Probabilistic Model-based Multisensor Image Fusion

Ravi Krishna Sharma
B.E., University of Bombay, 1992
M.S., Oregon Graduate Institute, 1997

A dissertation submitted to the faculty of the
Oregon Graduate Institute of Science and Technology
in partial fulfillment of the
requirements for the degree
Doctor of Philosophy
in
Electrical and Computer Engineering

October 1999

The dissertation "Probabilistic Model-based Multisensor Image Fusion" by Ravi Krishna Sharma has been examined and approved by the following Examination Committee:

Misha Pavel
Professor
Thesis Research Adviser

Todd K. Leen
Professor
Thesis Research Adviser

Dan Hammerstrom
Professor

Rajeeb Hazra
Intel Corporation

Hynek Hermansky
Professor

Dedication

To

my wife Sangita

and

my family.

Acknowledgments

I would like to thank Dr. Misha Pavel for giving me the opportunity to work on my doctorate under his guidance. The work in this dissertation would not have been possible without his direction and encouragement. I deeply appreciate his advice and support. I would also to thank Dr. Todd Leen for advising me in Misha's absence, and for his guidance and encouragement during the last two years. The numerous discussions we had, helped make the work presented in this dissertation more rigorous. I have been fortunate to have the benefit of learning from two advisors.

I would like to thank the other members of my thesis committee, Dr. Hynek Hermansky, Dr. Dan Hammerstrom and Dr. Rajeeb Hazra for reviewing my work. Their valuable suggestions have helped me improve both the style and the contents of this dissertation. Special thanks to Dr. B. Yegnanarayana for his input on the introductory chapter.

My internship at Intel during the summer of 1997 was a unique experience and a welcome break from graduate studies. I am grateful to Dr. Mike Macon and Dr. Tom Gardos for making me aware of this opportunity. I would like to thank my mentor at Intel, Dr. Rajeeb Hazra, for his encouragement and advice during my internship and thereafter.

I have been fortunate to find several friends including Priya, Javed, Shafqat, Radhika, Neena, Anurag, Bala, Sai, Ujwal, Ajit, Naren, Sachin, Pratibha, Brian and Paul. We have had wonderful times together and their cheerful and helpful presence has been invaluable. Special thanks to Don Johansen, John Hunt, Mary Hultine, Barbara Olsen and Amy Todd for making my experience at OGI more memorable.

I thank my family members, especially my parents and sister, for their love, support, patience and understanding. Last, but not the least, I would like to express my gratitude to my wife, Sangita, for her love, constant support, and encouragement. She has been a friend, critic, colleague and companion throughout this endeavor.

Finally, I would like to thank OGI, NASA Ames Research Center, and FLIR Systems Inc., for providing support for my research.

Contents

List of Figures

Abstract

Probabilistic Model-based Multisensor Image Fusion

Ravi Krishna Sharma

Supervising Professors: Misha Pavel and Todd K. Leen

Image fusion is the process of combining images of a scene obtained from multiple sensors to obtain a single composite image. The goal is to reliably integrate image information from multisensor images to aid in tasks such as navigation guidance, object detection and recognition, medical diagnosis and data compression. The main challenges in fusion are caused primarily by local contrast reversals, mismatched sensor-specific image features and noise present in multisensor images. One or more of these conditions adversely affect existing fusion techniques.

In this thesis, we present a probabilistic model-based approach for the fusion of multisensor images that addresses the shortcomings of existing solutions. We formulate the fusion task as a problem of estimating an underlying true scene from the sensor images. We model the sensor images as noisy, locally affine functions of this true scene. The parameters of the affine functions explicitly incorporate reversal in local contrast and the presence of sensor-specific image features in the sensor images.

Given this model, we use a Bayesian framework to provide either maximum likelihood or maximum a posteriori estimates of the true scene from the sensor images. The estimate of the true scene constitutes our probabilistic fusion rule which resembles principal component projections. The fused image obtained by this rule is a locally weighted linear combination of the sensor images. The weights depend upon the parameters of the affine functions and the noise. The weights scale the sensor images according to the signal and noise content. We derive estimates of the model parameters from the sensor images. The least squares estimates of the affine parameters are based on the local covariance of the image data, and are related to local principal components analysis.

Our fusion approach also incorporates prior image information about the scene. The contribution of the prior image information is locally weighted and added to the combination of the sensor images. The weighting determines the confidence in the prior. The inclusion of the prior provides the ability to obtain reliable fusion results when the sensor images are unreliable.

We demonstrate the efficacy of our fusion approach on real and simulated images from visible-band and infrared sensors. We compare the results and computational complexity with those of the existing fusion techniques which are based on selection and averaging strategies. The results presented in this thesis illustrate that our probabilistic approach yields results that are similar to existing techniques when the noise is low and performs better than existing techniques when the noise is high. Common features and contrast reversed features are preserved, and sensor-specific features from each sensor image are retained in the fused image. The results using prior image information demonstrate that inclusion of prior information produces more reliable fused images.

Chapter 1

Introduction

Image fusion is the combination of images of an underlying scene captured by multiple sensors to synthesize a composite image. Advances in sensing devices have fueled the deployment of multiple imaging sensors. The different sensors (e.g. visible-band, forward looking infrared, millimeter wave radar etc.) provide different information about the scene and are effective in different environmental conditions. The use of sensors from multiple modalities can increase utility and reliability in comparison to single sensor systems. For effective practical use of multiple sensors it is often necessary to fuse the sensor images into a single composite image for interpretation. For example, visible-band and infrared images may be fused to aid pilots landing aircraft in poor visibility, or to aid in object detection. Image fusion techniques are used in computational vision[1] applications such as navigation guidance [72, 39, 24, 55], medical imaging [74], data compression [54], object detection [56] and recognition [22], classification [54, 41], and in integrating multifocus and multiexposure imagery [16].

Existing fusion techniques produce unsatisfactory results in the presence of noise, or in case of a mismatch in contrast or image features between the sensor images. This dissertation focuses on developing a methodology for reliable fusion of multisensor images in situations that tax existing techniques. Our proposed solution is to construct a probabilistic model of the process by which the underlying scene gives rise to the sensor images. Given the model, we use a Bayesian framework to estimate the most likely scene from

[1] Computational vision involves extraction of information about the real world from data captured in the form of images. Image data about a real world scene can be obtained by using a variety of imaging sensors [33].

the sensor images. The scene estimates constitute our *fusion rules.* We demonstrate the efficacy of our approach on both real and simulated sensor images. The results show that our approach overcomes the drawbacks faced by existing techniques while retaining their advantages. The fused images produced by our techniques are relatively less noisy and show better contrast and feature retention than those produced by existing techniques. Moreover, our approach has a provision to include prior image information about the scene into the fused image. We demonstrate that prior information can be used to obtain more reliable fusion results.

We also deal with three issues closely related to fusion — conformal geometric representations, multisensor image registration and fused image display. Solution of the first two issues is necessary to be able to perform fusion whereas addressing the last issue is important to be able to benefit from fusion. Multisensor images often have different geometric representations which have to be transformed to a conformal (common) representation for fusion. We describe a new geometric representation for fusion of radar images with visible-band or infrared images. This representation retains the better resolution of either sensor. Another issue that we address is the alignment (registration) of multisensor images. Multisensor registration is also affected by the differences in the sensor images. We extend a technique used for registration of same-sensor images to overcome the difficulties caused by multisensor images. Lastly, we explore techniques that allow sensor-specific details to be introduced in a display showing a fused image. These techniques investigate the use of color to identify which sensor gave rise to features appearing in the fused image.

In this chapter, Section 1.1 reviews single sensor computational vision systems. Section 1.2 argues that multisensor image fusion is a viable alternative to overcome the drawbacks of using a single sensor and describes an application of fusion. Section 1.3 highlights the issues involved in fusion of imagery generated by multiple imaging sensors. Section 1.4 describes the scope of this dissertation and our main research contributions. The last section of this chapter provides an overview of the organization of this document.

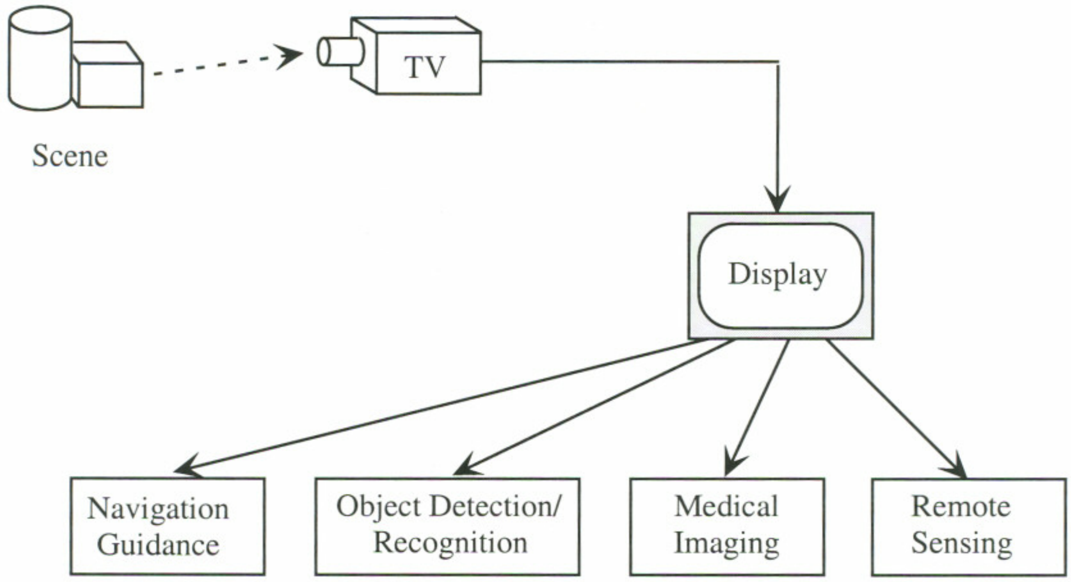


Figure 1.1: Single sensor computational vision system

1.1 Single Sensor Computational Vision System

A computational vision system consists of an imaging sensor, and components for processing the generated sensor images. An illustration of a single sensor computational vision system is shown in Figure 1.1. The imaging sensor shown is a *visible-band* sensor such as a charge coupled device (CCD) used in a television (TV) camera. The sensor captures the real world scene as an image. A sequence of images from an imaging sensor comprise a video sequence. This video sequence is then used either by a human operator or by a machine to perform some task. For example, in navigation guidance applications, a human operator views the scene on a display to aid in navigation of an aircraft or vehicle. In object detection, a human operator searches the scene on the display to detect objects such as enemy tanks. Tasks such as detection and recognition may be performed by a machine, in which case there may not be a need for a display. Other applications that use such vision systems are remote sensing, surveillance, industrial manufacturing and inspection, and medical diagnosis.

One limitation of a computational vision system is the capability of the imaging sensor that is being used. The conditions under which the system can operate, the dynamic range, resolution, visual angle, range of visibility, and the type of information obtained about the scene are all limited by the capability of the sensor. For example, the CCD sensor in the TV camera is well suited for a brightly illuminated environment (daylight or studio). The same sensor is usually not appropriate for poorly illuminated situations such as at night, or under different environmental conditions such as in fog or rain. In some applications (recognition and detection tasks, surveillance) it is essential that the vision system match and even surpass the limits of the human eye. But in many respects (e.g. dynamic range, resolution) the human visual system still exceeds the capabilities of current sensing devices [73, 33].

One possible approach to improve the range of operation of vision systems is to build a sophisticated sensor that meets all the specifications. For example, if it is necessary to match or surpass the abilities of the human visual system, one could build a sophisticated sensor with capabilities that match or exceed those of the human eye. However, such an approach would be quite expensive if not infeasible.

1.2 Multisensor Image Fusion System

An alternate approach to overcome the limitations of a single sensor vision system is to deploy multiple sensors, and to combine (*fuse*) the images from these sensors to obtain a composite (*fused*) image. The goal of such a system is to combine the image information in multisensor images such that salient image features (e.g. edges) from each sensor image are preserved in the fused image. This approach mimics biological systems. For example the human visual system fuses visual information from the three color cones and also the rods. An illustration of a multisensor image fusion system is shown in Figure 1.2. Comparing with Figure 1.1, the TV camera is supplemented by an infrared (IR) camera and their outputs are fused to produce a fused image. Although the TV camera may not produce a useful image in poor illumination, the IR camera can. The fusion algorithm would then emphasize the contribution from the IR camera and the system would be able to function even in poor illumination.

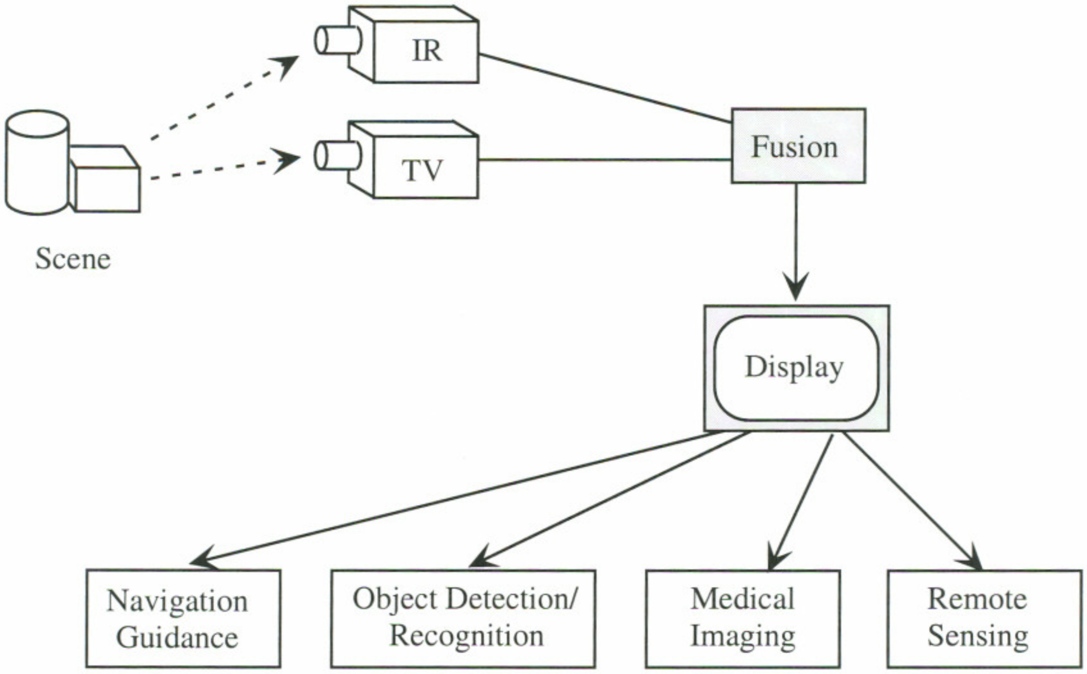


Figure 1.2: Multisensor image fusion

The benefits of multisensor image fusion [28] include:

1. Extended range of operation — multiple sensors that operate under different operating conditions can be deployed to extend the effective range of operation. For example different sensors can be used for day/night operation.

2. Extended spatial and temporal coverage — joint information from sensors that differ in spatial extent and spatial resolution can increase the spatial coverage. The same is true for the temporal dimension.

3. Reduced uncertainty — joint information from multiple sensors can reduce the uncertainty associated with the sensing or decision process.

4. Increased reliability — the fusion of multiple measurements can reduce noise and therefore improve the reliability of the measured quantity.

5. Robust system performance — redundancy in multiple measurements can help in

system robustness. In case one or more sensors fail or the performance of a particular sensor deteriorates, the system can depend on the other sensors.

6. Compact representation of information — fusion leads to compact representations. For example, in remote sensing, instead of storing imagery from several spectral bands, it is comparatively more efficient to store the fused information.

Deployment of multiple sensors for performing the same task is becoming increasingly popular because of two main reasons. The first reason is the advance in sensing devices that operate in different bands of the electromagnetic spectrum. Sensors operating in the infrared, ultraviolet and millimeter wavelengths provide the capability to *see* what the human eye cannot [33, 42]. The second reason is the increase in available computing power, which makes it possible to process the images from the multiple sensors.

The idea of combining information from multiple sources is not new — the human visual system utilizes information gathered from both eyes for depth perception. Another example is human color perception [73], where trichromatic components are combined to create the perception of color. The use of multiple sensors is also not uncommon in the field of computational vision. Two or more similar sensors are required in stereo vision applications to estimate the depth of objects in the scene [2]. Multiple imaging sensors are widely used in remote sensing applications [58]. Other applications of multisensor image fusion include navigation guidance [24, 39, 55, 72] medical imaging [74], object detection [56] and recognition [22]. Image fusion is also used for compression of multisensor (or hyperspectral) images. The fused image can be stored or transmitted at a fraction of the cost of the original images [26].

The term *image fusion* is also used in the context of fusing images obtained from the same sensor. For example, image fusion can be used to extend the depth of focus of a camera, by fusing two views of the same scene having different depth of fields [16, 43]. This is called multifocus fusion. Similarly, fusion can be used to overcome camera limitations such as limited dynamic range by fusing images obtained from the same camera at different exposure settings [16, 47], also referred to as multiexposure fusion. Images obtained from the same sensor at different times can be enhanced using fusion. Techniques that obtain enhanced images using temporal processing (i.e., using sequences of images over time) are also called *superresolution* techniques [35, 65].

In this dissertation we address the problem of multisensor image fusion. Our focus is to develop a fusion approach that addresses the problems arising out of the use of diverse multiple sensors. However our proposed fusion solution is general and can also be used to combine multiple images from the same or similar sensors. Our fusion approach also applies to *video fusion*, the fusion of image sequences from multiple sensors. However, throughout this dissertation we will use the term image fusion to maintain consistency.

1.2.1 Application of fusion for navigation guidance in aviation

The application of fusion that we use as an example throughout this dissertation is the autonomous landing guidance (ALG) system in aviation [72]. However, the solution for fusion that we develop is not specific to this particular application. Autonomous landing guidance refers to the use of synthetic or enhanced vision systems for landing aircraft autonomously in inclement weather without the help of ground aids [8]. Another system that is used for landing of aircraft in bad weather is the instrument landing system (ILS). The ILS requires special equipment at the airport and in the aircraft. Even with ILS, the pilot has no direct information about the positions of other aircraft or objects on the runway. Air traffic experiences delays because the air traffic control enforces greater separation between aircraft during bad weather. At airports which do not operate ILS, operations cease completely when the visibility conditions drop below a specified minimum. ALG employs multiple imaging sensors placed in the nose cone of the aircraft to provide navigation guidance to pilots for landing the aircraft in low visibility conditions. The goal is to display the landing scene to the pilot on a suitable electronic device[2] in the cockpit. Such a system could support operation in low visibility, resulting in significant benefits to airlines and passengers. Since the equipment for ALG (sensors, processing modules) is on board the aircraft, it can improve the safety of landing operations even at small airports. In addition, the ALG system would permit shorter separation between aircraft and at the

[2] Generally a heads up display (HUD) or a heads down display (HDD) is used for this purpose.

same time enable the pilot to verify clear runway conditions.

Although several different sensors have been studied for use in ALG, visible-band, forward looking infrared (FLIR) and millimeter wave radar (MMWR) based imaging sensors are the most common [34]. These sensors have been found to be effective in providing imagery that can support reliable navigation in unpredictable environmental conditions. FLIR sensors are passive sensors based on the reflection of thermal energy, and can produce relatively high resolution images when imaging at night as well as through haze and some types of fog [40, 64]. MMWR sensors are active sensors based on the reflection of transmitted radio waves. MMWR can penetrate fog and provides the least attenuation in rain [40]. The concept of the ALG system is illustrated in Figure 1.3. The system consists of FLIR and MMWR sensors in addition to a visible-band TV camera. In addition to the imaging sensors, the ALG system uses information from the global positioning system (GPS) and an inertial navigation system (INS). The ALG system sometimes uses a terrain database of the landing scene, if it is available. The terrain database provides prior information about the scene [51].

The pilot can be provided with one display for each of the sensors. However, humans are not effective at integrating visual information by viewing multiple displays separately [71]. Another alternative is to have a single display and provide a switch that allows the pilot to select which sensor to view at any time. However, this solution can require frequent switching in certain situations, for example, in a scenario where the aircraft is breaking in and out of clouds. In addition, it is not desirable to increase the workload of the pilot in time critical tasks such as landing.

It is therefore desirable to fuse the images in the sensor image sequences and obtain a fused sequence. The fused sequence should ideally provide all the required visual information to the pilot for landing the aircraft safely. For this purpose, one needs to develop techniques for automatically and reliably fusing multisensor images. The fused sequence can then be displayed to the pilot as illustrated in Figure 1.3.

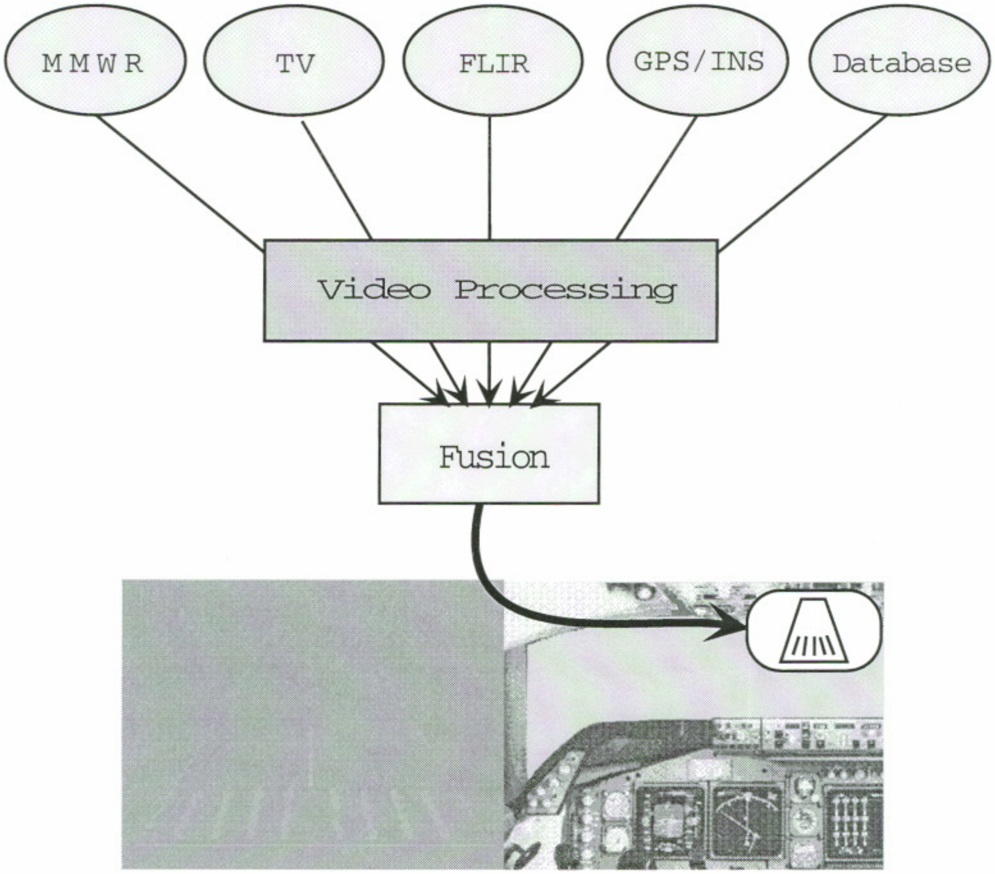


Figure 1.3: Application of image fusion in aviation

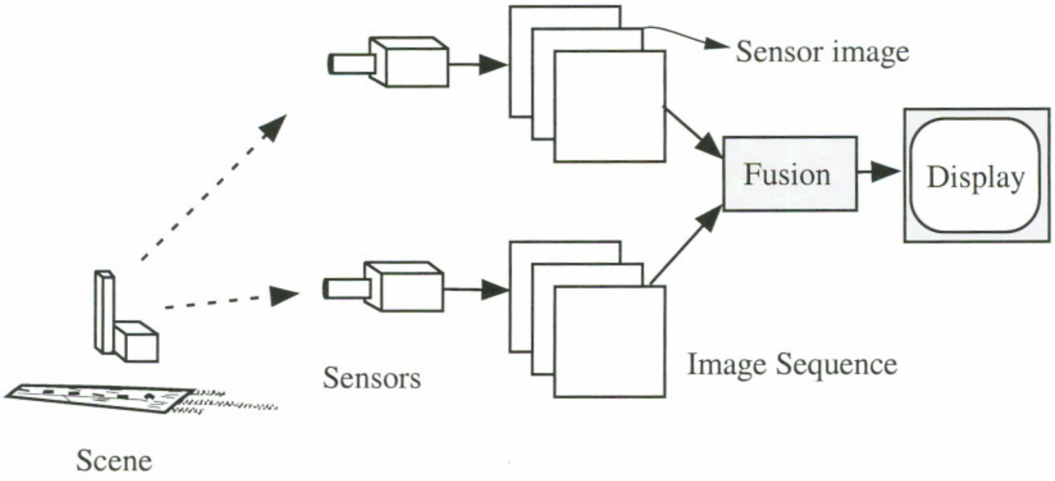


Figure 1.4: Multisensor fusion system

1.3 Issues in Fusing Imagery from Multiple Sensors

Figure 1.4 shows a schematic of multisensor image fusion. Two different sensors (for example, visible-band and infrared) capture the same scene and generate two different sequences of sensor images. The issues to be addressed before performing fusion are concerned with converting the sensor images into representations in which they can be compared and fused. The central issue is how to combine the images. Finally, there are issues concerning display of fused images. Most of the issues are common to both fusion of images as well as fusion of image sequences, and the exceptions are pointed out. The important issues are discussed below:

1.3.1 Mismatch in features

There is often a mismatch in image features between images from different sensors. This mismatch makes it difficult to compare the sensor images and therefore causes difficulties in fusion. The mismatch arises mainly because of the sensors used. For example, visible-band and infrared sensors are based on different physical phenomena and their sensing processes are different. The images produced by these sensors have distinct graylevel appearances [16, 40, 44], giving rise to a mismatch in the features[3] in the images. The polarity of local contrast is often reversed between visible-band and infrared images. We call this *local polarity reversal.* For example, dark objects in a scene that are warmer than the surroundings appear dark in visible-band images and appear bright in infrared images. Sometimes, image features present in one image are missing in another image. We call such sensor-specific features *complementary features.* For example, objects obscured by fog in visible-band imagery are visible in radar imagery. In addition, the images have different noise characteristics depending upon the sensors used.

