance. For example, if an actor's head moves forward, the pixel intensity at that region would change. As long as the actor keeps his head in that region, the pixel intensity would remain the same with some small variance. The pixel intensity would return to the background pixel intensity (with some variance) as the head moves back to its original position. Hence there are two clear segments of pixel intensities: one before the head moved in and the second is after the head moved back to its original position.

Thus a complete video is divided into smaller segments and each segment is treated as a batch. Hence the video segment length determines the number of data-points in one batch. The segmentation was done by visually inspecting changes in pixel intensity over time and watching the corresponding sections of the video clip.

The medium predictability region of videos was manually segmented in time, where each segment corresponds to a local motion in the video. If a video had multiple motions, it is divided into multiple segments. On the other hand, high and low predictability regions show similar variance behaviour throughout the video. Hence the entire video was treated as one segment.

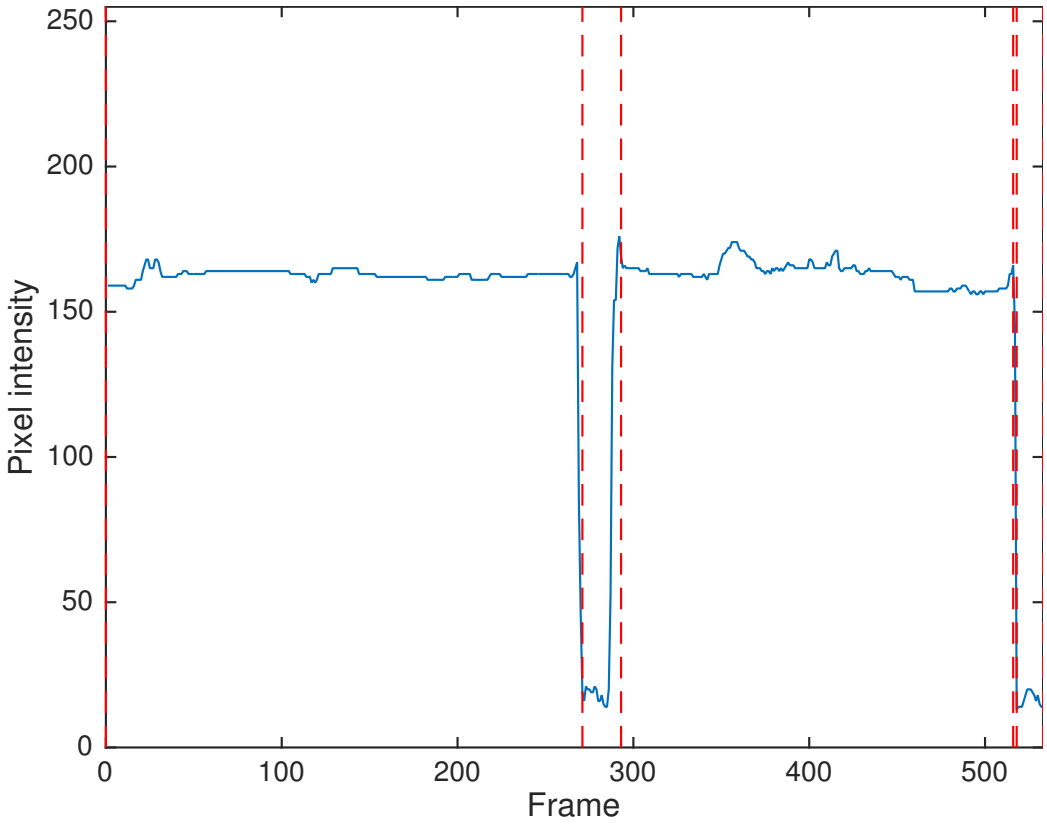


Figure 5.15: Plot of pixel intensity over time. Six vertical dashed lines depict five segments of the time series. The figure shows how pixel intensity time series is grouped into smaller batches. Notice the last vertical dashed red line overlaps the figure axis.

Figure 5.15 is a plot of the medium predictability time-series from the 'Faces-clip 46' video clip. Dashed vertical red lines overlaid on top of the

time-series plot mark time divisions. The section of the time-series that is in between two lines is one batch. Notice that there are five batches shown in the plot.

The first batch is from the beginning of time series ($t = 0$ to $t = 268$) which includes a steep fall in pixel intensity at the end. The steep fall at the end was included in the batch to evaluate how the MLE algorithm performs on sharp changes. The second batch involves a sharp change with less data than the first batch. The third batch is long and there are no sharp changes in the batch. The fourth batch includes only two data points and there is a sharp change in pixel intensity in the batch. Finally, the fifth batch is a small batch and does not involve any sharp changes.

Responses of each of the estimators (MLE and recursive) were recorded and compared with true statistical variance within the batch. The area under the curve between true variance and estimated variance is the error in estimation. The integrated error over time is total error and the total error divided by total time (i.e. error per unit time) is the average error.

$$E_{tot} = \int_{t=a}^{t=b} |e(t)|dt \tag{5.39a}$$

$$E_{avg} = \frac{\int_{t=a}^{t=b} |e(t)|dt}{b-a} \tag{5.39b}$$

where a, b are limits of integration that marks the beginning and the end of the time duration of the integral, $e(t)$ is instantaneous error, E_{tot} is total error and E_{avg} is average error. As the process noise estimation starts at the beginning of the video and runs up to the end, the lower limit of integration $a = 0$ and the upper limit is b is the total number of frames of the video.

As the estimated variance curves are irregular in shape, the error in estimation is computed numerically as an approximated integral (approximation is done using trapezoidal method, with unit spacing in time axis) of difference in area between the true and the estimated process noise.

The integrated area under the curve between the true and the estimate indicates total error and is a measure of performance of the estimators. As the MLE is a batch estimator, its estimation remains constant for the duration of a batch, whereas the recursive estimation varies at every time step (See figure: 5.16).

It might seem from figure 5.16 that the MLE estimator performs better than the recursive estimator as visually the area shown in green is less than the area shown in blue. But these error curves depend on specific selection of measurement noise, choice of γ, and on the accuracy of the estimated mean. Hence error curves change between runs and there could be instances where the recursive estimator performs better than the MLE estimator.

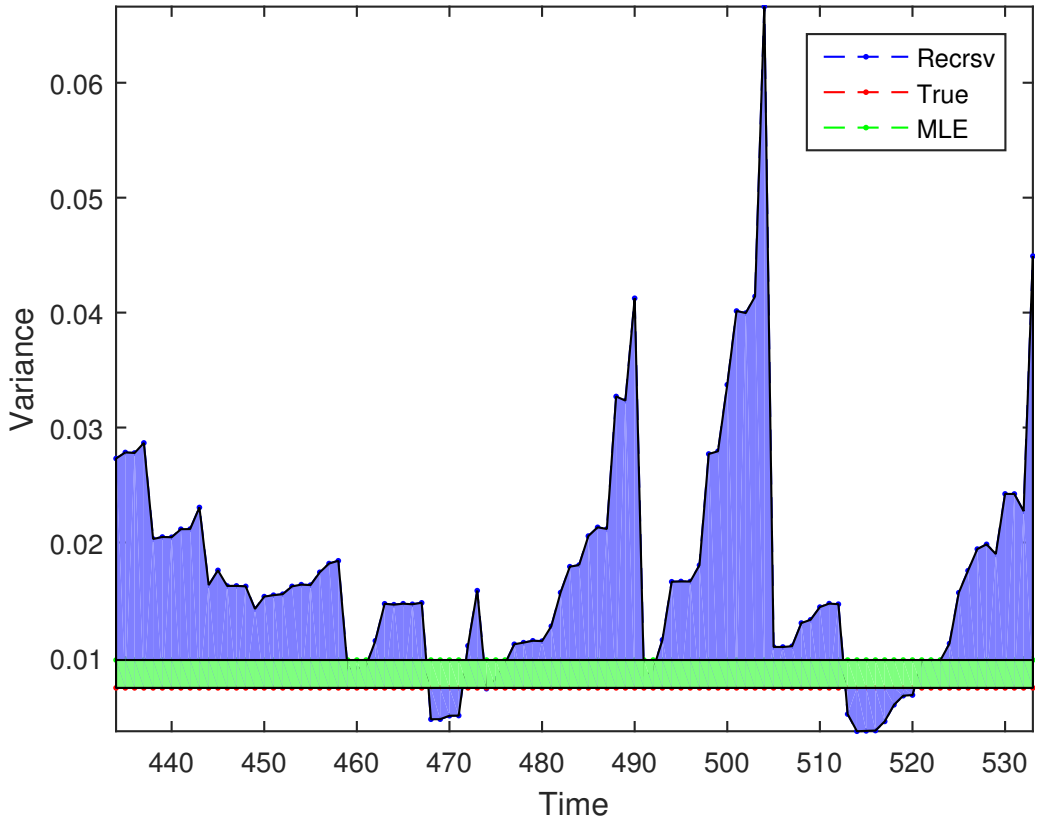


Figure 5.16: Plot of estimated variance and true variance with estimation error shaded in colour. The area shaded in blue indicates error in estimation by the recursive estimator and the green area shows the error in MLE estimation. Notice that the error for MLE is constant, whereas the error of the recursive estimator changes with time. Integration over the entire shaded area for a given estimator is a measure of total error in estimation.

While evaluating the maximum likelihood estimator on synthetic data it was observed that performance of MLE suffers when there are not enough data points in a batch(see figure: 5.4). It is expected that the MLE would produce erroneous result for the fifth batch and its performance would be better for the other divisions. Also, the effect of choice of response-speed(γ) on the recursive estimator and effect of size of dataset on maximum likelihood estimator was noticeable. It is described below.

γ	Total error in recursive estimation
0.95	17.841
0.85	31.190

Table 5.2: Table showing total error in estimation for two values of γ. A low choice of γ results in more total error and a high choice results in less total error.

5.4.6 Results of Process Noise Estimation

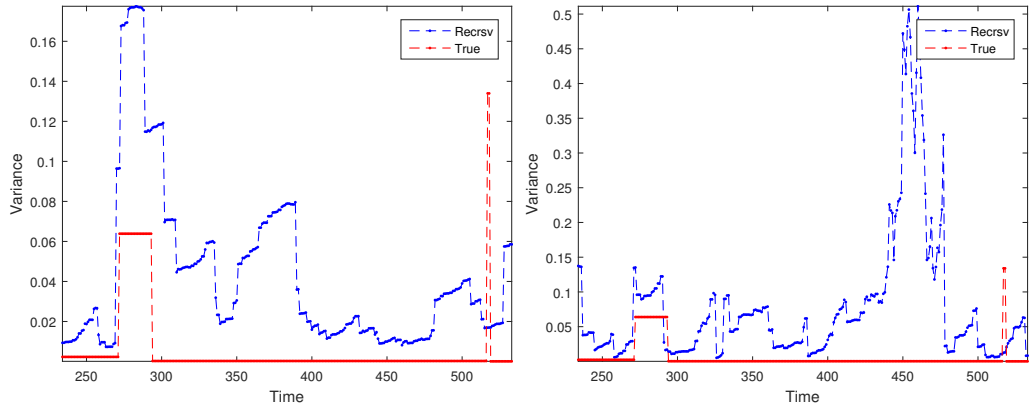
Effect of γ on recursive estimation

Figure 5.17 is a plot of recursive estimation for 300 time instances, overlaid with the true variance estimation. The height of the red depicts changes in true variance changes. The blue line shows the estimate and the red line shows true value. The left panel shows the estimate with $\gamma = 0.95$ and right panel shows the estimate with $\gamma = 0.85$ on the same data.

If the choice of γ is high ($\gamma = 0.95$), the recursive estimator trusts past estimates more than the recent measurements and fails to catch up with short term changes in the time series. This results in a slow response in estimation from the recursive estimator. Notice that the estimator took around 50 time steps to reflect the change in variance (5.17, left panel,$\gamma = 0.95$,$t = 270$ to $t = 320$). In contrast, the recursive estimator takes around 26 time steps to reflect the change.

Low γ causes the estimator to trust measurements more. This causes the estimator to follow the real time-series data closely but the recursive variance estimation is easily affected by noisy measurements and the estimate fluctuates. Estimate fluctuations result in more total error ($E_{tot} = 31.190$) compared to more non-reactive version of the estimator ($\gamma = 0.95$), which results in total error of $E_{tot} = 17.841$.

Figure 5.18 shows plots of pixel intensities over time and their corresponding estimations of variance. The left column shows pixel intensi-



(a) $\gamma = 0.95$ (b) $\gamma = 0.85$

Figure 5.17: Plot of estimated variance and true variance using the recursive estimator with two different values of γ. The horizontal axis shows the total number of data-points used in the estimation and the vertical axis shows the value of estimated variance. The plot in blue shows the estimated variance and the red dashed line shows the true variance.

ties where upper, middle and bottom rows show predictable, medium predictable and unpredictable pixel correspondingly. The right column shows estimations of variance of the corresponding pixel shown in the left column. All the pixels shown in this plot are chosen from the 'Faces-46' video. The γ for the recursive estimator was chosen as 0.85 as low choice of γ allows the recursive estimator to follow the changes in variance better.

As MLE is a batch estimation method, its estimation is constant within a time window whereas the output of recursive estimation changes at each time step. A small number of data points affect the MLE negatively causes it to overestimate variance, which can be observed in the middle panel. The distance between the green line and the true variance is high, indicating higher error in estimation(around $t = 282$ and $t = 517$). Where the data size is big MLE performs better. Average estimation error is 0.6381 with 2 data points and 0.0929 with 22 data points in contrast with 0.0002 with 223 data points. In general, MLE seems to have performed better than the

recursive estimator and results in less error in estimation.

Table 5.19 shows a summary of average estimation errors of MLE and recursive methods. Each of the estimators was run on three choices of pixels on each of the five videos choices from the Coutrot video database 1 [38]. The first, second and third rows in the table show average errors in estimated variance of low medium and noisy pixels respectively.

Unlike the previous measurement of error presented in table 5.2, total error cannot be used to compare performance across videos of different lengths. The total error is a function of the video length and the video lengths of the five chosen videos are different. Hence the effect of video length on error calculation needs to be nullified. To bring error estimation on a similar footing, average error per unit time was chosen over total error.

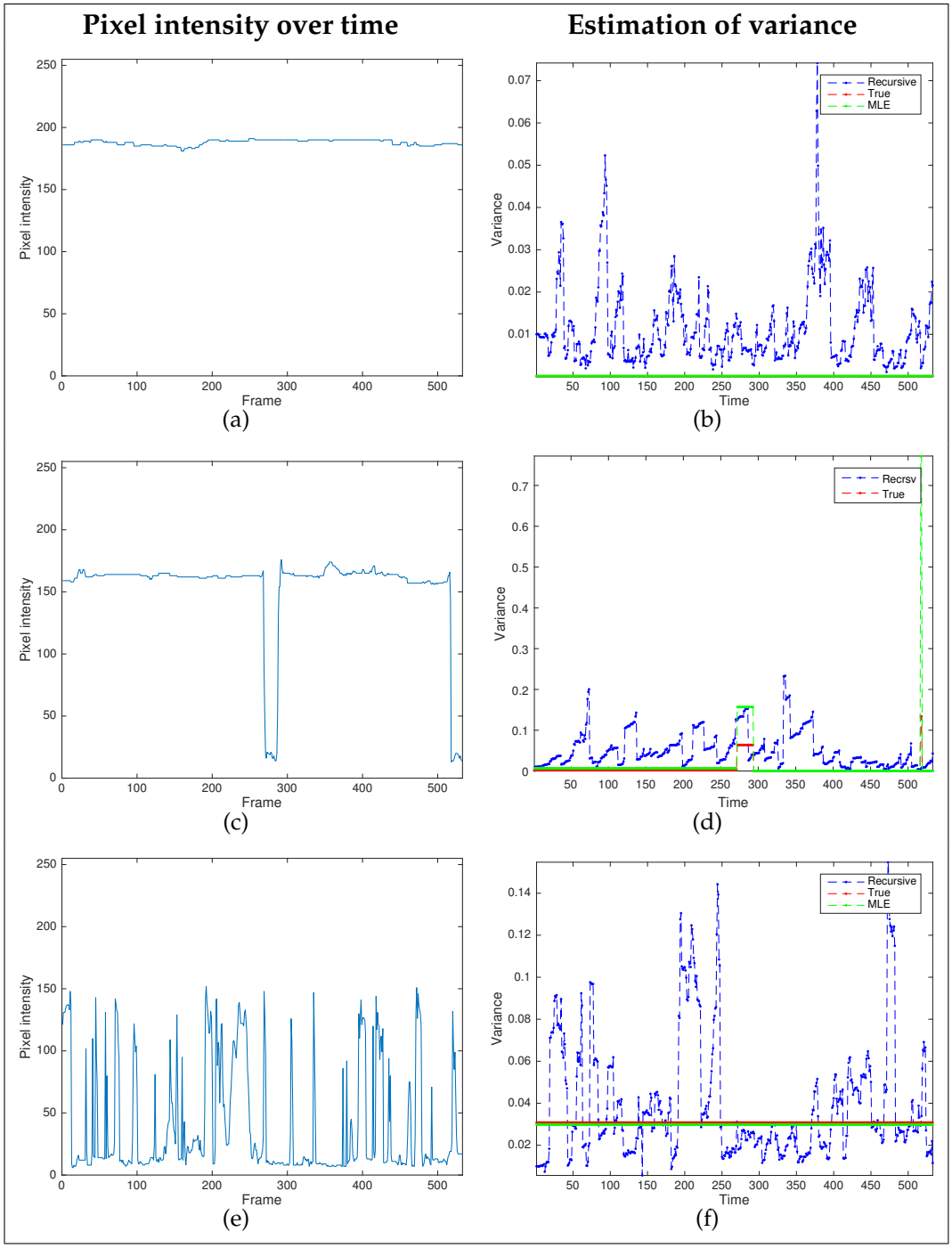


Figure 5.18: The left column shows plot of pixel intensities over time and the right column shows estimated variance by MLE and recursive estimators for that pixel. Blue, green lines show recursive and MLE estimations respectively. The true estimation is shown in red.

Video name	Pixel characteristics	E_{avg}	
		Max likelihood estimator	Recursive estimator
Faces-46	Low variance	0.0024	0.0128
	Medium variance	0.0025	0.0128
	High variance	0.0023	0.0033
Faces-51	Low variance	0.0007	0.0153
	Medium variance	0.0077	0.0162
	High variance	0.0271	0.0310
Faces-53	Low variance	0.0000	0.0150
	Medium variance	0.0050	0.0133
	High variance	0.0101	0.0249
Faces-55	Low variance	0.0002	0.0094
	Medium variance	0.0052	0.0183
	High variance	0.0015	0.0031
Faces-57	Low variance	0.0009	0.0146
	Medium variance	0.0063	0.0210
	High variance	0.0141	0.0197

Figure 5.19: Summary of average estimation errors of MLE and recursive methods.

Notice that on average the MLE performs better than the recursive variance. Amongst the three types of pixels, estimating the variance of the unpredictable pixel produces the highest error.

A point worth noticing is that in the case of maximum likelihood based batch estimation of variance, the sequence of sampling in a batch of data is irrelevant as the set of data-points is treated as a whole. In contrast, the sequence in which individual measurements occur affects the trajectory of the recursive estimator.

5.5 Chapter Discussion

This chapter contributes two novel algorithms that can learn process noise from observations made with varying measurement noise. To the best of the author's knowledge the two proposed algorithms are the first to learn process noise from observations made with varying measurement noise. This two process noise learning methods are intended to help the Kalman filter based visual target selector adapt to a new dynamic visual scene.

In the context of the saliency map literature, this chapter presented the first demonstration of adapting inhibition time according to the temporal characteristics of the visual scene.

The drawback of the MLE estimator proposed is that it cannot follow instantaneous changes in variance as it is a batch processing method. However, the recursive estimator can follow instantaneous changes, given its response-speed is chosen to be sufficiently short, but is less accurate than the MLE. Choice of γ is an important design parameter for this estimator. Low γ allows the recursive estimator to follow changes in variance closely but is easily affected by noise. High γ lowers the recursive estimator's response-speed, therefore, the past estimates are trusted more and estimation is robust to noise. The drawback of high γ is the estimator cannot follow quick changes.

It is important to note that the aim of this variance estimation is not to produce an accurate estimate of the variance in the video. Instead, the aim here is to make an estimate of the process noise so that a foveation profile can be driven sensibly. Hence an accurate estimation is not necessary as long as the estimation is within a limit that keeps the corresponding pixel within its original category of low medium or high predictability.

It was assumed that visual regions with their characteristic predictability do not change over space. However, in real life, characteristic predictability of scene regions might change. This change in temporal characteristic should not hinder the performance of the two estimators proposed

here.

These two variance estimators will be used in the next chapter (chapter 6) along with the proposal of the novel utility function.

Chapter 6

Utility functions

6.1 Introduction

In this chapter, the proposed Kalman filter based visual target selector will be experimented with using multiple utility functions to select visual targets in a dynamic visual scene and the model's behaviour will be studied. The experiments will be carried out in three stages, where with each stage progressively more complexity will be added.

The first stage presents how the proposed Kalman filter based target selector prevents fixation without using the classical IOR mechanism. This stage proposes three utility functions and studies the resulting system behaviours. The proposed model is tested with a set of simple videos without using the Itti saliency map to study the model's behaviour. In these experiments, pixel intensity of the videos is used for the internal belief states.

Finally, in the third stage the proposed model will be applied to the Itti model. This stage will present a comparative study between the ability of the Itti model and the proposed model to predict human fixation for a popular video dataset. The ability to predict human fixation of a video is an important application of saliency models as this has many important real-life applications like video segmentation, video encoding, visual

SLAM.

6.2 Experimental Method

Visual scene sampling behaviours of an agent with each of the five utility functions were explored. The results of each of the utility functions are presented in their corresponding sections of this chapter. Each section presents an analytical explanation of the agent's behaviour using the help of Kalman filter equations and the utility function. Attention distribution plots are used to show how the agent distributes attention over the five chosen videos from the Coutrot video database 1 [38].

Similar methods are used for experimenting with each of the utility functions. So, a general description of experimental methods is discussed here. Where necessary, each subsequent section presents a small additional description, explaining aspects of the experiments related to that section only.

All the proposed utility functions were implemented using the MATLAB® programming language on a Linux platform. In each experiment, an agent running Kalman filters that follow the pixel intensities of a video makes its decision of where to look in the visual scene based on one of the five utility functions discussed above. Given a visual scene and a utility function, how the agent distributes its visual attention is noted at each time step. Then this data is used to plot an attention distribution histogram, and to show the agents decision at each time step.

An attention distribution histogram is a plot of the number of visits to a visual location versus the location it visited. The number of visits is colour coded into a $2D$ plot to show a heatmap of the frequency of visit to all possible locations within a video. The $2D$ heat map is overlaid with the first frame of the corresponding video for a better reference for visualisation of which location the agent visited the most. This plot depicts what region of the video got how much visual attention.

Another plot shows the histogram of the visual attention distribution versus process noise. This plot describes what process noise attracted how much visual attention. This histogram is necessary to depict how the agent distributes visual attention based on the process noise matrix of the video.

It is expected that the distribution of visual attention over process noise will be different based on the choice of utility function. For example, when an agent aims to reduce internal uncertainty it should preferentially visit rapidly changing visual locations. On the other hand, if an agent aims to ignore the predictable and unpredictable elements in the scene, its attention should be paid towards visual regions with a medium predictability that lies in between predictable and unpredictable.

The two process noise learning algorithms presented in the earlier chapter 5 are implemented. The MLE based learning algorithm learns process noise from batches of data of predefined size. In contrast, the recursive learning algorithm learns process noise at every time step of algorithm execution.

It is important to note that the entire process noise matrix is replaced with a new one at the end of a learning window in case of the maximum likelihood estimator. Hence the agent operates based on the last updated version of the $\boldsymbol{Q}$ matrix. At any point of time, an agent with MLE estimator operates based on the $\boldsymbol{Q}$ matrix it learnt at the end of the last learning window. Any attention distribution plot would reflect the result of only the last learnt process noise matrix. Hence it was decided that the results of only the first learning window and its resulting attention distribution will be presented in this chapter.

Recall as discussed in the proposed model chapter 3, the pixel at the profile centre is observed with the lowest measurement noise. The remainder of the visual scene is observed with increasing measurement noise from the centre. Although the generation of foveation profile was discussed in an earlier chapter, a short description is provided next as a recapitulation.

6.2.1 Choice of Foveation Profile

The foveation profile is the distribution of measurement noise over space under the observable region of the agent's field of view. A biologically plausible foveation profile has minimum measurement noise at the centre and measurement noise increases towards the periphery.

A simple exponentially increasing function was used to model the outwardly increasing measurement noise property This function is:

$$f(d;\rho) = 1 - \exp^{-\frac{d^2}{\rho^2}} \tag{6.1}$$

where d is the distance from the profile centre and the foveation profile parameter ρ^2 determines the rate of increase in measurement noise with distance from the profile centre. A smaller value of ρ^2 results in a sharp rise in the measurement noise from the profile centre and a large value results in a slowly rising measurement noise that covers more visual region with low measurement noise.

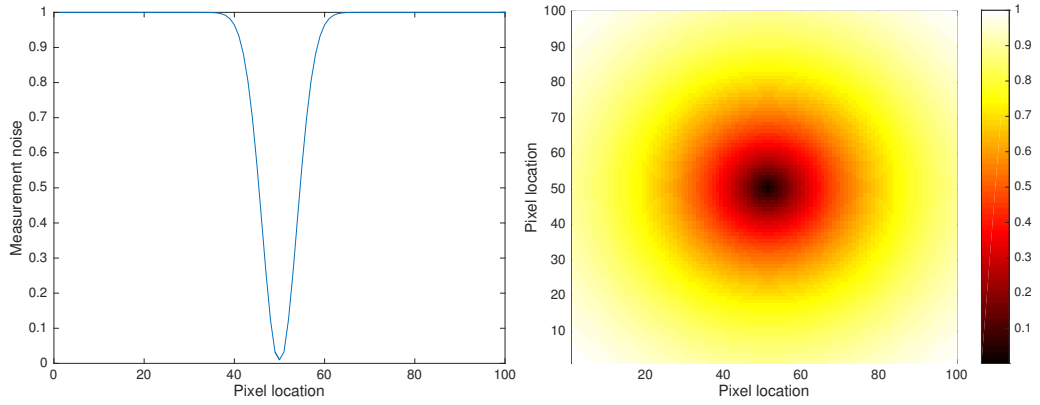
Figure 6.1 shows two foveation profiles with $\rho^2 = 30$. The profile shown on the left is a one dimensional foveation profile, whereas the one on the right is a two dimensional profile. The horizontal axis of the left panel shows pixel number and the vertical axis shows the measurement noise. For the right panel, both the horizontal and the vertical axis shows pixel location and the measurement noise is colour coded. Both the foveation profiles have their profile centre at the centre of the plot. Notice that the measurement noise increases symmetrically with distance from the profile centre.