1.3.2 Combination of multisensor images

The most important issue concerning image fusion is to determine how to combine (fuse) the sensor images. This task is complicated particularly when there is a mismatch between the sensor images as discussed above. In recent years, several image fusion techniques have been proposed. Pixel-based methods such as pixel-wise averaging operate on pixels of sensor images to obtain the fused image. Averaging works well when images are similar, but causes a reduction in contrast when there are complementary features or local polarity reversals. Feature-based methods [11, 16, 51, 68, 70, 78] use selection as the criterion for fusion and operate in a multiresolution pyramid transform domain (e.g., Laplacian pyramids, contrast pyramids). These methods construct a fused pyramid by selecting the most salient coefficient (based on, for example, maximum magnitude or local energy) at each pyramid location. Since features are selected rather than averaged, they are rendered at full contrast in the fused image. However, selection techniques have drawbacks. They do not explicitly account for noise in the sensor images and may thus select large noise spikes in the fused image. In addition, these techniques do not have the ability to adapt to changing sensor characteristics. To improve upon these techniques and reliably integrate image information from multiple sensors, one needs to determine the exact nature of the relationships between the image features and the noise characteristics, and then use this knowledge for fusion. One important aspect concerning the combination of sensor images

[3] *Image features* are patterns in the image that arise due to objects and materials in the scene, environmental factors, and the sensing process.

is the potential of using temporal information, available in the form of a sequence of images from each sensor over time (i.e., video sequences), for improving fusion results. This aspect has been largely ignored in the existing literature.

1.3.3 Geometric representations of imagery from different sensors

Images from different sensors may have different geometric representations. Before performing fusion, all sensor images must be transformed to a conformal (common) geometric representation to facilitate comparison and fusion [10]. The imaging geometry depends upon the physical mechanism underlying the sensor. For example, passive sensors such as visible-band sensors or FLIR sensors have a projective geometry and produce projective images (images are formed by a 2-D projection of a 3-D scene on the image plane). Active sensors such as MMWR sensors have a range geometry (a radar image is the intensity of the reflected radio wave as a function of the azimuth angle and the range). Converting an image from one geometric representation to another involves reconstruction of the 3-D scene geometry from a 2-D image followed by transforming the 3-D geometry to the new representation.

1.3.4 Registration of misaligned image features

The goal of registration is to establish a spatial correspondence between the sensor images and determine a spatial geometric transformation (warping) that aligns the images. After registration, corresponding image features in the sensor images are perfectly aligned when superimposed. Misalignment of image features is caused by several factors including differences in imaging geometries of the sensors, different spatial positions of the sensors, different temporal capture rates of the sensors and the inherent misalignment of the sensing elements [7]. Registration is carried out either using optical-flow techniques [1, 4, 5, 27, 32, 48] or feature-matching techniques [44, 46, 60, 80]. These techniques align the images by exploiting the similarities (in graylevels or features) between the sensor images. The mismatch of image features in multisensor images reduces the similarities between the images and makes it difficult to establish the correspondence between the images.

1.3.5 Spatial resolution of different sensors

There is often a difference in spatial resolution between the images produced by different sensors. FLIR and MMWR sensors usually have a lower resolution than visible-band sensors [40]. One way to combine image data with different spatial resolutions is to use superresolution techniques [18, 35, 59], when possible, to improve resolution. Another approach is to use multiresolution image representations so that the lower resolution imagery does not adversely affect the higher resolution imagery.

1.3.6 Differences in frame rate

This issue is specific to video fusion where a sequence of images from each sensor are to be fused. The temporal sampling rates of the sensors need to be matched and synchronized for fusion so that images captured at the same time instant are combined. However, different types of sensors can have different frame rates. For example, visible-band sensors are capable of generating 30 video frames (i.e. images) every second. However, imaging radars usually generate around 10 to 15 frames per second [8]. There are two approaches to match the frame-rate of the sensors. One approach consists of synthetically increasing the frame-rate of the slower sensor by using video frame interpolation techniques to estimate the missing frames. Another approach is to discard the excess frames of the faster sensor. In the latter case, the resulting fused video sequence may have poor temporal resolution.

1.3.7 Display of fused images

Although fusion aims to preserve salient information from the sensor images, the source of the information is lost. For example, it is difficult to determine whether a bright patch in a fused image came from a visible-band sensor or an infrared sensor. Display issues are concerned with reinstating the knowledge about the source of the image features in the fused image [71, 76] to provide an easily interpretable display of fused images.

1.4 Problem Definition and Research Contribution

The focus of this thesis is methods for fusion, i.e., combination of sensor images. The most important contribution of this thesis is the development of a probabilistic model-based approach for performing fusion that circumvents the difficulties in fusion due to mismatched image features and noise. We provide a rigorous theoretical foundation for this approach and demonstrate its efficacy using several examples. We also develop techniques to achieve conformal geometric representations, multisensor image registration, and interpretable display of fused images. The major contributions of this thesis are summarized by the following points:

- **Probabilistic model-based approach for fusion:**
  We construct a probabilistic model of the process by which an underlying true scene gives rise to the sensor images. The probabilistic nature of the model takes into account the uncertainties of the sensing process. The sensor images are modeled as noisy, locally affine functions of the underlying true scene. Although the model is simple, it explicitly captures the local relationships between the sensor images — local polarity reversals and complementary features. The model lays the foundation for a principled approach to image fusion.

- **PCA-like fusion rules:**
  The sensor images and the model are used to derive maximum likelihood and maximum a posteriori estimates of the underlying true scene, within a Bayesian framework. These estimates constitute our probabilistic fusion rules, which are locally weighted additive combinations of the sensor images. The weights depend upon the model parameters and determine the contribution of each sensor image based upon the signal and noise content. We describe factor analysis techniques to estimate the model parameters from the sensor images. With the model parameters estimated, the probabilistic fusion rules resemble PCA-like projections.

- **Use of prior information for fusion:**

  Our fusion approach also provides a principled way to combine prior image information about the scene with the sensor images. The contribution from the prior is locally weighted and added to the combination of the sensor images. The weighting determines the confidence in the prior. The inclusion of the prior provides the ability to obtain reliable fusion results when the prior information is more reliable than the sensor images.

- **Demonstration of fusion approach on real and simulated images:**

  We present results of several fusion experiments on both real and simulated multisensor images. The experiments illustrate the different ways of using our fusion approach. The results indicate that our approach addresses the problems faced by existing methods such as selection and averaging. The fused images produced by probabilistic fusion have relatively lower noise, and show better contrast and feature retention.

- **An approach for registration of multisensor images:**

  We develop an approach for registration of multisensor images that have mismatched image features. We have extended the gradient-based registration technique [5] for this purpose. The image representations that we use for gradient-based registration are invariant to graylevel differences and local polarity reversals and, thereby facilitate registration.

- **A novel conformal geometric representation of multisensor images:**

  We develop a conformal geometric representation called M-scope. The M-scope representation preserves the best resolution of the sensor images at each spatial location and is well-suited for fusion of radar images with images from visible-band or infrared sensors.

- **Interpretable display of fused images** We present several examples of techniques for identifying the source of sensor-specific image features in a fused image. We investigate the mapping of data from the fused image and the sensor images onto color dimensions, and develop three mapping methods. We show that pseudocolor mapping can help identify sensor-specific details in a fused display.

1.5 Organization of the Dissertation

The organization of this dissertation is as follows. In Chapter 2 we review important techniques for image fusion that are described in literature. These include feature based approaches based on selection and their variants. We outline the advantages and drawbacks of these existing fusion techniques. In Chapter 3 we describe our probabilistic model-based approach to fusion. We define our model and derive our probabilistic fusion rules using a Bayesian framework. In Chapter 4 we describe techniques to estimate the parameters of the model from the sensor images. In Chapter 5 we demonstrate different ways of using the probabilistic fusion rules and describes results of experiments using these techniques. In Chapter 6 we present conclusions and suggest directions for future work. Fusion issues relating to geometric representations, registration and display are organized as appendices. Appendix A deals with conformal geometric representation of multisensor images. Appendix B describes techniques for multisensor image registration. In Appendix C we describe display techniques to facilitate the identification of the source of image features in a fused image.

Chapter 2

Review of Image Fusion Techniques

2.1 Introduction

In Chapter 1 we introduced the concept of multisensor image fusion. We discussed the motivation for fusion and its benefits. We also described the various issues that need to be addressed in order to develop a solution to the fusion problem. In this chapter, we review the most important existing fusion techniques that have been described in literature. A detailed understanding of these techniques is essential to develop better fusion algorithms. From this perspective, we outline the advantages as well as the disadvantages of the existing techniques and set the stage for our proposed fusion solution presented in following chapters.

Section 2.2 states the typical assumptions that are made by fusion algorithms. Section 2.3 reviews some of the simplest fusion techniques that are based on combining the images a pixel by pixel. Section 2.4 reviews fusion techniques that decompose images into feature representations and then select features from one image or another to generate the fused image. The Laplacian pyramid transform, which we later use in examples using our proposed fusion solution, and its significance in fusion by selection are explained in detail. Sections 2.5 and 2.6 review various techniques that employ variants of the selection strategy, use different feature representations, and neural networks for performing fusion. In Section 2.7, we draw attention to the shortcomings of the fusion techniques discussed in this chapter. Section 2.8 is a brief discussion that highlights the specific issues we will address in the following chapters.

2.2 Typical Assumptions for Image Fusion

Multiple imaging sensors capturing the same scene usually generate image data that are not in a form that is directly suitable for performing fusion. For example, the sensor data may have different geometric representations and may be mis-registered. In order to be able to perform fusion, the sensor data must be converted to a conformal geometric representation and perfectly registered. As discussed in Section 1.3, conformal representations and registration are challenging problems. Appendix A and B deal in detail with these issues. For the discussion in this chapter and in Chapters 3, 4 and 5, we assume that the images from the sensors are in a conformal geometric representation and perfectly registered. Recall from Section 1.3 that additional processing may be required before performing fusion (e.g., solving frame rate issues for fusion of image sequences from sensors with unequal frame rates), but these are beyond the scope of this dissertation.

2.3 Pixel-based Approach to Fusion

The simplest techniques employed for fusion are direct approaches that synthesize the fused image from pixels of the sensor images. Hence these direct approaches are also called pixel-based approaches.

2.3.1 Fusion by averaging

A simple approach for fusion, based on the assumption of additive Gaussian noise, consists of synthesizing the fused image by averaging corresponding pixels of the sensor images as shown in Figure 2.1. Averaging works well when the images to be fused are from the same type of sensor and contain additive noise. If the variance of noise in q sensor images is equal then averaging them reduces the variance of noise in the fused image by a factor of q. Figure 2.2 is an example that illustrates fusion by averaging. The images in Figure 2.2(a) and 2.2(b) are synthetic images that simulate noisy visible-band images of a runway scene. The fused image in Figure 2.2(c) is less noisy than either of the original sensor images. Another advantage of using averaging for fusion is that it is computationally inexpensive as discussed in Appendix I.

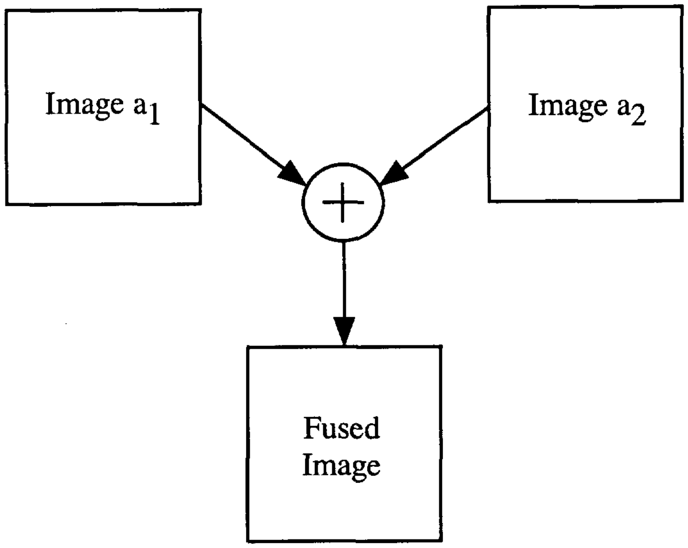


Figure 2.1: Schematic diagram of fusion by averaging

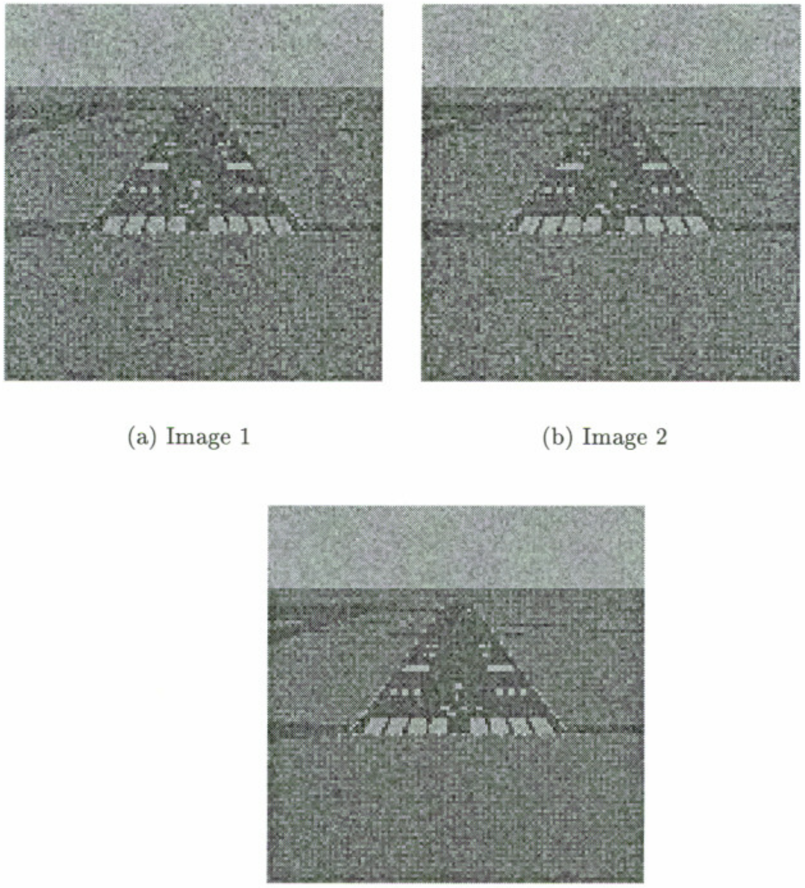
2.3.2 Image merging

Another direct approach to fusion is image merging. This technique consists of generating a composite or fused image by merging relevant regions of the sensor images. Image merging generally involves identifying regions of interest from each of the sensor images and inserting them into the fused image. This is followed by blurring to smooth the boundaries caused by inserting regions from different images. Burt and Adelson [15] performed image merging using the multiresolution Laplacian pyramid[1] representation, followed by smoothing using splines to obtain a smooth merge.

2.4 Feature-based Approach to Fusion

Since the essential goal of fusion is to preserve the image features in the sensor images, a logical extension of pixel-based fusion is to transform the images into a representation that decomposes the images into "features" such as edges, and perform fusion in this domain. Such a decomposition or transformation can be obtained in terms of basis functions that capture the particular image features. Researchers have shown that fusion techniques that

[1] The Laplacian pyramid representation is discussed in Section 2.4.1.



(a) Image 1

(b) Image 2

(c) Fused image

Figure 2.2: Example of fusion by averaging

operate on such features in the transform domain yield subjectively better fused images than pixel-based techniques [11, 13, 16, 51, 68, 69].

The need to preserve image features in the fused image imposes certain requirements for the transform domain representation to satisfy. The transform should separate the features with respect to resolution at different scales and maintain the location information [51]. Different features in an image are important at different scales. Relevant details about each image feature generally exist over a restricted range of scales and resolutions. Sometimes only coarse image features are important, in other cases fine detail is important. For fusion (as in other computational vision tasks), it is essential that the scale and resolution of the analysis (and synthesis) match that of image features within the scene [12]. A multiscale representation facilitates this type of analysis using a fixed size analysis window which is then used over different scales of the representation. The transform domain representation should also separate the information in the images according to resolution. This facilitates combination of image information at the matching resolution when images with different resolutions are to be fused. The need for maintaining location information arises because the relationships between multisensor image features changes with location (this point is discussed further in Section 3.2.1). With location information maintained, features at different locations in the sensor images can be combined appropriately.

A multiresolution pyramid transformation decomposes an image into multiple resolutions at different scales while preserving location information [11, 68]. A pyramid is a sequence of images (or levels) in which each level is a filtered and subsampled copy of the predecessor. A schematic of a multiresolution representation is shown in Figure 2.3. The lowest level of the pyramid has the same scale as the original image and contains the highest resolution information. Higher levels of the pyramid are reduced resolution and increased scale versions of the original image. Multiresolution pyramid representations contain descriptive information about edges, zero crossings, gradients, contrast etc. in the image. The successive levels of the pyramid from the lowest level to the highest level represent increasingly coarse approximations to these features. Fusion using the Laplacian pyramid representation is described in the following section.

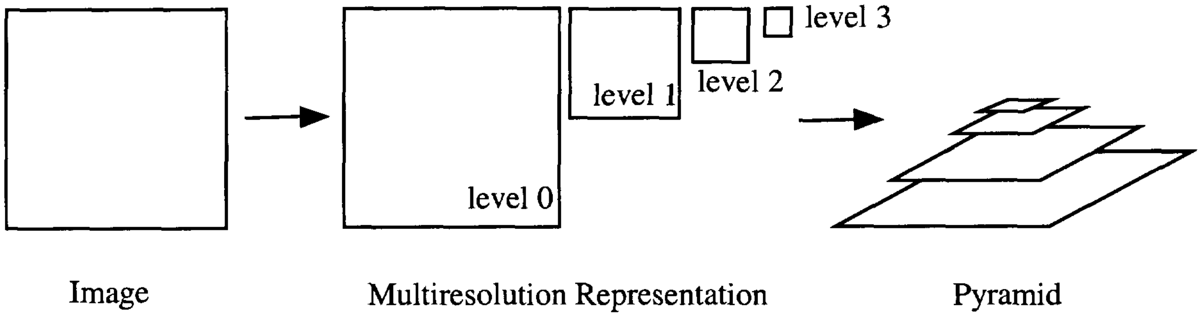


Figure 2.3: Multiresolution image representation

2.4.1 Fusion using selection in the Laplacian pyramid domain

Burt [11] first proposed selection-based fusion as a model for binocular vision. The selection[2] approach to fusion consists of three main steps:

1. Each sensor image is decomposed into a multiresolution pyramid representation to obtain the sensor pyramids.

2. A fused pyramid is constructed from the sensor pyramids by selecting the most salient coefficient (hyperpixel[3]) from the sensor pyramids at each location of the pyramid.

3. The inverse pyramid transform is then applied to the fused pyramid to obtain the fused image.

These three steps are illustrated in Figure 2.4.

Burt performed the selection operation in the Laplacian pyramid transform domain. The Laplacian pyramid is obtained from the Gaussian pyramid [11]. Let G^k be the k^{th} $(k = 0, \ldots, N)$ level of the Gaussian pyramid for an image I. Then,

$$G^0 = I$$

$$G^{k+1} = [w * G^k]_{\downarrow 2} \qquad \text{for} \quad k = 1 \ldots N-1$$

[2] The pyramid-based selection approach is also known as "pattern-selective" fusion [13, 70] since it selects patterns (i.e., image features) in the images that are isolated by the pyramid transform.

[3] In this dissertation we refer to the the coefficients at different levels of the pyramid as *hyperpixels*.

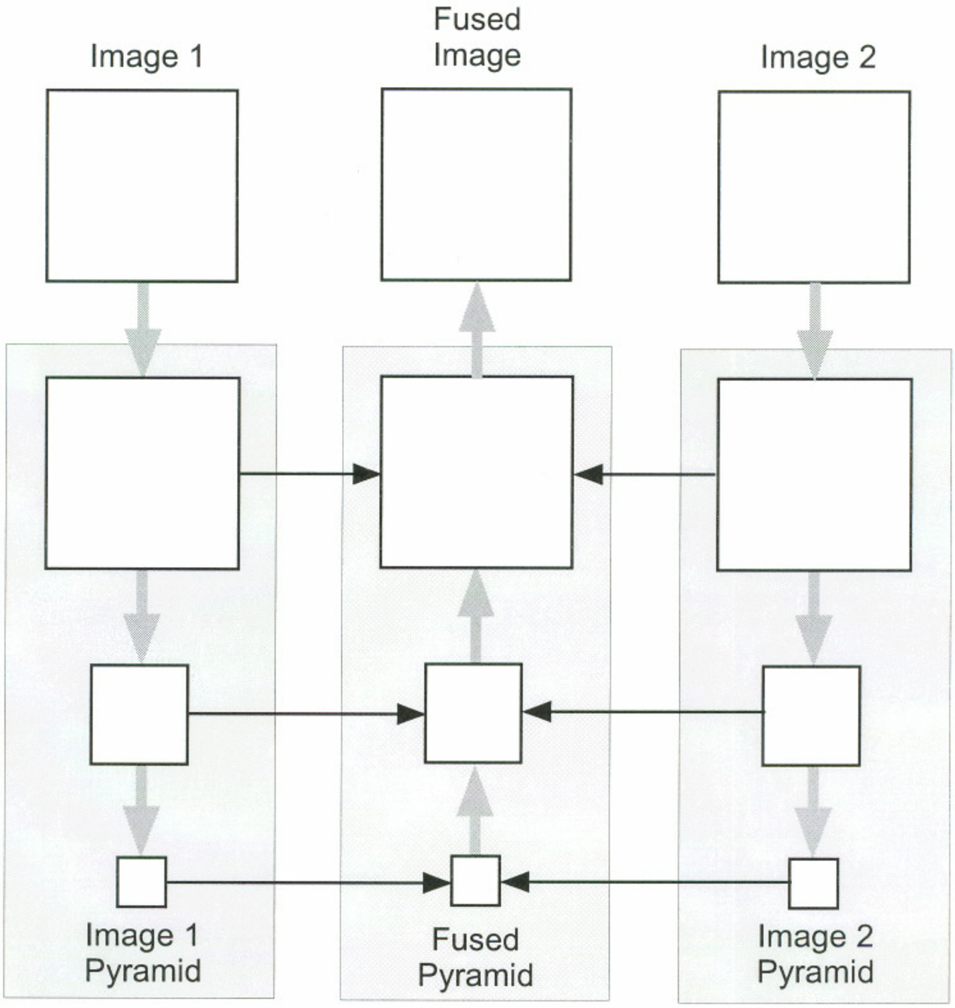


Figure 2.4: Fusion using multiresolution pyramids

where w is a kernel that is a digital approximation of the Gaussian probability distribution function[4], $*$ denotes two dimensional convolution and the notation $[...]_{\downarrow 2}$ indicates that the image in brackets is downsampled by 2 (in both the horizontal and vertical directions). The Gaussian kernel w is used as a convolution mask for filtering the image. The downsampling operation is accomplished by selecting every other point in the filtered image. The Gaussian pyramid is a set of low-passed filtered copies of the image, each with a cut-off frequency one octave[5] lower than its predecessor.

The levels of the Laplacian pyramid are obtained as

$$L^N = G^N$$

$$L^k = G^k - 4w * [G^{k+1}]_{\uparrow 2} \qquad \text{for} \quad k = 0 \ldots N-1$$

where the notation $[...]_{\uparrow 2}$ indicates that the image inside the brackets is upsampled by 2 (in both the horizontal and vertical directions). Here, convolution by the Gaussian kernel has the effect of interpolation by a low-pass filter. Each level in the Laplacian pyramid represents the result of convolving the original image with a difference of two Gaussian functions [11]. The difference of Gaussians resembles the Laplacian operator[6] commonly used in image processing [26, 63], and hence the name Laplacian pyramid. The Laplacian pyramid transform decomposes the image into multiple levels. Each successive level is a band-passed[7], sub-sampled and scaled version of the original image.

The Laplacian pyramid has the perfect reconstruction property – the original image can be reconstructed by reversing the Laplacian pyramid operations:

$$\widehat{G}^N = L^N$$

[4] Burt [11] gave several constraints in the spatial domain for the kernel w including separability and symmetry. As a result the two dimensional convolution can be implemented as two one dimensional convolutions. The k^{th} level of the Gaussian pyramid can also be obtained by filtering with an equivalent kernel w_k and appropriate downsampling. For a choice of $w = [1, 4, 6, 4, 1]$, the equivalent kernels w_k resemble the Gaussian probability density function and hence the name Gaussian pyramid.

[5] Convolving by w does not result in a perfect half-band filtering operation and consequently there is some aliasing due to downsampling. However, the aliased component is cancelled when the image is reconstructed from the Laplacian pyramid.

[6] The Laplacian operator is a second derivative operator. Zero crossings of the image obtained after applying the Laplacian operator give the location of edges in the image [26].

[7] For a frequency domain analysis of the Laplacian pyramid operations, see [53].

$$\widehat{G}^k = L^k + 4w * [\widehat{G}^{k+1}]_{\uparrow 2} \qquad \text{for} \quad k = 0 \ldots N-1$$

$\widehat{G}^0$ is identical to the original image I.

Fusion is performed in the Laplacian pyramid domain by constructing a fused pyramid. The pyramid coefficient (or hyperpixel) at each location in the fused pyramid is obtained by selecting the hyperpixel of the sensor pyramid that has the largest absolute value. Let L_A and L_B be the Laplacian pyramids of two images A and B. Let L_F be the fused pyramid. Then,

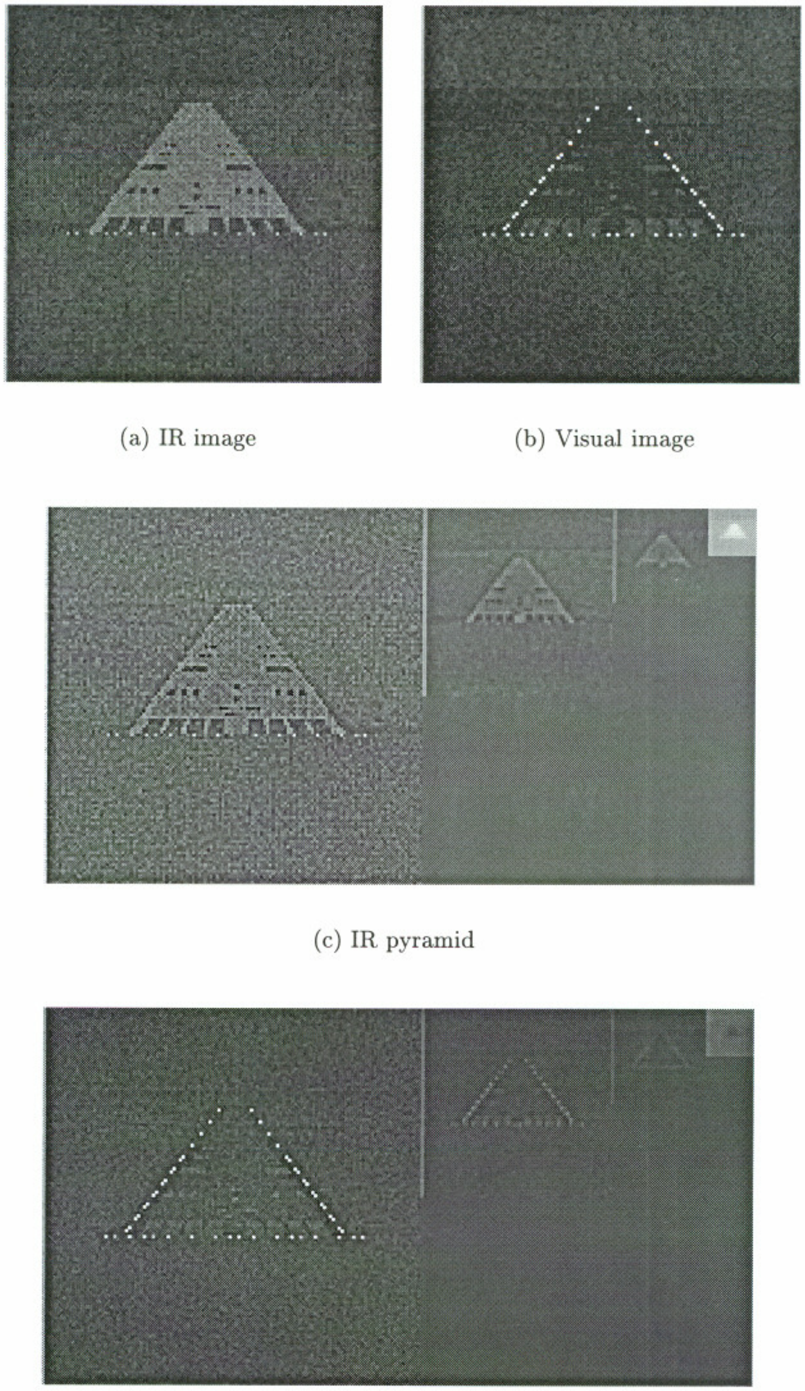
$$L_F^k(i,j) = \begin{cases} L_A^k(i,j) & \text{if} \qquad |L_A^k(i,j)| > |L_B^k(i,j)| \\ L_B^k(i,j) & \text{otherwise} \end{cases}$$

where k is the level of the pyramid and (i,j) denotes a hyperpixel at that level. The computational complexity of fusion by selection using Laplacian pyramids is discussed in Appendix I.

Figure 2.5 shows simulated infrared and simulated visible-band images of a runway scene and their corresponding Laplacian pyramids. Figure 2.6 shows the fused pyramid synthesized with the selection rule described above. Image features from both the sensor images are preserved in the fused image. The runway structure from the infrared image as well as the runway lights from the visible-band image appear in the fused image. Moreover, the contrast of features that are present in only one of the sensor images (for example the runway lights) appears to be retained in the fused image.

2.5 Variants of Feature-based Approaches

Several variations of feature-based approaches using multiresolution representations have been described in literature. These techniques differ from pattern-selective fusion using the Laplacian pyramid described in Section 2.4.1 in the type of pyramid transform used and in the rule used to synthesize the hyperpixels in the fused pyramid. A brief description of the important techniques follows.



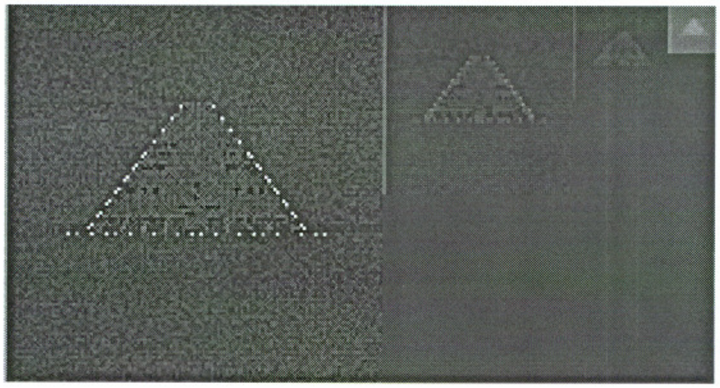
(a) IR image

(b) Visual image

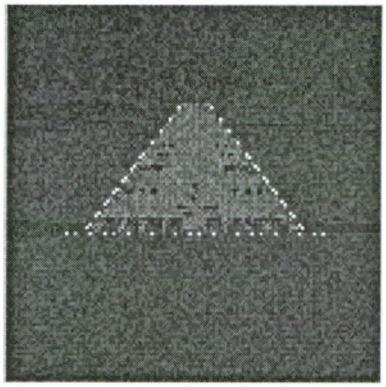
(c) IR pyramid

(d) Visual pyramid

Figure 2.5: Infrared and visual images and their Laplacian pyramids



(a) Fused pyramid



(b) Fused image

Figure 2.6: Fusion by selection using Laplacian pyramids

2.5.1 Fusion using contrast pyramids

Toet et al. [70, 68, 69] introduced an image fusion technique which preserves local luminance contrast in the sensor images. The technique is based on selection of image features with maximum contrast rather than maximum magnitude. It is motivated by the fact that the human visual system is based on contrast and hence the resulting fused image will provide better details to a human observer. The pyramid decomposition used for this technique is related to luminance processing in the early stages of the human visual system which are sensitive to local luminance contrast [70]. Fusion is performed using the multiresolution contrast pyramid.

The contrast pyramid is obtained from levels of the Gaussian pyramid (see Section 2.4.1). The k^{th} level R_k of the contrast pyramid is obtained as:

$$R_k = \frac{G_k}{4w * [G_{k+1}]_{\uparrow 2}} \qquad \text{for } k = 1, \dots, N-1$$

$$R_N = G_N$$

The hyperpixels of the contrast pyramid R are related to the local luminance contrast. Luminance contrast C is defined as

$$C = \frac{L - L_b}{L_b} = \frac{L}{L_b} - 1$$

where L is the luminance at a certain location in the image and L_b is the luminance of the local background [68]. The denominator in the equation for R_k represents the upsampled and interpolated version of G_{k+1}. A hyperpixel in this interpolated image corresponds to a weighted local average in the neighborhood of the hyperpixel at the same location in G_k. Hence, the denominator in R_k is proportional to L_b whereas the numerator is proportional to L. Therefore the pyramid whose levels are $R_k - I_k$ (where I_k is the k^{th} level of the unit pyramid with all hyperpixels having value 1), represents a contrast pyramid[8]. The original image can be perfectly reconstructed by reversing the pyramid generation operations described above.

[8] R is also known as the ratio of low pass (ROLP) pyramid because of the manner in which it is constructed from low-pass filtered copies of the original image [68].

The fused contrast pyramid R_F is formed from the contrast pyramids R_A and R_B of the images A and B by using the selection rule,

$$R_F^k(i,j) = \begin{cases} R_A^k(i,j) & \text{if} \qquad |R_A^k(i,j)| > |R_B^k(i,j)| \\ R_B^k(i,j) & \text{otherwise} \end{cases}$$

where k is the level of the pyramid and (i, j) denote the hyperpixels at that level. The fusion rule selects the hyperpixels corresponding to the largest local luminance contrast. As a result, image features with high contrast are preserved in the fused image.

2.5.2 Noise-based fusion rule

Pavel et al. [51] developed this approach for fusion of images from passive millimeter wave radar (PMMW[9]) with computer generated imagery (synthetic images) from a terrain database. Images from PMMW have a much lower resolution than the synthetic images (or even visible-band images). In this technique, the effects related to the generation of the PMMW image $m(\overline{x})$, such as limited spatial resolution, atmospheric attenuation and noise are described by a set of operations on a gray-level image of the scene $s(\overline{x})$:

$$m(\overline{x}) = h * [a(z(\overline{x}))b(\overline{x})s(\overline{x})] + n(\overline{x})$$

where, $\overline{x}$ is the spatial location, $a(z)$ is the atmospheric attenuation over a distance z, h represents the low-pass sensor characteristics, $*$ denotes convolution, n is the noise due to the environment and the sensor, and b is a mapping between the PMMW image and the desired gray level image s.

The database image contains the high frequency features expected in the scene, but does not have any information about the presence of obstacles in the scene. The database image $d(\overline{x})$ is related to the desired scene by a multiplicative function $c(\overline{x})$ that indicates the presence or absence of unexpected objects $g\,(\overline{x})$,

$$d(\overline{x}) = [1 - c\,(\overline{x})]\, s\,(\overline{x}) + c\,(\overline{x})\, g\,(\overline{x})\,,$$

where $c(\overline{x}) \in \{0, 1\}$.

[9] PMMW is a passive sensor based on millimeter wavelength radiation [61].

Fusion is performed using the Laplacian pyramid. The fusion algorithm is based on a weighted sum of the PMMW and database pyramids,

$$\widehat{s}\left(\overline{x}\right) = \alpha m\left(\overline{x}\right) + \beta d\left(\overline{x}\right)$$

with weights that change with the level and location within the pyramid. The weights α and β are determined by local estimates of the signal-to-noise ratio of the two sensor images. The technique approximates the atmospheric attenuation a using regression analysis. The mapping $b(\overline{x})$ is estimated in the minimum mean squared error sense assuming $c(\overline{x}) = 0$ everywhere. The residual error is used to estimate the standard deviation of noise, σ_m. The residual errors are then compared to σ_m and excessive values are interpreted as potential obstacles, $c = 1$. Estimates of variability in small spatial and temporal neighborhoods are used to determine the weights at each hyperpixel location. The authors assert that this technique is effective in fusing synthetic imagery with lower resolution PMMW imagery.

2.5.3 Fusion by combination of selection and averaging

This approach proposed by Burt and Kolczynski [16], is a modification of the selection rule using Laplacian pyramids described in Section 2.4.1. Fusion is performed by a combination of selection and averaging to improve the noise immunity and to address the case of features with opposite contrast. The pyramid representation used in this approach is the gradient pyramid where the basis functions are gradient of Gaussian patterns (i.e., gradient operators applied to Gaussian kernels).

The fusion step consists of two operations — selection and averaging. At the hyperpixels where the source images are distinctly different, the most salient hyperpixel is selected into the fused pyramid. At the hyperpixels where source images are similar, their average is computed and assigned to the fused pyramid. Similarity and salience are determined by the match measure and the salience measure. The salience is determined by the local energy within a neighborhood p of each hyperpixel. The salience S at a hyperpixel (m, n)

at orientation o and level k of the gradient pyramid[10] of an image I is computed as

$$S_I(m,n,o,k) = \sum_{m'n' \in p} D_I(m+m', n+n', o, k)^2 p(m', n')$$

where D_I denotes the gradient pyramid of the image. The analysis window defined by the neighborhood p is typically the hyperpixel itself or a 3×3 or 5×5 array of hyperpixels surrounding the hyperpixel.

The match measure is determined by the normalized local correlation between the source pyramids within the neighborhood p.

$$M_{AB}(m,n,o,k) = \frac{2 \sum\limits_{m'n' \in p} D_A(m+m', n+n', o, k) D_B(m+m', n+n', o, k) p(m', n')}{S_A(m,n,o,k) S_B(m,n,o,k)}$$

The normalized local correlation M_{AB} has a value 1 for identical patterns, -1 for patterns that are identical with opposite signs and a value between 1 and -1 for all other patterns.

The fusion rule consists of a weighted average

$$D_C(\vec{m}) = W_A(\vec{m}) D_A(\vec{m}) + W_B(\vec{m}) D_B(\vec{m})$$

where the weights are determined by the match and salience measures as

$$\begin{array}{llll} \text{if } (M_{AB} \leq \alpha), & W_{\text{min}} & = 0 & \text{and} \quad W_{\text{max}} = 1 \\ \text{otherwise} & W_{\text{min}} & = \frac{1}{2} - \frac{1}{2}\frac{(1-M_{AB})}{1-\alpha} & \text{and} \quad W_{\text{max}} = 1 - W_{\text{min}} \end{array}$$

$$\begin{array}{llll} \text{if } (S_A > S_B), & W_A & = W_{\text{max}} & \text{and} \quad W_B = W_{\text{min}} \\ \text{otherwise} & W_A & = W_{\text{min}} & \text{and} \quad W_B = W_{\text{max}} \end{array}$$

The weights change between the extremes $(0, 1)$ in selection mode and are roughly 0.5 in averaging mode when the correlation is near 1. The weights as a function of the normalized correlation are shown in Figure 2.7 for $\alpha = 0.85$.

The fused image is then constructed from the gradient pyramid D by first constructing a Laplacian pyramid from the gradient pyramid [16], and then following the steps in Section 2.4.1. Burt and Kolczynski [16] assert that this approach provides a partial solution

[10] For details of the gradient pyramid see [16].

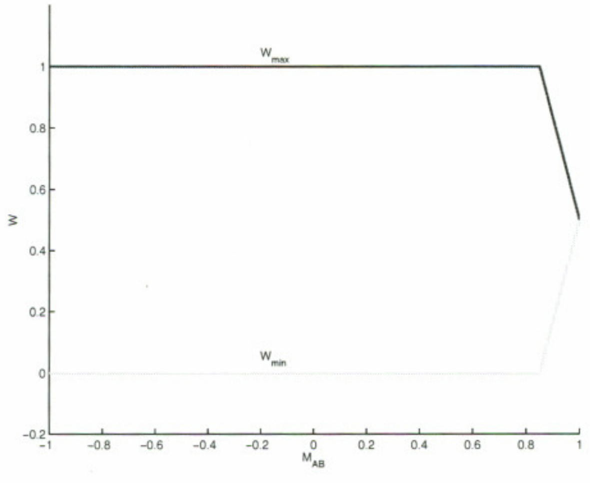


Figure 2.7: Sensor weights as a function of normalized correlation

to the problem of combining features that have reversed contrast, since such patterns are always combined by selection. In addition, the area-based salience measure and the gradient pyramid provide greater stability in noise, compared to Laplacian pyramid based fusion.

2.5.4 Wavelets based approaches

An alternative to fusion using pyramid based multiresolution representations is fusion in the wavelet transform domain [43]. The wavelet transform decomposes the image into low-high, high-low, high-high spatial frequency bands at different scales and the low-low band at the coarsest scale. The low-low band contains the average image information whereas the other bands contain directional information due to spatial orientation. Higher absolute values of wavelet coefficients in the high bands correspond to salient features such as edges, lines, etc. Li et al. [43] perform fusion in the wavelet transform using a selection-based rule. Fusion is performed by selecting, at each location of the wavelet transform, the wavelet coefficient which has the maximum absolute value within a small analysis window, since it represents the presence of a dominant feature in the local area. This is followed by a consistency verification stage in which the selection of a hyperpixel in image A is changed to image B if a majority of the surrounding pixels are from image B. The authors showed that this scheme performs better than Laplacian pyramid based fusion due to the compactness, directional selectivity and orthogonality of the wavelet transform.

Wilson et al. [78] suggested an extension to wavelet-based fusion using a perceptual-based weighting. The wavelet coefficients from each sensor image are combined using a weighted average. The weighting of each coefficient is based on a salience measure that is determined by the contrast sensitivity function[11] of the human visual system. As a result, higher weighting is given to coefficients that are salient to the human visual system. The salience measure is computed using a neighborhood of wavelet coefficients, as the contrast sensitivity weighted sum of the Fourier transform coefficients of the wavelet coefficients. Wilson et al. showed that this technique produces fused images that are visually better than the fusion techniques based on the gradient pyramid (Section 2.5.3) or the contrast pyramid (Section 2.5.1).

2.6 Other Approaches to Image Fusion

Fechner and Godlewski [23] have used artificial neural networks to combine a low light level television (LLLTV) image with a FLIR image at the pixel-level. This technique uses the LLLTV image as the base image. Important details of the FLIR image, as selected by a multi-layer perceptron (MLP), are then superimposed on the base image. The MLP generates a binary mask which designates which regions in the FLIR image should pass into the fused image. The MLP is trained with input features such as edges, contrast, variance, etc., as well as task specific features such as roads and busyness[12] determined by small spatial filters. The output masks for training are prepared manually. Fechner and Godlewski showed that this technique has the potential to work better than selection-based fusion.

[11] Contrast sensitivity is the reciprocal of the threshold contrast required for a given spatial frequency to be perceived. The contrast sensitivity function is a plot of contrast sensitivity as a function of spatial frequency [17, 57, 73].

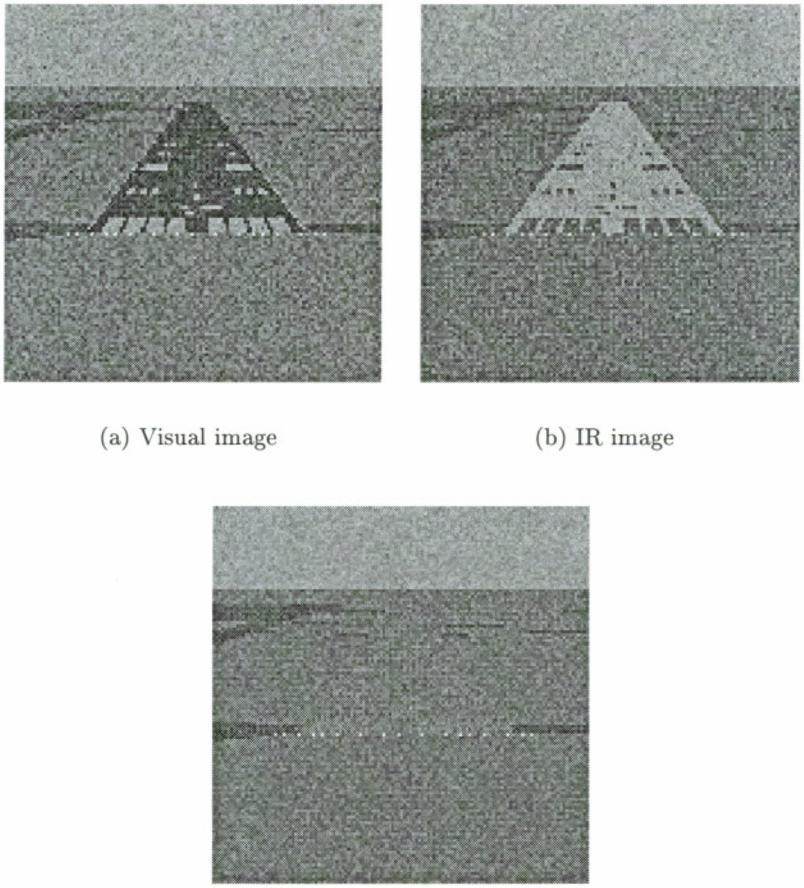
[12] For details, see the references within [23].

2.7 Shortcomings of Existing Fusion Approaches

Fusion by averaging can produce unsatisfactory results when there is a mismatch in image features in the sensor images (such as local polarity reversals and complementary features mentioned in Section 1.3). Features that appear with local polarity reversed contrast in the sensor images can appear cancelled out in the fused image as a result of averaging. An example of fusing images with mismatched features using averaging is shown in Figure 2.8. Figure 2.8(a) and 2.8(b) are simulated visible-band and IR images, respectively, of a runway scene. The runway surface in the IR image has reversed contrast relative to the visible-band image. The fusion result using averaging is shown in Figure 2.8(c). Averaging the pixels in the contrast-reversed runway region cancels out the runway features making it difficult to identify the runway in the fused image. Another drawback is caused by complementary image features that appear in one sensor image but not the others. Such features are rendered in the fused image at reduced contrast relative to the sensor image in which they appear. The issues relating to the mismatch in image features are discussed further in Chapter 3.

The main disadvantage of using image merging is that, even in ideal situations the signal from one image is completely ignored. In addition, it is difficult to automatically select regions of interest from the sensor images. Image merging can lead to false edges or discontinuities between regions taken from different images [68]. Such artifacts can create problems for a human observer viewing the fused image.

The various approaches to image fusion based on pattern-selection in the pyramid transform domain (i.e. the selection rule and variations of this rule) differ mostly in the image representation used for fusion. The selection-based approaches to fusion work well under the assumption that at each image location, only one of the sensor images provides the most useful information. This assumption is often not valid. For example, when infrared and visible-band images are to be fused, image features in the infrared image are often similar to those in the visible-band image but with reversed contrast. The strategy of selecting pyramid coefficients from one image or another degrades fusion results when the images contain features with reversed contrast. There is a possibility of



(a) Visual image

(b) IR image

(c) Fusion by averaging

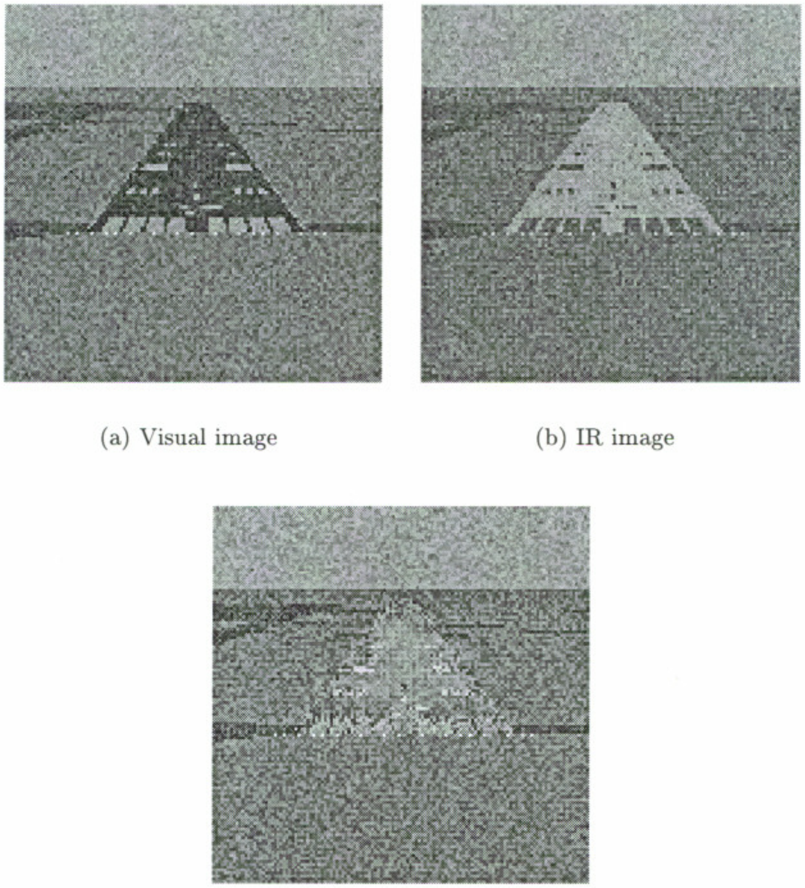
Figure 2.8: Cancellation of image features due to fusion by averaging

feature cancellation when the inverse pyramid transform is applied to obtain the fused image.

Most of the existing approaches do not explicitly discriminate between signal and noise. In selection-based techniques the salience metrics that determine the selection of features are sensitive to noise in the sensor images. The salience metrics are usually based either on the energy of the pyramid coefficients or the energy of coefficients in a small local neighborhood. When noise in the sensor images is high, it tends to get selected into the fused image. An example of fusion by selection using an area based selection rule on noisy images is shown in Figure 2.9. Figure 2.9(a) and 2.9(b) are simulated visible-band and infrared images of a runway scene. The result of fusion using selection in the Laplacian pyramid domain is shown in Figure 2.9(c). The salience of each hyperpixel in this case is computed as the sum of squared hyperpixel values in a 5×5 area surrounding the hyperpixel. This area based salience measure reduces the effect of noise. However, the fused image is still noisy because the noise spikes in the sensor images appear as salient hyperpixels to the selection algorithm.

The modifications to selection-based fusion approaches, which include techniques that perform a mix of averaging and selection (see Section 2.5.3) aim to overcome some of the drawbacks of the above techniques. However, the decision to average or select is dependent upon certain thresholds that are difficult to determine automatically since they are image-dependent. Any hard threshold will be sub-optimal when the sensor image characteristics change over time. Moreover, the weighting of the two sensors based on the match and salience metrics is somewhat ad-hoc.

Most of the existing fusion approaches deal with static fusion, i.e., fusion of a set of images of the same scene obtained from multiple sensors. Temporal information available in a sequence of images (video sequences) from the sensors is not utilized for the purpose of fusion. Moreover, a large part of the research on image fusion has focussed on choosing an appropriate image representation to facilitate better pattern-selection. An important issue that has not been given due emphasis is the characterization of noise in the imagery obtained from the sensors. As a result the fusion techniques based on selection are not capable of adapting to changing environmental conditions or changing noise characteristics.