The foveation profile is normalised so that the maximum value of measurement noise is 1. To avoid numerical problems arising from using zero uncertainty at the profile centre the minimum value was set to 0.0001. The two dimensional profile has been used throughout all the experiments presented in this chapter.

The main intent of this chapter is to understand the overall system behaviour. It is independent of the specific choices of minimum or maxi-

mum value of the measurement noise. Rather it depends on the choice of the utility function.

Note that the foveation profile is characteristic of a particular sensor. A system designer is not free to choose the profile used but must select one that is matched to the actual sensor for the purposes of experimental demonstration in a simulation.



(a) (b)

Figure 6.1: Plots of one dimensional and two dimensional foveation profiles. Each of the profiles has $\rho^2 = 30$. The horizontal axis shows pixel location. The fixation points for the foveation profiles shown are in the middle of the scene. For the left plot, the vertical axis shows the measurement noise, whereas the measurement noise is colour coded for the $2D$ foveation profile plot. Please note that the Gaussian distribution was normalised to $0\ 1$ for both the plots. This can be noticed on the vertical axis of the plot. On the right pane, the maximum height of the 2D Gaussian is 1.

Sometimes it is easier to describe the output of a utility function on a smaller dimensional world. In such cases, the utility function is applied on a small synthetic visual scene made of only ten pixels to describe the output. Then the utility function is applied to the videos from the database.

6.2.2 Generation of Synthetic Saliency Map for Initial Tests

A method was designed for creating synthetic saliency maps with a set number of saliency peaks and their widths.

Ten one-dimensional Gaussian bumps of fixed size in a 100 pixels wide visual scene (see figure 6.2) was used for saliency map generation. The following equation was used to generate saliency peaks:

$$f(x_b) = \frac{1}{\sigma_b\sqrt{2\pi}} e^{-(x_b-\mu)^2/2\sigma_b^2} \tag{6.2}$$

μ describes the location of the peak in the above equation and σ_b determines how wide the curve is around the peak.

Different values of σ_b (one σ_b for each peak) were chosen by randomly sampling from a uniform distribution with predefined boundaries. A narrow Gaussian produces a higher saliency peak than a wide Gaussian.

Similarly, the locations of those peaks were chosen by random sampling from another uniform distribution, which spans the same limits as the visual scene. Finally, all the Gaussian distributions were normalised to the range 0–1 to obtain the synthetic saliency map.

As this process of generating synthetic saliency maps involves random sampling, individual instances of saliency maps will be different.

Figure 6.2 shows an example of such a saliency map used in our experiments. Notice the peak saliency value is 1 and there are 10 distinct peaks.

For the final testing of the utility function on videos, each of the five chosen videos from the Coutrot video database 1 [38] were down sampled into a 64×64 video. The agent operates on the down-sampled video. each pixel of the video is modelled with an independent Kalman filter and the chosen utility function makes a decision to look at a particular pixel based on the internal states of the Kalman filter.

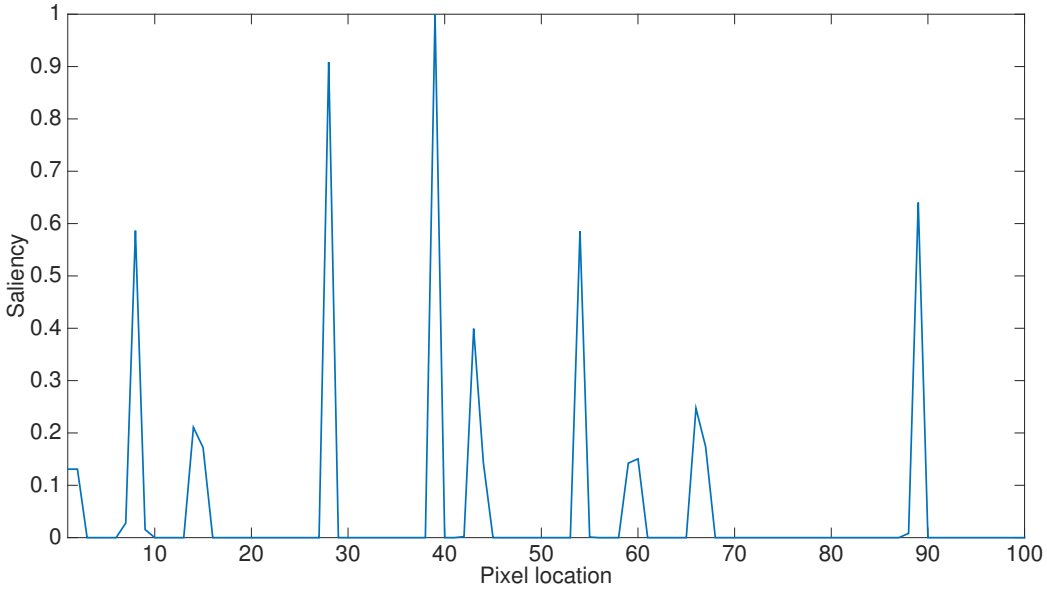


Figure 6.2: A saliency map: There are 10 distinct peaks which stretch from 1 to 100 in a horizontal scale (the visual scene). The height of the maximum peak is 1. The distortion in shape is due to sampling.

6.3 Utility Function 1: Targeted Point Uncertainty Reduction

A useful observation strategy to keep an agent updated about the surrounding world is to look at parts of the visual scene that the agent is uncertain about. An agent operating on this principle would give maximum importance to a belief state that has the highest uncertainty and the corresponding visual location would have the highest utility.

At every time instant, the agent finds the location of the visual scene with highest internal uncertainty by a simple maximum finding operation. Then that location is chosen as the next saccade target.

Hence the utility function should find the internal state with the highest uncertainty.

$$u_{max} = \max(\sigma_i^2) \tag{6.3}$$

where σ_i^2 is the internal uncertainty of the agent's i^{th} belief state and u_{max} is the corresponding utility and each of the σ_i^2 comes from the corresponding element in the $\boldsymbol{P}$ matrix. The visual location corresponding to the belief state with the highest uncertainty becomes the target location for the next time step. The remainder of the visual scene is observed with increasing measurement noise from this centre.

Equation 3.3b is the guiding principle for how the internal uncertainty of a belief state changes over time. In that equation, $\boldsymbol{P}_{t+1|t}$ is the uncertainty associated with a prediction and $\boldsymbol{P}_{t|t}$ is the current level of uncertainty. While taking action to reduce internal uncertainty, the uncertainty growth rate $\boldsymbol{Q}$ dictates how often any given region is re-observed. Large values of $\boldsymbol{Q}$ results in uncertainty growing more quickly hence a previously observed location offers higher utility compared to other regions. This forces the agent to re-observe that location. Hence $\boldsymbol{Q}$ determines the time gap between two subsequent observations and the $\boldsymbol{R}$ determines the improvement after observation.

To simplify initial experiments with this utility function, synthetic saliency maps were used rather than saliency maps derived from input videos. A simple 100 pixel visual scene was used with a foveation profile having $\rho = 30$.

6.3.1 Results

Figure 6.3 shows a plot of the utility versus the target saccade locations. The vertical axis shows the utility of a location and the horizontal axis shows pixel location. As the internal uncertainty is used as the utility, the vertical axis also shows the internal uncertainty of the agent. Notice that the highest utility is offered by the pixel with the highest uncertainty.

Pixel 28 offers the highest utility in figure 6.3. The highest value is marked with a red circle.

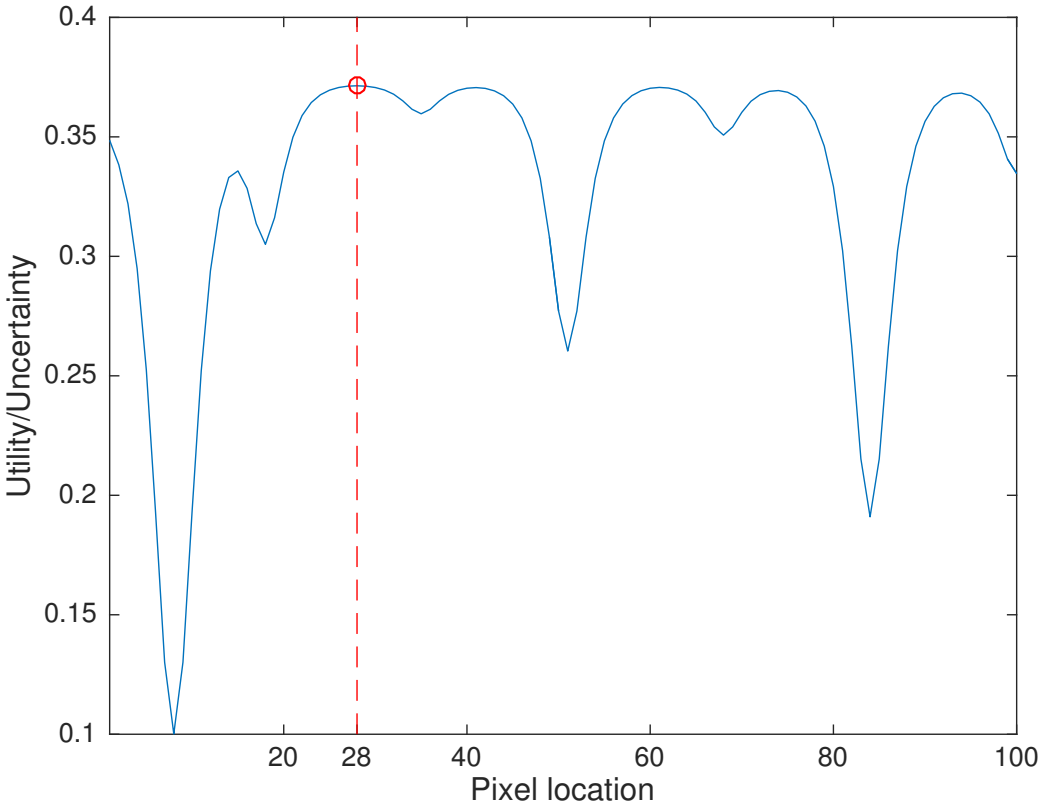


Figure 6.3: Utility of looking at 100 candidate saccade locations. The utility used is the internal uncertainty of the agent.

When applied to real-life videos it is expected that locations with the highest process noise will gain most of the attention.

Figure 6.4 shows the outputs of this utility function applied to the five videos. The MLE based variance estimator was used for learning process noise from observations. A learning window of 30 time steps was chosen, which means that the process noise matrix was updated at every 30 time step. The effect of learning window length on the MLE estimator was discussed earlier in chapter 5.

The outputs presented in this figure 6.4 are after the first 30 time steps.

The left column of the figure shows the learnt process noise image of the video. Each pixel in the process noise image shows the process noise of a corresponding pixel in the down-sampled video.

The middle column of the figure shows the histogram plot. This plot has two separate graphs superimposed. The axis on the left shows the frequency of visit. The height of the blue coloured bar plots shows frequency of a number of visits versus the process noise shown on the horizontal axis. Notice that the pixels with the highest process noise get observed the most.

On the same histogram plot, the histogram of pixels versus their process noise is plotted. This plot uses the right-hand axis and is shown in red colour. Notice that the pixel count is high towards the low process noise, suggesting that most pixels have low processing noise. This can be intuitively understood as the majority of a visual scene is made of stationary objects like wall or furniture.

The right hand column shows, the attention distribution of the agent superimposed on the first frame of the corresponding video. Notice that the pixels with high process noise, shown as white in the process noise image, got the most visual attention. A direct fixation means that the foveation profile centre was directly placed on that pixel. A correlation can be seen between the process noise image and the visual locations that got the highest visual attention.

Figure 6.5 shows the output of this utility function with the recursive method as the process noise learning mechanism. The agent operated on the entire video using the recursive process noise learner. Hence the variance images presented in the first column is the output of the learner at the end of the entire length of the corresponding video.

As the recursive learner is adaptive, the process noise matrix has different process noise associated with each pixel at different times. Hence the histogram, which is computed at the end of the entire run is not an accurate description of which process noise got the most attention. Although

the presented histograms show an average result of attention distribution.

Differences in the variance image between the MLE and the recursive are noticeable. Although the visual locations with the highest uncertainty are almost the same between the videos, the recursively estimated process noise image shows an average of process noise over the length of the video.

A one to one correspondence between the locations with high process noise can be found with the locations that gained the highest visual attention.

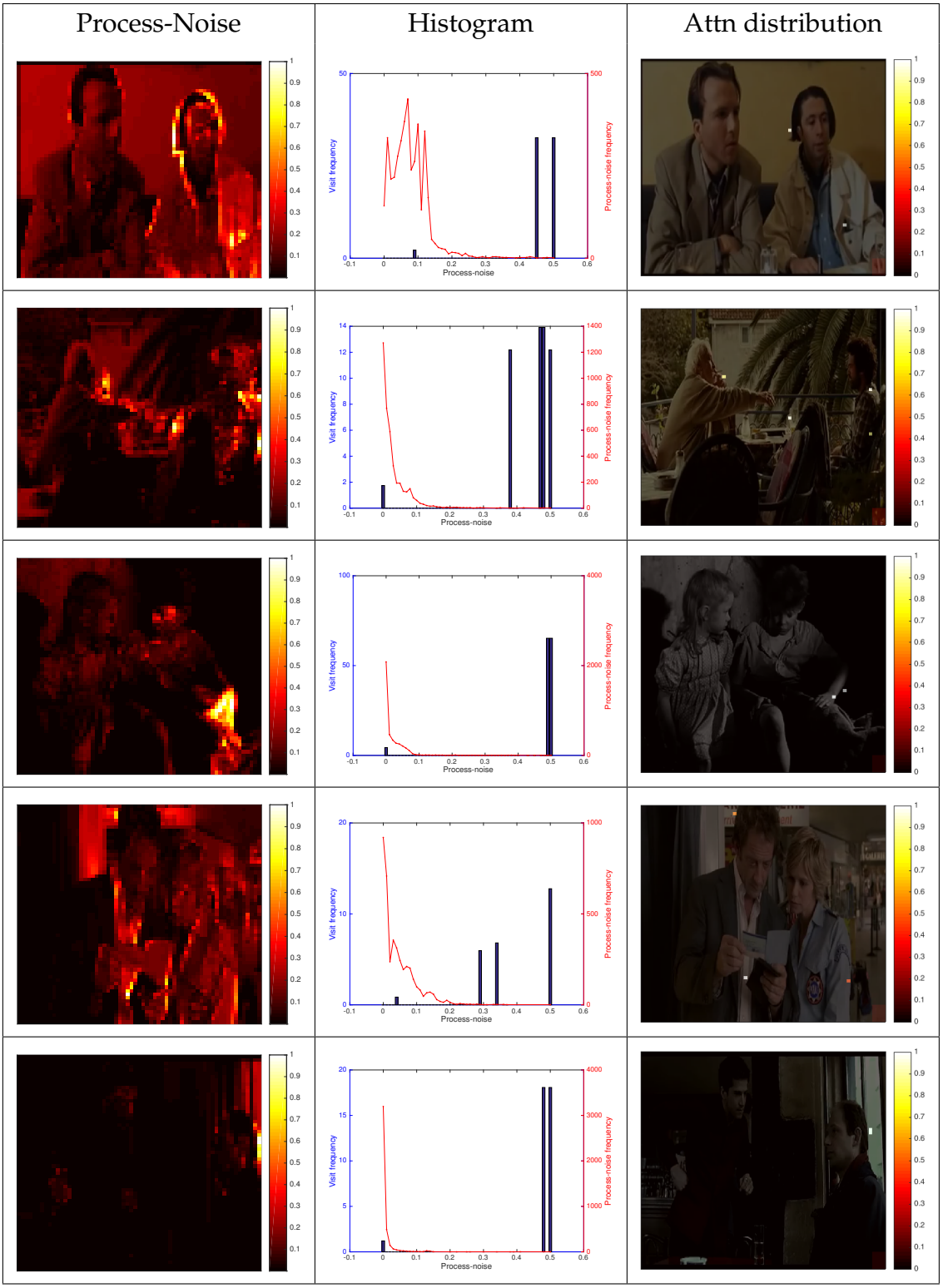


Figure 6.4: Estimated process noise using MLE of all the five videos, and the corresponding visual attention distribution. The left, middle and right column shows the process noise image, histogram plot and the visual attention distribution respectively. Notice that the variance image was normalised between 0 1. Notice in the colorbar, the highest varying pixel has been coded white.

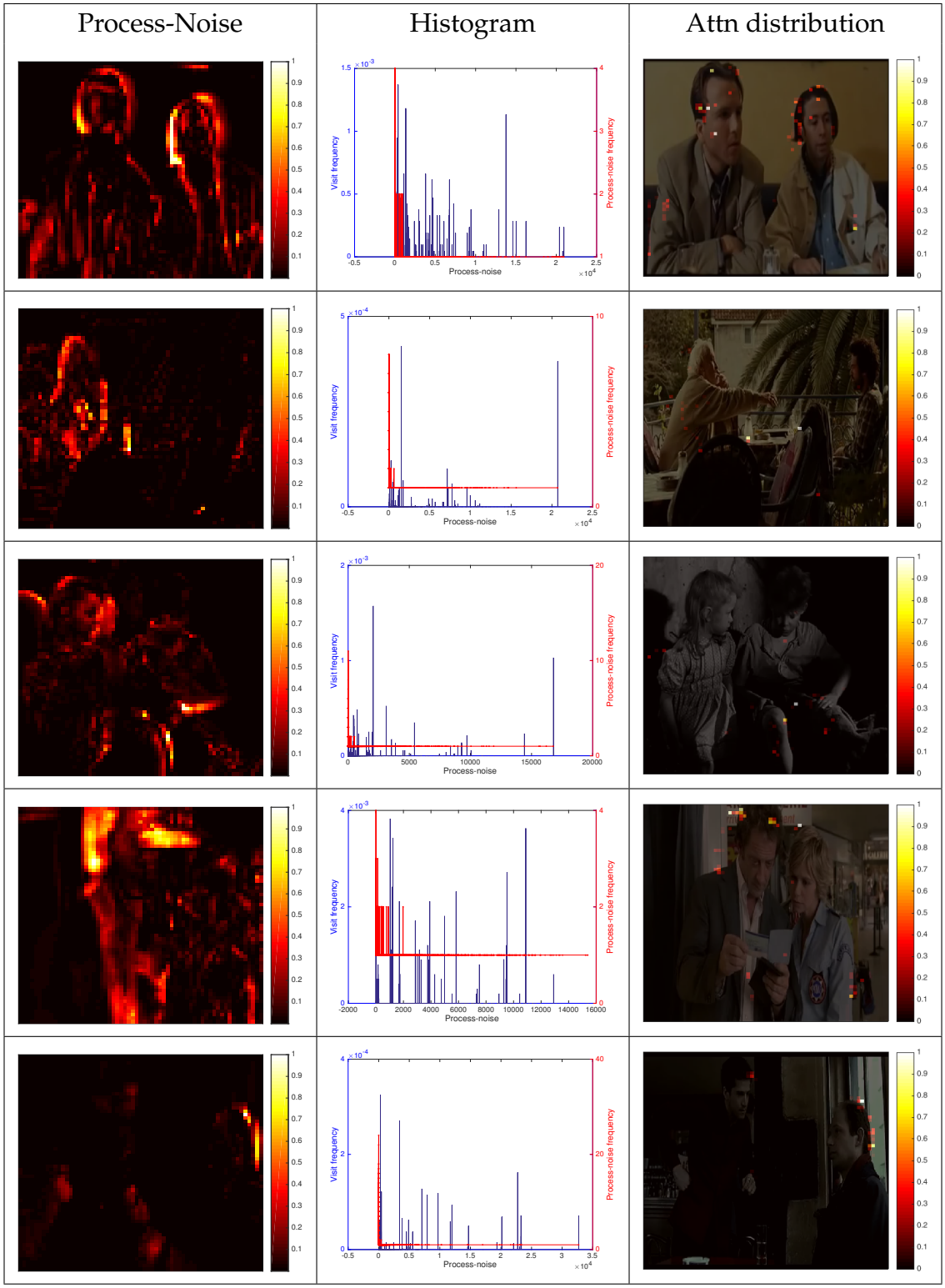


Figure 6.5: Estimated process noise using the recursive estimator of all five videos, and the corresponding visual attention distribution. The left, middle and right column shows the process noise image, histogram plot and the visual attention distribution respectively. Notice that the variance image was normalised to have values between 0 1. The colorbar shows that the highest varying pixel is displayed in white color.

6.4 Utility Function 2: Total Uncertainty Reduction

Another desired agent behaviour is to take measurements to reduce the average internal uncertainty over all the belief states. In this, the agent considers the sum of reduction in uncertainty over all the locations of the visual scene. In contrast, the previous section 6.3 considered the reduction in uncertainty associated with only the targeted location of the scene.

At every time instant, the agent simulates placing the foveation profile centre all the possible locations. At every location of the visual scene, the agent computes the total uncertainty reduction over the entire scene. And finally looks at the visual location that offers maximum reduction in total uncertainty. This would result in an optimal attention distribution in the sense that it would keep the agent optimally updated about the whole visual world.

Such a system should maximise a utility function of the form

$$u_{avg} = \frac{1}{n}\sum_{i=1}^{n} u_i \tag{6.4}$$

$$= \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\sigma_i^2} \tag{6.5}$$

where the u_i is the utility measure ($1/uncertainty$ in this case) of the individual regions of interest (ROI), u_{avg} is the average utility of considering one action and n is the total number of regions that could be observed.

A visual scene with n regions of interest has n average utility measures one for each potential future fixation point. All the average utility measures were arranged into a vector $\boldsymbol{u} \in \mathbb{R}^n$. Each element in this vector is the average utility of fixating at the corresponding region.

An action a is the act of looking at a target region in the visual scene with a given foveation profile. Looking at a region means placing the profile-centre of the foveation profile on a target region. The visual region

which is observed with the lowest measurement noise results in maximum reduction in the corresponding uncertainty associated with its belief state. The nearby regions are observed with progressively higher measurement noise.

Hence the problem of selecting the next saccade target can be expressed as a maximisation problem shown below:

$$a^{\star} = \arg\max_{a} f(a) \tag{6.6}$$

where a is the anticipated future action of choosing a point of fixation and $a^{\star}$ is the optimal action that provides maximum utility.

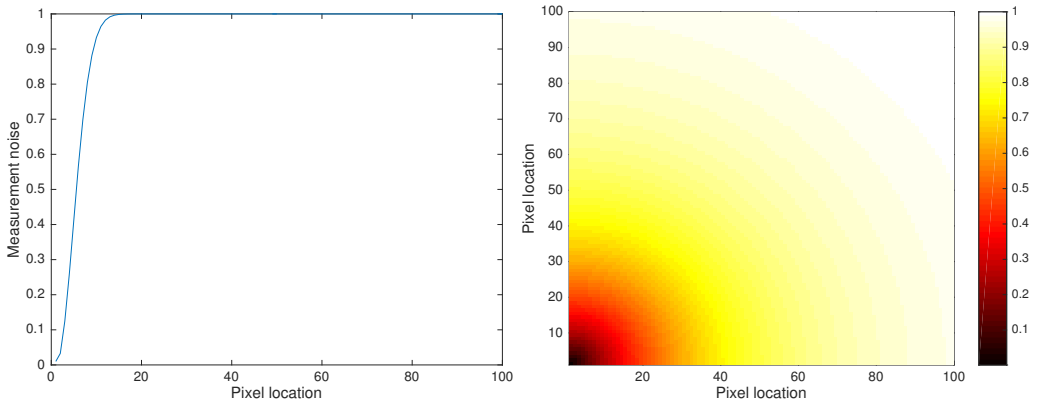
At each time step, the Kalman filter prediction is used to estimate the world one-step-ahead and then assume placing the centre of the foveation profile at target fixation points. With each placement the variance of state estimates is considered for all the possible future measurements is calculated. For a given scene of n regions of interests, there are n means and n uncertainties. Potential future actions are compared against each other based on a given utility function, and the observation location having the highest utility is selected.

Figure 6.6 shows plots of foveation profiles that are looking at the first pixel in the scene. Notice that portion of the foveation profile that is outside the scene does not contribute to reduction in uncertainty. Hence the visual locations towards the edge offer low utility.

Results

Initially, this utility function was used to distribute visual attention in a small visual scene with 100 pixel. Figure 6.7 shows the utility of each pixel.

This figure plots two graphs, the utility and the internal uncertainty, each plotted with its own vertical axis. The left vertical axis shows the utility of the observation of a target saccade location and the right vertical axis shows the internal uncertainty.



(a) (b)

Figure 6.6: Plots of one dimensional and two dimensional foveation profiles. Each of the profiles has $\rho^2 = 30$. The fixation points for the foveation profiles shown are in the middle of scene. For the left plot, the vertical axis shows the measurement noise, whereas the measurement noise is colour coded for the $2D$ foveation profile plot. Notice that the measurement noise increases symmetrically with distance from the profile centre.

Notice that this utility function differs from the earlier one, in that the utility of looking at a pixel does not directly reflect the internal uncertainty. Figure 6.7 shows that pixel number 45 which is towards the centre of the scene has the highest utility. The highest utility value is marked with a red circle in the utility plot. The corresponding state uncertainty is also marked with a red circle. Clearly, the utility is not highest at the locations where state uncertainty is the highest.

The internal uncertainty is high towards the edges of the scene but the utility is very low at the edges. This behaviour is due to the averaging of reduction in uncertainty. As this utility computes the overall reduction in uncertainty instead of the reduction in uncertainty of the targeted pixel, the edges of a scene provide the lowest utility. In other words, if the agent looked at a pixel which is at the edge of a visual scene, half of the area of the foveation profile that is beyond the edge does not contribute to uncer-

tainty reduction.