(a) Visual image

(b) IR image

(c) Fusion by selection

Figure 2.9: Fusion by selection on noisy images

The noise based selection rule of Section 2.5.2 is a first step towards more robust fusion techniques. The explicit use of knowledge gained from the study of the human visual system has been lacking, as also the need for training the fusion algorithms on known data. Approaches such as the ones described in Sections 2.5.4 and 2.6 are beginning to address these issues.

2.8 Discussion

In this chapter, we reviewed important fusion techniques described in literature. We discussed the advantages as well as the shortcomings of these techniques. In this dissertation we will present a fusion approach that overcomes some of the shortcomings of the existing approaches. In the next chapter we model the process of sensor image formation and explicitly deal with noise, local polarity reversals and complementary features. Using this model, we develop a probabilistic fusion rule that combines the sensor images based upon the signal and noise content. The fusion rule has a provision to use prior image information about the imaged scene. In Chapter 4 we describe how the parameters of the model are obtained from the sensor images. We also show how temporal information from multiple frames can be used to estimate the sensor noise. Our proposed fusion solution is illustrated with several examples in Chapter 5.

Chapter 3

Model-based Approach to Image Fusion

3.1 Introduction

In this chapter we introduce our model-based probabilistic approach to fusion. We define an image formation model that is based upon a detailed analysis of images of the same scene captured by different sensors. This analysis explains the different relationships that exist between image features in multisensor imagery. In our image formation model, the sensor images are noisy, locally affine functions of a *true scene*. The model is an approximation of the nonlinear mapping that exists between the scene and the sensor images. Yet, it captures the important relationships between the features in the scene and the sensor images. The model is defined within a multiresolution pyramid representation (see Section 2.4).

The image formation model is then inverted within a Bayesian framework to provide either maximum likelihood or maximum a posteriori estimates of the true scene. These estimates constitute our rule for fusion of the sensor images. We relate these fusion rules to the existing fusion techniques (discussed in Chapter 2) and show that these rules address their drawbacks while retaining their advantages. The fusion rules require estimates of the parameters of the image formation model. This estimation process is described in Chapter 4.

Section 3.2, analyzes the fusion problem by examining the relationships between multisensor image features. Based on this analysis, we introduce our model for fusion in Section 3.3. Section 3.4 describes our Bayesian approach to fusion and derives the fusion rules based on maximum likelihood and maximum a posteriori estimation. The derived fusion rules are then compared with existing approaches. Section 3.5 summarizes the discussion in this chapter. Throughout this chapter, as in Chapter 2, we assume that the sensor images have been transformed to a conformal geometric representation and are perfectly registered.

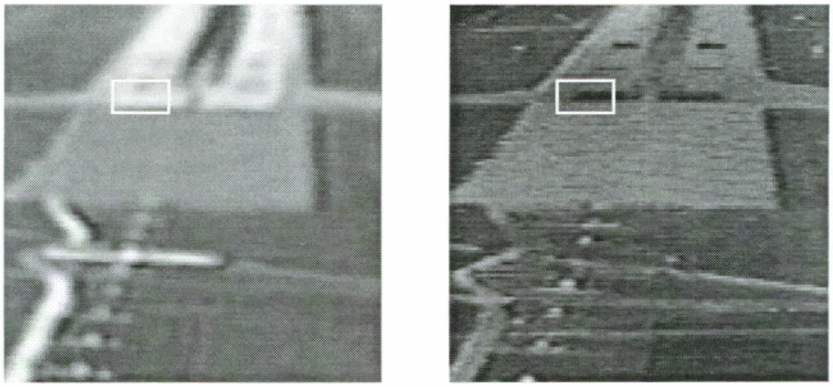
3.2 Fusion Problem Analysis

Visible-band, infrared and radar-based imaging sensors (e.g. TV, FLIR and MMWR) operate in different bands of the electromagnetic spectrum and have different spectral sensitivities. The sensing principles of these sensors are based on different physical phenomena. As a result, characteristics of the image data generated by each of these sensors are different. Objects and materials in a scene, environmental factors, and the sensing process, give rise to patterns or image features in the sensor images. Due to the differences in the sensors, the features in images of the same scene are likely to be different from one sensor image to another, but are usually closely related. However, the relationships between image features can sometimes be complex. In order to appropriately fuse images from multiple sensors, it is necessary to understand the nature of the relationships between image features at corresponding locations in different sensor images. The benefits obtained from fusion depend upon extracting knowledge about these relationships and meaningfully representing them in the fused image. To understand the relationships between image features, we analyzed images obtained from different types of sensors.

3.2.1 Relationships between multisensor image features

As stated above, due to differences in the sensing processes, different sensors generate distinct images of the same scene. Differences also arise due to the different material characteristics of the objects in the scene and their thermal properties. The differences are further accentuated by factors such as the time of the day and environmental conditions such as fog, cloud cover and rain.

The polarity of local contrast is often reversed between visible-band and IR images [16, 44] of the same scene. As a result, an IR image looks like the negative version of the



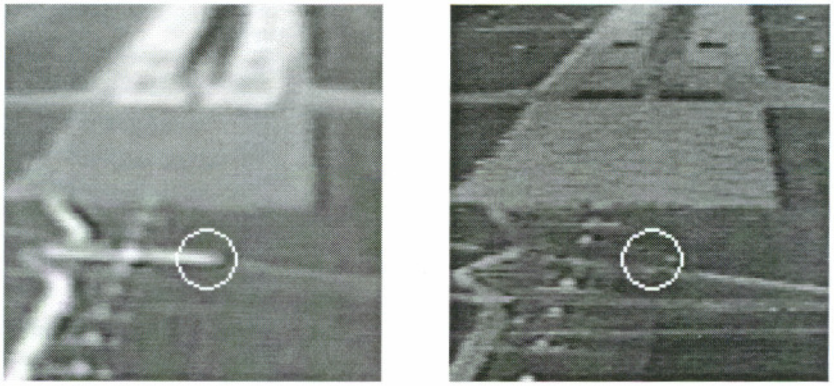
(a) Visible-band image

(b) Infrared image

Figure 3.1: Example of polarity reversal

visible-band image. However, the reversal in the polarity of contrast is not necessarily a global effect throughout the image. Sometimes only specific local patches or patterns in the infrared image have reversed polarity of contrast. We call this effect *local polarity reversal.* Figure 3.1 shows an example of this effect in the ALG application. Figure 3.1(a) and 3.1(b) are visible-band and IR images respectively that show a runway scene as an aircraft is approaching to land. A region that has a local polarity reversal is marked with white rectangles in both images. During daytime, an asphalt runway with white markings is imaged as a bright runway with dark markings in the IR image. This is because the asphalt runway becomes hot whereas the white paint of the markings stays relatively cool. However in the visible-band image the runway appears dark with bright markings. Such polarity reversals are often encountered when IR sensors are used.

IR and MMWR images may sometimes contain image features that are absent in visible-band images or vice versa [64]. Such disparities can be caused by thermal differences or strongly reflecting man-made objects in the scene. We call such sensor-specific features in the images *complementary features.* The reason these features are termed complementary is because they complement each other in tasks such as detection and recognition. Figure 3.2 shows an example of images containing complementary features. Figures 3.2(a) and 3.2(b) are the same images as in Figure 3.1 and show complementary features marked with white circles.



(a) Visible-band image (b) Infrared image

Figure 3.2: Example of complementary features

In addition to the discrepancies in the image features, the noise characteristics may be different for each sensor image. The source of the noise could be either the sensor itself or environmental factors such as fog or rain. The sensor images may thus have different signal-to-noise ratios. The noise characteristics may also change from region to region within the same sensor image. Figure 3.3 illustrates these effects, again with the help of an example from the ALG application. The image in Figure 3.3(a) is another image of the same scene captured by MMWR. The radar image is noisy compared to the infrared image. The signal-to-noise ratio in the radar image varies from one region to another.

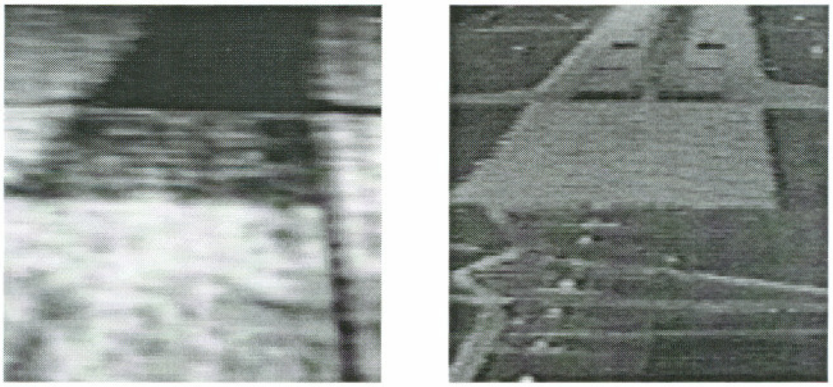
Based on the above analysis, image features in multisensor images can be generally categorized into the following components:

1. Common features

   These are the features that are common to all sensor images. Common features include features that appear similar as well as features that are polarity reversed. An appropriate combination of the common features increases the signal-to-noise ratio and is beneficial for fusion.

2. Complementary features

   Complementary features are those features that are unique to a particular sensor image. Such features convey important information about the scene being sensed and



(a) Radar image

(b) Infrared image

Figure 3.3: Example of noisy sensor images

can be used in detection, identification and guidance. Examples of complementary features are bright features that appear in infrared imagery due to hot objects.

3. Irrelevant complementary features

   Irrelevant complementary features contain information that is irrelevant to the task being performed and could cause undue interference to the human operator or vision system. For example, consider a shadow of an aircraft on a runway. The shadow cools a patch of the runway. Once the aircraft departs, this patch will appear dark in an IR image whereas there would be no such feature in a visible-band image. Isolation of irrelevant complementary features from complementary features generally requires domain knowledge and task specific processing.

4. Noise

   Noise is the randomly varying component that does not carry any information about the scene that is being captured by the sensors. Noise changes from sensor to sensor and from region to region within a sensor image.

It is important to note that the relationships between the image features are local in nature and change from location to location in the sensor images. The design of effective fusion algorithms is difficult because the optimality of combining different images depends on the local relationships between sensor images. For example, fusion based on averaging works well for regions in the sensor images which are essentially the same except for the additive noise (see Section 2.3.1). However, averaging can cause a reduction in contrast in the regions in the sensor images that contain complementary features. In this case, a simple local selection of the sensor with more salient features will preserve the contrast (see Section 2.4.1).

Due to the local nature of the relationships between the features in the sensor images, it is desirable that a fusion algorithm combine different local regions according to the local relationships. Furthermore, to maximize the benefits obtained from using multiple sensors, a fusion algorithm should enhance and display the common features, display the useful complementary features and suppress or eliminate noise and irrelevant features. In order to do that we would need to estimate each of these components in each sensor image.

3.3 Model-based Approach

A model-based approach to fusion allows the inclusion of knowledge about the sensor processes into a fusion algorithm [20]. Our approach to fusion is based on modeling the formation of the sensor images from the scene by defining an image formation model. The model provides a framework to incorporate knowledge about the relationships between the features in the sensor images and the variability caused by the noise. This model-based approach also allows the inclusion of additional information about the scene in the form of prior knowledge. Our image formation model explicitly accounts for the different components outlined in Section 3.2[1].

3.3.1 Image formation model

We assume that there exists an actual physical scene (true scene) s that is being imaged by multiple sensors. For example, for an application such as automatic landing guidance in aviation (see Section 1.2.1), s would be an image of the landing scene under conditions of uniform lighting, unlimited visibility and perfect sensors.

[1] Here, both relevant and irrelevant complementary features are treated as complementary features. Appendix C describes approaches to enable a human observer to discriminate between relevant and irrelevant complementary features.

The true scene gives rise to a sensor image through a nonlinear and noninvertible mapping. We approximate the mapping between the true scene and each sensor image by a locally affine transformation. This transformation is defined at every hyperpixel (see Section 2.4.1) of the multiresolution Laplacian pyramid. Using the true scene s, this transformation is given by,

$$a_i(\vec{l},t) = \beta_i(\vec{l},t)s(\vec{l},t) + \alpha_i(\vec{l},t) + \epsilon_i(\vec{l},t) \tag{3.1}$$

where,

$i = 1, \ldots, q$ indexes the sensors,

$\vec{l} \equiv (x, y, k)$ is the hyperpixel location,

where,

x, y are the pixel coordinates, and

k is the level of the pyramid,

t is the time,

a is the sensor image,

s is the true scene,

α is the sensor bias (captures the effects of dysfunctional sensor elements),

β is the sensor gain (captures local polarity reversals and complementary features),

ϵ is the (zero-mean) sensor noise having variance $\sigma^2_{\epsilon_i}$,

and we assume that s, a, β, α and ϵ are all real valued, and x, y, k and t are all discrete[2]. This is the image formation model or the sensor model[3]. The model describes how the

[2] The true scene is continuous. However, we assume point sampling with no aliasing.

[3] A commonly used image formation model [26] used in the field of image processing consists of

$$a_i(x,y) = \sum_{m=-M/2}^{M/2} \sum_{n=-N/2}^{N/2} h_i(m,n)s(x-m, y-n) + \epsilon_i(x,y)$$

where (x, y) is the pixel location and h is the blur caused by the imaging process. Our image formation model given by Equation (3.1) is a simplification of this model where we assume $h_i(m,n) = \beta_i$ for $m = 0, n = 0$ and is zero otherwise. That is, we assume that the effects of blur have already been corrected.

sensing elements give rise to the sensor data. The probabilistic nature of the model and the assumptions about the scene and the noise are described in detail in the next section. The image formation model is local because it is defined at each hyperpixel location at each time frame. We do assume, however, that the image formation parameters and sensor noise distribution vary *slowly* from one spatial location to another. Specifically, the parameters vary slowly on the spatiotemporal scales over which the true scene s may exhibit large variations. Hence, a particular set of parameters is considered to hold true over a spatial region of several square hyperpixels. We use this assumption implicitly when we estimate these parameters from the sensor images (see Chapter 4).

For the general case in which there are q sensors, the model in Equation 3.1 can be expressed in matrix form as,

$$\boldsymbol{a} = \boldsymbol{\beta} s + \boldsymbol{\alpha} + \boldsymbol{\epsilon} \tag{3.2}$$

where $\boldsymbol{a} = [a_1, a_2, \ldots, a_q]^{\mathrm{T}}$, $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_q]^{\mathrm{T}}$, $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_q]^{\mathrm{T}}$, s is a scalar and $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_q]^{\mathrm{T}}$. The reference to location and time has been dropped for simplicity and will not be made explicit henceforth unless necessary.

Note that the image formation model is a locally linear, first degree approximation of the actual, possibly nonlinear, mapping that exists between the visual information in the scene and the output generated by an imaging sensor. A more complete model would capture nonlinearities, and model other attributes such as sampling effects and low pass sensor characteristics (blur) [51].

The model in Equation (3.1) incorporates components that are important from the fusion point of view. Since the image formation parameters $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$, and the sensor noise covariance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ can vary from hyperpixel to hyperpixel, the model can express local polarity reversals, complementary features and spatial variation of sensor gain. In addition, the model explicitly accounts for the noise characteristics of each sensor.

3.3.2 Representation used for performing fusion

We employ a multiresolution pyramid representation of the sensor images based on the Laplacian pyramid transform. The advantages of using multiresolution representations for fusion are outlined in Section 2.4. The Laplacian pyramid transform decomposes an image into multiple levels such that each successive level is a bandpassed, subsampled and scaled version of the original image. Section 2.4.1 gives a detailed description of the Laplacian pyramid transform.

Although we have chosen the Laplacian pyramid as the representation for performing fusion, our model-based approach may also be applied to other representations. The focus of our technique is on modeling the relationships between the sensor images with an objective to fuse the sensor images in a principled manner. Any of a number of multiresolution representations may be used for this purpose.

3.4 Bayesian Fusion

We have defined a model that describes how the sensor images are obtained from the true scene. The goal of our fusion algorithm is to invert this model to obtain an estimate of the true scene from the sensor images. We use a Bayesian approach for estimating the true scene. Clark and Yuille [20] have described the advantages of Bayesian methods for data fusion, and applied such methods to several problems including stereo vision and shape from shading. Given the sensor intensities, $\boldsymbol{a}$, we estimate the true scene s using a Bayesian framework. We assume that the *apriori* probability density function of s is a Gaussian with (locally varying) mean s_0 and (locally varying) variance σ_s^2,

$$\mathcal{P}(s) = \frac{1}{(2\pi\sigma_s^2)^{\frac{1}{2}}} \exp\left[-\frac{1}{2}\frac{(s-s_0)^2}{\sigma_s^2}\right] \tag{3.3}$$

where the location dependence of the parameters is implicit. The noise density is also assumed to be Gaussian with zero mean and a (locally varying) diagonal covariance $\boldsymbol{\Sigma_\epsilon} = \mathrm{diag}[\sigma_{\epsilon_1}^2, \sigma_{\epsilon_2}^2, \ldots, \sigma_{\epsilon_q}^2]$, where $\sigma_{\epsilon_i}^2$ is the noise variance at a particular hyperpixel location of the i^{th} sensor. Noise in one sensor image is assumed to be uncorrelated with noise in other sensor images and uncorrelated with the scene s.

The density on the sensor images, conditioned on the true scene is,

$$\mathcal{P}(\boldsymbol{a}|s) = \frac{1}{(2\pi)^{\frac{q}{2}}\,|\boldsymbol{\Sigma_\epsilon}|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\boldsymbol{a}-\boldsymbol{\beta}s-\boldsymbol{\alpha})^{\mathrm{T}}\boldsymbol{\Sigma}_\epsilon^{-1}(\boldsymbol{a}-\boldsymbol{\beta}s-\boldsymbol{\alpha})\right] \tag{3.4}$$

The conditional density $\mathcal{P}(a|s)$ corresponds to the image formation model which describes how the scene gives rise to the sensor data depending upon the sensing devices and other factors. The marginal density $\mathcal{P}(\boldsymbol{a})$ describes the distribution of the sensor images $\boldsymbol{a}$ (therefore we also refer to it as the model distribution) and is given by,

$$\begin{aligned}\mathcal{P}(\boldsymbol{a}) &= \int_{-\infty}^{\infty} \mathcal{P}(\boldsymbol{a}|s)\mathcal{P}(s)ds \\ &= \frac{1}{(2\pi)^{\frac{q}{2}}\left|\mathbf{C}\right|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\boldsymbol{a}-\boldsymbol{\mu}_m)^{\mathrm{T}}\mathbf{C}^{-1}(\boldsymbol{a}-\boldsymbol{\mu}_m)\right] \qquad (3.5)\end{aligned}$$

where $\boldsymbol{\mu}_m$ is the model mean given by

$$\boldsymbol{\mu}_m = \boldsymbol{\beta} s_0 + \boldsymbol{\alpha} \qquad (3.6)$$

and $\mathbf{C}$ is the model covariance given by

$$\mathbf{C} = \sigma_s^2 \boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\epsilon} \qquad (3.7)$$

The posterior density of the latent variables s, given the observed sensor data $\boldsymbol{a}$, is obtained by Bayes' rule,

$$\mathcal{P}(s|\boldsymbol{a}) = \frac{\mathcal{P}(\boldsymbol{a}|s)\mathcal{P}(s)}{\mathcal{P}(\boldsymbol{a})} \qquad (3.8)$$

Substituting Equations (3.3), (3.4) and (3.5) and simplifying,

$$\mathcal{P}(s|\boldsymbol{a}) = \frac{1}{\sqrt{2\pi\mathbf{M}^{-1}}} \exp\left[-\frac{1}{2}(s-\mu_{s|a})\mathbf{M}(s-\mu_{s|a})\right] \qquad (3.9)$$

where,

$$\mathbf{M}^{-1} = \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta} + \frac{1}{\sigma_s^2}\right]^{-1} \qquad (3.10)$$

is the posterior covariance, and,

$$\mu_{s|a} \equiv E\{s|a\} = \mathbf{M}^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a}-\boldsymbol{\alpha}) + \frac{s_0}{\sigma_s^2}\right\} \qquad (3.11)$$

is the conditional mean. The operator $E\{\cdot\}$ denotes expectation.

3.4.1 Maximum likelihood estimate of fused image

The maximum likelihood (ML) estimate of the scene can be obtained from the density of the sensor images conditioned on the scene, given in Equation (3.4). Maximizing the logarithm of $\mathcal{P}(\boldsymbol{a}|s)$ with respect to s yields

$$\hat{s}_{\mathrm{ML}} = \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta}\right]^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a}-\boldsymbol{\alpha})\right\} \tag{3.12}$$

This is the maximum likelihood rule for fusion. For two sensors, this fusion rule reads

$$\begin{aligned} \hat{s}_{\mathrm{ML}} &= \frac{\frac{\beta_1(a_1-\alpha_1)}{\sigma_{\epsilon_1}^2}+\frac{\beta_2(a_2-\alpha_2)}{\sigma_{\epsilon_2}^2}}{\frac{\beta_1^2}{\sigma_{\epsilon_1}^2}+\frac{\beta_2^2}{\sigma_{\epsilon_2}^2}} \\ &= \sum_{i=1}^{2} w_i(a_i-\alpha_i) \end{aligned} \tag{3.13}$$

The fused image $\hat{s}_{\mathrm{ML}}$ is a weighted linear combination of the sensor images. The weights w_i change from hyperpixel to hyperpixel and through time as a result of the spatiotemporal variations in $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$.

3.4.2 Maximum a posteriori estimate of fused image

When prior knowledge about the scene is available, the maximum a posteriori (MAP) estimate of the true scene can be obtained. The MAP estimate, $\hat{s}_{\mathrm{MAP}}$, of the true scene is obtained by maximizing the logarithm of the posterior density in Equation (3.9) with respect to s. For our assumption of Gaussian distributions, this estimate is simply the posterior mean from Equation (3.11).

$$\hat{s}_{\mathrm{MAP}} = E\{s|a\} = \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta}+\frac{1}{\sigma_s^2}\right]^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a}-\boldsymbol{\alpha})+\frac{s_0}{\sigma_s^2}\right\} \tag{3.14}$$

This is the maximum a posteriori rule for fusion. For two sensors, it reads

$$\begin{aligned} \hat{s}_{\mathrm{MAP}} &= \frac{\frac{\beta_1(a_1-\alpha_1)}{\sigma_{\epsilon_1}^2}+\frac{\beta_2(a_2-\alpha_2)}{\sigma_{\epsilon_2}^2}+\frac{s_0}{\sigma_s^2}}{\frac{\beta_1^2}{\sigma_{\epsilon_1}^2}+\frac{\beta_2^2}{\sigma_{\epsilon_2}^2}+\frac{1}{\sigma_s^2}} \\ &= \sum_{i=1}^{2} w_i(a_i-\alpha_i)+w_0 s_0 \end{aligned} \tag{3.15}$$

From Equation (3.15) one can observe that the fused image $\hat{s}$ is a weighted linear combination of the sensor images and the prior image of the scene. As in the maximum likelihood case, the weights w_i change from hyperpixel to hyperpixel and through time as a result of the spatiotemporal variations in $\boldsymbol{\beta}$, $\boldsymbol{\Sigma_\epsilon}$ and σ_s^2.

The MAP fusion rule is equivalent to the ML fusion rule when $\sigma_s^2 = \infty$, i.e., with no prior knowledge. In Section 3.4.3 we discuss how the fusion rules achieve the goals for fusion in the presence of polarity reversed and complementary features as well as in noise. The fused image at each time frame can be computed using either Equation (3.14) or (3.12) only if the parameters $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\Sigma_\epsilon}$ are known. In reality, the parameters $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\Sigma_\epsilon}$ are unknown and need to be estimated from the sensor images. In Chapter 4 we describe different techniques to estimate these parameters, and in Chapter 5 we present results using the ML and MAP fusion rules.

Interpretation of the priors

The parameters s_0 and σ_s^2 denote prior knowledge about the scene. For example, in the ALG application in aviation, prior knowledge about the landing scene may be available in the form of an ensemble of scenes of previous landing approaches to the runway. In this case, s_0 and σ_s^2 are given by the mean and the variance of this ensemble. One cause for variations in the ensemble is that thermal properties of materials in the scene affect the appearance of the scene at different times of the day. Another cause is registration errors in the ensemble. Sometimes, prior knowledge may be available from a terrain database of the scene. The parameter σ_s^2 determines the weighting given to the prior image in the fused image. For example, in Equation (3.15), if σ_s^2 is high then the weighting given to s_0 is decreased and the weighting given to the sensor images is increased. Similarly, a low value of σ_s^2 increases the contribution of s_0 to the fused image and decreases the contribution of the sensor images. The parameter σ_s^2 can therefore be used as a knob to control the amount of prior knowledge that should pass on to the fused image.

3.4.3 Comparison of fusion rule with existing fusion techniques

The fusion rules in Equations (3.14) and (3.12) can properly respond to situations that tax averaging and selection methods. This can be easily understood from Equation (3.15). As an example, consider the case where the second sensor has a polarity reversal. β_2 is then negative and the contribution of the second sensor is *negated* before it is added to the contribution of the first sensor. On the other hand if the polarity is the same, then the two sensor contributions are added. Thus, in either case, the fusion rule performs the averaging operation using the correct polarity to add or subtract the sensor contributions. This ensures that there is no loss of detail due to reduction in contrast as in simple averaging, yet there is an increase in the signal-to-noise ratio due to averaging.

Now consider the case of complementary features, where a feature is missing from sensor 1. This corresponds to $\beta_1 = 0$. The model compensates by accentuating the contribution from sensor 2. In this case the result is the same as selection of the sensor in which the feature is present. Thus, when complementary features are present, the fusion rule resembles the feature selection techniques.