The maximum total uncertainty reduction is offered by the pixels that are more central in the scene. Hence, the natural tendency of an agent running this utility function is to look away from the edges and towards the centre of the scene.

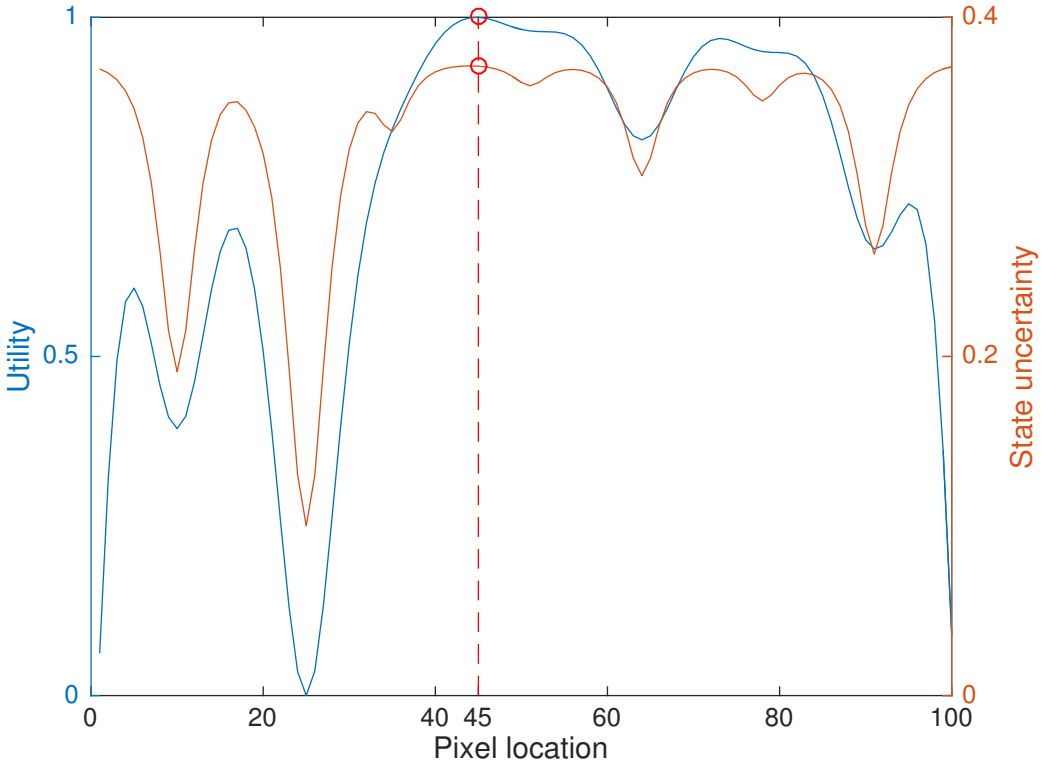


Figure 6.7: Utility of looking at 100 candidate saccade locations. The utility of each target location is the sum of reduction in uncertainty of all locations. Notice that the visual location offering the highest utility is not the same as the visual location with the highest internal uncertainty.

If there is a pixel with high uncertainty, the agent's attention would be shifted towards that pixel but unlike the previous utility function, the agent would not directly look at the pixel. Instead, it is expected that the agent would tend to look at a nearby region that offers the highest total

reduction in uncertainty. Also, it is expected that an agent would have a bias towards looking at the centre of the visual scene when this utility function is used.

Figure 6.8 shows the process noise image, histogram and the attention distribution plot generated while using this utility function. The left, middle and the right column shows the process noise image, histogram and the visual attention distribution respectively. Like the earlier utility function, the MLE learner was run for the first learning window.

Notice that a bias towards the centre of the visual scene is present for all the videos.

The agent did not look directly at the pixel with the highest uncertainty. Clusters of attended locations can be found at the centre of the scene.

Compared to the previous utility function, notice that histogram plot shows that the most attended locations do not have the highest process-noise.

Figure 6.9 shows the output of the same utility function with recursive process noise learning. As the recursive learner was run for the entire length of the video, there are more counts for the visual attention distribution. Therefore the central bias is more prominent in the attention distribution images. The histograms for the recursive learner is an average over the length of the video hence they differ from the MLE version.

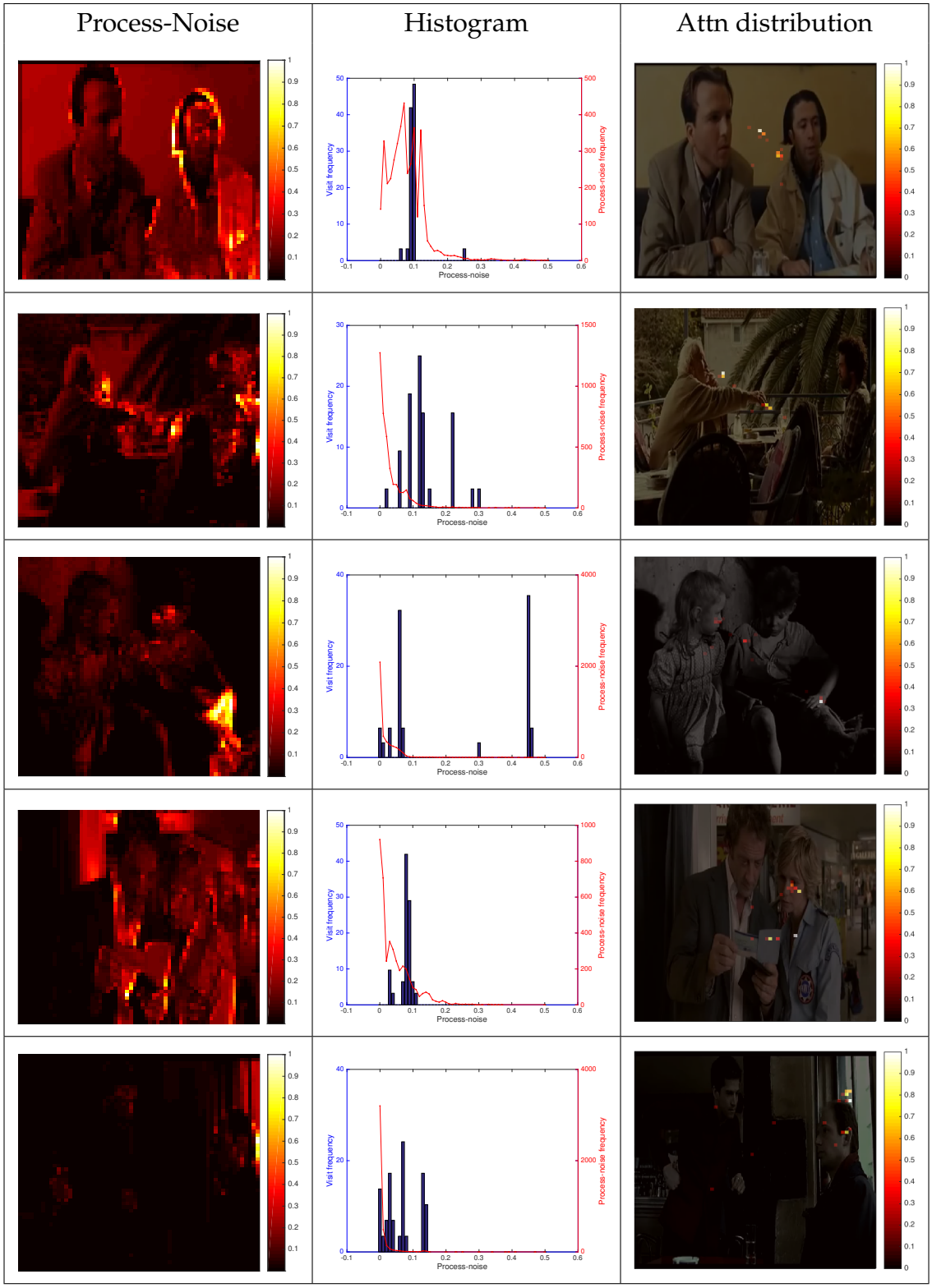


Figure 6.8: Estimated process noise using MLE of all the five videos, and the corresponding visual attention distribution. The left, middle and right column shows the process noise image, histogram plot and the visual attention distribution respectively.

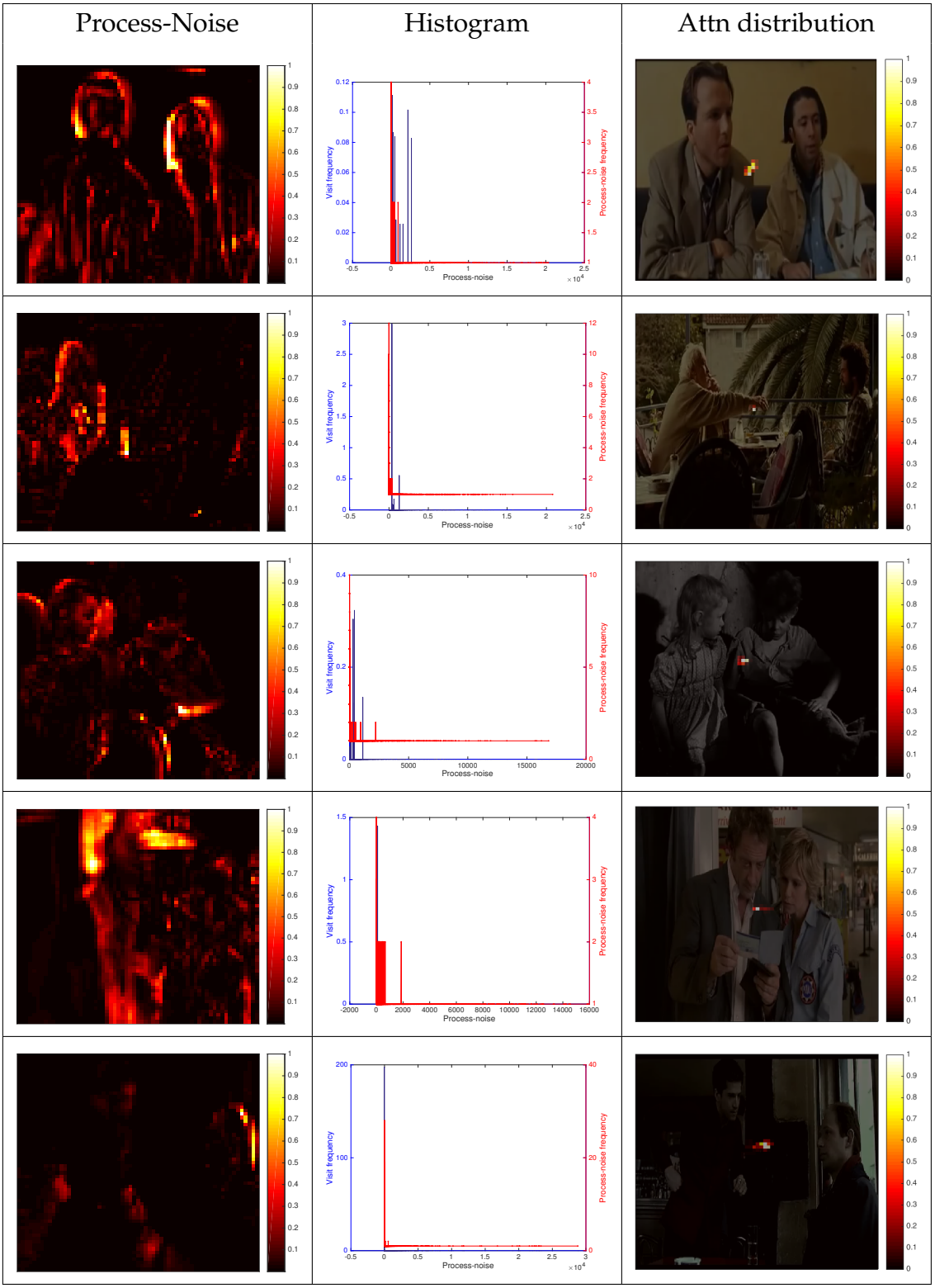


Figure 6.9: Estimated process noise using the recursive estimator of all the five videos, and the corresponding visual attention distribution. The left, middle and right column shows the process noise image, histogram plot and the visual attention distribution respectively.

This utility function is computationally demanding as it involves the process of simulating profile-centre placement at every pixel in the scene. Also, looking at the centre of a visual is a sensible way to reduce the most amount of total uncertainty in one time step. Hence central bias results in gaining the most amount of information at a glance.

The problem with central bias is that the agent's internal confidence about visual areas at the periphery is low. In case the uncertainty associated with the peripheral regions grow really big, the agent's attention would show shifts away from the central area and towards the edges of the scene.

The aim of the last two utility functions discussed is to reduce internal uncertainty (either of individual states or the average of all the states). Hence an agent operating solely based on reduction of uncertainty would distribute most of its visual attention to unpredictable sections of a visual scene.

These two utility functions are useful in scenarios where an agent must keep itself updated about the most unpredictable elements in the scene. In case there are too many unpredictable regions, the agent might not be able to directly attend them all and it will mostly ignore the predictable regions.

In contrast in a real life scenario the unpredictable visual regions are often not important and are ignored. For example, human observers will not look at TV snow. The following section presents a strategy for visual scene sampling that avoids unpredictable sections of the visual scene.

6.5 Utility Function 3: Avoiding Unpredictable Regions

Utilities presented in sections 6.3 and 6.4 get fixated on noise. Therefore in environments where there is a noise source present (a detuned TV set), the utility requires modification.

A deeper insight into the uncertainty based sampling strategy reveals that although the instantaneous states of unpredictable elements offer knowledge gain (instantaneous reduction in uncertainty), the newly gained knowledge is not usable (cannot be used to predict the world) for long. As an example, observation of an unpredictable state like the white noise of TV snow produces zero future usability of the newly gained knowledge.

In case of a predictable state, one observation is usable for a long period of time. For example, if a predictable state is observed, the knowledge gained about the state is good enough for the agent for a long time before the state needs to be observed again.

There is an inherent link between predictability, time difference between two subsequent observations of the same state, knowledge gained from one observation and the usability of the observation. For the same amount of time difference, an unpredictable region offers more gain in knowledge but less usability compared to the predictable state. Whereas one observation of a predictable state offers more usability than the unpredictable state. A visual sampling strategy that considers the usability of an observation can be used to avoid unpredictable elements of a scene.

The usability of an observation can be computed given the process noise matrix ($\boldsymbol{Q}$) using the Kalman filter equations.

Usually, elements of the visual scene are observed with different measurement noise depending on how far it is located from the location choice of the profile-centre for that time instant. The worst case of observing a visual scene element is when it is always observed with the highest measurement noise of the foveation profile. Such a state would have high

internal uncertainty. Also as this state has been observed with a constant measurement noise, it's internal uncertainty would reach a steady state value. This steady state of the internal uncertainty indicates the agent's maximum level of possible uncertainty regarding a world state which has only been observed with the highest measurement noise.

The worst case internal uncertainty specific to the belief state and can be computed using steady state Kalman filter equations given by the following:

$$\boldsymbol{P}_{ij}^{s} = \frac{-\boldsymbol{Q}_{ij} + \sqrt{\boldsymbol{Q}_{ij}^{2} + 4\boldsymbol{Q}_{ij}\boldsymbol{R}_{ij}}}{2} \tag{6.7}$$

where $\boldsymbol{P}_{ij}^{s}$ is the ij^{th} element of the steady state error matrix, and $\boldsymbol{Q}_{ij}$ and $\boldsymbol{R}_{ij}$ are ij^{th} element of the process noise and the measurement noise matrices respectively.

In reality, an agent's internal uncertainty would be less than the worst case scenario as visual states are typically not constantly observed with the highest measurement noise. As the measurement noise changes at each time step, a steady state uncertainty for the internal belief state cannot be computed analytically. Hence the worst case uncertainty can be used as a baseline to measure how much a new measurement improves on the internal uncertainty. If the new measurement is made using the highest measurement noise, there is no improvement as the internal uncertainty is already at a steady state.

Any observation of that state made with a lower measurement noise would reduce the internal uncertainty. The amount of reduction in internal uncertainty from the worst case steady state uncertainty is the gain in knowledge due to the observation. Hence it is the usability of taking that observation.

Figure 6.10 shows plots of how the internal uncertainties of three belief states increase over time. The plot shows change in uncertainty from a low state to steady state of high (yellow colour), medium (brown colour) and low (blue colour) predictability states. Also, notice that each of the

states reaches a steady state value and that the steady state uncertainty is different for each of the states. It is the highest for the unpredictable region and the lowest for the predictable region.

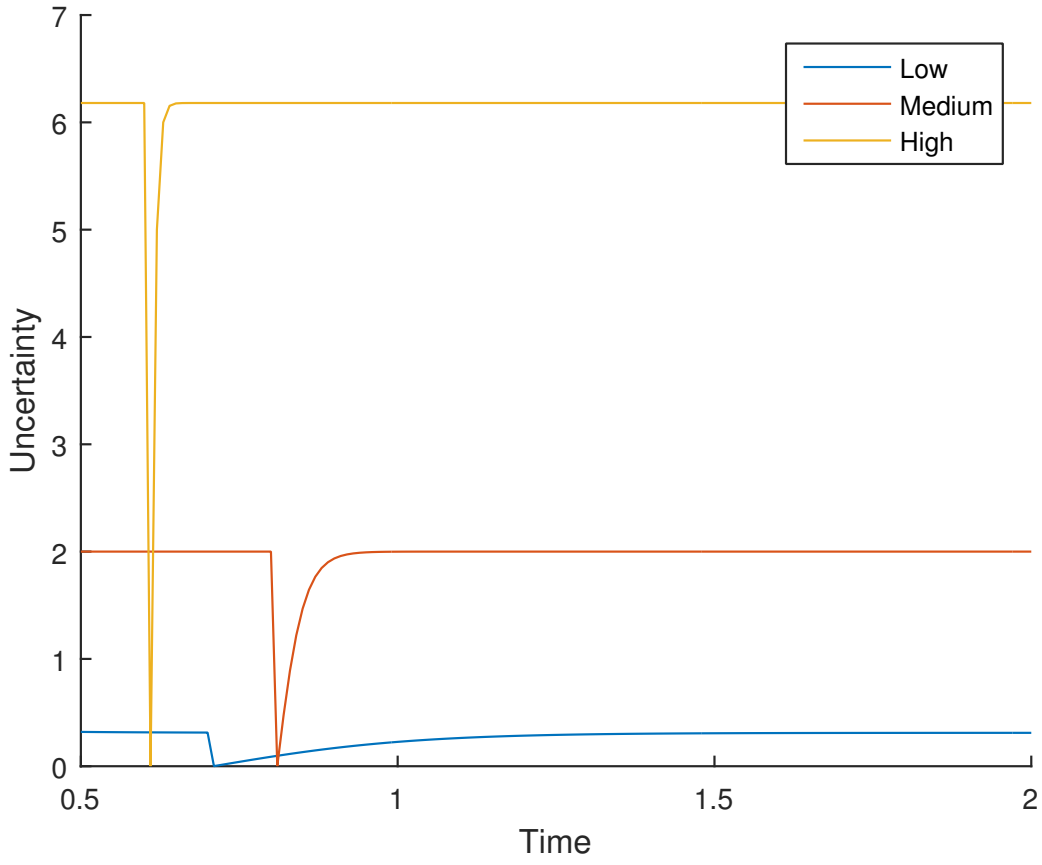


Figure 6.10: Plot of how the internal uncertainties of three states increase over time when all the states are observed with a constant measurement noise. The vertical axis represents uncertainty and the horizontal axis represents time. The three lines show change in uncertainty for high (yellow colour), medium (brown colour) and low (blue colour) predictability states respectively. Notice that the internal uncertainties for all three states reach steady state values.

Notice in figure 6.10 that each of the three states was observed once

directly (at different times). The uncertainty reaches its minimum immediately after a state is observed and it grows to the steady state afterwards. The reduction in uncertainty from the steady state until it reaches back to the steady state is due to the observation. In this work, the usability of the observation is defined as the area under the curve between the time instances of observing the state and the uncertainty returning to its steady state.

Figure 6.11 shows the uncertainty of the same three internal belief states discussed above. The area shaded in blue is the usability of that observation.

In reality, the internal state of uncertainty might be lower than the worst case steady state. Hence a new observation does not reduce the uncertainty from the worst case, instead, it will reduce it from the current state of uncertainty of the internal state.

Also, the same state may be observed again before its internal uncertainty grows back to the worst level. As it is not possible to know when that state will be observed again, the state's future uncertainty level is not known in advance and the exact uncertainty area between two observations cannot be calculated. In contrast, the worst case uncertainty can be theoretically computed.

The total gain in knowledge (or reduction in uncertainty) from the current state of uncertainty up to the worst case uncertainty is the best case usability and that will be used as the utility function. Hence, overall usability of an observation is the area under the curve between the time instances of observation and uncertainty growing back to worst case steady state.

For a given state (process noise is known), the current state of uncertainty and the process noise of the state decides the usability of a new observation. For an unpredictable state, the usability is low as a new measurement cannot reduce uncertainty for a long period of time. In comparison, for a predictable state, the very first measurement has high usability

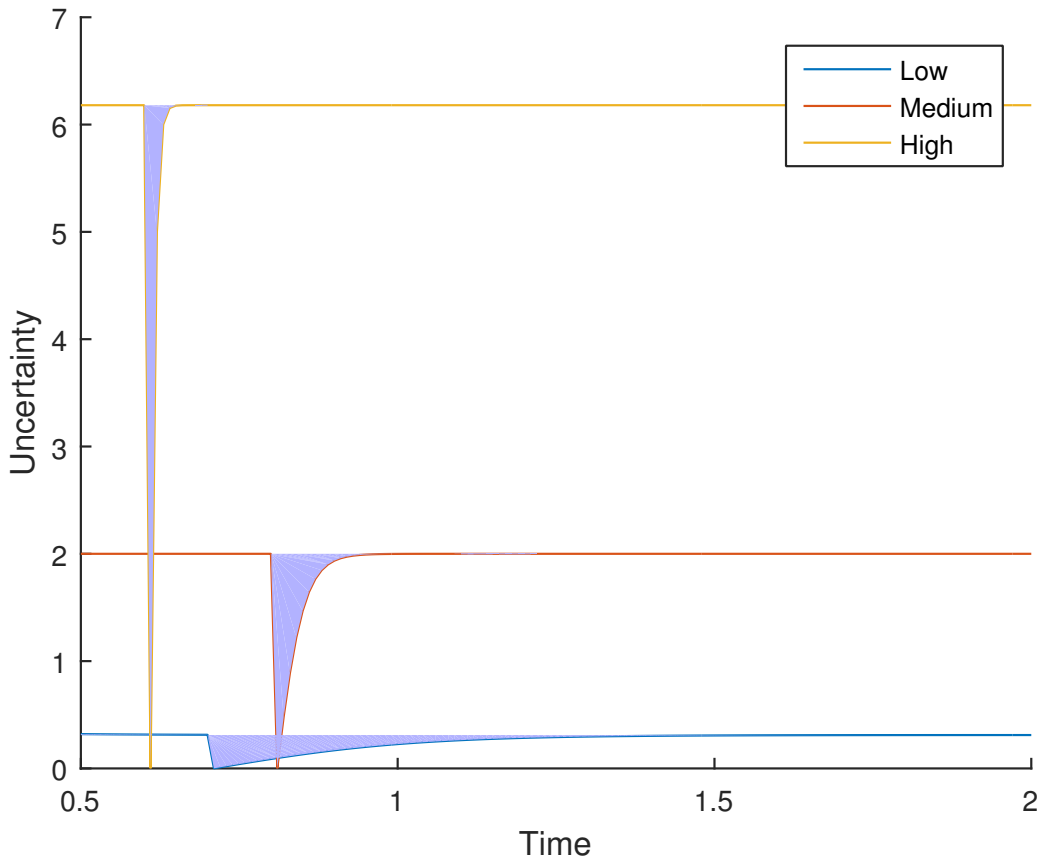


Figure 6.11: The area shaded in blue shows the usability of an observation. The vertical axis represents uncertainty and the horizontal axis represents time.

but the immediate measurements after the first one do not have much usability as the uncertainty is already low. For example, if a predictable state has just been observed, a re-observation of the same state in the following time step will not produce high amount of usability as there is not enough improvement to make over the last observation. Over time the uncertainty of the predictable state will grow high and the usability of observing that location rises again. Hence usability of an observation is a function of the time gap between two observations of a state.

The usability of an observation cannot be analytically computed as the uncertainty of internal states at the moment of a potential future observation are not known beforehand. Hence in this work, it is computed numerically using trapezoidal integration method from the instant of observation until the internal uncertainty reaches the worst case steady state value.

Similarity Between 'Usability of Observation' and 'Uncertainty Reduction'

Usability of observation discussed in this section is similar to the reduction in uncertainty approach discussed earlier in a sense that usability is an extended version of uncertainty. Reduction in uncertainty measures the instantaneous gain in knowledge, whereas the usability of one observation over time measures the instantaneous gain in knowledge plus how long that knowledge is retained by the agent.

From this point of view usability of an observation is a time integral (or area under the curve) with the lower limit as the time instant when the observation is taken and the higher limit as the time instant when the internal state reaches the worst case steady state uncertainty. If the higher limit of the integration is imagined to be one time step, then the usability of an observation is the reduction in uncertainty discussed in earlier two sections.

Results

Figure 6.12 shows plots of usability of observing low, medium or high predictability pixels. The left and the right columns show usability plots with $\Delta t = 5$ and $\Delta t = 200$ respectively. The first row shows the usability as a blue shaded area and the second row shows bar plots of usability.

A short time gap (Δt) between two observations of a predictable state offers low usability, comparatively higher usability for the medium pre-

dictability and low usability for the unpredictable state. A longer time gap between observations will offer the highest usability of observing a predictable state.

Notice on the left top panel ($\Delta t = 5$) the area shaded in blue is the smallest for the predictable region and highest for the medium predictable region. On the right top panel ($\Delta t = 200$) the shaded area is highest for the predictable region. Hence an agent running on this utility function will ignore the unpredictable elements of the visual scene and will observe the predictable regions only when their uncertainties are high. Hence a predictable region, for example, a wall, will be re-observed only after a long period without any observation, and the unpredictable region, for example, TV noise, will never be observed.

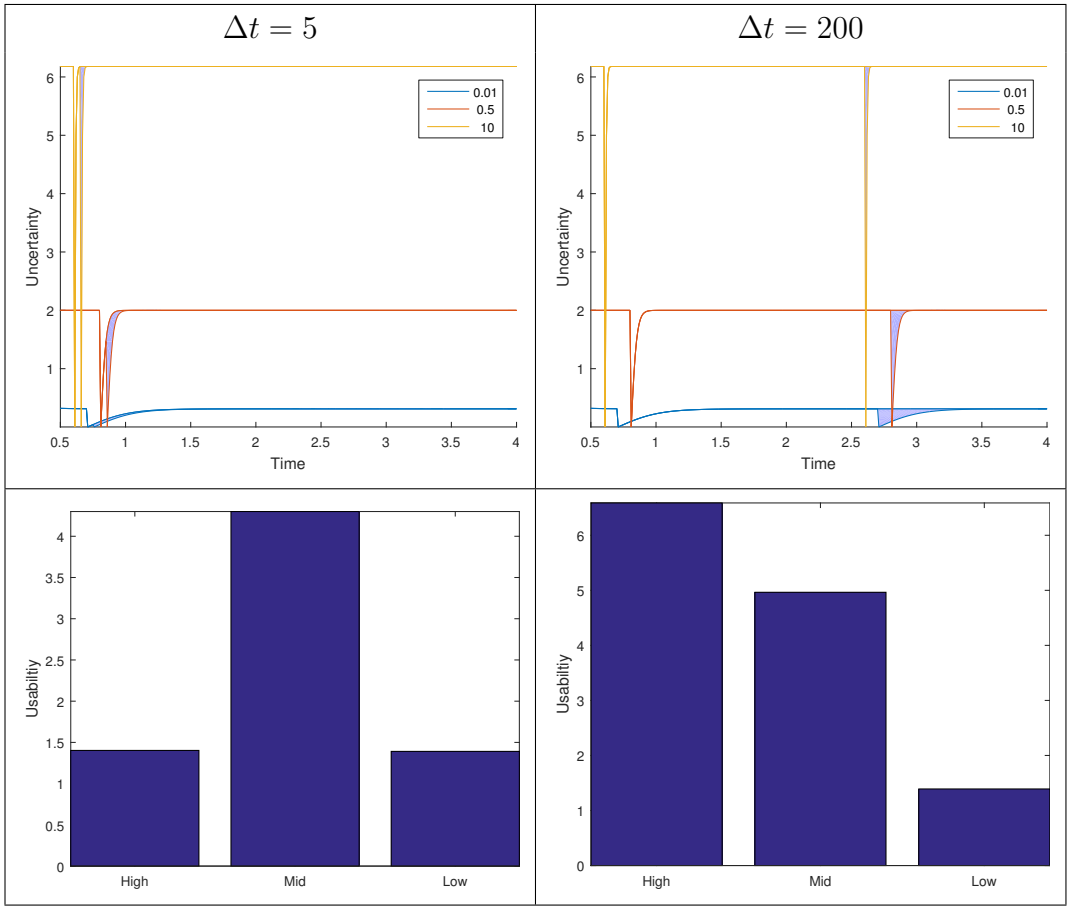


Figure 6.12: Plots of usability of an observation. The left column shows plots of usability with a short time gap ($\Delta t = 5$) and the right column shows the usability with a longer time gap ($\Delta t = 200$). The top row shows usability as blue shaded area and the bottom row shows a bar plot for visual comparison of usabilities. Notice that medium predictability regions offer more usability in short time gaps and the high predictability regions offer high usability only after a long time gap.

It is seen that the usability changes with the time gap between two subsequent observations. A summary of how usability changes with time gap is shown in figure 6.13. The vertical axis of the plot shows usability and the horizontal axis shows time gap between two observations.

Notice that at Δt close to zero, medium predictable regions gain importance whereas the unpredictable region gains higher importance as the time gap increases and the unpredictable regions are not observed.

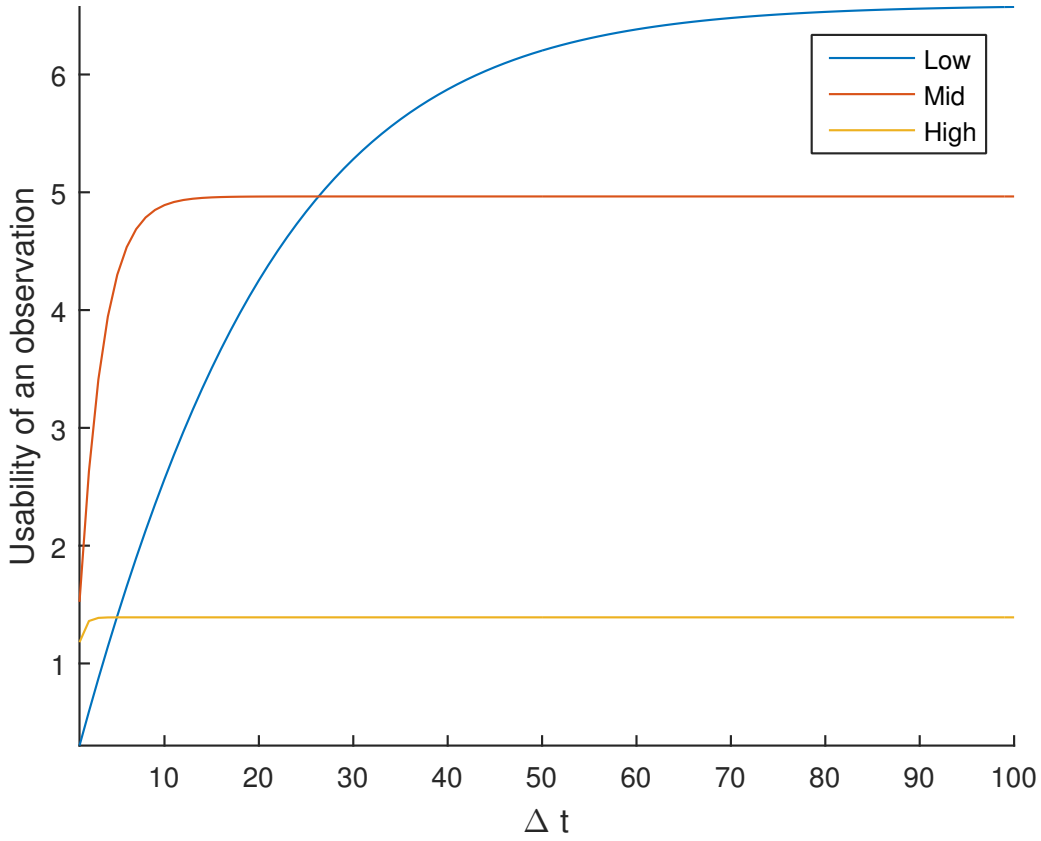


Figure 6.13: Plot of usability of an observation with change in Δt. The horizontal axis shows time gap between two observations and the vertical axis shows usability. Usability of predictable visual regions is very low at small time gaps and usability of medium region is highest. Whereas the usability of predictable regions is the highest at longer time gaps.

Figure 6.14 shows a plot of the usability of an observation versus the process noise of the corresponding element of the visual scene with $\Delta t = 5$. The horizontal axis shows the process noise and the vertical axis shows usability. Notice that the usability grows with increase in process noise

until a maximum after which the usability drops. This indicates that an agent using this utility function will avoid both the unpredictable and the very predictable regions of the visual scene.

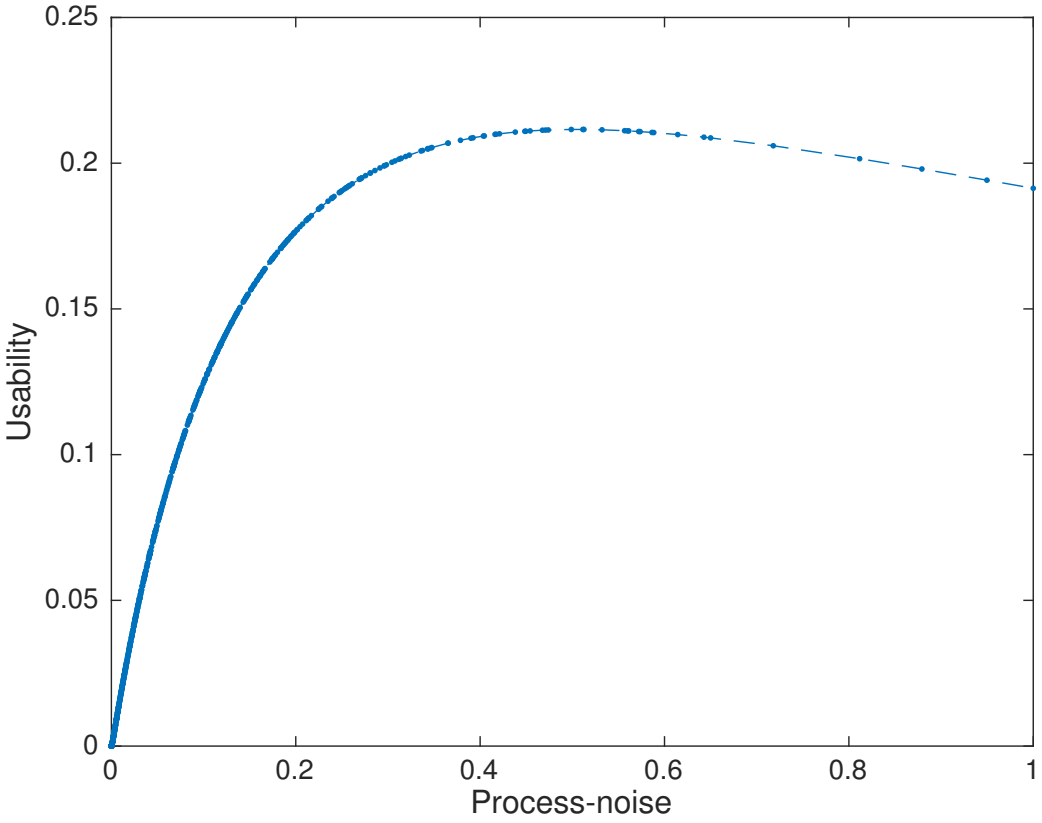


Figure 6.14: Plot of usability of an observation versus the process noise.

Figure 6.15 shows the process noise image, histogram and attention distribution of running this utility on the five videos from the Coutrot video database 1 [38]. Notice that the agent ignores regions with high process noise. Visual locations shown in white colour (high process noise) and black colour (low process noise) in the process noise image are ignored. The histogram plot shows that most of the visual attention was given to visual locations with process noise that is in between the high and the low process noise.

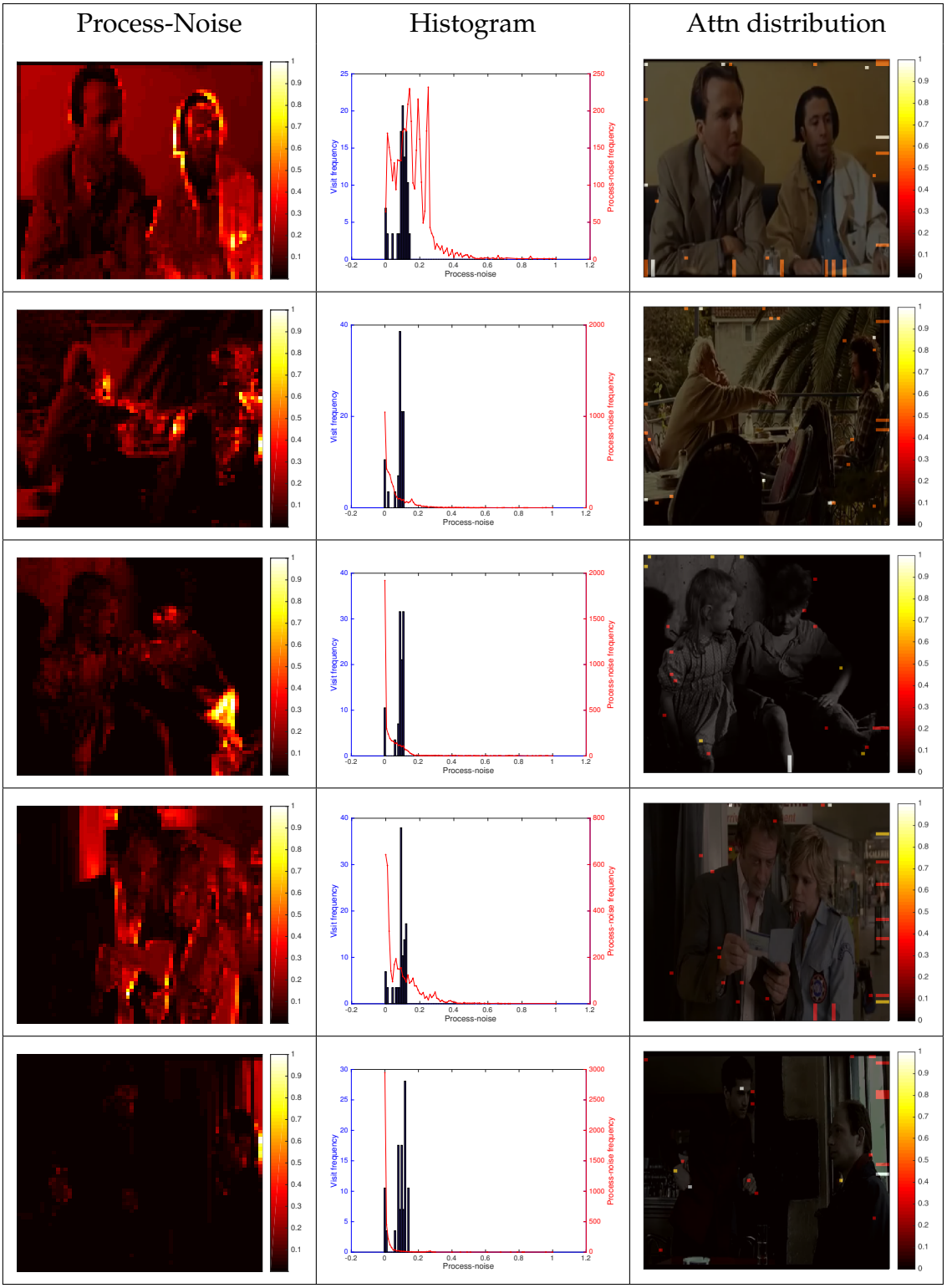


Figure 6.15: Estimated process noise using MLE on all five videos, and the corresponding visual attention distribution. The left, middle and right column shows the process noise image, histogram plot and the visual attention distribution respectively.

Figure 6.16 shows the outputs of the agent's behaviour while learning the elements of the $\boldsymbol{Q}$ matrix using the recursive estimation algorithm. As discussed in the earlier sections (section: 6.4, 6.3), the histogram presents an overall average of visual attention distribution. As expected, it can be seen that the agent ignores the high and low process noise regions of the scene.

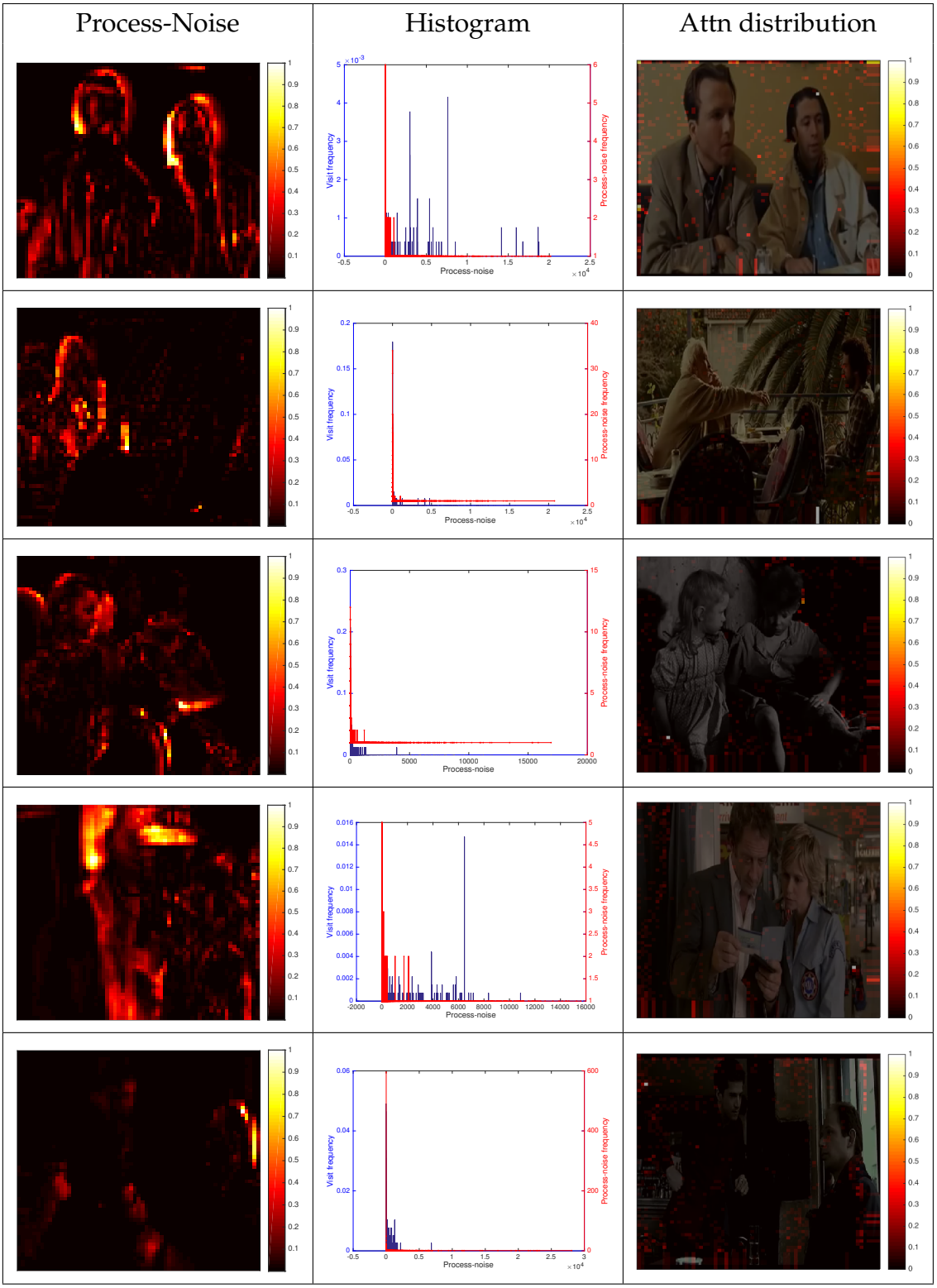


Figure 6.16: Estimated process noise using recursive estimator on all five videos, and the corresponding visual attention distribution. The left, middle and right column shows the process noise image, histogram plot and the visual attention distribution respectively.

6.6 Improved Ability to Predict Human Fixations

The aim of this set of experiments is to compare the proposed model with the baseline Itti model [90, 93] in their ability to predict human fixation. The comparison in the ability to predict human fixations was tested using the CRCNS eye-1 eye-tracking 'original' video dataset [94]. This dataset records eight human volunteers' eye fixations data while they freely viewed 50 complex video stimuli (TV programs, outdoors videos, video games rooftop bar etc.). In this set of experiments, the uncertainty reduction utility function is used as it resembles the free viewing for which the human fixation data is available.

The Itti model was chosen to be compared with for two reasons: a) This seminal work is well known and has been applied to videos previously [113, 129]. b) The implementation of the model is publicly available.

Grayscale saliency maps are first obtained for each individual frame of each video using the L.Itt model using the implementation proposed by [191]. The magnitudes of the entries in these maps denote how much attention each pixel is predicted to attract in the frame. Further, these saliency maps were used as input for our proposed algorithm with the uncertainty reduction utility function.

Each model was evaluated using two different metrics for accuracy and the evaluation scores were compared.

An accuracy score is determined for each frame and averaged across all frames to provide a measure of performance. For each sequence, the scores of all methods are compared and analysed statistically to determine if there is a clear winner for that sequence.

Different measurement metrics have previously been proposed [28] in the past where saliency evaluation metrics have been adopted from the field of information theory and signal detection and also including a crowd-sourced perceptual experiment where human participants ranked saliency models based on how similar an estimated a saliency map is to

the ground-truth saliency map [112]. There is substantial disagreement between researchers regarding which metric to use to compare saliency maps. No metric captures all the aspects of saliency map comparison, for example, the difference in information between two saliency maps, accuracy in predicting human scanpath, accuracy in predicting human fixation etc. Objectively determining which model offers the best approximation to human eye fixations still remains a challenge.

A detailed study by Zoya et al. suggests [28] to use information gain (IG) as a metric to compare probabilistic saliency models. This applies for the proposed model due to its probabilistic nature. A visual location scored higher by the information gain metric does not ensure that the location would gain visual attention. Ideally, better saliency maps should have higher saliency compared with the baseline model at the human fixated (ground truth) locations and should be able to attract attention at those locations, i.e. those locations must have high saliency within the saliency map itself. The IG metric is inadequate to capture if a location would actually gain visual attention. Hence, two measurement metrics are used to compare the proposed saliency maps with the baseline Itti maps.

The information gain metric calculates the difference in saliency–measured in bits–at the human fixated locations between two saliency maps for each frame of the video. Given a binary map of ground truth fixations Q^B, a saliency map U, and another baseline saliency map V, information gain is computed as:

$$IG = \frac{1}{N} \sum_{i}^{N} Q_i^B \Big[\log_2(\epsilon + U_i) - \log_2(\epsilon + V_i) \Big] \tag{6.8}$$

where i indexes over pixels, N is the total number of fixated pixels, ϵ is a small regularization constant, and information gain is measured in bits per fixation. Then an average information score is computed for the whole video.

The information gain metric expects valid probability distributions as input. Hence, the saliency map was normalized accordingly to have a

valid probability distribution.

Although higher IG score indicates higher saliency, it does not ensure gaining visual attention for the video frame. On the other hand, the AUC measures how successfully a location in a saliency map was to attract visual attention in competition with other random locations in the same map. In other words, it measures the relative saliency map values at ground truth fixation locations. This is computed by varying a hypothetical threshold and computing the trade-off between true and false positives. Models that place high valued predictions at human fixated locations receive high scores.

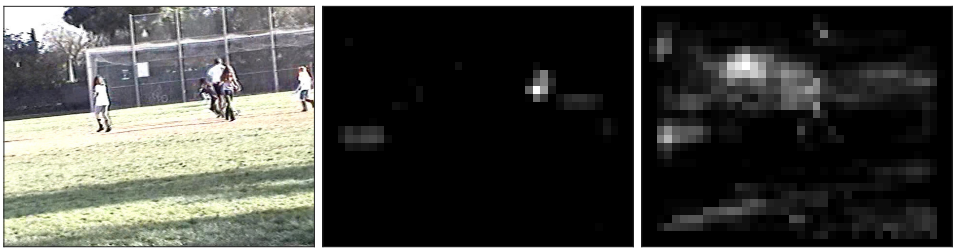
A high scores on both the metrics will indicate that the proposed saliency map has higher saliency and attracts visual attention at human fixated locations more than the baseline saliency model.

The rest of the results section is divided into three parts. Each paragraph presents one of the three utility functions described above and the resulting sampling distribution of visual attention.

6.6.1 Results

It is not possible to display all the saliency maps produced by the two competing methods within the limited space of this article. The video, called 'beverly03', was chosen as an example to display and visually compare the saliency maps produced by the two methods. This video shows children playing in a field. In this video, the left half of the window displays most of the movements and actions. Figure 6.17 shows the 100^{th} frame of the input video and the two corresponding saliency maps. Notice that, the saliency map from the new method highlights the overall motions in the video (high intensity on the left). In comparison the baseline model highlights an area with high contrast in that frame.

Table 6.1 shows the information gain score between the Itti and the proposed model. It can be noticed that the proposed model achieves a



(a) Input video frame. (b) Saliency map produced by Itti model. (c) Saliency map produced by the proposed model.

Figure 6.17: Input image frame and the resulting saliency maps form the two competing models.

higher score in all the videos.

Table 6.2 shows an AUC score comparison between the two models. The AUC computation method proposed in [23] is followed to compute AUC score for each frame of the video, then the AUC scores were averaged over the entire sequence. The proposed model outperforms the Itti model in 35 videos based on the average AUC score, while the standard deviation in the frame-wise AUC scores for any given video sequence remains comparable between the two models. This indicates a steady performance improvement for all the frames in the video.

Table 6.1: This table displays average Information Gain score of videos. The information gain score on each frame of a video has been averaged over the entire video sequence. A positive number indicates that the proposed system assigns higher saliency on an average for the video. Notice that the IG score is positive for all the videos.

Video file	Avg. IG score
beverly01	31.5
beverly03	35.8
beverly05	32.1
beverly06	28.1
beverly07	30.5
beverly08	27.1
gamecube02	32.7
gamecube04	31.6
gamecube05	28.8
gamecube06	30.3
gamecube13	36.1
gamecube16	33.6
gamecube17	28.6
gamecube18	32.5
gamecube23	24.3
monica03	35.5
monica04	29.6
monica05	30.8
monica06	28.2
saccadetest	12.2
standard01	28.9
standard02	33.9
standard03	38.1
standard04	35.2
standard05	35.3
standard06	34.5
standard07	32.2
tv-action01	32.7
tv-ads01	35.4
tv-ads02	31.4
tv-ads03	34.5
tv-ads04	30.5
tv-announce01	37.3
tv-music01	35.8
tv-news01	27.7
tv-news02	32.8
tv-news03	30.3
tv-news04	31.3
tv-news05	34.4
tv-news06	32.3
tv-news09	32.6
tv-sports01	37.2
tv-sports02	34.9
tv-sports03	37.3
tv-sports04	36.7
tv-sports05	33.9
tv-talk01	30.3
tv-talk03	33.6
tv-talk04	25.4

Table 6.2: AUC score of the proposed and the Itti model. The proposed model achieves better AUC score in 70% of the videos.