Finally, consider the case where the sensor images are noisy. If the first sensor has high noise (large $\sigma^2_{\epsilon_1}$) content, its contribution to the fused image is attenuated. At the same time, the contribution of the second sensor is increased. If both sensors are equally noisy, then the noise variance weights their contributions equally. Since the fusion rule attenuates the contribution of the sensors which are noisy, it overcomes the drawback of the feature selection techniques which often tend to select the more noisy sensor.

Thus the MAP fusion rule not only responds properly to situations that challenge the previous fusion techniques, but it also retains the advantages of the existing fusion techniques (this is true for the ML fusion rule as well). An important difference as compared to existing fusion techniques is that these fusion rules explicitly account for noise in the sensor images. In the next chapter we will describe how ML and MAP fused images can be obtained using spatial as well as temporal information from the sensor images to estimate the model parameters.

3.5 Summary

In this chapter, we analyzed the relationships between image features in multisensor imagery to understand the fusion problem. This analysis provided an understanding of the problems that need to be addressed by a fusion solution and motivated our approach to fusion. The approach is based on a probabilistic model which approximates the sensor images as local affine functions of the true scene. The model is based on a multiresolution representation of the images which decomposes the images into constituent patterns or features. The parameters of these affine functions describe how features in the sensor images are related to features in the scene. A Bayesian framework provides either maximum likelihood or maximum a posteriori estimates of the true scene from the sensor images — these estimates are our rules for fusion. We described how the fusion rules extend and improve upon the existing fusion techniques such as averaging and selection. However, to compute the fused image using the fusion rules, one needs to estimate the parameters of the model. The estimation of these parameters is described in Chapter 4 and fusion results are presented in Chapter 5.

Chapter 4

Estimation of Model Parameters

4.1 Introduction

In Chapter 3, we derived fusion rules based on a probabilistic image formation model; the model being a set of locally affine functions that map the scene into the sensor images. The fusion rules generate the fused pyramid as a locally weighted linear combination of the pyramids of the sensor images. The weighting takes into account the strength of the signal attributed to the scene as well as the noise in the sensor images. We showed how these rules can retain the advantages of existing fusion techniques based on selection and averaging, while overcoming their drawbacks.

The weights of the probabilistic combination rules for fusion depend upon the parameters of the image formation model — the affine parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, and the noise covariance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$. These parameters are typically not known and need to be estimated from the sensor images and a priori knowledge about the scene. In this chapter we describe techniques for estimating these parameters. Section 4.2 describes the concept of using a local analysis window to estimate the model parameters. Section 4.3 describes two methods for estimating the noise variance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ in the sensor images. The first method estimates the noise variance using one image from each sensor. The second method utilizes multiple images from each sensor. In Section 4.4 we describe a simple approach to estimate the affine parameters from a reference image of the scene. This approach uses regression to estimate the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, but has drawbacks. In Section 4.5 we describe a probability model for sensor image data in the analysis window. Based on this model we derive estimates of the affine parameters using least squares factor analysis. Section 4.6 shows that with the parameters estimated, the probabilistic fusion rules of Chapter 3 are closely related to local principal components analysis. Section 4.7 concludes with a summary of the material covered in this chapter.

4.2 Local Analysis Window for Parameter Estimation

The image formation model of Equation (3.1) is defined for every hyperpixel at each level of a multiresolution Laplacian pyramid. However, with the exception of discontinuities, the relationship between two sensor images (e.g., local polarity reversals, complementary features) generally varies slowly between neighboring spatial locations (a few pixels apart, say 5 to 10 pixels) within an image. Therefore we assume that the image formation parameters and the noise characteristics vary *slower* with spatial location than the scene, within each pyramid level. Specifically, the parameters vary slowly over the spatiotemporal scales over which the true scene s may exhibit large variations. Hence, a particular set of model parameters ($\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$) can be considered to hold true over a spatial region of several square hyperpixels.

To estimate the model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, we define a local analysis window, $\mathcal{R}_{\mathcal{L}}$, of $h \times h$ hyperpixels. We assume that the parameters are constant over this analysis window and estimate the parameters using the statistics computed from all the hyperpixels in the window. Essentially, this is an assumption of spatial ergodicity, where ensemble averages are replaced by spatial averages (carried out locally over regions in which the statistics are approximately constant). Ideally, $\mathcal{R}_{\mathcal{L}}$ should be small enough so that the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are indeed constant in the window. However, the analysis window should be large enough to contain enough sensor data to estimate the parameters reliably. For the results in this thesis we have chosen as $\mathcal{R}_{\mathcal{L}}$ a region of 5×5 hyperpixels around the hyperpixel for which parameters are to be estimated. This choice is a tradeoff between the two conflicting requirements on the local analysis window.

Similarly, we also define an analysis window to estimate the noise variance in the sensor pyramids. To estimate noise variance from a single image from each sensor, we assume that the noise is identically distributed at all spatial locations in a pyramid level.

The noise variance at a pyramid level is then estimated from the statistics of local spatial variance at that level(see Section 4.3.1). When multiple video frames from each sensor are available, we obtain an estimate of the noise variance at each hyperpixel (see Section 4.3.2). We assume that the noise is identically distributed at the same object location in the multiple frames. Therefore, if motion compensated successive frames are available, the noise variance can be estimated using hyperpixels in the successive frames at the same physical object location. In addition, we assume that the distribution of noise varies slowly spatially. Hence, noise variance estimates in a region of 5×5 hyperpixels can be averaged to obtain robust estimates at each hyperpixel.

4.3 Estimation of Noise Variance

We describe two techniques for estimating the variability due to the noise in the sensor images. The first technique can be applied when only a single image is available from each sensor. The second technique is based on the assumption that multiple motion compensated video frames are available from each sensor.

4.3.1 Estimation using single frame

In this technique we obtain an estimate of the noise variance using a single frame (i.e. a single image) from each sensor. This technique is based on two assumptions. The first assumption is that the distribution of noise is identical at different spatial locations at a particular level of a sensor pyramid. The second assumption is that the sensor image contains mostly flat regions with few discontinuities in the form of edges and lines. This technique can be used to estimate the noise variance when only one image frame is available from each sensor image, as long as the assumptions stated above are valid. An independent estimate of the noise variance is obtained for each pyramid level in two steps.

The first step consists of estimating the spatial variance in small spatial regions of the image at the pyramid level of interest. The contribution to spatial variance in a small spatial region comes from two sources — the variance due to the noise and the variance due to edges and lines in the scene. In flat regions in the scene the spatial variance is

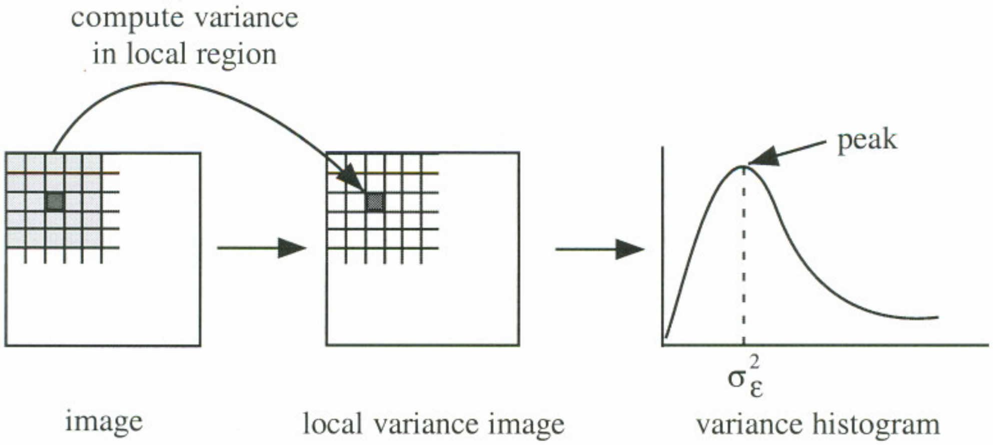


Figure 4.1: Schematic diagram of noise estimation using a single frame

almost entirely due to noise. In other regions the spatial variance is from both sources. We estimate the spatial variance at each point in the image using a 5×5 region surrounding the point as shown in Figure 4.1. This operation produces a spatial variance image of the same size as the pyramid level.

The second step consists of estimating the noise variance by analyzing the distribution of variance values in the spatial variance image. We plot a histogram of the variance values as shown in Figure 4.1. If the image was flat throughout, then the histogram would correspond to the distribution of noise variance estimates. In that case the noise variance estimate could be obtained by computing the mean of the distribution. However, since the image also contains edges and lines, the histogram will resemble that of Figure 4.1, and the mean of the distribution would overestimate the noise variance. We estimate the noise variance as the spatial variance corresponding to the mode (peak) of the distribution. Note that this approximation tends to underestimate the noise variance in the case where the image is essentially flat with very few edges.

The above two steps are repeated to obtain an estimate of noise variance at each level of the Laplacian pyramid for all the sensors. Figure 4.2 shows the local variance images and the variance histograms of level 0 of the Laplacian pyramid of two sensor images corrupted with additive Gaussian noise. The noise covariance matrix $\mathbf{\Sigma}_\epsilon$ is computed

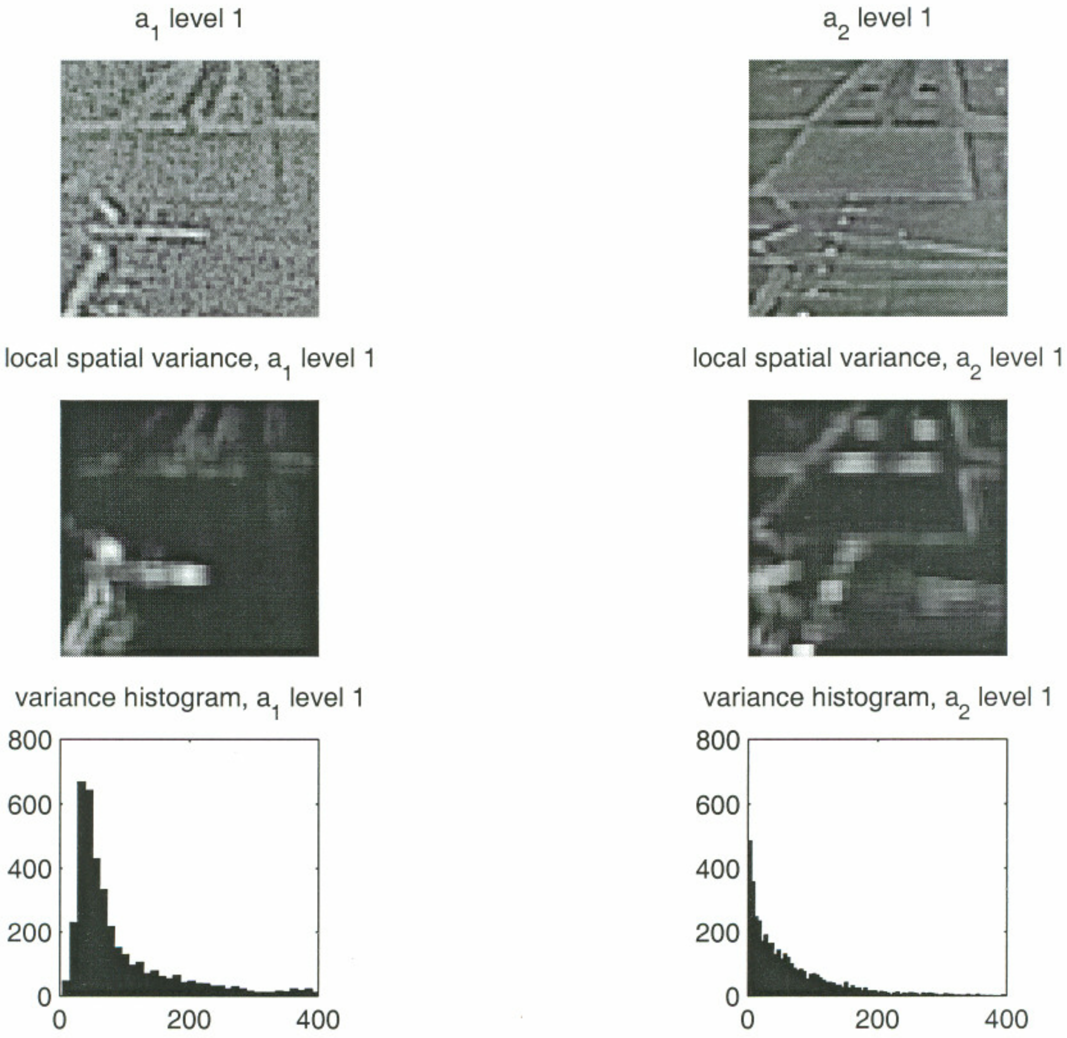


Figure 4.2: Noise estimation using a single frame

from the noise variance of each sensor as defined in Section 3.4. Although this method is useful to estimate the noise from just one image, its drawback is that it obtains a single estimate of noise variance for a pyramid level. Hence, this method is not appropriate for sensors such as MMWR in which the noise distribution is likely to change from one spatial region to another within a pyramid level.

4.3.2 Adaptive estimation of noise using multiple frames

When a sequence of video frames are available from each sensor, we employ an alternative approach that uses multiple frames to estimate the noise characteristics at each hyperpixel in the Laplacian pyramid. This approach consists of adaptively estimating the noise variance from multiple frames. Adaptive estimation allows for tracking the change in the noise characteristics over time.

We assume that the noise is identically distributed at the same object location in multiple frames. We further assume that successive video frames from the sensors can be motion compensated (i.e., registered, see Appendix B) so that identical object locations in multiple frames refer to the the same spatial locations. Several techniques for motion compensation are available in video processing and machine vision literature [7, 63, 65][1].

We compute the variance due to noise at each hyperpixel in the Laplacian pyramid. This approach is advantageous when the noise distribution is likely to change from region to region. We decompose the motion compensated frames into Laplacian pyramids as illustrated in Figure 4.3. We then compute the noise variance at each location in the pyramid from hyperpixels at that location in successive frames. The variance computation is performed adaptively by recursively estimating the average and the sum of squares at each hyperpixel. The estimate of the average value at each location is computed using the previous estimate of the average and the hyperpixel value at that location in the current frame. For the initial t_0 frames, this estimation utilizes all available frames to compute the average. For subsequent frames, the estimate of the average at each location is computed using an exponential decay in the form of a leaky integrator to forget the effect of past

[1] In the aviation example of Chapter 1, a knowledge of the motion of the platform containing the sensors can aid in motion compensation.

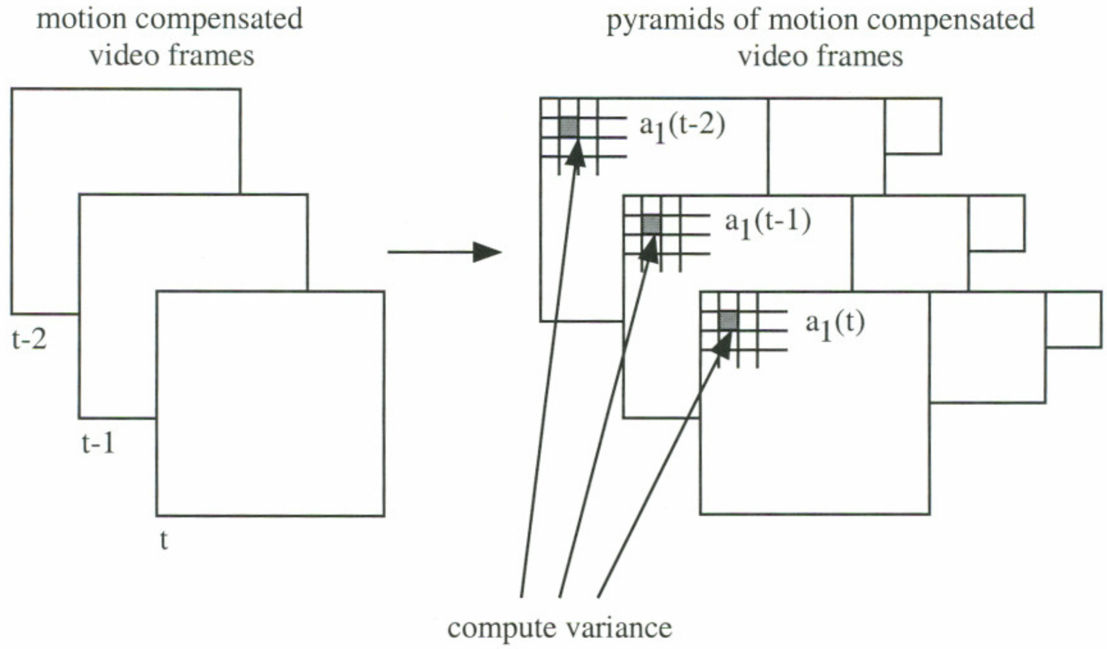

motion compensated video frames
t-2
t-1
t
pyramids of motion compensated video frames
a1(t-2)
a1(t-1)
a1(t)
compute variance

Figure 4.3: Noise estimation using multiple frames

frames,

$$\overline{a_i}(t) = \begin{cases} \frac{t-1}{t}\overline{a_i}(t-1) + \frac{1}{t}a_i(t) & \text{for } 1 < t \leqslant t_0 \\ \\ \frac{t_0-1}{t_0}\overline{a_i}(t-1) + \frac{1}{t_0}a_i(t) & \text{otherwise} \end{cases} \tag{4.1}$$

where $a_i(t)$ is the value at a particular location (here we have dropped the notation referring to location) of sensor i at time t, $\overline{a_i}(t)$ is the estimate of the average value at that location at time t, $\overline{a_i}(0) = a_i(0)$, and $t_0 > 1$ is a time constant. The average sum of squares is also estimated in a similar manner,

$$\overline{a_i^2}(t) = \begin{cases} \frac{t-1}{t}\overline{a_i^2}(t-1) + \frac{1}{t}a_i^2(t) & \text{for } 1 < t \leqslant t_0 \\ \\ \frac{t_0-1}{t_0}\overline{a_i^2}(t-1) + \frac{1}{t_0}a_i^2(t) & \text{otherwise} \end{cases} \tag{4.2}$$

with $\overline{a_i^2}(0) = a_i^2(0)$. The variance due to the noise at each hyperpixel for each sensor is then computed as

$$\sigma^2_{\epsilon_i}(t) = \overline{a_i^2}(t) - \overline{a_i}^2(t) \tag{4.3}$$

The noise covariance matrix for q sensors is then given by

$$\mathbf{\Sigma_\epsilon} = \begin{bmatrix} \sigma^2_{\epsilon_1} & 0 & \dots & 0 \\ 0 & \sigma^2_{\epsilon_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma^2_{\epsilon_q} \end{bmatrix} \tag{4.4}$$

A separate estimate of the noise covariance $\mathbf{\Sigma_\epsilon}$ is obtained at every hyperpixel location within each sensor pyramid. As discussed in Section 4.2, the estimates of noise variance of neighboring hyperpixels are usually similar. Hence, the noise variance estimates over a 5 × 5 neighborhood can be averaged to obtain an estimate for the noise variance at each hyperpixel.

The constant t_0 determines the forgetting factor or the effective sequence length used to compute the variance. A lower value of t_0 results in a shorter effective length, and the effect of past frames is quickly reduced. Whereas, a higher value of t_0 results in a longer effective length and the effect of past frames is slowly reduced. The choice of t_0 should take into account the dynamics of the changes in the image features due to motion, from frame to frame. For example, if the platform motion is fast, motion compensation will cause artifacts due to old scene information moving out of the video frames. Consider a sensor moving away from a scene. If the previous frame is motion compensated with respect to the current frame, it will have blank patches corresponding to the new scene regions that appear due to the away motion in the current frame. These artifacts are prominent for video frames that are further apart in time (say 4 frames apart). In our experiments we used $t_0 = 5$, so that the influence of frames prior to the last 5 video frames (i.e., 1/6th of a second, if there are 30 frames per second) on the estimation of noise variance, is rapidly reduced.

Discussion

We compared the single frame noise estimation technique with the adaptive estimation given below on a image sequence obtained by adding Gaussian noise to a simulated image. One frame of this sequence is shown in Figure 5.6(a). Both the assumptions made in the single frame estimation were valid for an image frame from this noisy sequence. The variance estimate from the single frame estimation was comparable to the mean of the variance estimates obtained from the adaptive estimation over the pyramid level. However, at higher pyramid levels, the single frame estimation overestimated the noise, as the number of hyperpixels available for estimation was small. The adaptive estimation over multiple frames generates a variance estimate at each hyperpixel location, and therefore is capable of providing finer control over the weighting given to the sensor images in the probabilistic fusion rules.

4.4 Reference Image Approach for Estimating Model Parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

One approach to estimate the parameters $\boldsymbol{\beta}$ (i.e. $\beta_1, \beta_2, \dots, \beta_q$) and $\boldsymbol{\alpha}$ (i.e. $\alpha_1, \alpha_2, \dots, \alpha_q$) of the image formation model, is to correlate the sensor images with a known reference image of the scene. The reference image can be a prior image of the scene, for example an image of the scene obtained from a terrain database. In the absence of a prior image, one of the sensor images can be considered as the reference image. The choice of which sensor image to use as the reference image may depend on the application domain. For example, in ALG a visible-band image would be preferred since it generally approximates the desired scene. Let the reference image be represented by d. Then from Equation 3.1, a hyperpixel a_i of the i^{th} sensor image a can be described in terms of the hyperpixel of the reference image as,

$$a_i = \beta_i d + \alpha_i + \epsilon_i \tag{4.5}$$

where the notation referring to location and time has been dropped. As explained in Section 4.2, we assume that the parameters β_i and α_i are constant in a 5×5 hyperpixel spatial neighborhood. These parameters are then obtained by regression by minimizing the squared error E given by

$$E = \sum_{n=1}^{N} (a_{in} - \beta_i d_n - \alpha_i)^2 \tag{4.6}$$

where $N = 25$ is the number of hyperpixels in a 5×5 neighborhood surrounding the hyperpixel for which the parameters β_i and α_i have to be estimated. The parameter estimates are obtained by minimizing E. Differentiating E with respect to α_i and β_i and equating to zero gives,

$$\alpha_i = \frac{1}{N}\sum_{n=1}^{N} a_{in} - \beta_i \frac{1}{N}\sum_{n=1}^{N} d_n \tag{4.7}$$

and,

$$\beta_i = \frac{\frac{1}{N}\sum_{n=1}^{N} a_{in} d_n - \frac{1}{N}\sum_{n=1}^{N} a_{in} \frac{1}{N}\sum_{n=1}^{N} d_n}{\frac{1}{N}\sum_{n=1}^{N} d_n^2 - \frac{1}{N}\sum_{n=1}^{N} d_n \frac{1}{N}\sum_{n=1}^{N} d_n} \tag{4.8}$$

The parameters are thus obtained by regressing the sensor images onto the reference image. If one of the sensor images is used as a reference, then the parameters α_i and β_i for that sensor will be zero and one respectively since the parameters are obtained by regressing this sensor image onto itself. Once the parameters β_i and α_i and the noise covariance are estimated at each hyperpixel, the fused image is obtained using the fusion rules derived in Section 3.4.2.

This approach of estimating the affine parameters using a reference image has drawbacks. The estimation of the affine parameters by regression is sensitive to the noise in the reference image. If a sensor image used as a reference is noisy, the parameter estimates are also noisy. Another drawback of this approach relates to complementary features present in a sensor image but absent in the reference image. Since, the parameter β_i is estimated by regressing a_i on to d, the value of β_i is close to zero when there is a feature in a_i that is absent in d. Consequently, the graylevel intensity component of a_i that is absent in d is assigned to α_i. Although α_i will contain the complementary features in the sensor image that is missing in the reference image, these features will not appear in the fused image if

β_i is close to zero (from Equation (3.15), a sensor does not contribute to the fused image if β_i is zero). In this situation, complementary features appear with reduced contrast in the fused image.

4.5 Probabilistic Approach for Estimating Model Parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

We now describe an approach to estimate the affine parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ using the probabilistic image formation model described in Chapter 3. The image formation model defined in Section 3.3 is

$$\boldsymbol{a}(\vec{l}\,) = \boldsymbol{\beta}(\vec{l}\,)\ \ s(\vec{l}\,)\ +\ \boldsymbol{\alpha}(\vec{l}\,)\ +\ \boldsymbol{\epsilon}(\vec{l}\,)\ \ , \tag{4.9}$$

where $\boldsymbol{a}$ corresponds to the sensor images, s is the true scene, $\boldsymbol{\epsilon}$ is the noise, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are the affine parameters that capture the sensor gain and the sensor offset respectively, and we have reintroduced the notation referring to the location $\vec{l}$ in the pyramid.

$$\vec{l} \equiv (x, y, k) \tag{4.10}$$

with k the level of the pyramid and (x, y) the location of the hyperpixel at that level.