Video file	Itti	Proposed
beverly01	0.553 ± 0.09	0.654 ± 0.15
beverly03	0.514 ± 0.06	0.633 ± 0.14
beverly05	0.539 ± 0.09	0.619 ± 0.14
beverly06	0.587 ± 0.12	0.681 ± 0.17
beverly07	0.478 ± 0.04	0.453 ± 0.03
beverly08	0.477 ± 0.04	0.461 ± 0.04
gamecube02	0.55 ± 0.11	0.637 ± 0.16
gamecube04	0.571 ± 0.13	0.712 ± 0.2
gamecube05	0.564 ± 0.08	0.584 ± 0.11
gamecube06	0.577 ± 0.11	0.743 ± 0.13
gamecube13	0.504 ± 0.07	0.474 ± 0.1
gamecube16	0.538 ± 0.1	0.458 ± 0.14
gamecube17	0.581 ± 0.1	0.597 ± 0.14
gamecube18	0.564 ± 0.1	0.684 ± 0.16
gamecube23	0.638 ± 0.18	0.709 ± 0.17
monica03	0.515 ± 0.07	0.522 ± 0.13
monica04	0.572 ± 0.11	0.589 ± 0.17
monica05	0.562 ± 0.1	0.592 ± 0.16
monica06	0.565 ± 0.1	0.592 ± 0.13
saccadetest	0.758 ± 0.16	0.704 ± 0.17
standard01	0.564 ± 0.12	0.594 ± 0.19
standard02	0.541 ± 0.09	0.586 ± 0.13
standard03	0.486 ± 0.03	0.467 ± 0.12
standard04	0.517 ± 0.07	0.488 ± 0.13
standard05	0.524 ± 0.09	0.627 ± 0.15
standard06	0.526 ± 0.08	0.496 ± 0.15
standard07	0.557 ± 0.1	0.656 ± 0.17
tv-action01	0.543 ± 0.1	0.657 ± 0.18
tv-ads01	0.506 ± 0.08	0.497 ± 0.11
tv-ads02	0.533 ± 0.1	0.479 ± 0.12
tv-ads03	0.527 ± 0.08	0.55 ± 0.14
tv-ads04	0.523 ± 0.08	0.511 ± 0.11
tv-announce01	0.49 ± 0.05	0.493 ± 0.15
tv-music01	0.524 ± 0.09	0.603 ± 0.16
tv-news01	0.545 ± 0.09	0.632 ± 0.16
tv-news02	0.504 ± 0.06	0.51 ± 0.08
tv-news03	0.519 ± 0.08	0.533 ± 0.1
tv-news04	0.544 ± 0.1	0.512 ± 0.11
tv-news05	0.516 ± 0.08	0.538 ± 0.12
tv-news06	0.502 ± 0.06	0.541 ± 0.12
tv-news09	0.508 ± 0.07	0.516 ± 0.1
tv-sports01	0.501 ± 0.06	0.528 ± 0.15
tv-sports02	0.509 ± 0.08	0.493 ± 0.11
tv-sports03	0.507 ± 0.07	0.589 ± 0.2
tv-sports04	0.515 ± 0.08	0.539 ± 0.17
tv-sports05	0.541 ± 0.09	0.616 ± 0.15
tv-talk01	0.554 ± 0.12	0.556 ± 0.14
tv-talk03	0.53 ± 0.09	0.495 ± 0.13
tv-talk04	0.506 ± 0.06	0.502 ± 0.07

6.7 Chapter Discussion

This chapter presented two important contributions of the thesis as below.

1. The first contribution of this chapter is the three novel utility functions that were proposed for selecting visual targets. This is the first time in the field of saliency map literature a varied range of useful behaviours has been demonstrated while operating on the same input.

2. This chapter presented an improvement of the classical Itti model by replacing the traditional winner-take-all and inhibition of return mechanisms with the novel Kalman filter aided epistemic target selector. It was shown that the improved Itti model could predict human fixations better compared to the classical Itti model according to two standard measurement metrics.

One drawback of using Kalman filter aided epistemic target selector methods is that the computational complexity of the algorithm is $\mathcal{O}(n^3)$ due to the matrix inversion involved with the Kalman Filter equation. Under the assumption that the visual regions of interest are independent of each other, the computational cost was reduced. Nevertheless, a more general case where the interdependencies of the pixel intensities are considered needs to be explored. The extent to which computational performances constrain real-world exploitations of our proposed method also remains to be explored.

Due to the simplicity of initial experiments, any spatial dependency in the visual scene was not considered. As the proposed method does not make any assumptions on the size of input dimensions it should principally work on a spatially dependent scenario without modification.

Three utility functions were proposed in this chapter, but other strategies could also be formed. Different strategies would result in different agent behaviours, which might be necessary to operate as per the need of different situations.

All three utility functions proposed in this chapter operate solely on the uncertainty in the state estimate. Often visual locations with a desired set of features need to be given more attention. In such scenarios, the mean of the agent's belief distribution about the presence of that feature at a specific location needs to be included in the utility function. Such a utility function will be proposed in chapter 7. This one combines the mean and the uncertainty in a way that higher uncertainty regions that have the desired features gain the most visual attention.

Chapter 7

Prioritisation of High Salience Targets

This chapter presents a utility function that can achieve the task of reducing uncertainty about visual targets with desired features. This utility function is important for scenarios where reduction in knowledge about every element of the scene is not important. Instead, it is desired that the agent reduces uncertainty about sections of the visual scene that have relevant features. Such an agent's internal belief states would represent the intensity of the feature by the mean of the belief distribution and the confidence as the variance of the distribution.

A desired agent behaviour is searching for a known set of features in a visual scene. As an example, in an image processing application it could be important to be able to locate the pedestrians in a scene. In such case, the agent should be driven by some combination of where it thinks the target is likely to be (mean of saliency), plus at areas where it is not sure. Most importantly, the agent does not want to look at locations where it is confident that there is not a target. That is, if the mean (the saliency) and the variance (the confidence on saliency) are both low then the agent should avoid that region. The aim of the agent looking for desired features in a visual scene would be to reduce uncertainty about the belief distributions at

locations that have high values for the desired features. This would generate a behaviour where the agent will look preferentially at visual regions with a high degree of desired features.

It is important to notice that the process nose estimators proposed in the previous chapter 5 will be utilised in this chapter to compute the elements of the process noise online. This will allow the Kalman filter based target selector to adapt to the visual scene.

7.1 Searching for Desired Features in a Visual Scene

The internal belief states of an agent are its representation of the world. In the presented case it was assumed that the internal belief states are Gaussian distributed. Hence they are completely defined by the mean of the distribution and the variance of the distribution.

In such scenarios, reduction in knowledge about every element of the scene is not important. Instead, it is desired that the agent reduces uncertainty about sections of the visual scene that have relevant features. Hence the sections of the visual scene that have the relevant feature need to gain higher utility than the rest of the scene.

These task dependent features would form the constituent parts of the saliency map, just as colour, orientation and intensity formed the parts of the original Itti model. In this way, the combination of task specific target features is rendered salient by the initial processing of the saliency system. An agent can then operate on this modified saliency map to seek areas of desirable properties (i.e. areas with high saliency).

Such an agent's internal belief states would represent the intensity of the feature by the mean of the distribution and the confidence as the variance of the distribution. Given the mean and the variance of the belief states, the aim of an agent looking for desired features in a visual scene

would be to reduce uncertainty about the belief distributions that have high values for the desired features. This would generate a behaviour where the agent will look at visual regions with a high degree of desired features.

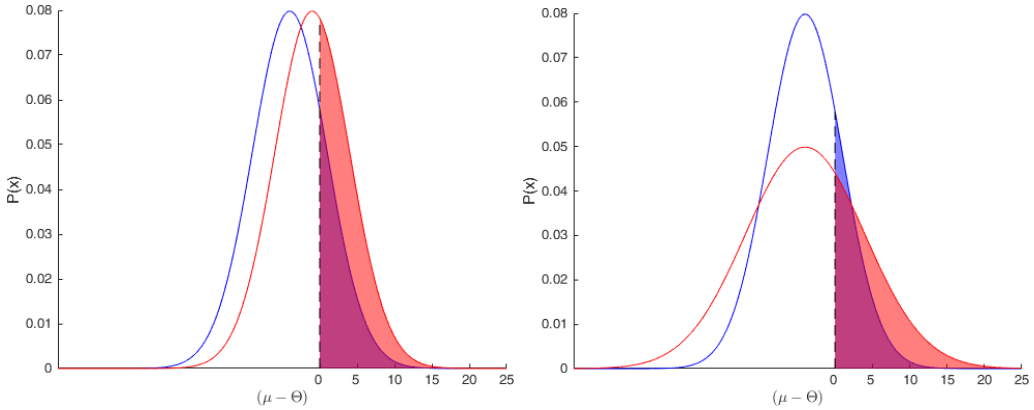
An agent needs proper instruments to measure the presence of desired features in a scene. For example a camera with relevant feature detector. Such an agent's internal belief states would represent the intensity of the feature by the mean of the distribution and the confidence as the variance of the distribution.

Given the mean and the variance of the belief states, the aim if an agent looking to desired features in a visual scene would be to reduce uncertainty about the belief distributions that have high mean.

This behaviour can be achieved by finding the mass of a belief probability distribution that is above a given threshold. That is, it would only investigate areas that appear to offer pay-off for the current task. For example, consider figure 7.1 where blue and the red lines show two independent Gaussian distributions. The black vertical dashed line shows a chosen threshold and the area shaded in blue and red shows the area under the distribution above the threshold.

The left panel shows that the red distribution has more variance than the blue one and it can be noticed that the area in blue is smaller than the area in red. On the other hand, the right panel shows that the red distribution has a higher mean than the blue distribution. The red area under the curve above the threshold is more than the blue area under the curve.

These two figures show that if the area under the curve above a threshold is used as a utility function, the agent will minimise uncertainty only regarding the sections of the visual scene that have a higher mean than the threshold. This is known as the expected improvement above a threshold.



(a) Red distribution has higher mean than the blue

(b) Red distribution has higher uncertainty than the blue

Figure 7.1: Figure shows the area under the curve of two Gaussian distributions above a threshold. The threshold is shown as a black dashed line.

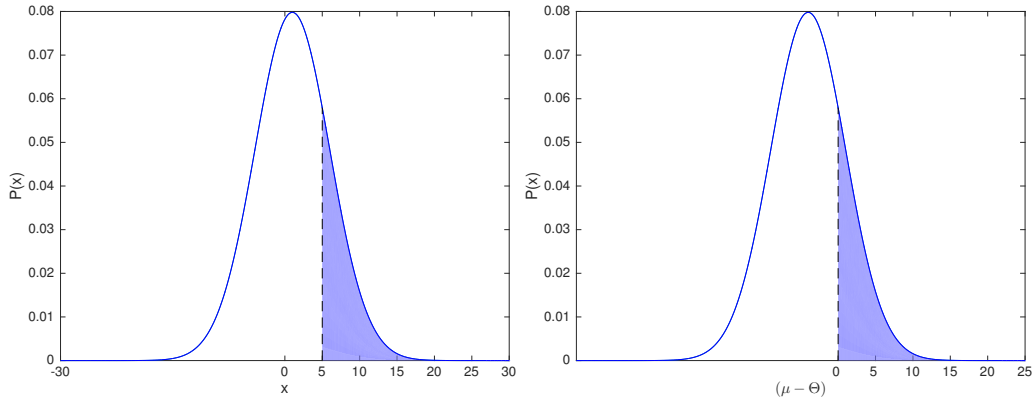
Instead of simply finding the probability that there will be some improvement, the amount of expected improvement can be calculated by finding the expected value of the mass of the probability distribution above the threshold.

Figure 7.2 shows two Gaussian distributions. The left panel shows the probability distribution and the area shaded in blue shows the area under the curve above a threshold of 5. The right panel shows the same belief state distribution but now the horizontal axis marks the improvement over the threshold 5. Notice the change in the marking of the horizontal axis in the two plots.

This can be achieved by finding out the mass of a belief probability distribution that is above given threshold. The following equation is used to compute the mass of the probability distribution above a threshold.

$$EI = \int_{\Theta}^{\infty} P(x)(x - \Theta)dx \tag{7.1}$$

where EI is the expectation of an improvement, Θ is a chosen threshold



(a) Expectation above a threshold

(b) Expectation of improvement above a threshold

Figure 7.2: The left and the right panels show the expectation above a threshold and the expected improvement above a threshold. Notice that the horizontal axis on the right panel shows improvement.

beyond which an improvement is sought, $P(x)$ is the probability distribution of a belief state and x is the mean of the distribution.

The integration mentioned in equation 7.1 is tedious to solve. A compliment of this equation that finds the expected improvement by integrating from a threshold to $-\infty$ already exists in the Gaussian process based efficient global optimisation literature [59, 98].

As this existing integral finds the expectation of the distribution from a threshold up-to the negative infinite, this integral can be used to form our proposed equation 7.1 as a minimization problem over all the belief states. Hence the visual state with the highest expected improvement over threshold Θ is given by

$$EI^{\star} = \arg\min_{x} \left(\int_{-\Theta}^{-\infty} P(x)(-x - \Theta)dx \right) \tag{7.2}$$

where $EI^{\star}$ is the optimum choice.

Equation 7.2 is evaluated using the error function as below:

$$EI(x) = (-\Theta + x)\left[\frac{1}{2} + \frac{1}{2}erf(\frac{-\Theta + x}{\sqrt{2}\sigma})\right] + \sigma\frac{1}{\sqrt{2\pi}}\exp\left[\frac{-(-\Theta + x)^2}{2\sigma^2}\right] \quad (7.3)$$

This evaluation is a standard result of the Gaussian process and efficient global optimisation literature [59,98].

The above equation 7.3 computes the expectation of a distribution above a given threshold. The threshold is a choice of the agent. Figure 7.3 shows two heat-map plots of the proposed utility function with two different thresholds. Each plot shows how the utility changes with a change in variance and the mean of a distribution. The vertical axis shows the variance and the horizontal axis shows the mean of the distribution. White indicates high values and black colour is low values.

It is important to note that the mass of the probability distribution above the threshold is the probability of improvement. The Gaussian distributions represent the uncertainty in the belief state and the part of the distributions that is above the dotted line indicates the possibility of improving above the threshold. Hence it is the area enclosed by the Gaussian belief state above the threshold value.

Equation 7.1 computes the expectation of a distribution above a given threshold. The threshold is a choice of the agent. The choice of the threshold produces different visual scene sampling strategy.

Method

For the utility function that searches for desired features in the visual scene, skin colour was used as the desired feature. How close a pixel is to a skin colour was decided by computing the Euclidean distance between the pixel and a reference point for skin colour in the L*a*b colour space. This method is commonly referred to as the ΔE colour difference com-

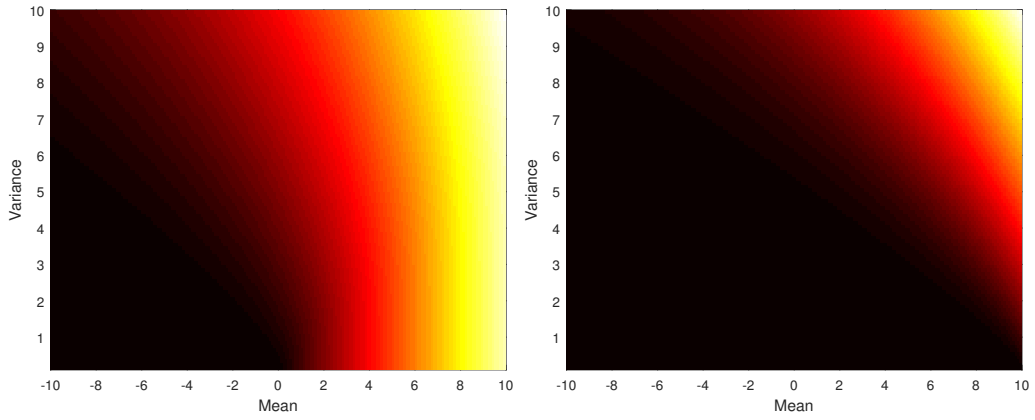
putation [86] The reference pixel is chosen manually for each video. This pixel is always chosen from an exposed part of the skin.

The distance in the L*a*b colourspace of all the pixels in each frame of the videos from the reference pixel was computed. Then the individual frames were put back in the same sequence as the original video.

Results

Figure 7.3 shows two heatmap plots of the proposed utility function with two different thresholds. Each plot shows how the utility changes with a change in variance and the mean of a distribution. The vertical axis shows the variance and the horizontal axis shows mean of the distribution. The white colour shows high value and black colour shows low value.

The left panel shows the utility plot with $\Theta = 0$ and the right panel shows the utility plot with $\Theta = 10$. Notice that the high variance region above the set threshold offers the highest utility.



(a) $\Theta = 0$ (b) $\Theta = 10$

Figure 7.3: Two plots of how the utility changes with a change in the mean and the variance of the distribution. White colour shows the high utility and black shows the low utility. Notice that only the high variance region above the threshold offers high utility (white colour).

Figure 7.4 shows the results of using an expected improvement based utility function. MLE was used to learn the process noise matrix. The first second and the third column of the figure shows the first frame of the skin-colorness, process noise matrix and the visual attention distribution respectively.

The process noise matrix image differs from the earlier process noise matrices as the agent computes the variance in change in the coloruness, not in the pixel intensity.

Notice that the agent visited the visual regions that have a similar colour to the reference colour. The colorness image can be used as a reference for understanding the visual attention distribution. Notice that the visual regions attended have high values in the colorness image. For example, the area on the left side of the forehead of the actor on the right of the Faces-46 video has high values. The corresponding regions attracted high visual attention, which can be noticed in the attention distribution plot. Similar correspondences can be found for all the other videos.

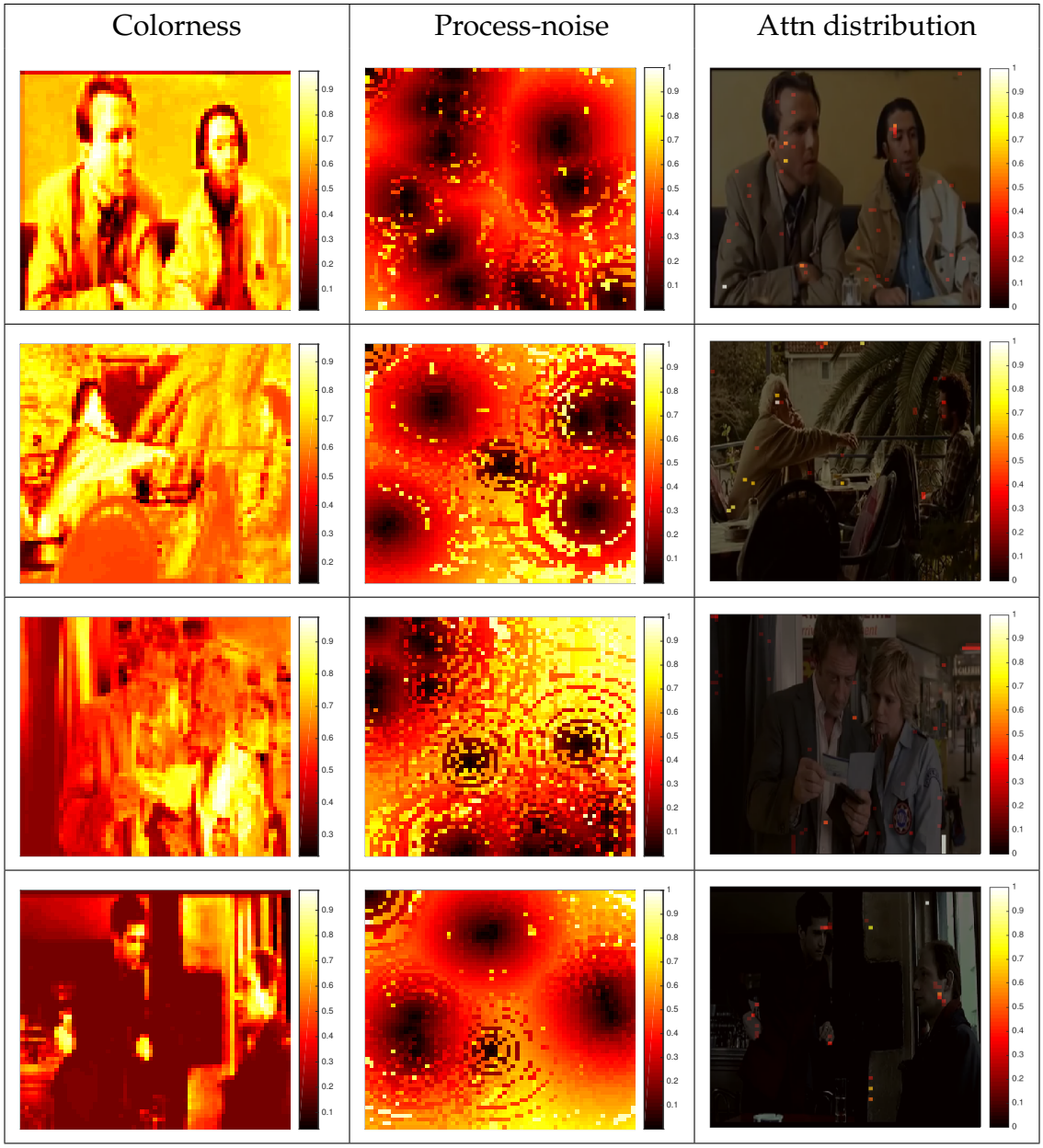


Figure 7.4: Estimated process noise using MLE of all the five videos, and the corresponding visual attention distribution. The left, middle and right columns show the skin-colorness image, histogram plot and the visual attention distribution respectively.

Figure 7.5 shows the outputs of the expected improvement based utility function while the $\boldsymbol{Q}$ matrix was learnt using the recursive method. The left, middle and the right columns show the colourness, process noise and the visual attention distributions plots respectively. Notice that the visual regions with high values of skin colour attracted most of the visual attention which is the desired behaviour.

Figure 7.5: Estimated process noise using recursive estimator of all the five videos, and the corresponding visual attention distribution. The left, middle and right column shows the skin-colorness image, histogram plot and the visual attention distribution respectively.

7.2 Chapter Discussion

This chapter contributed a novel utility function which can generate the behaviour of reducing uncertainty about desired visual targets. This behaviour could not be achieved using the three utility functions in chapter 6. Also, different values of the threshold can be used to generate different behaviours.

Chapter 8

Surprise Detection

This chapter presents a novel surprise detection mechanism that works along with the Kalman filter based epistemic target selector. The combination of these systems will behave as before until it detects a surprise. In case of a surprise, the system will direct attention to the surprising location for further inspection.

The proposed model chapter 3 presented the Kalman filter based visual target selector. At every time step the Kalman filter updates the internal belief states as a weighted average of its observation and internal prediction. The limitation of this approach is that in case the measurement significantly varies from the prediction of the Kalman filter, the Kalman filter will still find a weighted average between the measurement and the prediction to update the internal belief state. Whereas, in real life surprises need immediate attention for further detailed inspection.

A view is taken that a surprise is an unpredicted or sudden change in the real-world, which cannot be predicted by the internal world model matrix. That is, a surprise is an occasion for which the model does not adequately predict what has been observed. Hence surprising events should be considerably different from the prediction of the agent's internal belief.

8.1 Detecting Surprising Events in a Visual Scene

The aim is to add surprise detection ability on top of one of the previously discussed utility functions. Therefore the system will behave as before until it detects a surprise.

A view is taken that a surprise is an unpredicted or sudden change in the real world, which can not be predicted by the internal world model matrix ($\boldsymbol{A}$). That is, a surprise is an occasion for which the model does not adequately predict what has been observed. Hence surprising events should be considerably different from the prediction of the agent's internal belief.

The difference between the internal belief and a new measurement can be described as a difference between prediction and the actual measurement.

$$\boldsymbol{\mathcal{I}}_{t+1} = \boldsymbol{y}_{t+1} - \hat{\boldsymbol{x}}_{t+1|t} \tag{8.1}$$

where $\boldsymbol{\mathcal{I}}_{t+1}$ is called the innovation, $\boldsymbol{y}_{t+1}$ is the new measurement, $\hat{\boldsymbol{x}}_{t+1|t}$ is the prediction. The magnitude of the innovation will be high for surprising measurements.

The proposed model measures each visual region with different measurement noise. A measurement with higher measurement noise will inherently give rise to higher innovation due to statistical variation. However as it was measured with higher measurement noise, the measurement itself is not trustworthy. Hence a measure of surprise should be weighted by the uncertainty associated with the innovation.

The innovation $\boldsymbol{\mathcal{I}}_{t+1}$ is a zero mean, white Gaussian noise signal whose covariance is the sum of the measurement noise covariance and the uncertainty in the internal belief. As model uncertainty and the measurement noise are uncorrelated, the uncertainty in innovation is given by the equation below:

$$\mathrm{E}[\boldsymbol{\mathcal{I}}_{t+1}\boldsymbol{\mathcal{I}}_{t+1}^{\mathsf{T}}] = \boldsymbol{P}_{t+1|t} + \boldsymbol{R}_{t+1} \tag{8.2}$$

An individual instance of innovation (described as a difference in equa-

tion 8.1) can be thought of as a sample from a Gaussian distribution with zero mean and the variance given by equation 8.2. Given a sample from the innovation distribution defined as above, the distance between the sample and the distribution in terms of 'how many standard deviations' can be calculated. This distance measure is known as the Mahalanobis Distance [124], which is adopted as a measure of surprise in this work. This measure is given by the following equation:

$$\boldsymbol{s}_{t+1|t} = \sqrt{\boldsymbol{\mathcal{I}}_{t+1}^{\mathsf{T}} \left(\boldsymbol{P}_{t+1|t} + \boldsymbol{R}_{t+1}\right)^{-1} \boldsymbol{\mathcal{I}}_{t+1}} \tag{8.3}$$

where $\boldsymbol{s}$ is the measure of surprise. Elements of the vector $\boldsymbol{s}$ in equation 8.3 will be small for the regions with unsurprising measurements but will be high for the case of measurement that is inconsistent with the internal model. Note that this calculation doesn't require the inclusion of the current measurement into the belief state, unlike the approach of Itti and Baldi [88].