We now briefly review the probabilistic framework of the image formation model, noting the dependence on the location $\vec{l}$ wherever necessary. We assumed that the a priori probability density of s is Gaussian with mean $s_0(\vec{l}\,)$ and variance $\sigma_s^2(\vec{l}\,)$ as given by Equation (3.3). We also assumed that the noise density is Gaussian with zero mean and covariance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}(\vec{l}\,)$. Therefore the density on the sensor image hyperpixels conditioned on the true scene s at the location $\vec{l}$, $\mathcal{P}(\boldsymbol{a}|s, \vec{l}\,)$ is Gaussian with mean

$$\boldsymbol{\beta}(\vec{l}\,)\ s(\vec{l}\,)\ +\ \boldsymbol{\alpha}(\vec{l}\,) \tag{4.11}$$

and covariance

$$\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}(\vec{l}\,)\ , \tag{4.12}$$

as given by Equation (3.4). From Equation (3.5), we also know that the marginal density on the sensor images at the location $\vec{l}$, $\mathcal{P}(\boldsymbol{a}|\vec{l}\,)$ is also Gaussian with mean

$$\boldsymbol{\mu}_m(\vec{l}\,) = \boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,)\ , \tag{4.13}$$

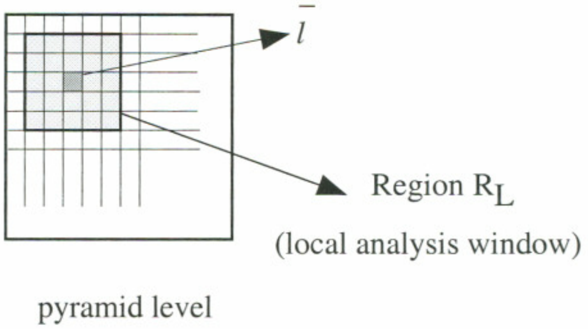


Figure 4.4: Region $\mathcal{R}_{\mathcal{L}}$ used to obtain the probability distribution of the sensor hyperpixels $\boldsymbol{a}$ for parameter estimation

and covariance

$$\mathbf{C}(\vec{l}\,) \;=\; \boldsymbol{\beta}(\vec{l}\,)\,\boldsymbol{\beta}^{\mathrm{T}}(\vec{l}\,)\,\sigma_s^2(\vec{l}\,) \;+\; \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}(\vec{l}\,) \;\;. \tag{4.14}$$

The image formation model fits the framework of the *factor analysis* model in statistics [3, 29, 67]. Factor analysis refers to a number of statistical techniques for the resolution of a set of observed variables in terms of a smaller number of variables called factors or latent variables. Factor analysis attempts to explain the correlations between the set of variables, and yields the minimum number of underlying factors (linear combinations of observed variables) that contain *all* the essential information about the linear relationships between the observed variables [38]. In our image formation model, the hyperpixel values of the true scene s are the latent variables or common factors, $\boldsymbol{\beta}$ contains the factor loadings, and the sensor noise $\boldsymbol{\epsilon}$ values are the independent factors. Estimating the parameters $\boldsymbol{\beta}$ is then equivalent to estimating the factor loadings from the observations $\boldsymbol{a}$.

Following the discussion in Section 4.2, in order to estimate the model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ at a location $\vec{l}$ in the pyramid we assume that the parameters are constant over a small region $\mathcal{R}_{\mathcal{L}}$ shown in Figure 4.4. The region $\mathcal{R}_{\mathcal{L}}$ constitutes a local analysis window of 5×5 hyperpixels. Assuming ergodicity, we replace ensemble averages by spatial averages. The factor analysis parameter estimation of $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ involves expressing the first and second order statistics of the observed sensor data in the local analysis window in terms of the parameters of the image formation model. To do this, one needs to obtain the probability density of the sensor image hyperpixels over the local analysis window defined by the region $\mathcal{R}_{\mathcal{L}}$.

The probability density function $\mathcal{P}(\boldsymbol{a}|\vec{l}\,)$ (see Equation (3.5) and Equations (4.13) and (4.14)) above) specifies the distribution of $\boldsymbol{a}$ at the location $\vec{l}$. To obtain the distribution of $\boldsymbol{a}$ over the region $\mathcal{R}_{\mathcal{L}}$, one must integrate over the region $\mathcal{R}_{\mathcal{L}}$. Therefore, the distribution of the sensor hyperpixels $\boldsymbol{a}$ over the region $\mathcal{R}_{\mathcal{L}}$ is given by,

$$\mathcal{P}(\boldsymbol{a}|\mathcal{R}) = \int_{\mathcal{R}} \mathcal{P}(\boldsymbol{a}|\vec{l}\,)\;\; \mathcal{P}(\vec{l}\,)\;\; d\vec{l}\;\; , \tag{4.15}$$

where $\mathcal{P}(\vec{l}\,)$ is the probability of sampling from the area $d\vec{l}$ about $\vec{l}$ in the region $\mathcal{R}_{\mathcal{L}}$, and the random variable $\mathcal{R}$ denotes membership in $\mathcal{R}_{\mathcal{L}}$. We assume $\mathcal{P}(\vec{l}\,)$ to be uniform in the region $\mathcal{R}_{\mathcal{L}}$ and zero outside,

$$\mathcal{P}(\vec{l}\,) = \begin{cases} \dfrac{1}{\displaystyle\int_{\mathcal{R}_{\mathcal{L}}} d\vec{l}} & \text{if } \vec{l} \in \mathcal{R}_{\mathcal{L}} \\ 0 & \text{otherwise} \end{cases} \quad . \tag{4.16}$$

The probability density function $\mathcal{P}(\boldsymbol{a}|\mathcal{R})$ gives the distribution of the sensor hyperpixels given the image formation model in the region $\mathcal{R}_{\mathcal{L}}$. Hence we refer to $\mathcal{P}(\boldsymbol{a}|\mathcal{R})$ as the model distribution over $\mathcal{R}_{\mathcal{L}}$. At each location in $\mathcal{R}_{\mathcal{L}}$ the sensor hyperpixels $\boldsymbol{a}$ can be assumed to be independently[2] and identically distributed according to the density function $\mathcal{P}(\boldsymbol{a}|\mathcal{R})$. Note that the model distribution over $\mathcal{R}_{\mathcal{L}}$ is different from the distribution of $\boldsymbol{a}$ at a location $\vec{l}$ given by Equation (3.5). The former is the average probability density of $\boldsymbol{a}$ over the region $\mathcal{R}_{\mathcal{L}}$. Since the probability distribution $\mathcal{P}(\boldsymbol{a})$ is a Gaussian, $\mathcal{P}(\boldsymbol{a}|\mathcal{R})$ is also Gaussian.

To be able to estimate the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ of the image formation model one needs to know the parameters of the model distribution over the region $\mathcal{R}$, in particular its mean and covariance. The mean of the model distribution over the region $\mathcal{R}$ is obtained

[2] The assumption of independence can be justified since $\boldsymbol{a}$ is an hyperpixel within a Laplacian pyramid scheme and corresponds to a prediction residual [14].

as follows.

$$\begin{aligned}
\boldsymbol{\mu}_m(\boldsymbol{a}|\mathcal{R}) &\equiv E\{\boldsymbol{a}|\mathcal{R}\} \\
&= \int \boldsymbol{a}\mathcal{P}(\boldsymbol{a})d\boldsymbol{a} \\
&= \int d\boldsymbol{a} \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\boldsymbol{a}\mathcal{P}(\boldsymbol{a}|\vec{l}\,)\mathcal{P}(\vec{l}\,) \qquad (4.17) \\
&= \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\mathcal{P}(\vec{l}\,) \int \boldsymbol{a}\mathcal{P}(\boldsymbol{a}|\vec{l}\,)d\boldsymbol{a} \\
&= \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\mathcal{P}(\vec{l}\,)\boldsymbol{\mu}_m(\boldsymbol{a}|\vec{l}\,) \\
&= \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\mathcal{P}(\vec{l}\,)\left\{\boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,)\right\}
\end{aligned}$$

Assuming that $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are constant over the region $\mathcal{R}$, we obtain

$$\begin{aligned}
\boldsymbol{\mu}_m(\boldsymbol{a}|\mathcal{R}) &= \boldsymbol{\beta}\langle s_0(\vec{l}\,)\rangle_{\vec{l}} + \boldsymbol{\alpha} \\
&= \boldsymbol{\beta}\mu_{s_0} + \boldsymbol{\alpha} \qquad (4.18)
\end{aligned}$$

where

$$\mu_{s_0} \equiv \langle s_0(\vec{l}\,)\rangle_{\vec{l}} = E_{\vec{l}}\{s_0(\vec{l}\,)\} \qquad (4.19)$$

is the expected value of s_0 in the spatial region defined by $\mathcal{R}$.

To derive the model covariance over the region $\mathcal{R}_\mathcal{L}$, $\mathbf{C}(\boldsymbol{a}|\mathcal{R})$, we make use of a general result concerning conditional covariance (derived in Appendix G) and obtain,

$$\begin{aligned}
\mathbf{C}(\boldsymbol{a}|\mathcal{R}) &\equiv \mathrm{cov}(\boldsymbol{a}|\mathcal{R}) \\
&= E_{\vec{l}}[\mathrm{cov}(\boldsymbol{a}|\vec{l}\,)] + \mathrm{cov}_{\vec{l}}(E[\boldsymbol{a}|\vec{l}\,]) \\
&= \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\mathcal{P}(\vec{l}\,)\mathrm{cov}(a|\vec{l}\,) + \mathrm{cov}_{\vec{l}}(\boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,)) \\
&= \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\mathcal{P}(\vec{l}\,)\left\{\boldsymbol{\beta}(\vec{l}\,)\boldsymbol{\beta}^{\mathrm{T}}(\vec{l}\,)\sigma_s^2(\vec{l}\,) + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}(\vec{l}\,)\right\} + \\
&\quad \int_{\mathcal{R}_\mathcal{L}} d\vec{l}\,\mathcal{P}(\vec{l}\,)\left[\left\{\boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,) - \langle\boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,)\rangle_{\vec{l}}\right\}\right. \\
&\quad \left.\left\{\boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,) - \langle\boldsymbol{\beta}(\vec{l}\,)s_0(\vec{l}\,) + \boldsymbol{\alpha}(\vec{l}\,)\rangle_{\vec{l}}\right\}^{\mathrm{T}}\right] \qquad (4.20)
\end{aligned}$$

Assuming $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$, $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ and σ_s^2 to be constant over the region $\mathcal{R}_{\mathcal{L}}$, we obtain

$$\begin{aligned} \mathbf{C}(\boldsymbol{a}|\mathcal{R}) &= \left\{\sigma_s^2 + \mathrm{var}_{\vec{l}}(s_0)\right\} \boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \\ &= \left\{\sigma_s^2 + \sigma_{s_0}^2\right\} \boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \\ &= \sigma_{s,s_0}^2 \boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \end{aligned} \tag{4.21}$$

where

$$\sigma_{s,s_0}^2 = \sigma_s^2 + \sigma_{s_0}^2 \tag{4.22}$$

and

$$\sigma_{s_0}^2 \equiv \mathrm{var}_{\vec{l}}(s_0) = \int d\vec{l}\,\mathcal{P}(\vec{l}\,)\left\{s_0(\vec{l}\,) - \langle s_0\rangle_{\vec{l}}\right\}^2 \tag{4.23}$$

is the spatial variance of s_0 over the region $\mathcal{R}_{\mathcal{L}}$. In the local analysis window $\mathcal{R}_{\mathcal{L}}$, $\mathcal{P}(\boldsymbol{a}|\mathcal{R})$ is Gaussian with mean $\boldsymbol{\mu}_m(\boldsymbol{a}|\mathcal{R})$ and covariance $\mathbf{C}(\boldsymbol{a}|\mathcal{R})$ as given in Equations (4.18) and (4.21) respectively.

4.5.1 Least squares factor analysis estimation of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

In this approach, we derive the estimates of the affine parameters from the first and second order statistics on the sensor image data in the local analysis window. Let there be N individual hyperpixels in the local analysis window (in this case, $N = 25$). Let $\boldsymbol{a}_n$ be the vector of sensor intensity values from the n^{th} hyperpixel ($1 \le n \le N$) in the 5×5 region ($\boldsymbol{a}_n = [a_{1n}, a_{2n}, \ldots, a_{qn}]^{\mathrm{T}}$, where $a_1, a_2, \ldots, a_q$ are the hyperpixel values of sensors $1, 2, \ldots, q$ respectively). From these observations, the sample mean vector $\boldsymbol{\mu}_a$ and the sample covariance matrix $\boldsymbol{\Sigma}_a$ are computed as,

$$\boldsymbol{\mu}_a \equiv \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{a}_n \tag{4.24}$$

$$\boldsymbol{\Sigma}_a \equiv \frac{1}{N}\sum_{n=1}^{N} (\boldsymbol{a}_n - \boldsymbol{\mu}_a)(\boldsymbol{a}_n - \boldsymbol{\mu}_a)^{\mathrm{T}} \tag{4.25}$$

Least squares factor analysis consists of obtaining estimates of the affine parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and the noise variance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ by fitting the model to the observed sensor data. We have already described separate estimation procedures for the noise variance in Section 4.3. Here we develop the least squares estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$.

To obtain the least squares estimate of $\boldsymbol{\alpha}$, consider the squared norm of the difference between the model mean $\boldsymbol{\mu}_m(\boldsymbol{a}|\mathcal{R})$ given by Equation (4.18) and the data mean $\boldsymbol{\mu}_a$ given by Equation (4.24),

$$E_\alpha = \|\boldsymbol{\mu}_a - \boldsymbol{\mu}_m(\boldsymbol{a}|\mathcal{R})\|^2 \tag{4.26}$$

Minimizing E_α with respect to $\boldsymbol{\alpha}$ gives

$$\boldsymbol{\alpha}_{\mathrm{LS}} = \boldsymbol{\mu}_a - \boldsymbol{\beta}\mu_{s_0} \quad . \tag{4.27}$$

To obtain the least squares estimate of $\boldsymbol{\beta}$, consider the squared norm of the difference between the model covariance $\mathbf{C}(\boldsymbol{a}|\mathcal{R})$ given by Equation (4.21) and the data covariance $\boldsymbol{\Sigma}_a$ [3, 38] given by Equation (4.25),

$$\begin{aligned} E_\beta &= \|\boldsymbol{\Sigma}_a - \mathbf{C}(\boldsymbol{a}|\mathcal{R})\|^2 && (4.28) \\ &= \sum_{i,j}(\boldsymbol{\Sigma}_a - \mathbf{C}(\boldsymbol{a}|\mathcal{R}))^2_{i,j} \\ &\equiv \mathrm{tr}\{(\boldsymbol{\Sigma}_a - \mathbf{C}(\boldsymbol{a}|\mathcal{R}))^2\} && (4.29) \end{aligned}$$

E_β is the sum of squared differences between the elements of $\boldsymbol{\Sigma}_a$ and $\mathbf{C}(\boldsymbol{a}|\mathcal{R})$. Differentiating E_β with respect to $\boldsymbol{\beta}$ and equating to zero gives (see Appendix D),

$$(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_\epsilon)\boldsymbol{\beta} = \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta} \tag{4.30}$$

This equation imposes two constraints on $\boldsymbol{\beta}$ —

1. $\boldsymbol{\beta}$ is an eigenvector of $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_\epsilon)$, and

2. $\sigma^2_{s,s_0}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}$ is the corresponding eigenvalue.

The solution to $\boldsymbol{\beta}$ that satisfies both these constraints is,

$$\boldsymbol{\beta}_{\mathrm{LS}} = \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r \tag{4.31}$$

where $\mathbf{U}$ is an eigenvector, and λ is an eigenvalue of the noise-corrected covariance matrix $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_\epsilon)$. The variable $r = \pm 1$, and determines the polarity of contrast. The choice of

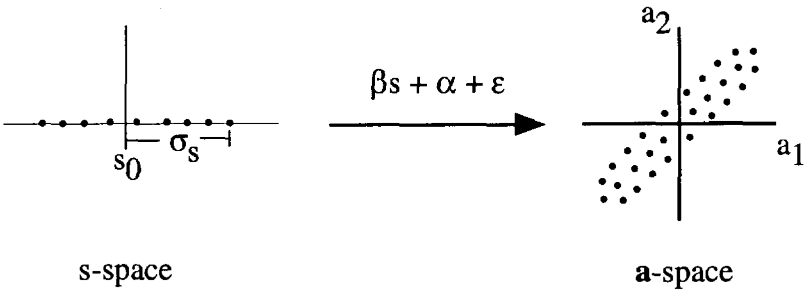


Figure 4.5: Mapping from s-space to $\boldsymbol{a}$-space

the sign of r is discussed later in this section. It is shown in Appendix D that the error metric in Equation (4.28) is minimum when $\mathbf{U}$ is the principal eigenvector and λ is the principal eigenvalue of the noise-corrected covariance matrix $(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon)$.

Link between the estimate of $\boldsymbol{\beta}$ and the relationship between the sensor images

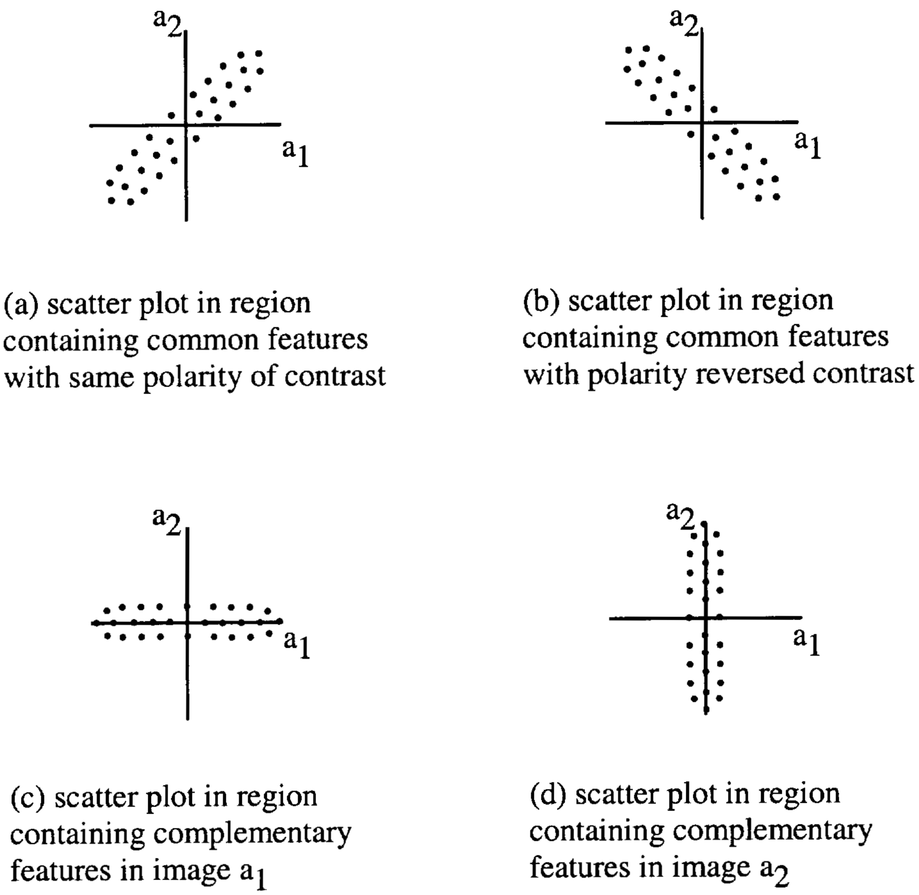
The image formation model of Equation (4.9) maps hyperpixels in s-space (true scene) into hyperpixels in $\boldsymbol{a}$-space (sensor images). Figure 4.5 illustrates this mapping in the case where there are two sensor images a_1 and a_2. The parameter $\boldsymbol{\beta}$ determines the underlying orientation of the cloud of hyperpixels in $\boldsymbol{a}$-space. $\boldsymbol{\alpha}$ determines the shift of the cloud from the origin in $\boldsymbol{a}$-space. Both s and $\boldsymbol{\epsilon}$ contribute to the spread of the cloud. The contribution from s is along the direction of $\boldsymbol{\beta}$, whereas the contribution from $\boldsymbol{\epsilon}$ is along the a_1 and a_2 axes. Although $\boldsymbol{\beta}$ determines the underlying orientation of hyperpixels in $\boldsymbol{a}$-space, the noise variance in each sensor pushes the sensor hyperpixel values along the direction of the a_1 and a_2 axes. Therefore, in the general case, where the noise variance in each sensor is different, the orientation of the cloud in $\boldsymbol{a}$-space is different than that of $\boldsymbol{\beta}$.

The relationship between the image features in the sensor images a_1 and a_2 can be illustrated from the scatter plot of hyperpixel values in $\boldsymbol{a}$-space. Figure 4.6(a) is an example of a scatter plot when there are common image features (see Section 3.2) in a_1 and a_2, and these features have the same polarity. The values in a_2 are high when a_1 values are high and low when a_1 values are low. Figure 4.6(b) shows a scatter plot of image features that are common to a_1 and a_2, but having reversed polarity. The values in a_2 are low when values in a_1 are high and vice versa. Figure 4.6(c) shows the scatter plot when a feature is present in a_1 but not in a_2. There is a large variation in a_1 values due to the presence of the complementary features, whereas there is little variation in a_2. Figure 4.6(d) shows the converse case when there are complementary features in a_2.

Noise in a_1 and a_2 can cause the orientation of the data cloud in $\boldsymbol{a}$-space to change. The general case where the noise covariance is heteroscedastic (different noise in each sensor) is illustrated in Figure 4.7(a). The least squares factor analysis estimate of $\boldsymbol{\beta}$ points along $\mathbf{U}$, which is the principal eigenvector of the corrected data covariance matrix $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_\epsilon)$. The contribution of the noise terms is suppressed and the direction (of $\mathbf{U}$) that contains the contribution from s is captured. Thus, the principal eigenvector $\mathbf{U}$ captures the relationship between the image features as discussed above. In general, this direction is different from the direction of maximum variance in $\boldsymbol{a}$-space. However when the noise covariance is homoscedastic (equal noise in all sensors), the scatter plots look like the one in Figure 4.7(b). Here, the direction of $\mathbf{U}$ is also the direction of maximum variance in $\boldsymbol{a}$-space.

Choice of the sign r of the eigenvector U

The sign parameter r in Equation (4.31) represents the sign (i.e. the direction) of the eigenvector $\mathbf{U}$ and therefore the sign of $\boldsymbol{\beta}$. Therefore it determines the sign of $\hat{s}$ in the fusion rule of Equation (3.14), and hence also determines the polarity of contrast in the fused image. The least squares estimation specifies the orientation of the $\mathbf{U}$ but not its sign (direction). This is because either choice of the sign results in the same fit of the model covariance matrix $\mathbf{C}(\boldsymbol{a}|\mathcal{R})$ to the data covariance matrix $\boldsymbol{\Sigma}_a$. In the following discussion we assume that $\mathbf{U}$ computed from $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_\epsilon)$ has an arbitrary sign, and we need to choose an appropriate sign r for $\mathbf{U}$. The choice of r would not pose any problem if the image formation model was global. However, our model consists of several local models — one for each hyperpixel. An arbitrary choice of r for each local model would result in arbitrary reversals in the polarity of local contrast in the fused image. In order to be able to properly piece together our local models, we cannot allow arbitrary sign



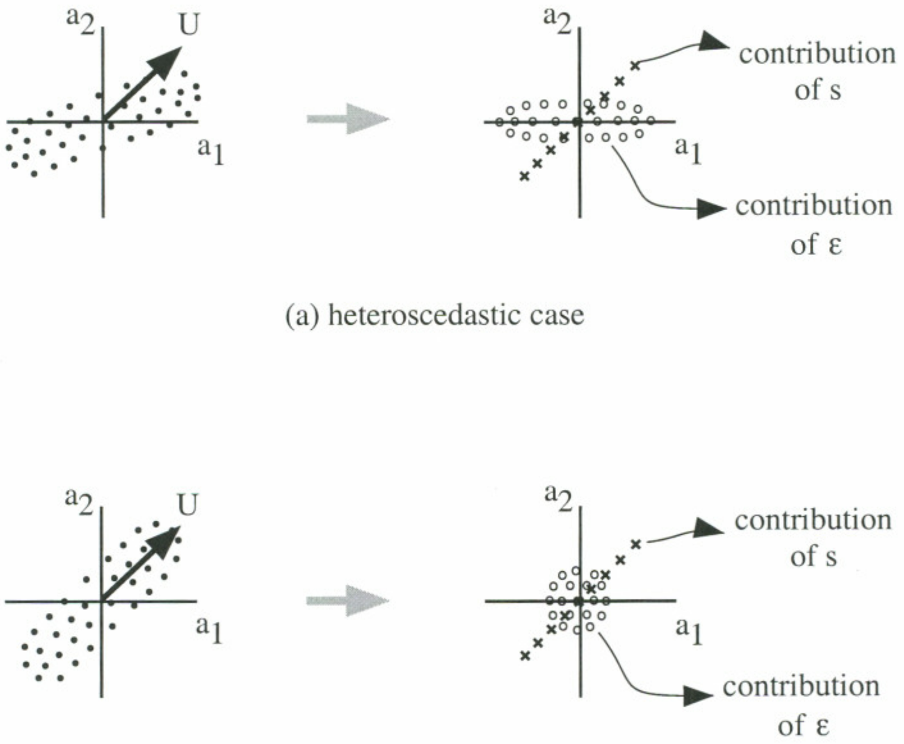
(a) scatter plot in region containing common features with same polarity of contrast

(b) scatter plot in region containing common features with polarity reversed contrast

(c) scatter plot in region containing complementary features in image a_1

(d) scatter plot in region containing complementary features in image a_2

Figure 4.6: Scatter plots in $\boldsymbol{a}$-space for different relationships between image features in a_1 and a_2



(a) heteroscedastic case

(b) homoscedastic case

Figure 4.7: Direction of **U** obtained by least squares estimation

reversals between neighboring local regions. To ensure this, we must correctly choose the sign parameter r at each hyperpixel.