With the assumption where the regions of interests in a visual scene are independent, it is found that the surprise at a region i is given by

$$s_{t+1|t} = \sqrt{\frac{\mathcal{I}_{t+1,i}^2}{\left(P_{t+1|t,i} + R_{t+1,i}\right)}} \tag{8.4}$$

where $\mathcal{I}$, $P_{t+1|t,i}$, $R_{t+1,i}$ are corresponding elements of the innovation, uncertainty and measurement noise matrix.

Equation 8.4 can be thought of as a ratio of 'innovation' (numerator) to 'confidence' (denominator). Hence high innovation alone does not result in high surprise; the value of $(P_{t+1|t,i} + R_{t+1,i})$ have to be low at the same time. The ideal condition for the surprise to be high is when there is a low value in denominator and high value in the numerator. An intuitive interpretation is that a system is surprised only when it is confident about the future state of a region and the actual measurement of the state is considerably different from the prediction.

8.1.1 System Performance Measurement Metric

In the light of equation 8.3 a surprise after observation is:

$$\boldsymbol{s}_{t+1|t+1} = \sqrt{\boldsymbol{\mathcal{I}}_{t+1}^{\mathsf{T}} \left(\boldsymbol{P}_{t+1|t+1} + \boldsymbol{R}_{t+1}\right)^{-1} \boldsymbol{\mathcal{I}}_{t+1}} \tag{8.5}$$

where $\boldsymbol{s}_{t+1|t+1}$ is the surprise after observation at time instant $t+1$, $\boldsymbol{\mathcal{I}}$ is the innovation, $\boldsymbol{P}$ is the covariance and $\boldsymbol{R}$ is the measurement noise. Notice the change in the notation of covariance matrix compared to equation 8.3. Here the internal uncertainty is updated to the current time step as the interest is in measured surprise after the observation in contrast to the detection of surprise, which is detected before the observation.

Surprise detection can be thought of as a problem of statistical hypothesis testing. The value of $\boldsymbol{s}$ is low when there is no surprise and it is high when the agent detects surprise. A decision threshold is used to verify whether or not a hypothesis (e.g. there was a surprise) is true or false. A hypothesis $\mathcal{H} : \boldsymbol{s}(\boldsymbol{i}) \geq \theta$ can be formulated, for when there is a surprise, where i is the corresponding element of the surprise vector. Here θ, the decision threshold, is a positive number. If θ is set too low, a noisy measurement can render $\mathcal{H}$ to be true. Alternatively, if θ is too high, the agent might not detect any surprise at all. A noisy measurement triggering $\mathcal{H}$ is a false alarm, known as False Positive (FP), whereas a true surprise escaping the test is a deficient performance, known as False Negative (FN).

A low threshold guarantees detection of all the true surprises, known as True Positives (TP), but potentially along with a few false positives (FP). Whereas setting a very high threshold would cause the system to ignore many true surprises. Deciding on a value of the decision threshold is a trade-off between TP and FP. A receiver operating curve (ROC) depicts the sensitivity of the decision threshold when faced with balancing this trade-off.

A ROC plots false positive on the horizontal axis and true positive on the vertical axis (the surprises intended to be identified). It illustrates the complete trade-off between false positives and false negatives over a range

of decision thresholds. Each point on the ROC corresponds to a unique decision threshold.

A good algorithm detects a significant proportion of true positives with a few false positives. When plotted, a good algorithm results in a ROC that climbs rapidly towards the upper left corner of the graph. Originally, ROC was designed for two class problems but has been extended to multi-class problems, which suits our problem of surprise detection. This uses the one-vs-rest strategy described in [54].

One algorithm results in a single ROC, so by comparing these curves the performances of an algorithm with different settings can be compared. One data point on the ROC gives us an 'operating point' of an algorithm and the whole curve captures the overall quality of the algorithm. Figure 8.4 shows three ROCs, plotted in red, green and blue. The area under the ROC, called AUC, is used to quantify how quickly the ROC rises to the upper left corner, which in turn is a quantified measure of the system's performance. A larger value of AUC represents better ability in detecting surprise. The result section analyses these curves in details.

8.1.2 Experimental Method

The goal of the experiments with a surprise detector is to determine how long an agent takes to detect a novel event (or to measure the system's reaction time). Here a novel event is a highly salient element that appears at a random location within the visual scene, with saliency higher than the usual maximum saliency. This happens only after the system has sampled the entire visual scene at least once.

A computer program records the system's reaction time, measured in a number of iterations the system takes to detect the novel event after the novel event has occurred. This test is run for 1000 trials in order to record the reaction time for each run.

Each test run, called a trial, involves the salient event occurring at a

random location which is fixed only through that trial. Hence over the course of 1000 trials, the system's reaction time is measured in detecting surprises occurring at different locations of the visual scene. Finally, a histogram of the reaction times will be plotted to show the sensitivity of the system to surprise and the foveation profile. Each histogram was later normalised to have an area under the curve as 1.

The choice of foveation profile affects the system's reaction time. These choices are discussed in this section. The implementation method for the traditional saliency map idea and the measurement metric used to evaluate the proposed method is presented.

As discussed in chapter 4, an agent with a narrow foveation profile is designed to reduce uncertainty only at the point of fixation. For such a system, the overall uncertainty of its own knowledge will remain high. Its lack of internal confidence makes it difficult for the system to detect any surprise.

It is important to bias the agent's decision in the case of a low internal confidence. Hence the surprise ($\boldsymbol{s}$) is added to the utility function to make the system aware of the unexpected changes. It was decided to compare the augmented utility function with that described in section 6.4. That is the surprise detection performance of the total uncertainty reduction utility function was probed.

$$u_p = P_{i,i} \tag{8.6a}$$

$$u_{sp} = P_{i,i} + s_{t+1|t,i} \tag{8.6b}$$

The first equation involves only uncertainty and the second equation adds surprise to the uncertainty. $P_{i,i}$ are diagonal elements of the matrix $\boldsymbol{P}_{t+1|t}$. $\boldsymbol{P}_{t+1|t}$ is the projected uncertainty and $s_{t+1|t,i}$ is the corresponding element from surprise vector. It is evident that the agent's decision should be influenced by surprise for the second choice of utility function.

1000 trials were tested with the three foveation profiles (section 6.2.1) for each of the two utility functions introduced in equation 8.6. It is pos-

tulated from equation 8.6 that adding surprise would result in shorter response time.

8.1.3 Traditional Saliency Map

Our method is compared with the traditional method of sampling saliency maps, where important regions of a visual scene are sampled serially in decreasing order of their salience. It is important to inhibit a previously observed salient location in traditional saliency maps otherwise the system would keep fixating forever on the most salient location. Hence after each observation, a small section of the saliency map around the just observed peak is set to zero saliency. As a result, the next higher saliency gains visual attention. Named after its functionality, this mechanism is called Inhibition of Return (IOR). The human visual system is believed to have an after observation inhibition effect of around 3 seconds [160].

The performance of an artificial system, designed to detect surprise, will depend on how long a previously observed location is blocked (the IOR blocking time). It is expected that a longer IOR blocking time will increase the system's reaction time to a surprising event, as such events are likely to occur at blocked locations.

The area of sampling (number of pixels to be viewed at each step) was chosen as 5% of the saliency map area with an equal number of pixels on either side of the fixation point. That gives us $(100 \times 5)/100 \approx 5$ pixels to sample at every iteration, centred at the location of the most salient region on the visual scene. This can be imagined as a 5 pixel wide movable spotlight within a visual scene of 100 pixels, which would take 20 steps to sample the complete visual scene. The IOR blocking time was chosen as 20, so that ideally, the system can explore the whole visual scene before re-observation of any location. When part of the 'spotlight' went outside the saliency map boundary, the hanging section was trimmed accordingly.

8.1.4 Results

This section presents the performance of the proposed algorithm under six different evaluation conditions described below. The end of this section presents a quantitative evaluation of response time with a traditional saliency map approach with varying IOR blocking time and the proposed method. Experimentation with both the approaches use the same saliency map (see figure 6.2).

8.1.5 Results from Kalman Filter Based Saliency Map

Figures 8.2 shows the histograms of the distribution of reaction times with three foveation profile parameters ($\rho^2 = 0.001, 10, 100$). For each of these parameters, the histogram of the system's performance was plotted for both utility functions described in equation 8.6. The three panels in the first column of the figure show the system's output without surprise in the utility function, whereas the three panels of the right column show the system's output with surprise included in the utility function. The surprise detection threshold was set to 3 for all the cases.

With a narrow foveation profile ($\rho^2 = 0.001$) any observation results in a reduction in uncertainty associated only with the point of focus in the scene, so this strategy does not reduce the uncertainty of any neighbouring pixels. A slow growing rate ($\boldsymbol{Q} = 0.001$) for uncertainty causes the system to place the profile-centre of the foveation profile at every pixel in the scene in turn. The top panel of figure 8.1 shows the system's attention distribution for 10 time steps with narrow foveation profile. The horizontal axis shows time, the vertical axis shows locations. The red dots show the attended location. The system attended each pixel serially.

The introduction of surprise in the utility equation reduces the worst-case reaction time from 386 time steps to 139 time steps, which is evident from the limit of the horizontal axis in the right-top panel in figure 8.2. Also, the system failed to detect surprise 18 times out of the total number

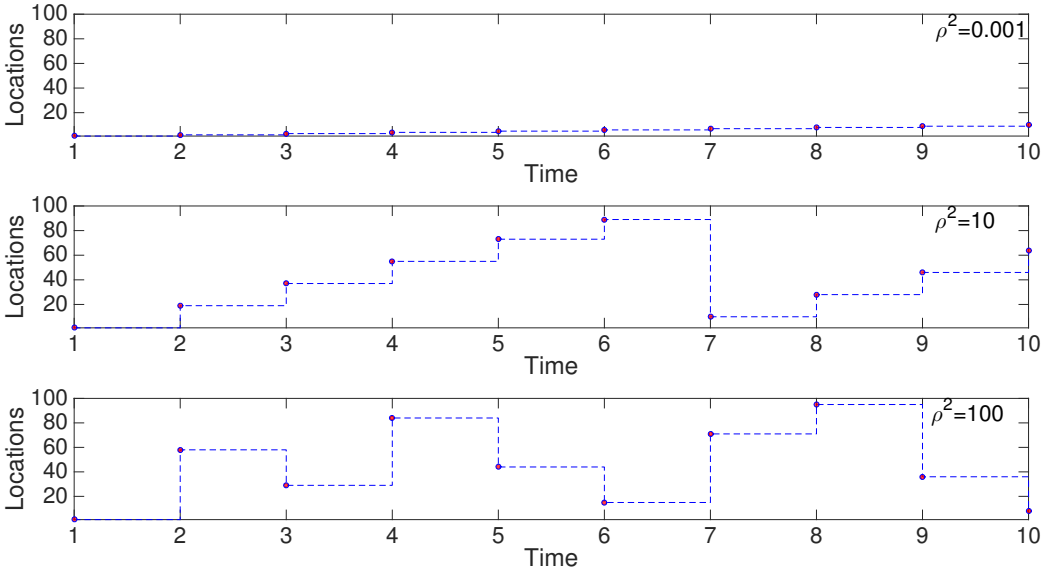


Figure 8.1: Staircase plot of the agent's attention distribution over time with the three foveation profiles. The Y axis shows the visual location attended and the horizontal axis shows time. The red dots show the attended locations. The height of each step increases as the foveation profile becomes wider. This is because a wider foveation profile reduces uncertainty related to a larger group of pixels. Hence the system skips the whole group of pixels.

of trials when it was not biased by previous measures of surprise, i.e. only the uncertainty was used as the utility (see equation: 8.6a).

The system fails to detect surprise because while operating with the narrow foveation profile the system's average level of uncertainty remains high. Hence the system is not confident about detecting surprise. This can be thought of as the large denominator (due to high internal uncertainty) in equation 8.5, which results in a very low value of measured surprise.

With a medium foveation profile ($\rho^2 = 10$) one observation results in a reduction in uncertainty associated with a small group of neighbouring pixels. Hence when the system is reducing only uncertainty (equation: 8.6a), many of the pixels are never directly observed. This is because

each observation adequately reduces the uncertainty of neighbouring pixels. This is evident from the increase in the height of the step in attention distribution (figure: 8.1, middle panel). The step height is greater compared with $\rho^2 = 0.001$ (the top panel). The second row, first column of figure 8.2 shows the histogram of the system's performance with the medium foveation profile. Reduction in worst case reaction time (maximum reaction time decreased from 290 time-steps to 30 time-steps) is observed when surprise was included in the utility function (shown in the corresponding histogram in the right of the figure 8.2). Under this foveation profile, the system failed to detect surprise only once with uncertainty as the utility function. As expected, this is a better performance than the narrow foveation profile as the system more successfully detected a larger number of surprises.

With a wide foveation profile ($\rho^2 = 100$) one observation results in a reduction in uncertainty associated with a larger number of neighbouring pixels. This results in bigger jumps between two successive fixation locations (Figure: 8.1 bottom panel). In this case, most of the pixels are never directly observed again as the observation of a neighbouring pixel reduces the uncertainty associated with it. The addition of surprise in the utility function does not reduce the maximum time taken to detect a surprise as much as in the earlier settings. The maximum response time with wide foveation profile shown in the third row of figure 8.2 is almost the same with or without adding surprise. Due to overall low internal uncertainty, the system always detected surprise successfully. The wider foveation profile results in low overall uncertainty level (the system maintains a better knowledge of the world) and detects all the surprising events, hence it generates the most efficient system behaviour. However, having a wide foveation profile is more expensive.

Figure 8.3 shows the change of true state, internal state, and surprise related to region number 1 over time for comparison of how the internal state updates with two different foveation profiles. The left panel shows

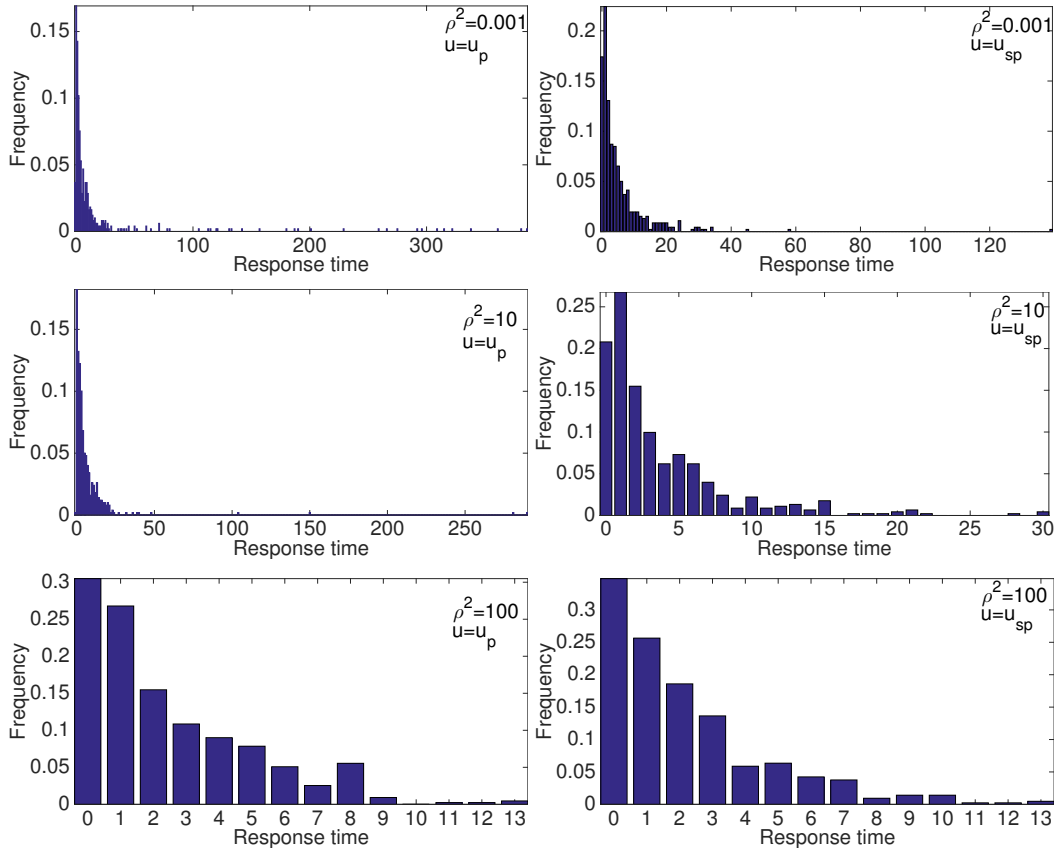


Figure 8.2: System's performance with three foveation profiles ($\rho^2 = 0.001, 10, 100$). Simulation outputs of the proposed algorithm under 6 evaluation conditions. The system's response time with two utility functions for three foveation profiles is shown column wise.

the states while operating with a narrow foveation profile ($\rho^2 = 0.001$) and the right panel shows a plot of a wide foveation profile($\rho^2 = 100$). Notice that the amplitude of measured surprise is considerably higher in the case of the wider foveation profile. This is due to the wide foveation profile reducing the overall internal uncertainty. Hence the system is more confident about the surprise in the external world. Notice also, that the internal belief state (the red line in the figure) reaches the true value, near time instant 100, in case of the wide foveation profile. This is because the

agent trusts its measurements more as they are less noisy compared with a narrow foveation profile. On the other hand, the agent mostly trusts its internal predictions rather than the measurements while operating with a narrow foveation profile. This results in slow rising of the internal belief state of the agent (refer to the left panel of figure 8.3).

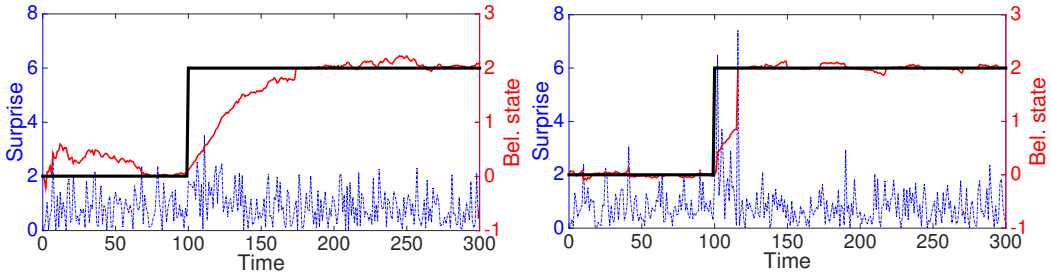


Figure 8.3: Plot of the agent's belief state, with perceived surprise compared to the true change in the visual scene over time. the left panel shows the agent operating with a narrow foveation profile and right panel shows operating with a wide foveation profile. The axis on the left hand side of each panel shows surprise whereas the right hand axis shows the internal belief state of the agent. Notice that the system fails to detect the surprise that comes just after the 100^{th} time instant while operating with narrow foveation profile.

Figure 8.4 shows the ROC for the surprise detection algorithm with three foveation profiles. The vertical axis shows the true positive and the horizontal axis shows the false positives. A larger area under the curve indicates better classification. Notice that the best classification is achieved when the agent operates with a wide foveation profile. This is because the wide profile observes a greater area of the visual scene with low measurement noise, which results in low internal uncertainty. Therefore the agent's internal uncertainty is low and can tell surprise apart from noisy measurements.

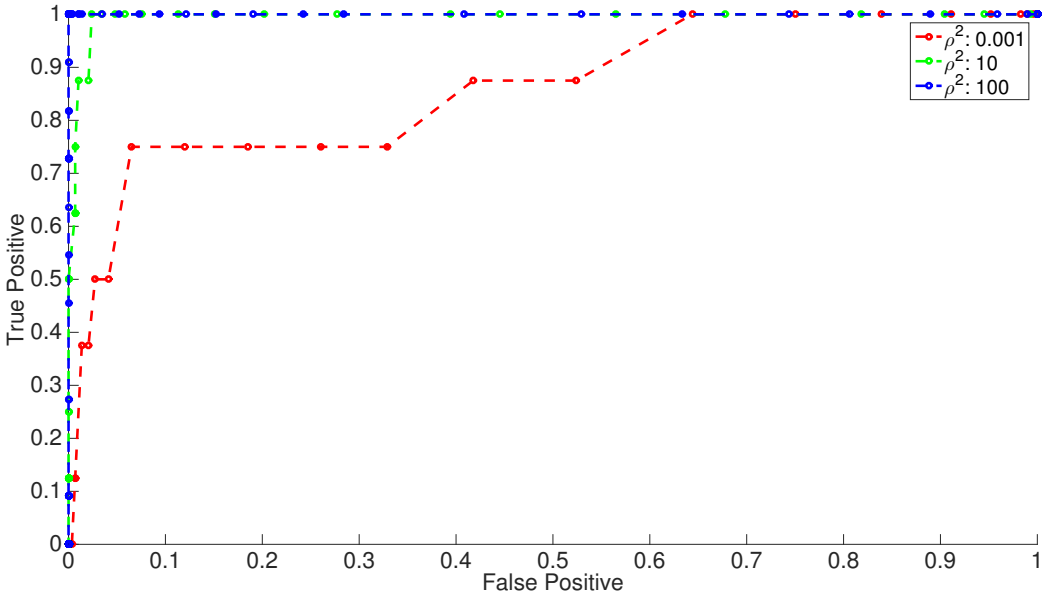


Figure 8.4: ROCs with three foveation profiles. The red curve shows the least area covered (AUC=0.86), the green curve shows AUC=0.98 and the blue curve shows AUC=1. They capture the system's ability to detect false positive compared to true positives. A larger area under the curve (AUC) represents better classification.

8.1.6 Results from Traditional Saliency Map

A simple version of the traditional saliency map with IOR was implemented. It showed how the average time to detect surprise increases as a function of the IOR blocking time. Figure 8.5 shows the output of a traditional Saliency map implementation with IOR blocking time varying from 20 to 35, and the graph shows a monotonic increase in surprise detection time (the response time) with an increase in IOR blocking time.

Although direct comparison is difficult the medium foveation profile has the closest resemblance of the IOR blocking area of 5 pixels wide. Hence the system behaviours of traditional saliency map and the proposed system operating with the medium foveation profile are compared. Figure 8.6 shows the response time histograms of the traditional and the

proposed method. On an average, the traditional method took 11.9 time-steps and the proposed method took 6.9 time-steps to detect the surprise. The traditional approach has a longer response time because it ignores the inhibited regions of the scene. Any surprise event occurring at the inhibited region remains undetected until it is released. Hence the maximum response time is determined by the IOR blocking time. The maximum response time taken by the proposed method was 290 time steps, which is longer than the maximum time taken by the traditional method (20 time steps). Observe that the histogram of the traditional approach is less skewed (standard deviation=5.5) compared to the histogram of the proposed system (standard deviation is=20.3). This is because with the medium foveation profile the system cannot maintain an average uncertainty level low enough to detect all the surprises (refer to equation 8.5).

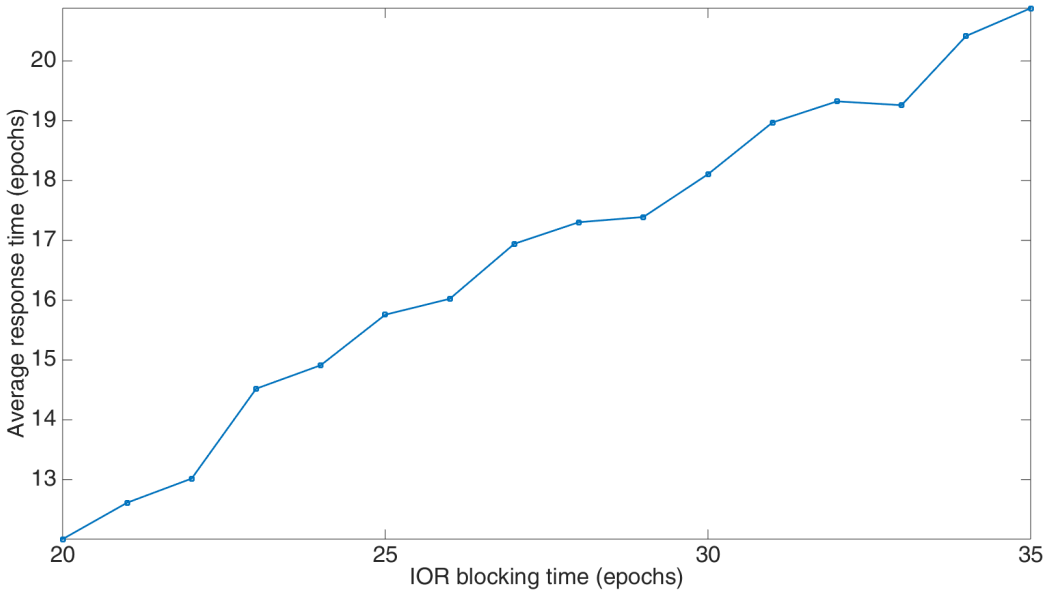


Figure 8.5: A plot showing the rise in average reaction time with the increase in the IOR blocking time for the traditional saliency system.

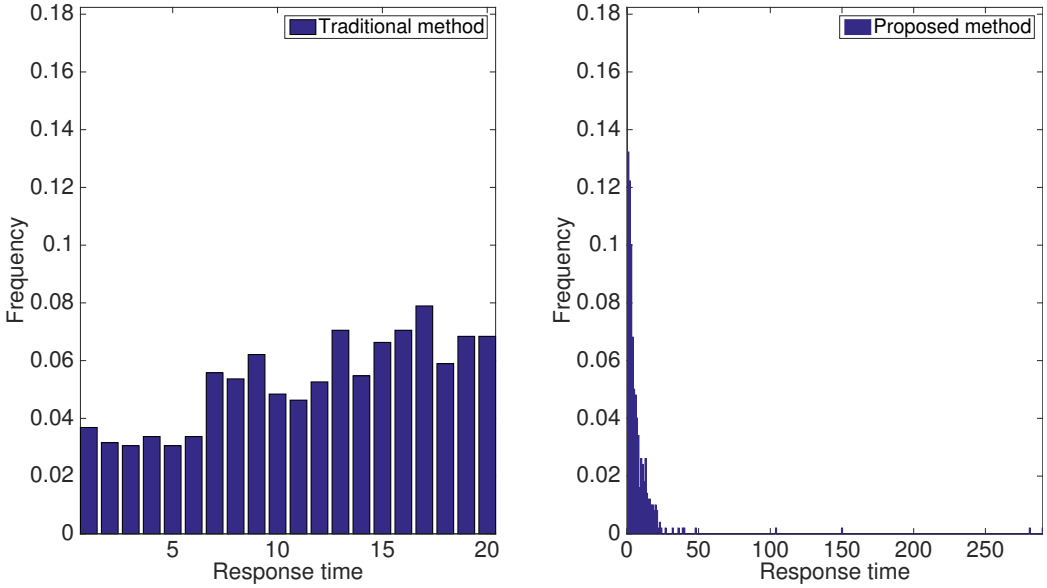


Figure 8.6: The left panel shows the histogram of the response time of 1000 runs of the traditional saliency map using inhibition of return. The right panel shows the output of the proposed system. The traditional saliency map algorithm took 11.9 epochs on an average to detect surprise whereas our proposed method took 6.9 epochs on average.

8.2 Chapter Discussion

This chapter contributes a novel surprise detector that fits along with the Kalman filter based epistemic target selector for detecting sudden event in the visual scene that could not be captured by the Kalman filter.

Although this utility was shown to operate in conjunction with the sum of uncertainty reduction utility function, it is not specific to the sum of uncertainty utility and can be used in conjunction with other utility functions.

In this chapter, the assumption of a one-dimensional visual world was intended for the purpose of demonstration. The proposed surprise de-

tection method is not limited to one dimension as any 2D videos can be rearranged into 1D and this method can be applied.

Chapter 9

Conclusion and Future Work

9.1 Discussions

This thesis focuses on saliency map based visual sampling in dynamic scenarios. The overall goal was to extend the traditional saliency map model to include an internal world model.

An internal model was used for utility based sampling of the visual scene. For example, states of uncertainty are used to guide sampling of a visual scene as uncertainty in an internal model represents its confidence in itself. Once a visual region is sampled its uncertainty goes down due to the addition of new measurement and visual attention is distributed to other locations with higher uncertainty. In general different visual scene sampling behaviours were achieved by designing a variety of appropriate utility functions.

Two supporting algorithms were also presented for learning process noise from observations. This allows the proposed system to adapt to a dynamic visual scene.

The traditional approach towards visual scene sampling is to find a suitable combination of visual features to compute a saliency map and observe the visual scene in descending order of the saliency map. This approach needs an additional mechanism called the IOR to prevent fixation.

IOR has the drawback of blocking previously observed visual regions. An IOR based mechanism is inappropriate for a dynamic visual scene as it cannot observe visual changes in the blocked region.

This work presented the first Kalman filter aided approach that allows preventing fixation without using the traditional IOR mechanism. More importantly, the novel mechanism proposed a conceptual framework that allows further development of utility functions and learning from visual observations.

The presented thesis which proposed the Kalman filter aided visual scene sampling approach achieved the following:

1. The first objective was to use a Bayesian framework to include uncertainty in an agent's decision making. Such a system should behaviourally inhibit previously observed locations, achieve different inhibition time for different parts of the scene and predict utility of a future observation. A novel Kalman filter aided epistemic visual target selector was presented in chapter 3. It was shown in chapter 4 using equations in section 4.2 and attention distribution plots that the proposed method could behaviourally achieve inhibition of previously observed locations, achieve different inhibition time for different parts of the scene and could act based on predicted utility of a future observation.

2. The second objective was to design a set of utility functions using which a variety of visual scene sampling behaviour could be produced. This objective was achieved in the chapters 4, 6 and 7. These three chapters proposed four novel utility functions that could be used to distribute visual attention in a dynamic scene. Varied visual scene sampling behaviours were displayed using visual attention distribution plots in those chapters.

3. The third objective was to use statistical methods to learn the variance of pixels from observations. Two novel estimators were pro-

posed in chapter 5. It was shown using the estimation error plots in that chapter that the two proposed variance learners could estimate the statistical variance of pixels form observations with varying measurement noise.

4. The fourth objective was to detect a sudden event, which could not be detected by the epistemic target selector. This was achieved in chapter 8 by adding a novel statistical distance based metric between a new observation and the prediction from the epistemic visual target selector. The proposed distance based surprise detector's ability was presented using histograms of its response-time in detecting a surprising event. Also, it was shown that the proposed system can detect surprising events in a visual scene faster than the traditional Itti model.

5. In addition, it was also shown in chapter 6 that, the proposed system improves human fixation prediction by replacing the traditional winner-take-all and inhibition of return mechanisms with the novel Kalman filter aided epistemic target selector as assessed by standard measurement metrics.

The proposed system along with the variance learning algorithm could be used in various engineering applications. The proposed model is not dependent on any specific feature detector although, the Itti bottom-up model and pixel intensity were used as feature detectors in this work. An application engineer could implement the proposed system on top of a suitable feature detector that he thinks useful. In such case he does not need to worry about choosing an inhibition of return blocking previously observed locations, inhibition time or inhibition shape. He needs to choose a suitable utility function to express what the agent cares about. He also

For example, an online pedestrian detection task would require 'prioritisation of high salience targets' as the utility function and a square foveation profile. The proposed saliency based algorithm can be utilised to reduce the search space whereas the canonical pedestrian detectors perform a brute force search of desired features in a visual scene (e.g. Viola-Jones algorithm [189]). This should result in reduction in pedestrian detection time.

Another application area that could benefit from the proposed method is video conferencing systems. Usually these systems use evenly spaced grid to reduce video resolution when transmitting. This often results in a very poor perceived video quality, especially when used over slow networks. The proposed mechanism can be used to selectively encode areas perceived as important by humans in high resolution. This would reduce the data transmission burden without compromising on the video quality experience. In this application, the uncertainty reduction utility along with the concave foveation profile could be used.

9.2 Future Work

Many different adaptations, tests, and experiments can be conducted based on the presented work. Future work should concern deeper analysis of particular mechanisms, new proposals to try different methods, or simply curiosity. Here is a list of ideas to be tried out as future work.

1. The presented work assumes a non changing world model for simplicity in presentation. However, a future work could involve more general Kalman filter equations to accommodate for complicated motions and different state transition matrices. This could be useful for applications such as tracking, motion stabilized camera systems.

   An internal model that describes motions across a visual scene (e.g. motion of a moving ball) can be used to distribute visual attention

amongst tracking multiple objects in the scene with limited processing. Such an agent will distribute more visual attention to an object whose motion it cannot describe with its internal model compared to an object with predictable motion path. This behaviour can be seen as an extension of the uncertainty based visual seen sampling strategy proposed in this work.

This work assumed that the entire visual scene has similar state transition model. In future it can be extended to describe different types of motions for different sections of a scene.

Also, it would be interesting to learn the state transition matrix from observations.

2. This work considered every pixel of a visual scene to be independent. Often those pixels can be grouped based on colour or similarities in temporal statistics, and observing any pixel of that group gives information about rest of the pixels in the group. In such cases, an input image could be segmented into super-pixels (groups of similar pixels) [114]. Hence the total number of internal states can be heavily reduced to a more manageable number.

   An immediate consequence is that an observation of any pixel in a group would give information about other pixels in that group. This would reduce the total number of visual targets in the scene and an agent should be able to explore a visual scene quicker.

   Usually, predictable regions (for example: wall, furniture, buildings etc.) take up the most space in a setting and medium and unpredictable regions take comparatively less space. Predictable objects have the most distinctive temporal characteristic and are easy to separate from the rest of the visual scene based on other features like colour. Therefore predictable regions are easier to group together and offers a big reduction in a total number of internal states.

As an introductory idea, the variance learning algorithm presented in chapter 5 can be modified to learn the covariance between different visual regions. This covariance information can be further used for perceptual grouping.

However, it can be challenging to find the exact size and shape of a visual region that can be grouped into one. A predefined set of features for example colour, texture, the temporal behaviour is necessary to be used as the criteria for grouping. This grouping of pixels is an important future direction to study.

3. This work presented four utility functions. Many other utility functions could be explored in the future. As an immediate example, two further visual scene sampling strategies based on the expected improvement can be thought of as an extension of the utility function presented in chapter 7. The agent could follow one of the following strategies: (i) give more importance to features values above a fixed threshold, and (ii) give more importance to feature values above the best known feature value. The first strategy should generate a behaviour where the agent would look at visual locations that have high intensity of the desired features. The second strategy would generate a behaviour where the agent would be looking at the visual location with the highest intensity of feature.

   It would also be interesting to work out what utility function can achieve what real-life robot tasks.

4. Another important future work could be to study the Kalman filter aided system's behaviour with a non Gaussian error assumption. As the Kalman filter non optimally works with non Gaussian error assumptions [6], it would be interesting to study the resulting visual attention distribution. Specially, it would be important to study how often the surprise signal is triggered.

5. The Kalman filter assumes that the immediate past random variable can sufficiently describe the future. Hence the filter keeps track of one prior state. Other richer algorithms like the Gaussian process can predict future based on all the past states of an agent. Including all past states is computationally more expensive, but can express more complicated temporal behaviours. A promising future work is to extend the presented Kalman filter based work to use a Gaussian process based reasoning algorithm.

6. The proposed work assumed that each visual scene part is observed at every time step with varying measurement noises. In practice, only a smaller part of the entire visual scene is observed at any time step. For example, when someone is looking towards the front, he is not watching part of the visual scene behind him.

   In such a case the unobserved visual states should be updated in time based on the internal model of the agent. The proposed method can be extended in future for such scenarios.

   In the case of partial observation of a visual scene, any given part of the scene is observed irregularly. Due to lack of regular observations, the recursive method of estimation process noise is not suitable. The MLE based batch processing must be adapted to deal with datasets consisting of observations at irregular time intervals.

7. For simplicity of initial experiments the proposed visual scene sampling scheme was demonstrated with pixel intensity and Itti based saliency. The proposed method is not specific to any choice of a feature representing the visual scene, hence it is not limited to any particular bottom-up model. Future attempts could be made to study the proposed systems behaviour in conjunction with other prominent bottom-up models in the Saliency field.

8. It was assumed that a single Gaussian distribution faithfully repre-

sents a real world state. It might be possible that a single Gaussian distribution is not a good model of the real world states of a video or a visual scene and a mixture of Gaussian distributions is a suitable approach for real life modelling of visual scenes. Therefore using a Gaussian mixture model would be an important future direction to study. It is known that given enough Gaussian distributions per world state, any world state can be modelled. Causes that influences the world states can also be accurately modelled with the mixture of Gaussian, where each cause is represented as one Gaussian distribution. Although it is accurate, having multiple Gaussian distributions representing one state has the drawback of being memory extensive and computationally demanding. Therefore the total number of states that can be practically tracked by an agent is limited.

Appendix A

Itti bottom-up Saliency Map Parameters

This chapter discusses the Itti saliency model's default parameters. The Itti saliency map is given every frame of the video dataset and it produces the corresponding saliency maps for each frames.

A.1 Spatial Sampling in Itti Saliency Generation

The saliency map computed by the Itti model has smaller pixel count than the input image. This is because, that the saliency is computed by the sum of differences in features in centre-surround windows. Hence the group of pixels coming under a centre surround window gets represented by one pixel in the final saliency map. As an outcome the saliency map has less number of pixels and each pixel in the saliency map corresponds to a region in the input image. The mapping between a salient pixel and the cosponsoring image coordinates is defined by the choice of feature detectors and how they are interconnected.

Parameters such as the number of feature detector windows and how they are interconnected can be chosen in the Itti model. Hence, the saliency map to image coordinate mapping depends on the choice of these param-

eters.

The toolbox provided by the creators of the Itti model allows the user to chose amongst a set of predefined set of model parameters or can custom define parameters. Amongst the predefined sets, there is a special set, which was hand-crafted by the designers of the Itti model called the 'default' parameters set. Table A.1 lists out some of the parameters in the default set. The entire list is not presented here as it is long and it can be found in the saliency toolbox. The 'default' parameters were chosen for all the experiments that used the Itti model.

Table A.1: Selected default saliency parameters of the Itti model.

pyramidType	'dyadic'
features	{'Color','Intensities','Orientations'}
weights	[1 1 1]
IORtype	'shape'
shapeMode	'shapeFM'
levelParams	[1×1 struct]
IORdecay	0.9999
gaborParams	[1×1 struct]
oriAngles	[0 45 90 135]

The mapping between the saliency map the input image was fixed for all the experiments as the same parameters (the default parameters set) were used. The saliency tool box also comes with a built-in function to compute this mapping between the saliency location and the corresponding image coordinates.

Due to this down-sampling in image size the epistemic target selector operates on a much smaller number of pixels than the total pixels in the input image. Similarly the epistemic target selector does not need to operate on each pixel of the initial input videos. Therefore, all the initial testing videos were downsampled by a Itti saliency map to input image like ratio.

The pyramidType parameter in table A.1 defines the interconnection between the feature detectors. In this example, the default 'dyadic' means that the model will create a Gaussian pyramid by blurring and subsampling a map by a factor of two repeatedly, as long as both image-width and image-height are larger than one. Notice how the inhibition of return decay time, called the IORdecay, has been hard coded into the set of parameters. In contrast this work would attempt to learn this parameter form observations.

Bibliography

[1] ACHANTA, R., AND SÜSSTRUNK, S. Saliency detection for content-aware image resizing. In *Image Processing (ICIP), 2009 16th IEEE International Conference on* (2009), IEEE, pp. 1005–1008.

[2] ÅKESSON, B. M., JØRGENSEN, J. B., POULSEN, N. K., AND JØRGENSEN, S. B. A generalized autocovariance least-squares method for Kalman filter tuning. *Journal of Process control 18*, 7 (2008), 769–779.

[3] ALEXE, B., DESELAERS, T., AND FERRARI, V. What is an object? In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), IEEE, pp. 73–80.

[4] ALSPACH, D. A parallel filtering algorithm for linear systems with unknown time varying noise statistics. *Automatic Control, IEEE Transactions on 19*, 5 (1974), 552–556.

[5] ANDRIEU, Q., DEHAIS, F., IZAUTE, A., LESIRE, C., AND TESSIER, C. Towards a dynamic computational model of visual attention. In *International Conference on Humans Operating Unmanned Systems (HUMOUS?08), Brest, France* (2008).

[6] ARULAMPALAM, M. S., MASKELL, S., GORDON, N., AND CLAPP, T. A tutorial on particle filters for online nonlinear/non-Gaussian bayesian tracking. *Signal Processing, IEEE Transactions on 50*, 2 (2002), 174–188.

[7] ATCHISON, D. A., SMITH, G., AND SMITH, G. Optics of the human eye.

[8] AWAD, D., COURBOULAY, V., AND REVEL, A. Saliency filtering of sift detectors: Application to cbir. In *International Conference on Advanced Concepts for Intelligent Vision Systems* (2012), Springer, pp. 290–300.

[9] BAI, X., FANG, Y., LIN, W., WANG, L., AND JU, B.-F. Saliency-based defect detection in industrial images by using phase spectrum. *IEEE Transactions on Industrial Informatics 10*, 4 (2014), 2135–2145.

[10] BAKHTARI, A., AND BENHABIB, B. An active vision system for multitarget surveillance in dynamic environments. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 37*, 1 (2007), 190–198.

[11] BAN, Z., LIU, J., AND CAO, L. Superpixel segmentation using gaussian mixture model. *IEEE Transactions on Image Processing 27*, 8 (2018), 4105–4117.

[12] BAVDEKAR, V. A., DESHPANDE, A. P., AND PATWARDHAN, S. C. Identification of process and measurement noise covariance for state and parameter estimation using extended Kalman filter. *Journal of Process control 21*, 4 (2011), 585–601.

[13] BAYLIS, G. C., AND DRIVER, J. Visual attention and objects: evidence for hierarchical coding of location. *Journal of Experimental Psychology: Human Perception and Performance 19*, 3 (1993), 451–470.

[14] BELARDINELLI, A. *Salience features selection: Deriving a model from human evidence*. PhD thesis, Ph. D. thesis, Sapienza Universita di Roma, Rome, Italy, 2008.

[15] BJÖRKMAN, M., AND KRAGIC, D. Active 3d scene segmentation and detection of unknown objects. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (2010), IEEE, pp. 3114–3120.

[16] BLASER, E., PYLYSHYN, Z. W., AND OHLCOMBE, A. O. Tracking an object through feature space. *Nature 408*, 6809 (2000), 196–199.

[17] BOIMAN, O., AND IRANI, M. Detecting irregularities in images and in video. *International journal of computer vision 74*, 1 (2007), 17–31.

[18] BOLLEN, B. What should the value of lambda be in the exponentially weighted moving average volatility model? *Applied Economics 47*, 8 (2015), 853–860.

[19] BOLLMANN, M., HOISCHEN, R., JESIKIEWICZ, M., JUSTKOWSKI, C., AND MERTSCHING, B. Playing domino: A case study for an active vision system. *Computer Vision Systems* (1999), 392–411.

[20] BOLLMANN, M., HOISCHEN, R., AND MERTSCHING, B. Integration of static and dynamic scene features guiding visual attention. *Mustererkennung 19* (1997), 483–490.

[21] BONNIN-PASCUAL, F., AND ORTIZ, A. A probabilistic approach for defect detection based on saliency mechanisms. In *Emerging Technology and Factory Automation (ETFA), 2014 IEEE* (2014), IEEE, pp. 1–4.

[22] BORJI, A., AHMADABADI, M. N., ARAABI, B. N., AND HAMIDI, M. Online learning of task-driven object-based visual attention control. *Image and Vision Computing 28*, 7 (2010), 1130–1145.

[23] BORJI, A., SIHITE, D. N., AND ITTI, L. Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study. *IEEE Transactions on Image Processing 22*, 1 (2013), 55–69.

[24] BORJI, A., TAVAKOLI, H. R., SIHITE, D. N., AND ITTI, L. Analysis of scores, datasets, and models in visual saliency prediction. In *Proceedings of the IEEE international conference on computer vision* (2013), pp. 921–928.

[25] BREAZEAL, C., EDSINGER, A., FITZPATRICK, P., AND SCASSELLATI, B. Active vision for sociable robots. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on 31*, 5 (2001), 443–453.

[26] BULUT, Y., VINES-CAVANAUGH, D., AND BERNAL, D. Process and measurement noise estimation for Kalman filtering. In *Structural Dynamics, Volume 3*. Springer, 2011, pp. 375–386.

[27] BYLINSKII, Z., JUDD, T., DURAND, F., OLIVA, A., AND TORRALBA, A. MIT saliency benchmark. http://saliency.mit.edu/.

[28] BYLINSKII, Z., JUDD, T., OLIVA, A., TORRALBA, A., AND DURAND, F. What do different evaluation metrics tell us about saliency models? *arXiv preprint arXiv:1604.03605* (2016).

[29] CAMPILLO, F., AND MEVEL, L. Recursive maximum likelihood estimation for structural health monitoring: tangent filter implementations. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on* (2005), IEEE, pp. 5923–5928.

[30] CASTELLÓ, P., CHOVER, M., SBERT, M., AND FEIXAS, M. Reducing complexity in polygonal meshes with view-based saliency. *Computer Aided Geometric Design 31*, 6 (2014), 279–293.

[31] CHEN, Q., CHEN, Z., GU, X., AND WANG, C. Attention-based adaptive intra refresh for error-prone video transmission. *Communications Magazine, IEEE 45*, 1 (2007), 52–60.

[32] CIOCCA, G., CUSANO, C., GASPARINI, F., AND SCHETTINI, R. Self-adaptive image cropping for small displays. *IEEE Transactions on Consumer Electronics 53*, 4 (2007).

[33] CLARK, J. J., AND FERRIER, N. J. Modal control of an attentive vision system. In *ICCV* (1988), pp. 514–523.

[34] CONNOR, C. E. Active vision and visual activation in area v4. *Neuron 40*, 6 (2003), 1056.

[35] CONNOR, C. E., EGETH, H. E., AND YANTIS, S. Visual attention: bottom-up versus top-down. *Current Biology 14*, 19 (2004), R850–R852.

[36] CORBETTA, M., AND SHULMAN, G. L. Control of goal-directed and stimulus-driven attention in the brain. *Nature reviews neuroscience 3*, 3 (2002), 201–215.

[37] COURTY, N., AND MARCHAND, E. Visual perception based on salient features. In *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on* (Oct 2003), vol. 1, pp. 1024–1029 vol.1.

[38] COUTROT, A., AND GUYADER, N. How saliency, faces, and sound influence gaze in dynamic social scenes. *Journal of Vision 14*, 8 (2014), 5.

[39] COUVREUR, L., BETTENS, F., HANCQ, J., AND MANCAS, M. Normalized auditory attention levels for automatic audio surveillance. *WIT Transactions on Information and Communication Technologies 39* (2008), 453–462.

[40] CULIBRK, D., MIRKOVIC, M., LUGONJA, P., AND CRNOJEVIC, V. Mining web videos for video quality assessment. In *Soft Computing and Pattern Recognition (SoCPaR), 2010 International Conference of* (2010), IEEE, pp. 75–80.

[41] CULIBRK, D., MIRKOVIC, M., ZLOKOLICA, V., POKRIC, M., CRNOJEVIC, V., AND KUKOLJ, D. Salient motion features for video quality assessment. *IEEE Transactions on Image Processing 20*, 4 (2011), 948–958.

[42] DAVISON, A. J., AND MURRAY, D. W. Simultaneous localization and map-building using active vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 24*, 7 (2002), 865–880.

[43] DAWKINS, M. S., AND WOODINGTON, A. Pattern recognition and active vision in chickens. *Nature 403*, 6770 (2000), 652–655.

[44] DENG, L., AND SHEN, X. Maximum likelihood in statistical estimation of dynamic systems: Decomposition algorithm and simulation results. *Signal Processing 57*, 1 (1997), 65–79.

[45] DHAVALE, N., AND ITTI, L. Saliency-based multifoveated MPEG compression. In *Signal processing and its applications, 2003. Proceedings. Seventh international symposium on* (2003), vol. 1, IEEE, pp. 229–232.

[46] DOMINEY, P. F., AND ARBIB, M. A. A cortico-subcortical model for generation of spatially accurate sequential saccades. *Cerebral cortex 2*, 2 (1992), 153–175.

[47] DONG, D. W., AND ATICK, J. J. Statistics of natural time-varying images. *Network: Computation in Neural Systems 6*, 3 (1995), 345–358.

[48] DUNIK, J., AND ŠIMANDL, M. Estimation of state and measurement noise covariance matrices by multi-step prediction. In *Proceedings of the 17th IFAC World Congress* (2008), pp. 3689–3694.

[49] DURRANT-WHYTE, H., AND BAILEY, T. Simultaneous localization and mapping (SLAM): Part i the essential algorithms. robotics

and automation magazine 13 (2): 99–110. doi: 10.1109/mra.2006.1638022. Tech. rep., Retrieved 2008-04-08, 2006.

[50] ENDRES, D., NEUMANN, H., KOLESNIK, M., AND GIESE, M. Hooligan detection: the effects of saliency and expert knowledge. In *Imaging for Crime Detection and Prevention 2011 (ICDP 2011), 4th International Conference on* (2011), IET, pp. 1–6.

[51] ENESCU, M., SIRBU, M., AND KOIVUNEN, V. Recursive estimation of noise statistics in Kalman filter based MIMO equalization. *the proceedings of XXVIIth General Assembly of the International Union of Radio Science (URSI), Maastricht the Netherlands* (2002), 17–24.

[52] ENGELHARD, N., ENDRES, F., HESS, J., STURM, J., AND BURGARD, W. Real-time 3d visual SLAM with a hand-held rgb-d camera. In *Proc. of the RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum, Vasteras, Sweden* (2011), vol. 180, pp. 1–15.

[53] ENGELKE, U., KAPRYKOWSKY, H., ZEPERNICK, H.-J., AND NDJIKI-NYA, P. Visual attention in quality assessment. *IEEE Signal Processing Magazine 28*, 6 (2011), 50–59.

[54] FAWCETT, T. An introduction to ROC analysis. *Pattern recognition letters 27*, 8 (2006), 861–874.

[55] FENG, B., FU, M., MA, H., XIA, Y., AND WANG, B. Kalman filter with recursive covariance estimationsequentially estimating process noise covariance. *Industrial Electronics, IEEE Transactions on 61*, 11 (2014), 6253–6263.

[56] FERREIRA, J. F., AND DIAS, J. Attentional mechanisms for socially interactive robots–a survey. *IEEE Transactions on Autonomous Mental Development 6*, 2 (2014), 110–125.

[57] FERZLI, R., AND KARAM, L. J. A no-reference objective image sharpness metric based on just-noticeable blur and probability summation. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on* (2007), vol. 3, IEEE, pp. III–445.

[58] FINDLAY, J. M., AND WALKER, R. A model of saccade generation based on parallel processing and competitive inhibition. *Behavioral and Brain Sciences 22*, 04 (1999), 661–674.

[59] FORRESTER, A., SOBESTER, A., AND KEANE, A. *Engineering Design via Surrogate Modelling: A Practical Guide*. Wiley, 2008.

[60] FRIEDMAN, N., AND RUSSELL, S. Image segmentation in video sequences: A probabilistic approach. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence* (1997), Morgan Kaufmann Publishers Inc., pp. 175–181.

[61] FRINTROP, S. The high repeatability of salient regions. In *Workshop on Vision in Action: Efficient strategies for cognitive agents in complex environments* (2008).

[62] FRINTROP, S. Towards attentive robots. *Paladyn 2*, 2 (2011), 64–70.

[63] FRINTROP, S., BACKER, G., AND ROME, E. Goal-directed search with a top-down modulated computational attention system. In *DAGM-Symposium* (2005), vol. 3663, Springer, pp. 117–124.

[64] FRINTROP, S., GARCÍA, G. M., AND CREMERS, A. B. A cognitive approach for object discovery. In *Pattern Recognition (ICPR), 2014 22nd International Conference on* (2014), IEEE, pp. 2329–2334.

[65] FRINTROP, S., AND JENSFELT, P. Attentional landmarks and active gaze control for visual SLAM. *Robotics, IEEE Transactions on 24*, 5 (2008), 1054–1065.

[66] FURRER, R., AND BENGTSSON, T. Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *Journal of Multivariate Analysis 98*, 2 (2007), 227–255.

[67] FURUKAWA, S. Y. T. Neural systems of vision. In *Advances in Neural Information Processing Systems 8: Proceedings of the 1995 Conference* (1996), vol. 8, MIT Press, p. 159.