To be consistent with our assumption that the parameters of the image formation model change slowly spatially, the signs for neighboring hyperpixels should be chosen such that the orientation of $\boldsymbol{\beta}$ changes slowly from one hyperpixel to another. To ensure a slow spatial change in the orientation of $\boldsymbol{\beta}$, we must choose the signs to minimize a metric of the form

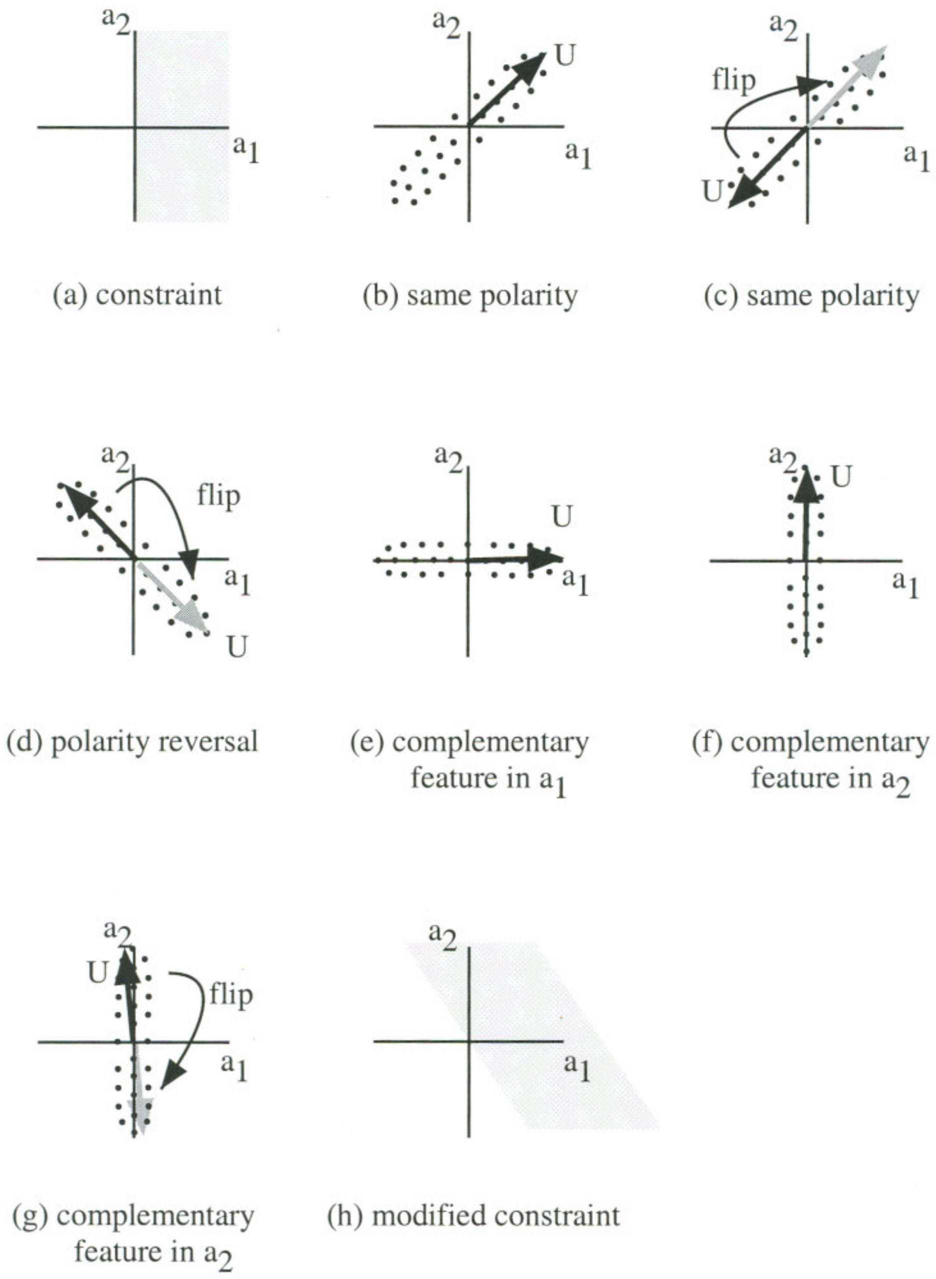
$$\mathcal{E}_r = \sum_{i,j} \|\boldsymbol{\beta}_i - \boldsymbol{\beta}_j\|^2 , \tag{4.32}$$

where the summation is over all pairs of adjacent hyperpixels i, j within a pyramid level. In addition, it is also necessary to ensure that the signs do not change arbitrarily at similar spatial locations from one pyramid level to another. This is a difficult combinatorial optimization problem and would require an iterative solution across the entire fused pyramid.

Instead of a complex optimization solution, we have developed a simple method to determine the signs. Empirically, we have found that this method gives good results. We choose a convenient reference region in $\boldsymbol{a}$-space and then choose the signs r such that $\boldsymbol{\beta}$ is forced to lie within the reference region for all hyperpixels. To understand how this method works, we refer to Figure 4.8. The shaded area in Figure 4.8(a) is one choice for the reference region for two sensors a_1 and a_2. Figure 4.8(b) is a scatter plot of hyperpixel values, a_1 and a_2, in two sensors. The plot shows the spread of hyperpixel values in a small, local spatial region that contains image features with the same polarity. The eigenvector $\mathbf{U}$ points in the direction of maximum variance, but can be oriented in either direction shown by the dark vectors in Figures 4.8(b) and 4.8(c). An arbitrary choice of the signs for eigenvectors in neighboring local regions can result in neighboring eigenvectors pointing in opposite directions. However, if we choose the signs to restrict the eigenvector to lie within the shaded region, the eigenvector in Figure 4.8(c) will have to be *flipped.* Note that if we assume $\mathbf{U}$ to have some orientation with an arbitrary sign, then a choice of $r = -1$ will cause $\boldsymbol{\beta}$ to point in the opposite direction to that of $\mathbf{U}$. The flipped eigenvector is shown by the gray vector. The constraint enforced by the shaded region effectively prevents arbitrary changes in the direction of $\boldsymbol{\beta}$ in neighboring regions.

The constraint imposed by the reference region also provides a way to bias the fused image such that polarity reversed image features appear with the polarity of a particular sensor. Figure 4.8(d) is a scatter plot of hyperpixels in a local region containing polarity reversals. The constraint will force $\mathbf{U}$ to lie in the shaded region. The components β_1 and β_2 of $\boldsymbol{\beta}$, for polarity reversed regions, will be such that β_1 is positive and β_2 is negative. From Equation 3.15, one can see that hyperpixel values a_2 of the second sensor will be negated and added to hyperpixel values a_1 of the first sensor to obtain the hyperpixel of the fused pyramid. Thus, the polarity of features of the first sensor will be preserved in the fused image. Figure 4.8(e) shows the scatter plot for a region in which there are complementary feature in sensor 1. The constraint ensures that the signs of a_1 are not reversed in the fused image.

However, there is an inherent indeterminacy in the constraint defined by the shaded region in Figure 4.8(a). At the borders of the shaded region (i.e. the a_2 axis), small variations in the orientation of $\mathbf{U}$ can cause an arbitrary choice of the sign r. Figure 4.8(f) shows the scatter plot for a region in which there are complementary features in sensor 2. If $\mathbf{U}$ is exactly along the vertical then there is no ambiguity, since it is still within the shaded region. However if $\mathbf{U}$ is slightly away from the vertical, the constraint will choose r that flips $\mathbf{U}$ as shown in Figure 4.8(g). Consequently, complementary features in a_2 can appear with arbitrary polarity in the fused image. We have a heuristic that addresses this problem. We choose the signs based on the shaded region shown in Figure 4.8(h). Complementary features in either sensor appear with their respective polarities in the fused image. Our heuristic does not overcome the indeterminacy though. One approach to resolve the indeterminacy is to look at the eigenvector directions in neighboring regions and then flip any eigenvectors that seem inconsistent. This solution, would require a relaxation technique such as the one mentioned above.



(a) constraint

(b) same polarity

(c) same polarity

(d) polarity reversal

(e) complementary feature in a_1

(f) complementary feature in a_2

(g) complementary feature in a_2

(h) modified constraint

Figure 4.8: Choice of the sign r of the eigenvector **U**

Choice of σ^2_{s,s_0} and μ_{s_0}

The least squares estimates of the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, given by Equations (4.31) and (4.27) respectively, depend upon the parameters σ^2_{s,s_0} and μ_{s_0}. The parameters σ^2_{s,s_0} and μ_{s_0} depend upon the priors σ^2_s and s_0 as shown in Equations (4.22) and (4.19). In the absence of prior knowledge about the scene, σ^2_{s,s_0} and μ_{s_0} must be appropriately chosen in order to estimate the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$.

The least squares approach does not provide an estimate of either σ^2_{s,s_0} or μ_{s_0}. As with the sign of r, this would pose no problem for a global model. But we must impose a constraint in order to smoothly piece together our local models. We impose that $\sigma^2_{s,s_0} = \lambda$, everywhere , motivated by the following reasoning: consider a small patch over which s varies but both σ^2_s and the image formation parameters $(\boldsymbol{\beta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ are constant. Any variation in sensor hyperpixel intensities in this region must arise from variations in the true scene s, and the noise. The leading eigenvalue λ of the noise-corrected covariance matrix $(\boldsymbol{\Sigma}_{a} - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ gives the scale of variations in $\boldsymbol{a}$ arising from variations in s. Thus, we should have $\lambda \propto \sigma^2_{s,s_0}$. To ensure consistency, the proportionality constant should be the same in all local regions. From the least squares solution of Equation (4.31), this proportionality constant is just $\| \boldsymbol{\beta} \|^2$. Hence we take $\| \boldsymbol{\beta} \|= 1$ everywhere, or $\sigma^2_{s,s_0} = \lambda$.

In the absence of prior knowledge about the scene, we assume that s_0 is zero everywhere, i.e., the prior distribution on s at each hyperpixel location has zero mean. Although this assumption is not valid in practice (this assumption attributes all variations in the scene s to the prior variance σ^2_s, and hence basically to noise), it allows us to obtain an approximation for the estimate of $\boldsymbol{\alpha}$ since we now have $\mu_{s_0} = 0$ from Equation (4.19). Note that the assumption does not change our choice for σ^2_{s,s_0} given above.

In view of the above assumptions made for choosing σ^2_{s,s_0} and μ_{s_0}, and assuming that the sign of r has been appropriately chosen, the least squares estimates of $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are,

$$\begin{aligned} \boldsymbol{\beta}_{\mathrm{LS}} &= \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}} \mathbf{U} r \\ &= \mathbf{U} \,, \end{aligned} \tag{4.33}$$

and

$$\begin{aligned}\boldsymbol{\alpha}_{\mathrm{LS}} &= \boldsymbol{\mu}_a - \boldsymbol{\beta}\mu_{s_0} \\ &= \boldsymbol{\mu}_a \ . \end{aligned} \tag{4.34}$$

4.6 Relation of Fusion to Principal Components Analysis

With the image formation model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ estimated using the factor analysis techniques described in Section 4.5.1, the MAP and ML fusion rules described in Chapter 3 are closely related to PCA. The factor analysis techniques estimate the affine parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ from an analysis of the covariance structure of hyperpixel intensities in the sensor image data. In principal components analysis (PCA), the principal components are obtained by analyzing the covariance structure of the data. However, factor analysis differs from PCA because factor analysis incorporates an a priori structure on the covariance matrix and an a priori structure on the noise terms. In factor analysis, the contribution of the noise covariance to the data covariance is removed before computing the principal components. This is achieved in least squares factor analysis by computing principal components of $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$.

To show the relationship between the estimate $\hat{s}$ of the true scene and PCA, we substitute the least squares estimates of the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ in Equations (4.33) and (4.34) into the fusion rule in Equation (3.14) to obtain,

$$\hat{s} = \left[\mathbf{U}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\mathbf{U} + \frac{1}{\sigma_s^2}\right]^{-1} \left\{\mathbf{U}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a} - \boldsymbol{\mu}_a) + \frac{s_0}{\sigma_s^2}\right\} \tag{4.35}$$

where $\mathbf{U}$ is the principal eigenvector of the corrected data covariance matrix, $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$. The estimate $\hat{s}$ is a scaled and shifted projection of the noise variance weighted sensor data onto the eigenvector $\mathbf{U}$.

The relationship between the MAP estimate $\hat{s}$ and PCA is clear when the noise is homoscedastic, $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} = \sigma_\epsilon^2\mathbf{I}$. Under this condition Equation (4.35) simplifies to

$$\hat{s} = \frac{1}{1 + \frac{\sigma_\epsilon^2}{\sigma_s^2}}\mathbf{U}_a^{\mathrm{T}}(\boldsymbol{a} - \boldsymbol{\mu}_a) + \frac{1}{1 + \frac{\sigma_s^2}{\sigma_\epsilon^2}}s_0 \tag{4.36}$$

where $\mathbf{U}_a$ is the principal eigenvector of the data covariance matrix $\mathbf{\Sigma}_a$[3]. The MAP estimate $\hat{s}$ is simply a scaled and shifted local PCA projection of the sensor data. Both the scaling and shift arise because the prior distribution on s tends to bias $\hat{s}$ towards s_0.

When the prior distribution on s is flat (uniform prior or $\sigma_s^2 = \infty$), the fused image is given by a simple local PCA projection,

$$\hat{s} = \mathbf{U}_a^{\mathrm{T}} \left(\boldsymbol{a} - \boldsymbol{\mu}_a\right) \quad . \tag{4.37}$$

Equivalently, if the noise variance vanishes ($\sigma_\epsilon^2 = 0$), the fusion rule reduces to the local PCA projection given by Equation (4.37).

Alternatively, using the ML fusion rule of Equation (3.12) and substituting the parameter estimates from Equations (4.33) and (4.34)

$$\hat{s} = \left[\mathbf{U}^{\mathrm{T}} \mathbf{\Sigma}_\epsilon^{-1} \mathbf{U}\right]^{-1} \left\{\mathbf{U}^{\mathrm{T}} \mathbf{\Sigma}_\epsilon^{-1} (\boldsymbol{a} - \boldsymbol{\mu}_a)\right\} \tag{4.38}$$

Now, in the case where the noise variance is homoscedastic, the fusion rule again reduces to the local PCA projection of Equation (4.37).

4.7 Summary and Discussion

In this chapter we described techniques to estimate the parameters $\mathbf{\Sigma}_\epsilon$, $\boldsymbol{\beta}$, and $\boldsymbol{\alpha}$ of the image formation model from the data available from the sensor images. We described two techniques for estimating the noise variance, one that uses a single image from each sensor, and another based on using multiple image frames from sensor video sequences. The single frame technique is based on a set of assumptions and does not provide local estimates of the noise variance, rather one estimate for each pyramid level. The multiple frame technique is adaptive and provides an estimate of the noise variance at each hyperpixel location. However, the multiple frame technique is based on the assumption of motion compensated frames being available. This makes it computationally much more expensive than the single frame technique in a practical application where motion compensation is likely to be expensive.

[3] Note that $\mathbf{U}_a$ is also the principal eigenvector of $(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon)$ when $\mathbf{\Sigma}_\epsilon = \sigma_\epsilon^2 \mathbf{I}$.

We described a technique for estimating the affine parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ using simple regression by making use of a reference image. However, we noted the drawbacks of this technique in the presence of noise in the reference image and when complementary features present in the sensor images are absent in the reference image. We then described a probabilistic technique to estimate the affine parameters. The parameters were derived using least squares factor analysis. The relation between the parameter estimates and the relationship between the sensor images was explained in detail. We also showed that with the parameters estimated using least squares factor analysis, the probabilistic fusion rules of Chapter 3 are closely related to local PCA. In the next chapter we will show examples of fused images obtained by using the probabilistic fusion rules and the estimated parameters.

Chapter 5

Experiments and Results

5.1 Introduction

In Chapter 3 we presented our probabilistic model for fusion and derived the fusion rules using a Bayesian framework. In Chapter 4 we derived estimates of the parameters of our probabilistic model — the affine parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, and the noise variance $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$. In this chapter we present several fusion experiments using the theory developed in Chapters 3 and 4. The experiments, performed on both real and simulated images, illustrate the different ways in which the model-based probabilistic fusion can be used to fuse images obtained from multiple sensors. In most of the experiments we compare the results obtained using the probabilistic fusion approach, with results of the averaging and selection based fusion rules discussed in Chapter 2. We also demonstrate the capability of our approach to combine prior information from a database with information from the sensor images. Finally, we describe an approach to quantitatively evaluate the result of fusion.

In Section 5.2 we describe experiments to fuse images from visible-band and IR sensors using the probabilistic fusion rules and associated parameter estimation described in Chapters 3 and 4. For comparison, we also show results using the traditional averaging and selection approaches. The effects of using different number of pyramid levels, different constraints for the signs and different sizes of local analysis windows for parameter estimation are illustrated using examples. Section 5.3 describes experiments for fusing images containing additive noise. Both the ML and MAP approaches are employed for combining the images, and the advantage of the MAP approach is discussed. Section 5.4

describes a set of experiments that utilize simulated computer graphics images to demonstrate approaches of combining image information from a database image with imagery from the sensors. Two different approaches using the MAP fusion rule to include database information are described. Section 5.5 describes a simple approach that we have developed for quantitative evaluation of fusion results. We discuss the advantages and drawbacks of this approach. Section 5.6 concludes with a summary of the experiments and results described in this chapter.

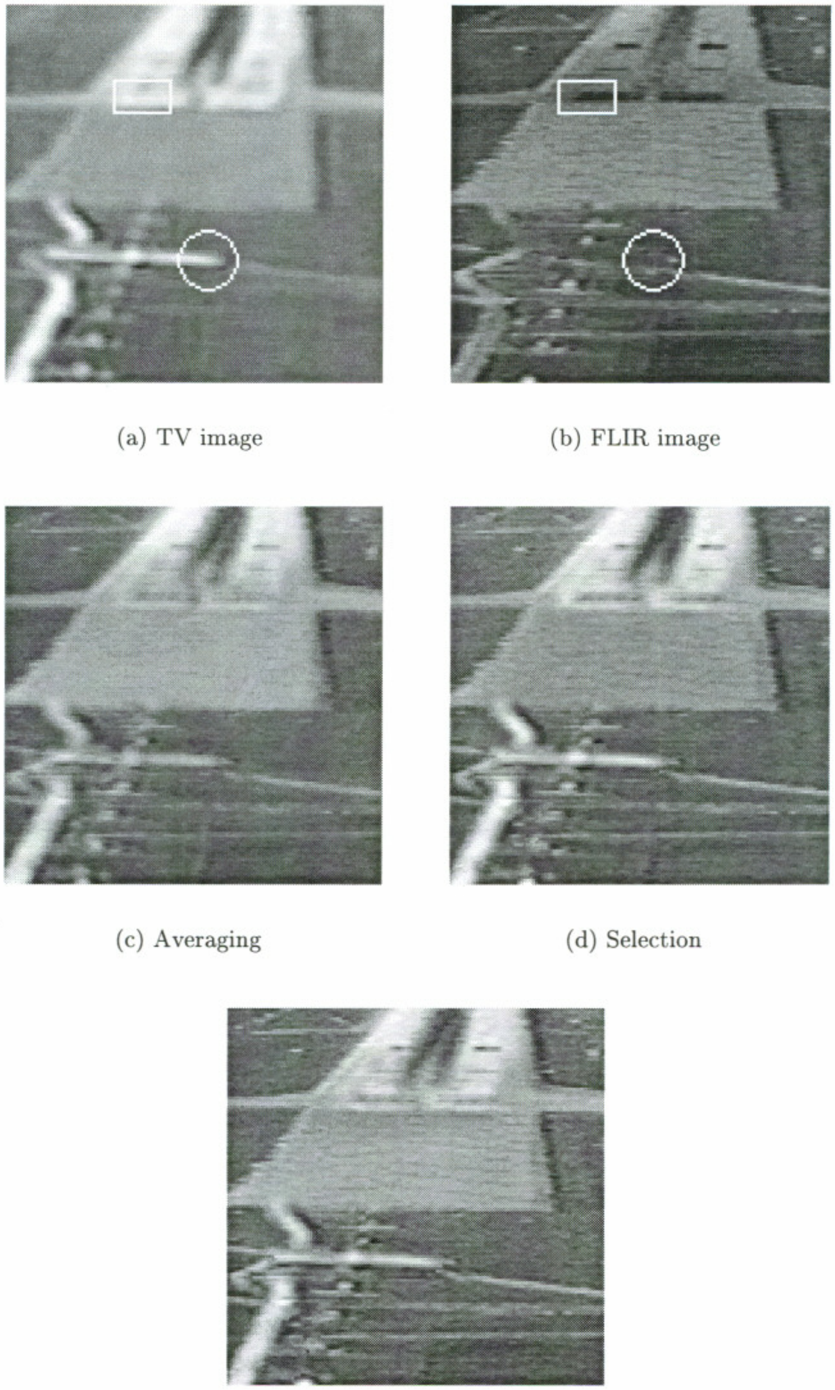
5.2 Fusion of Visible-band and Infrared Images

As discussed in Chapter 1, visible-band and IR sensors are commonly used in image fusion applications. The experiments described in this chapter illustrate fusion of visible-band and IR sensor images. Figure 5.1(a) is a visible-band image showing a runway scene as an aircraft is approaching to land. Figure 5.1(b) is an infrared image of the same scene. These images contain local polarity reversed and complementary features. The polarity of contrast of the markings on the runway is reversed between the two images (see the region highlighted by the rectangle). The visible-band image contains a bright horizontal patch just before the runway. This patch is missing in the IR image (see the region highlighted by the circle). The horizontal lines that are visible in the lower portion of the IR image are missing in the visible-band image.

5.2.1 Assuming equal noise variance in the sensor images

We fused the visible-band and IR images of Figures 5.1(a) and 5.1(b) using the ML fusion rule of Equations (3.12) and (3.13) given by,

$$
\begin{aligned}
\hat{s}_{\mathrm{ML}} &= \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta}\right]^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a}-\boldsymbol{\alpha})\right\} \\
&= \frac{\dfrac{\beta_1(a_1-\alpha_1)}{\sigma_{\epsilon_1}^2}+\dfrac{\beta_2(a_2-\alpha_2)}{\sigma_{\epsilon_2}^2}}{\dfrac{\beta_1^2}{\sigma_{\epsilon_1}^2}+\dfrac{\beta_2^2}{\sigma_{\epsilon_2}^2}}
\end{aligned}
\qquad (5.1)
$$



(a) TV image

(b) FLIR image

(c) Averaging

(d) Selection

(e) ML fusion

Figure 5.1: Fusion of TV and FLIR images. The rectangle highlights a region containing local polarity reversal. A region with complementary features is marked by the circle (Original data from the SVTD [9] project).

In this experiment, we assumed that the variance of noise in the sensor images is equal ($\mathbf{\Sigma}_{\boldsymbol{\epsilon}} = \sigma_\epsilon^2\mathbf{I}$) and close to zero ($\sigma_\epsilon^2 \approx 0$). Using the assumption of equal noise variance,

$$\begin{aligned} \hat{s}_{\text{ML}} &= \left[\boldsymbol{\beta}^{\text{T}}\boldsymbol{\beta}\right]^{-1}\boldsymbol{\beta}^{\text{T}}(\boldsymbol{a} - \boldsymbol{\alpha}) \\ &= \frac{\beta_1(a_1 - \alpha_1) + \beta_2(a_2 - \alpha_2)}{\beta_1^2 + \beta_2^2} \end{aligned} \tag{5.2}$$

The parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are computed using the least squares estimates of Equations (4.27) and (4.31), given by

$$\begin{aligned} \boldsymbol{\alpha}_{\text{LS}} &= \boldsymbol{\mu}_a - \boldsymbol{\beta}\mu_{s_0} \text{ and} \\ \boldsymbol{\beta}_{\text{LS}} &= \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r \end{aligned} \tag{5.3}$$

where $\boldsymbol{\mu}_a$ is the data mean, λ is the principal eigenvalue, $\mathbf{U}$ is the principal eigenvector of the noise corrected data covariance ($\mathbf{\Sigma}_a - \sigma_\epsilon^2\mathbf{I}$), and $\mathbf{\Sigma}_a$ is the data covariance matrix. The principal eigenvector $\mathbf{U}$ is identical to the principal eigenvector $\mathbf{U}_a$ of the data covariance matrix $\mathbf{\Sigma}_a$, since the noise variance is equal. Following the discussion in Section 4.5.1, values for μ_{s_0}, σ_{s,s_0}^2 and r need to be appropriately chosen. We assume $\mu_{s_0} = 0$, $\parallel \boldsymbol{\beta} \parallel = 1$, and choose an appropriate sign r to obtain,

$$\begin{aligned} \boldsymbol{\alpha} &= \boldsymbol{\mu}_a \text{ and} \\ \boldsymbol{\beta} &= \mathbf{U}_a \ . \end{aligned} \tag{5.4}$$

Using these parameter estimates, the ML fusion rule in Equation (5.1) simplifies to

$$\hat{s}_{\text{ML}} = \mathbf{U}_a^{\text{T}}(\boldsymbol{a} - \boldsymbol{\mu}_a) \tag{5.5}$$

which is a local PCA projection of the data on to the principal eigenvector $\mathbf{U}_a$ of the data covariance, as given by Equation (4.37). Note that when the noise variance in the sensor images is equal and zero, the MAP fusion rule of Equations (3.14) and (3.15) also simplifies to the local PCA projection above as explained in Section 4.6. However, when the noise variance is equal and high (the images are equally noisy), the MAP fusion rule can be applied in a different manner to obtain better results (see the results of MAP$_1$ fusion described in Section 5.4).

The experimental setup for computing the fused image, summarized in Table 5.1, consisted of decomposing each sensor image of size 128 × 128 pixels into 7 levels of the Laplacian pyramid using the operations described in Section 2.4.1. At each pyramid level a 5 × 5 hyperpixel local analysis window surrounding each hyperpixel was used to compute the data mean $\boldsymbol{\mu}_a$ and the data covariance matrix $\boldsymbol{\Sigma}_a$ as defined by Equations (4.24) and (4.25) respectively. At the border hyperpixels in the pyramid, the boundary was extended by reflecting the hyperpixels at the border, to obtain the 5 × 5 region. The principal eigenvector $\mathbf{U}_a$ was obtained through a eigen-decomposition of the data covariance matrix $\boldsymbol{\Sigma}_a$. The hyperpixels of the fused pyramid were obtained by computing $\hat{s}$ at each location using the computed eigenvector and the hyperpixels a_1 and a_2 of the sensor images. The direction for the eigenvector $\mathbf{U}_a$ (i.e. the sign r) was chosen such that $\mathbf{U}_a$ lay in the shaded region shown in Figure 4.8(a). The fused image was obtained from the fused pyramid formed by the $\hat{s}$ values using the inverse Laplacian pyramid transform described in Section 2.4.1.