[68] GARCÍA, G. M., POTAPOVA, E., WERNER, T., ZILLICH, M., VINCZE, M., AND FRINTROP, S. Saliency-based object discovery on rgb-d data with a late-fusion approach. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on* (2015), IEEE, pp. 1866–1873.

[69] GAZZANIGA, M. S. *The cognitive neurosciences*. MIT press, 2004.

[70] GILLESPIE, W., AND NGUYEN, T. Classification of video sequences in MPEG domain. In *Signal Processing for Telecommunications and Multimedia*. Springer, 2005, pp. 71–86.

[71] GOFERMAN, S., ZELNIK-MANOR, L., AND TAL, A. Context-aware saliency detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence 34*, 10 (2012), 1915–1926.

[72] GOULD, S., ARFVIDSSON, J., KAEHLER, A., SAPP, B., MESSNER, M., BRADSKI, G. R., BAUMSTARCK, P., CHUNG, S., NG, A. Y., ET AL. Peripheral-foveal vision for real-time object recognition and tracking in video. In *IJCAI* (2007), vol. 7, pp. 2115–2121.

[73] GRUNDMANN, M., AND KWATRA, V. Methods and systems for video retargeting using motion saliency, Oct. 20 2015. US Patent 9,167,221.

[74] GRUNDMANN, M., KWATRA, V., HAN, M., AND ESSA, I. Discontinuous seam-carving for video retargeting. In *Computer Vision and*

Pattern Recognition (CVPR), 2010 IEEE Conference on (2010), IEEE, pp. 569–576.

[75] GUO, C., MA, Q., AND ZHANG, L. Spatio-temporal saliency detection using phase spectrum of quaternion Fourier transform. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (2008), IEEE, pp. 1–8.

[76] GUO, C., AND ZHANG, L. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. *Image Processing, IEEE Transactions on 19*, 1 (2010), 185–198.

[77] GUPTA, R., AND CHAUDHURY, S. A scheme for attentional video compression. In *International Conference on Pattern Recognition and Machine Intelligence* (2011), Springer, pp. 458–465.

[78] HAREL, J., KOCH, C., AND PERONA, P. Graph-based visual saliency. In *Advances in neural information processing systems* (2007), pp. 545–552.

[79] HAYHOE, M. M. Advances in relating eye movements and cognition. *Infancy 6*, 2 (2004), 267–274.

[80] HEIDEMANN, G., RAE, R., BEKEL, H., BAX, I., AND RITTER, H. Integrating context-free and context-dependent attentional mechanisms for gestural object reference. *Computer Vision Systems* (2003), 22–33.

[81] HILBORN JR, C., AND LAINIOTIS, D. G. Optimal estimation in the presence of unknown parameters. *Systems Science and Cybernetics, IEEE Transactions on 5*, 1 (1969), 38–43.

[82] HORBERT, E., GARCÍA, G. M., FRINTROP, S., AND LEIBE, B. Sequence-level object candidates based on saliency for generic object

recognition on mobile systems. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on* (2015), IEEE, pp. 127–134.

[83] HOSANG, J., BENENSON, R., DOLLÁR, P., AND SCHIELE, B. What makes for effective detection proposals? *IEEE transactions on pattern analysis and machine intelligence 38*, 4 (2016), 814–830.

[84] HÜBNER, R., AND BACKER, G. Perceiving spatially inseparable objects: Evidence for feature-based object selection not mediated by location. *Journal of Experimental Psychology: Human Perception and Performance 25*, 6 (1999), 1556.

[85] HUELSE, M., MCBRIDE, S., AND LEE, M. Implementing inhibition of return: embodied visual memory for robotic systems.

[86] HUNTER, R. S. Photoelectric color difference meter. *J. Opt. Soc. Am. 48*, 12 (Dec 1958), 985–995.

[87] ITTI, L. Automatic foveation for video compression using a neurobiological model of visual attention. *Image Processing, IEEE Transactions on 13*, 10 (2004), 1304–1318.

[88] ITTI, L., AND BALDI, P. A principled approach to detecting surprising events in video. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (2005), vol. 1, IEEE, pp. 631–637.

[89] ITTI, L., AND KOCH, C. Comparison of feature combination strategies for saliency-based visual attention systems. In *Electronic Imaging'99* (1999), International Society for Optics and Photonics, pp. 473–482.

[90] ITTI, L., AND KOCH, C. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision research 40*, 10 (2000), 1489–1506.

[91] ITTI, L., AND KOCH, C. Computational modelling of visual attention. *Nature reviews neuroscience 2*, 3 (2001), 194–203.

[92] ITTI, L., AND KOCH, C. Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging 10*, 1 (2001), 161–169.

[93] ITTI, L., KOCH, C., AND NIEBUR, E. A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 20*, 11 (1998), 1254–1259.

[94] ITTI, LAURENT; CARMI, R. Eye-tracking data from human volunteers watching complex video stimuli. crcns.org., 2009.

[95] JAMES, W. *The principles of psychology*. New York: H. Holt and Company, 1890.

[96] JIANG, H., WANG, J., YUAN, Z., WU, Y., ZHENG, N., AND LI, S. Salient object detection: A discriminative regional feature integration approach. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (2013), IEEE, pp. 2083–2090.

[97] JOHNSON-ROBERSON, M., BOHG, J., BJÖRKMAN, M., AND KRAGIC, D. Attention-based active 3d point cloud segmentation. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on* (2010), IEEE, pp. 1165–1170.

[98] JONES, D. R., SCHONLAU, M., AND WELCH, W. J. Efficient global optimization of expensive black-box functions. *Journal of Global optimization 13*, 4 (1998), 455–492.

[99] KALMAN, R. E. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering 82*, 1 (1960), 35–45.

[100] KALMAN, R. E. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering 82*, Series D (1960), 35–45.

[101] KIRENKO, I. O. Reduction of coding artifacts using chrominance and luminance spatial analysis. In *Consumer Electronics, 2006. ICCE'06. 2006 Digest of Technical Papers. International Conference on* (2006), IEEE, pp. 209–210.

[102] KLEIN, R. M., AND MACINNES, W. J. Inhibition of return is a foraging facilitator in visual search. *Psychological science 10*, 4 (1999), 346–352.

[103] KOCH, C., AND ULLMAN, S. Shifts in selective visual attention: towards the underlying neural circuitry. *Hum Neurobiol 4*, 4 (1985), 219–27.

[104] KÖRDING, K. P., AND WOLPERT, D. M. Bayesian decision theory in sensorimotor control. *Trends in cognitive sciences 10*, 7 (2006), 319–326.

[105] KÜMMERER, M., WALLIS, T. S., AND BETHGE, M. Information-theoretic model comparison unifies saliency metrics. *Proceedings of the National Academy of Sciences 112*, 52 (2015), 16054–16059.

[106] LE MEUR, O., LE CALLET, P., AND BARBA, D. Construction dimages miniatures avec recadrage automatique basée sur un modele perceptuel bio-inspiré.

[107] LE MEUR, O., LE CALLET, P., BARBA, D., ET AL. A spatio-temporal model to predict visual fixation: description and assessment. *Vision research 47*, 19 (2007), 2483–2498.

[108] LE MEUR, O., LE CALLET, P., BARBA, D., AND THOREAU, D. A coherent computational approach to model bottom-up visual atten-

tion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 28*, 5 (2006), 802–817.

[109] LE MEUR, O., THOREAU, D., LE CALLET, P., AND BARBA, D. A spatio-temporal model of the selective human visual attention. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on* (2005), vol. 3, IEEE, pp. III–1188.

[110] LEE, C. H., VARSHNEY, A., AND JACOBS, D. W. Mesh saliency. In *ACM transactions on graphics (TOG)* (2005), vol. 24, ACM, pp. 659–666.

[111] LEVY, B. C., BENVENISTE, A., AND NIKOUKHAH, R. High-level primitives for recursive maximum likelihood estimation. *Automatic Control, IEEE Transactions on 41*, 8 (1996), 1125–1145.

[112] LI, J., XIA, C., SONG, Y., FANG, S., AND CHEN, X. A data-driven metric for comprehensive evaluation of saliency models. In *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 190–198.

[113] LI, Z., QIN, S., AND ITTI, L. Visual attention guided bit allocation in video compression. *Image and Vision Computing 29*, 1 (2011), 1–14.

[114] LI, Z., WU, X.-M., AND CHANG, S.-F. Segmentation using superpixels: A bipartite graph partitioning approach. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), IEEE, pp. 789–796.

[115] LIU, H., JIANG, S., HUANG, Q., XU, C., AND GAO, W. Region-based visual attention analysis with its application in image browsing on small displays. In *Proceedings of the 15th international conference on Multimedia* (2007), ACM, pp. 305–308.

[116] LIU, T., FENG, X., REIBMAN, A., AND WANG, Y. Saliency inspired modeling of packet-loss visibility in decoded videos. In *International Workshop VPQM* (2009), pp. 1–4.

[117] LIU, Z., YAN, H., SHEN, L., WANG, Y., AND ZHANG, Z. A motion attention model based rate control algorithm for h. 264/avc. In *Computer and Information Science, 2009. ICIS 2009. Eighth IEEE/ACIS International Conference on* (2009), IEEE, pp. 568–573.

[118] LOACH, D., FRISCHEN, A., BRUCE, N., AND TSOTSOS, J. K. An attentional mechanism for selecting appropriate actions afforded by graspable objects. *Psychological Science 19*, 12 (2008), 1253–1257.

[119] LONGFEI, Z., YUANDA, C., GANGYI, D., AND YONG, W. A computable visual attention model for video skimming. In *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on* (2008), IEEE, pp. 667–672.

[120] LONGHURST, P., DEBATTISTA, K., AND CHALMERS, A. A gpu based saliency map for high-fidelity selective rendering. In *Proceedings of the 4th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa* (2006), ACM, pp. 21–29.

[121] MA, Y., HUA, X., LU, L., AND ZHANG, H. A generic framework of user attention model and its application in video summarization. *Multimedia, IEEE Transactions on 7*, 5 (2005), 907–919.

[122] MA, Y., LU, L., ZHANG, H., AND LI, M. A user attention model for video summarization. In *Proceedings of the tenth ACM international conference on Multimedia* (2002), ACM, pp. 533–542.

[123] MAHADEVAN, V., AND VASCONCELOS, N. Spatiotemporal saliency in dynamic scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 32*, 1 (2010), 171–177.

[124] MAHALANOBIS, P. C. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta) 2* (1936), 49–55.

[125] MANCAS, M., COUVREUR, L., GOSSELIN, B., MACQ, B., ET AL. Computational attention for event detection. In *Proc. Fifth Intl Conf. Computer Vision Systems* (2007).

[126] MARAT, S., HO PHUOC, T., GRANJON, L., GUYADER, N., PELLERIN, D., AND GUÉRIN-DUGUÉ, A. Modelling spatio-temporal saliency to predict gaze direction for short videos. *International Journal of Computer Vision 82*, 3 (2009), 231–243.

[127] MARQUES, O., MAYRON, L. M., BORBA, G. B., AND GAMBA, H. R. An attention-driven model for grouping similar images with image retrieval applications. *EURASIP Journal on Applied Signal Processing 2007*, 1 (2007), 116–116.

[128] MASAKI, I. Vision-based vehicle guidance. In *Industrial Electronics, Control, Instrumentation, and Automation, 1992. Power Electronics and Motion Control., Proceedings of the 1992 International Conference on* (1992), IEEE, pp. 862–867.

[129] MATEESCU, V. A., HADIZADEH, H., AND BAJIĆ, I. V. Evaluation of several visual saliency models in terms of gaze prediction accuracy on video. In *Globecom Workshops (GC Wkshps), 2012 IEEE* (2012), IEEE, pp. 1304–1308.

[130] MEGER, D., FORSSÉN, P.-E., LAI, K., HELMER, S., MCCANN, S., SOUTHEY, T., BAUMANN, M., LITTLE, J. J., AND LOWE, D. G. Curious george: An attentive semantic robot. *Robotics and Autonomous Systems 56*, 6 (2008), 503–511.

[131] MEHRA, R. K. On the identification of variances and adaptive Kalman filtering. *Automatic Control, IEEE Transactions on 15*, 2 (1970), 175–184.

[132] MEHTA, A. D., ULBERT, I., AND SCHROEDER, C. E. Intermodal selective attention in monkeys. i: distribution and timing of effects across visual areas. *Cerebral Cortex 10*, 4 (2000), 343–358.

[133] MISHKIN, M., UNGERLEIDER, L. G., AND MACKO, K. A. Object vision and spatial vision: two cortical pathways. *Trends in neurosciences 6* (1983), 414–417.

[134] MISHNE, G., AND COHEN, I. Multi-channel wafer defect detection using diffusion maps. In *Electrical & Electronics Engineers in Israel (IEEEI), 2014 IEEE 28th Convention of* (2014), IEEE, pp. 1–5.

[135] MYERS, K., TAPLEY, B. D., ET AL. Adaptive sequential estimation with unknown noise statistics. *Automatic Control, IEEE Transactions on 21*, 4 (1976), 520–523.

[136] NAGAI, Y. From bottom-up visual attention to robot action learning. In *Development and Learning, 2009. ICDL 2009. IEEE 8th International Conference on* (2009), IEEE, pp. 1–6.

[137] NAVALPAKKAM, V., AND ITTI, L. Modeling the influence of task on attention. *Vision research 45*, 2 (2005), 205–231.

[138] NAVALPAKKAM, V., AND ITTI, L. An integrated model of top-down and bottom-up attention for optimizing detection speed. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* (2006), vol. 2, IEEE, pp. 2049–2056.

[139] NICKERSON, S., JASIOBEDZKI, P., WILKES, D., JENKIN, M., MILIOS, E., TSOTSOS, J., JEPSON, A., AND BAINS, O. The ark project: Autonomous mobile robots for known industrial environments. *Robotics and Autonomous Systems 25*, 1-2 (1998), 83–104.

[140] NORMAN, D. A. Toward a theory of memory and attention. *Psychological review 75*, 6 (1968), 522.

[141] NOTHDURFT, H.-C. Salience from feature contrast: Variations with texture density. *Vision Research 40*, 23 (2000), 3181–3200.

[142] ODELSON, B. J., LUTZ, A., AND RAWLINGS, J. B. The autocovariance least-squares method for estimating covariances: application to model-based control of chemical reactors. *Control Systems Technology, IEEE Transactions on 14*, 3 (2006), 532–540.

[143] ODELSON, B. J., RAJAMANI, M. R., AND RAWLINGS, J. B. A new autocovariance least-squares method for estimating noise covariances. *Automatica 42*, 2 (2006), 303–308.

[144] OLSHAUSEN, B. A., AND FIELD, D. J. Natural image statistics and efficient coding*. *Network: computation in neural systems 7*, 2 (1996), 333–339.

[145] OUERHANI, N., BUR, A., AND HÜGLI, H. Visual attention-based robot self-localization. In *In Proceeding of European Conference on Mobile Robotics* (2005), pp. 8–13.

[146] PARKHURST, D., LAW, K., AND NIEBUR, E. Modeling the role of salience in the allocation of overt visual attention. *Vision research 42*, 1 (2002), 107–123.

[147] PARKHURST, D. J. *Selective attention in natural vision: using computational models to quantify stimulus-driven attentional allocation*. PhD thesis, Johns Hopkins University, Baltimore, Maryland, 2002.

[148] PARKHURST, D. J., AND NIEBUR, E. Scene content selected by active vision. *Spatial vision 16*, 2 (2003), 125–154.

[149] PETERS, R., AND ITTI, L. Applying computational tools to predict gaze direction in interactive visual environments. *ACM Transactions on Applied Perception (TAP) 5*, 2 (2008), 9.

[150] POSNER, M. I., AND COHEN, Y. Components of visual orienting. *Attention and performance X: Control of language processes 32* (1984), 531–556.

[151] POSNER, M. I., RAFAL, R. D., CHOATE, L. S., AND VAUGHAN, J. Inhibition of return: Neural basis and function. *Cognitive neuropsychology 2*, 3 (1985), 211–228.

[152] POSNER, M. I., SNYDER, C. R., AND DAVIDSON, B. J. Attention and the detection of signals. *Journal of experimental psychology: General 109*, 2 (1980), 160.

[153] POTAPOVA, E., VARADARAJAN, K. M., RICHTSFELD, A., ZILLICH, M., AND VINCZE, M. Attention-driven object detection and segmentation of cluttered table scenes using 2.5 d symmetry. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on* (2014), IEEE, pp. 4946–4952.

[154] R. HARTER, M., AND AINE, C. Brain mechanisms of visual selective attention.

[155] RAJASHEKAR, U., CORMACK, L. K., AND BOVIK, A. C. Image features that draw fixations. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on* (2003), vol. 3, IEEE, pp. III–313.

[156] RASOLZADEH, B., BJÖRKMAN, M., HÜBNER, K., AND KRAGIC, D. An active vision system for detecting, fixating and manipulating objects in the real world. *The International Journal of Robotics Research 29*, 2-3 (2010), 133–154.

[157] ROTENSTEIN, A., ANDREOPOULOS, A., FAZL, E., JACOB, D., ROBINSON, M., SHUBINA, K., ZHU, Y., AND TSOTSOS, J. Towards the dream of intelligent, visually-guided wheelchairs. In *Proc. 2nd Intl Conf. on Technology and Aging* (2007), Citeseer.

[158] RUBINSTEIN, M., SHAMIR, A., AND AVIDAN, S. Improved seam carving for video retargeting. In *ACM transactions on graphics (TOG)* (2008), vol. 27, ACM, p. 16.

[159] RUDINAC, M., KOOTSTRA, G., KRAGIC, D., AND JONKER, P. P. Learning and recognition of objects inspired by early cognition. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (2012), IEEE, pp. 4177–4184.

[160] SAMUEL, A. G., AND KAT, D. Inhibition of return: A graphical meta-analysis of its time course and an empirical test of its temporal and spatial properties. *Psychonomic Bulletin & Review 10*, 4 (2003), 897–906.

[161] SANTELLA, A., AGRAWALA, M., DECARLO, D., SALESIN, D., AND COHEN, M. Gaze-based interaction for semi-automatic photo cropping. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (2006), ACM, pp. 771–780.

[162] SCASSELLATI, B. M. *Foundations for a Theory of Mind for a Humanoid Robot*. PhD thesis, Massachusetts Institute of Technology, 2001.

[163] SCHAUERTE, B., AND FINK, G. A. Focusing computational visual attention in multi-modal human-robot interaction. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction* (2010), ACM, p. 6.

[164] SCHAUERTE, B., RICHARZ, J., AND FINK, G. A. Saliency-based identification and recognition of pointed-at objects. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on* (2010), IEEE, pp. 4638–4643.

[165] SCHICK, A., BÄUML, M., AND STIEFELHAGEN, R. Improving foreground segmentations with probabilistic superpixel markov random fields. In *Computer Vision and Pattern Recognition Workshops*

(CVPRW), 2012 IEEE Computer Society Conference on (2012), IEEE, pp. 27–31.

[166] SCHILLACI, G., BODIROŽA, S., AND HAFNER, V. V. Evaluating the effect of saliency detection and attention manipulation in human-robot interaction. *International Journal of Social Robotics 5*, 1 (2013), 139–152.

[167] SERIES, B. Methodology for the subjective assessment of the quality of television pictures. *Recommendation ITU-R BT* (2012), 500–13.

[168] SESHADRINATHAN, K., AND BOVIK, A. C. Automatic prediction of perceptual quality of multimedia signalsa survey. *Multimedia Tools and Applications 51*, 1 (2011), 163–186.

[169] SIAGIAN, C., AND ITTI, L. Rapid biologically-inspired scene classification using features shared with visual attention. *IEEE transactions on pattern analysis and machine intelligence 29*, 2 (2007), 300–312.

[170] SIAGIAN, C., AND ITTI, L. Biologically inspired mobile robot vision localization. *IEEE Transactions on Robotics 25*, 4 (2009), 861–873.

[171] SIMON, D. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[172] SINGH, N., ARYA, R., AND AGRAWAL, R. A novel approach to combine features for salient object detection using constrained particle swarm optimization. *Pattern Recognition 47*, 4 (2014), 1731–1739.

[173] SOLOMON, J. A., AND SPERLING, G. 1st-and 2nd-order motion and texture resolution in central and peripheral vision. *Vision Research 35*, 1 (1995), 59–64.

[174] SPREIJ, P. Recursive approximate maximum likelihood estimation for a class of counting process models. *Journal of multivariate analysis 39*, 2 (1991), 236–245.

[175] SQUIRE, K., AND LEVINSON, S. E. Recursive maximum likelihood estimation for hidden semi-Markov models. In *Machine Learning for Signal Processing, 2005 IEEE Workshop on* (2005), IEEE, pp. 329–334.

[176] STAUFFER, C., AND GRIMSON, W. E. L. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.* (1999), vol. 2, IEEE.

[177] STENGEL, R. F. *Optimal control and estimation*. Courier Corporation, 2012.

[178] STENTIFORD, F. Attention based auto image cropping. In *The 5th International Conference on Computer Vision Systems, Bielefeld* (2007), Citeseer.

[179] STENTIFORD, F., AND BAMIDELE, A. Image recognition using maximal cliques of interest points. In *Image Processing (ICIP), 2010 17th IEEE International Conference on* (2010), IEEE, pp. 1121–1124.

[180] STENTIFORD, F. W. An attention based similarity measure with application to content based information retrieval. In *Storage and Retrieval for Media Databases* (2003), vol. 5021, pp. 221–232.

[181] STYLES, E. A. *Attention, perception and memory: an integrated introduction*. Psychology Press, 2005.

[182] SUH, B., LING, H., BEDERSON, B. B., AND JACOBS, D. W. Automatic thumbnail cropping and its effectiveness. In *Proceedings of the 16th annual ACM symposium on User interface software and technology* (2003), ACM, pp. 95–104.

[183] TAKAHASHI, S., FUJISHIRO, I., TAKESHIMA, Y., AND NISHITA, T. A feature-driven approach to locating optimal viewpoints for volume visualization. In *Visualization, 2005. VIS 05. IEEE* (2005), IEEE, pp. 495–502.

[184] TREISMAN, A. M., AND GELADE, G. A feature-integration theory of attention. *Cognitive psychology 12*, 1 (1980), 97–136.

[185] TSOTSOS, J., CULHANE, S., KEI WAI, W., LAI, Y., DAVIS, N., AND NUFLO, F. Modeling visual attention via selective tuning. *Artificial intelligence 78*, 1 (1995), 507–545.

[186] TSOTSOS, J. K., VERGHESE, G., DICKINSON, S., JENKIN, M., JEPSON, A., MILIOS, E., NUFLO, F., STEVENSON, S., BLACK, M., METAXAS, D., ET AL. Playbot a visually-guided robot for physically disabled children. *Image and Vision Computing 16*, 4 (1998), 275–292.

[187] TUYTELAARS, T., MIKOLAJCZYK, K., ET AL. Local invariant feature detectors: a survey. *Foundations and trends® in computer graphics and vision 3*, 3 (2008), 177–280.

[188] VIJAYAKUMAR, S., CONRADT, J., SHIBATA, T., AND SCHAAL, S. Overt visual attention for a humanoid robot. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on* (2001), vol. 4, IEEE, pp. 2332–2337.

[189] VIOLA, P., AND JONES, M. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on* (2001), vol. 1, IEEE, pp. I–I.

[190] WALSH, V., AND BUTLER, S. Different ways of looking at seeing. *Behavioural brain research 76* (1996), 1–3.

[191] WALTHER, D., AND KOCH, C. Modeling attention to salient proto-objects. *Neural networks 19*, 9 (2006), 1395–1407.

[192] WANDELL, B. A. *Foundations of vision.* Sinauer Associates, 1995.

[193] WANG, Z., AND KLEIN, R. M. Searching for inhibition of return in visual search: A review. *Vision research 50*, 2 (2010), 220–228.

[194] WANG, Z., AND LI, Q. Video quality assessment using a statistical model of human visual speed perception. *JOSA A 24*, 12 (2007), B61–B69.

[195] WANG, Z., SHEIKH, H. R., AND BOVIK, A. C. No-reference perceptual quality assessment of jpeg compressed images. In *Image Processing. 2002. Proceedings. 2002 International Conference on* (2002), vol. 1, IEEE, pp. I–I.

[196] WINKLER, S. Analysis of public image and video databases for quality assessment. *IEEE Journal of Selected Topics in Signal Processing 6*, 6 (2012), 616–625.

[197] WOLFE, J. Visual attention. *Seeing 2* (2000), 335–386.

[198] WOLFE, J. M. Guided search 2.0 a revised model of visual search. *Psychonomic bulletin & review 1*, 2 (1994), 202–238.

[199] WREN, C. R., AZARBAYEJANI, A., DARRELL, T., AND PENTLAND, A. P. Pfinder: Real-time tracking of the human body. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 19*, 7 (1997), 780–785.

[200] XU, K.-L., AND PHILLIPS, P. C. Adaptive estimation of autoregressive models with time-varying variances. *Journal of Econometrics 142*, 1 (2008), 265–280.

[201] XU, T., POTOTSCHNIG, T., KUHNLENZ, K., AND BUSS, M. A high-speed multi-gpu implementation of bottom-up attention using cuda. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on* (2009), IEEE, pp. 41–47.

[202] YANG, M. Y. Image segmentation by bilayer superpixel grouping. In *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on* (2013), IEEE, pp. 552–556.

[203] YANULEVSKAYA, V., UIJLINGS, J., GEUSEBROEK, J.-M., SEBE, N., AND SMEULDERS, A. A proto-object-based computational model for visual saliency. *Journal of vision 13*, 13 (2013), 27.

[204] ZDZIARSKI, Z., AND DAHYOT, R. Feature selection using visual saliency for content-based image retrieval.

[205] ZHANG, L., XIA, Y., MAO, K., MA, H., AND SHAN, Z. An effective video summarization framework toward handheld devices. *IEEE Transactions on Industrial Electronics 62*, 2 (2015), 1309–1316.

[206] ZHU, T., WANG, W., LIU, P., AND XIE, Y. Saliency-based adaptive scaling for image retargeting. In *Computational Intelligence and Security (CIS), 2011 Seventh International Conference on* (2011), IEEE, pp. 1201–1205.

[207] ZÜND, F., PRITCH, Y., HORNUNG, A. S., AND GROSS, T. Content-aware image compression method, Apr. 26 2016. US Patent 9,324,161.