One should note that the number of hyperpixels present at higher levels of the pyramid are insufficient to estimate the image formation model parameters. For example, for an image of size 128 × 128 pixels, the topmost level contains just 1 hyperpixel (the intensity value of this hyperpixel corresponds to the mean intensity of the graylevels in the image). It is not possible to estimate the model parameters from just 1 hyperpixel from each sensor. Hence the probabilistic fusion rules cannot be used to combine the topmost pyramid levels of the sensor images. In practice the sensor hyperpixels at the topmost level can be combined by averaging. Note that the topmost level of the fused pyramid would dictate the mean intensity in the fused image. The choice of the method used for combining the topmost hyperpixels is of no consequence if the graylevels in the fused image are scaled prior to display (since scaling would change the mean intensity level). Therefore, in our experiment we used 7 levels of the pyramid, meaning that the topmost pyramid level used was 2 × 2 hyperpixels. In practice, it may be adequate to stop at a pyramid level where there are a minimum number of hyperpixels to form a local analysis window.

Fusion by averaging, described in Section 2.3.1, consists of computing the fused image by averaging corresponding pixels of the sensor images. In our experiments, we have

Table 5.1: Experimental setup to obtain the ML-fused image of Figure 5.1

Size of images	128 × 128 pixels
Laplacian pyramid levels	7
Size of local analysis window	5 × 5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(h)
Noise variance	assumed equal in both sensors
Processing at borders	reflected hyperpixels to extend borders

performed fusion by averaging in the pyramid domain, to be consistent with the probabilistic fusion methods and fusion by selection. Each sensor image was decomposed into 7 levels of the Laplacian pyramid. Hyperpixels in corresponding locations in the sensor pyramids were averaged to compute the fused pyramid. The fused image was obtained by applying the inverse pyramid transform to the fused pyramid. Note that these operations are equivalent to averaging the images (since the Laplacian pyramid operations are linear operations).

Fusion by selection in the Laplacian pyramid domain is described in Section 2.4.1. The sensor images were decomposed into 7 levels of the Laplacian pyramid. The salience measure described in Section 2.4.1, used for deciding which of the sensor hyperpixels to select into the fused pyramid, is sensitive to noise since it is based on the energy at a single hyperpixel. Instead, we computed an area-based salience measure for each hyperpixel as in Section 2.7. The sum of squared hyperpixel values in a 5 × 5 hyperpixel region surrounding the hyperpixel of interest constituted the salience measure. The fused pyramid was constructed by selecting the most salient of the two sensor hyperpixels (i.e. selecting the hyperpixel that had the largest salience measure) at each hyperpixel location. The fused image was obtained by applying the inverse Laplacian pyramid transform to the fused pyramid.

The fused images obtained by averaging, selection and ML fusion are shown in Figure 5.1. For the purposes of displaying the fused images it is necessary to rescale the graylevels so that the images can be readily compared. Pavel and Ahumada [50] discuss display issues concerned with image quality in detail. We have used a simple scheme for displaying the fused images. The graylevels in all the fused images of Figure 5.1 were linearly scaled such that the average graylevel of each fused image and the standard deviation of graylevels in each fused image was identical. This ensures that the contrast and brightness computed over the entire image is same for each displayed fused image[1] (see the discussion in Section 5.5). The graylevels of the scaled images were then clipped between 0 and 255 before display.

The fused image obtained by averaging, Figure 5.1(c), has reduced contrast in the regions containing polarity reversed features and complementary features. The bright patch before the runway in the visible-band image and the horizontal lines in the lower portion of the IR image are rendered at reduced contrast in the fused image obtained by averaging, as compared to the contrast in the sensor images. Selection, Figure 5.1(d), does better than averaging. Contrast is preserved in regions containing complementary features. Selection causes pattern cancellation if the algorithm arbitrarily selects from one sensor image or the other, in regions containing polarity reversals (for example, see the selection fused image of Figure 2.9). Arbitrary selection is likely to occur if image features have equal but opposite contrast. Noise in the sensor images can then cause arbitrary switching between the sensor images. In the sensor images of Figures 5.1(a) and 5.1(b), the runway markings have opposite but unequal contrast. This prevents pattern cancellation.

The fused image obtained by ML fusion is shown in Figure 5.1(e). ML fusion retains complementary features from both the sensor images. The bright horizontal patch in the visible-band image, the image features at the top left in the IR image and the horizontal lines in the lower portion of the IR image are all visible at the original contrast in the fused image. Local polarity reversed features are combined by a weighted sum after reversing the polarity of one of the images. Overall, the ML-fused image is similar to the selection-fused image. Recall from Section 4.5.1 that the choice of the constraint used for choosing the sign r in Equation (5.3) determines the contrast with which local polarity

[1] If two images that were exact copies of each other were displayed on different monitors, then the contrast and brightness knobs of the monitor could be adjusted to match the two images. In case of two images that differ, the difference seen after matching the contrast and brightness would be the actual difference between them.

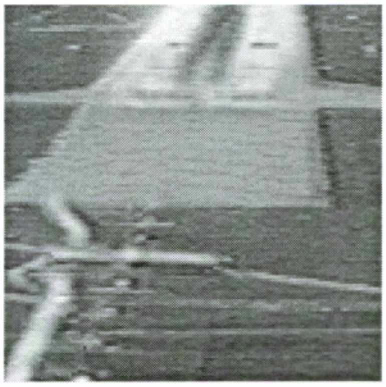


Figure 5.2: ML-fusion using 4 levels of the Laplacian pyramid

reversed and complementary features are rendered in ML fusion. This aspect is examined in Section 5.2.4.

5.2.2 Effect of number of Laplacian pyramid levels used

In Section 5.2.1, the 128 × 128 pixel visible-band and IR images were fused using 7 levels of the Laplacian pyramid. We now examine the effect of the number of Laplacian pyramid levels used for performing ML-fusion. In this experiment we decomposed the sensor images of Figures 5.1(a) and 5.1(b) into 4 levels of the Laplacian pyramid. As in Section 5.2.1, a 5 × 5 hyperpixel analysis window was used to compute the parameters $\boldsymbol{\mu}_a$ and $\mathbf{U}_a$ at each level of the Laplacian pyramid except the topmost level. The hyperpixels of the fused pyramid were obtained by computing $\hat{s}$ at each location using Equation (5.5). For a 4 level Laplacian pyramid, the topmost level (lowest resolution) consists of 16 × 16 hyperpixels. At this level, all the hyperpixels were used to estimate $\mathbf{U}_a$ and $\boldsymbol{\mu}_a$. However, a different $\hat{s}$ was computed for each hyperpixel of the topmost level. The experimental setup is summarized in Table 5.2. Figure 5.2 shows the result of fusion using 4 levels of the Laplacian pyramid. The parameters of the image formation model and the ML-fused image are computed as described in Section 5.2.1. Comparing the image in Figure 5.2 with Figure 5.1(e), there are a few noticeable differences. The dark marks appearing along the center line of the runway have higher contrast in the fused image obtained using 7 levels. These marks closely resemble the marks in the visible-band sensor image of Figure 5.1(a). In addition, the fused image of Figure 5.1(e) has a slightly higher overall contrast than

Table 5.2: Experimental setup to compute the ML-fused image in Figure 5.2

Size of images	128 × 128 pixels
Laplacian pyramid levels	4
Size of local analysis window	5 × 5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(h)
Noise variance	assumed equal in both sensors
Processing at borders	reflected hyperpixels to extend borders

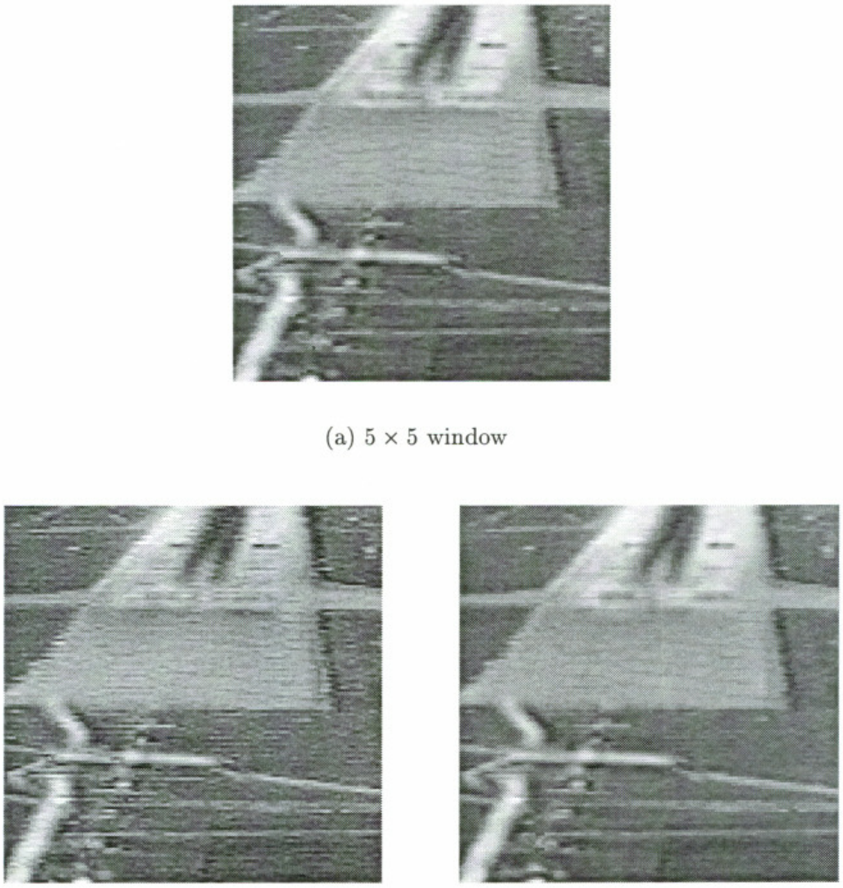
the ML-fused image of Figure 5.2.

This experiment indicates that using more Laplacian pyramid levels is beneficial for fusion. However the differences between the fused images are not significantly large. One can, therefore, make a choice on the number of levels to be used for performing fusion by making a tradeoff between the speed requirements of the application and the improvement in the fused image.

5.2.3 Effect of the size of the local analysis window used for parameter estimation

In order to estimate the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ of the image formation model, we assumed in Chapter 4 that these parameters are constant over a spatial region containing several hyperpixels. This spatial region constitutes a local analysis window from which the data mean $\boldsymbol{\mu}_a$ and the data covariance matrix Σ_a are computed. In the experiments described in the previous sections we used a local analysis window of 5 × 5 hyperpixels surrounding each hyperpixel for which the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ had to be estimated. In Section 5.2.1 we discussed why a 5 × 5 window is a practical choice. In this experiment we examine the effect of the size of the local analysis window on the fused image.

To examine the effect of changing the size of the local analysis window, we performed two experiments on the sensor images of Figures 5.1(a) and 5.1(b) using local analysis windows of sizes 3×3 and 7×7 hyperpixels. The experimental setup was the same as that in Table 5.1, except that the size of the analysis window was different for each experiment.



(a) 5 × 5 window

(b) 3 × 3 window

(c) 7 × 7 window

Figure 5.3: ML-fusion: effect of the size of the local analysis window used for parameter estimation

The images were fused using Equation (5.5) as in Section 5.2.1. Since changing the size of the analysis window affects the computation of the data mean $\boldsymbol{\mu}_a$ and the data covariance $\boldsymbol{\Sigma}_a$, it also affects the estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. Figure 5.3 shows the fused images using the different window sizes. All the images are scaled to have the same average value and variance for the purpose of display.

The fused image of Figure 5.3(a) is the same as that in Figure 5.1(e). The fused image in Figure 5.3(b), obtained using the 3×3 analysis window, appears more noisy than the fused image of Figure 5.3(a). It also seems to contain more noise compared to the sensor images of Figures 5.1(a) and 5.1(b). The 3×3 hyperpixel local analysis window contains only 9 data points. This data is insufficient to estimate the data mean and the data covariance. As a result, the estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are noisy and the fused image is noisy.

The fused image of Figure 5.3(c), obtained using a 7×7 analysis window, looks like a smoothed version of Figure 5.3(a) in some regions. The markings on the runway, the striation on the runway surface as well as the lines below the runway appear smoothed. The smoothing can be attributed to the fact that the data mean and the data covariance are computed from a larger analysis window. This means that the hyperpixels in the analysis window that lie further away from the hyperpixel of interest (the center hyperpixel), influence the computation of $\boldsymbol{\mu}_a$ and $\boldsymbol{\Sigma}_a$. Now consider two neighboring hyperpixels for which parameters are to be estimated. The percentage of area change in the local analysis window from one hyperpixel to another is 14% for a 7×7 window and 20% for a 5×5 window[2]. Therefore, $\boldsymbol{\Sigma}_a$ and $\boldsymbol{\mu}_a$, from hyperpixel to hyperpixel, change more slowly when a 7×7 hyperpixel window is used as compared to when a 5×5 hyperpixel window is used. This means that $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ also change slowly from hyperpixel to hyperpixel for a 7×7 window. Now consider high resolution levels of the Laplacian pyramids (that contain the sharp edge information). The slow change in $\boldsymbol{\beta}$ causes a smoothing effect from hyperpixel to hyperpixel in the fused image. As a result, the edges are smoothed and therefore the

[2] For a 7×7 hyperpixel analysis window, the total number of hyperpixels in the window is 49. When the analysis window is shifted by 1 hyperpixel location, 7 old hyperpixels are discarded and 7 new hyperpixels are introduced. However 42 hyperpixels remain unchanged. Hence the percentage of new hyperpixels is approximately 14%. Similarly for a 5×5 hyperpixel analysis window, 5 new hyperpixels or 20% new hyperpixels are introduced every time the analysis window is shifted by 1 hyperpixel.

Table 5.3: Experimental setup to compute the ML-fused image in Figure 5.4

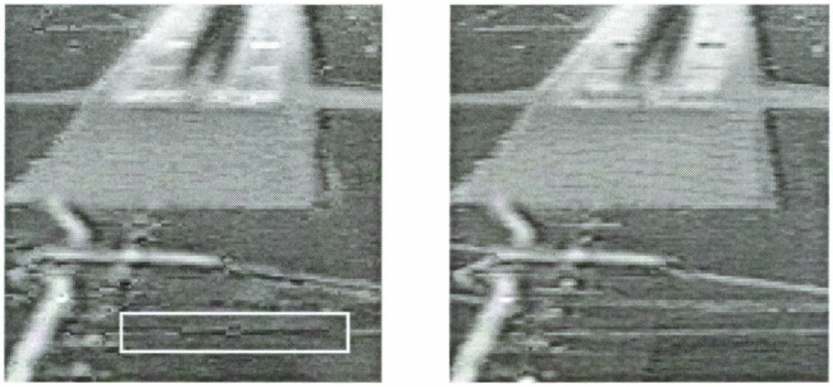
Size of images	128 × 128 pixels
Laplacian pyramid levels	7
Size of local analysis window	5 × 5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(a)
Noise variance	assumed equal in both sensors
Processing at borders	reflected hyperpixels to extend borders

fused image appears smoothed.

5.2.4 Effect of the choice of the constraint region used to determine the sign of $\boldsymbol{\beta}$

In the experiments described in Sections 5.2.1, 5.2.2 and 5.2.3, the constraint defined by the shaded region in Figure 4.8(h) was used to determine the direction of the eigenvector $\mathbf{U}_a$ (and therefore, also of $\boldsymbol{\beta}$). The reason for this constraint was that it addresses the problem of arbitrary sign reversals when there are complementary features in the sensor images (see the discussion in Section 4.5.1).

This experiment examines the effect of using the constraint defined by the shaded region in Figure 4.8(a) to determine the sign r of $\boldsymbol{\beta}$. The experimental setup is summarized in Table 5.3. Figure 5.4(a) shows the result of ML-fusion using this constraint. Recall that the constraint of Figure 4.8(a) ensures that polarity reversed features in the sensor images appear with the polarity of contrast of the first sensor image a_1. In this experiment, a_1 is the visible-band image. Therefore the runway markings appear white in the ML-fused image of Figure 5.4 just as they do in the visible-band image of Figure 5.1(a). However there is an indeterminacy of the sign of $\boldsymbol{\beta}$ at the borders of the constraint region. As a result complementary features may have arbitrary contrast reversals. This effect can be observed in the lower portion of the fused image. The horizontal line intermittently (see the highlighted box) changes contrast and appears bright and dark. The ML-fused image of Figure 5.1(e) that uses the constraint region defined in Figure 4.8(h) is shown again in



(a) Constraint of Figure 4.8(a)

(b) Constraint of Figure 4.8(h)

Figure 5.4: ML-fusion using different constraint regions to determine the sign of $\boldsymbol{\beta}$

Figure 5.4(b) for comparison.

5.3 Experiments With Images Containing Additive Noise

This experiment demonstrates our fusion technique in the presence of additive noise in the sensor images. We added samples from a Gaussian distribution to the graylevels of the visible-band and IR sensor images shown in Figures 5.1(a) and 5.1(b). The noisy images are shown in Figures 5.5(a) and 5.5(b) respectively. The visible-band image contains more noise than the IR image. The standard deviation of noise added to the visible-band image is 12 times that added to the IR image. We fused these images using the ML and MAP fusion rules[3] and compared the results with those of averaging and selection methods.

Figure 5.5(c) shows the result of fusing these images using averaging. The images were decomposed into 7 levels of the Laplacian pyramid. Hyperpixels of the sensor pyramids were averaged to obtain the fused pyramid, and the fused image was constructed by applying the inverse pyramid transform to the fused pyramid. The result of averaging suffers from the same drawback as before — reduced contrast in regions containing polarity reversed and complementary features.

Figure 5.5(d) shows the result of fusion using selection. Again 7 levels of the Laplacian

[3] A comparison of the ML and MAP fusion rules with fusion using local PCA is shown in Appendix J.

pyramid were used and a 5 × 5 area based salience measure was employed to decide which of the sensor hyperpixels to select into the fused pyramid. The result of selection is noisy. Comparing Figure 5.5(d) with the sensor images in Figures 5.5(a) and 5.5(b), one can observe that the selection technique selects features from the visible-band image in most locations of the scene. Noise in the visible-band image dominates the salience measure and noise spikes are confused as salient features.

ML fusion

The ML fusion approach uses the maximum likelihood fusion rule of Equations (3.12) and (3.13) given by

$$\begin{aligned}\hat{s}_{\mathrm{ML}} &= \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta}\right]^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a}-\boldsymbol{\alpha})\right\} \\ &= \frac{\frac{\beta_1(a_1-\alpha_1)}{\sigma_{\epsilon_1}^2}+\frac{\beta_2(a_2-\alpha_2)}{\sigma_{\epsilon_2}^2}}{\frac{\beta_1^2}{\sigma_{\epsilon_1}^2}+\frac{\beta_2^2}{\sigma_{\epsilon_2}^2}}\end{aligned} \tag{5.6}$$

and the least squares estimates of the image formation model parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ from Equations (4.27) and (4.31) given by

$$\begin{aligned}\boldsymbol{\alpha}_{\mathrm{LS}} &= \boldsymbol{\mu}_a - \boldsymbol{\beta}\mu_{s_0} \text{ and} \\ \boldsymbol{\beta}_{\mathrm{LS}} &= \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r\end{aligned} \tag{5.7}$$

As in Section 5.2.1 (also see the discussion in Section 4.5.1), we assume $\mu_{s_0} = 0$, $\| \boldsymbol{\beta} \| = 1$ and choose an appropriate sign of r to obtain,

$$\begin{aligned}\boldsymbol{\alpha} &= \boldsymbol{\mu}_a \text{ and} \\ \boldsymbol{\beta} &= \mathbf{U}\ .\end{aligned} \tag{5.8}$$

The experimental setup used for computing the fused image is summarized in Table 5.4. The ML-fused image is shown in Figure 5.5(e). Comparing with the results of averaging and selection in Figures 5.5(c) and 5.5(d), ML fusion does better than averaging and selection in regions containing polarity reversed and complementary features. The markings on the runway, the horizontal lines in the lower portion of the FLIR image and

Table 5.4: Experimental setup to obtain the ML and MAP-fused images of Figure 5.5

Size of images	128 × 128 pixels
Laplacian pyramid levels	7
Size of local analysis window	5 × 5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(h)
Noise variance	estimated from single frame
Processing at borders	reflected hyperpixels to extend borders

the bright horizontal patch just before the runway are all more distinct in the ML-fused image. The ML-fused image is less noisy compared to the averaging and selection-fused images. The ML fusion rule gives more weight to the IR image in regions where the image features are common between the visible-band and IR images, since the variance of noise in the IR image is lower than that in the visible-band image. However the ML fusion rule gives a high weight to the visible-band image in regions containing complementary features that are absent in the IR image. Consequently these complementary features are visible at high contrast in the fused image.

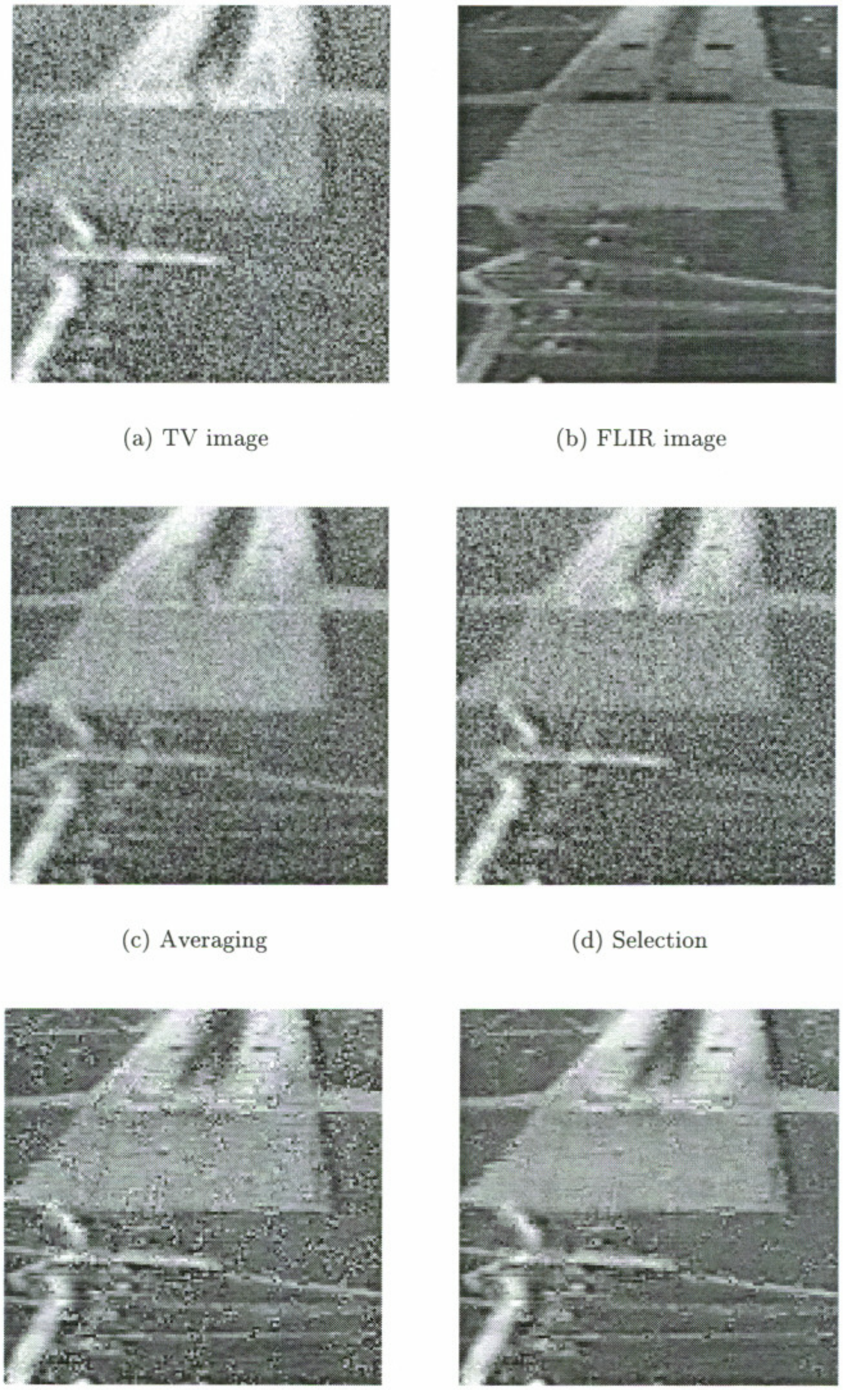
MAP fusion

The sensor images can also be combined by using the MAP fusion rule of Equations (3.14) and (3.15) given by

$$\begin{aligned}\hat{s}_{\mathrm{MAP}} &= \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta} + \frac{1}{\sigma_s^2}\right]^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\mathbf{a}-\boldsymbol{\alpha}) + \frac{s_0}{\sigma_s^2}\right\} \\ &= \frac{\dfrac{\beta_1(a_1-\alpha_1)}{\sigma_{\epsilon_1}^2} + \dfrac{\beta_2(a_2-\alpha_2)}{\sigma_{\epsilon_2}^2} + \dfrac{s_0}{\sigma_s^2}}{\dfrac{\beta_1^2}{\sigma_{\epsilon_1}^2} + \dfrac{\beta_2^2}{\sigma_{\epsilon_2}^2} + \dfrac{1}{\sigma_s^2}}\end{aligned} \tag{5.9}$$

with the least squares estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ given by

$$\begin{aligned}\boldsymbol{\alpha}_{\mathrm{LS}} &= \boldsymbol{\mu}_a - \boldsymbol{\beta}\mu_{s_0} \text{ and} \\ \boldsymbol{\beta}_{\mathrm{LS}} &= \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r\end{aligned} \tag{5.10}$$



(a) TV image

(b) FLIR image

(c) Averaging

(d) Selection

(e) ML fusion

(f) MAP fusion

Figure 5.5: Fusion of TV and FLIR images with additive Gaussian noise (Original data from SVTD [9] project).

Following the discussion in Section 4.5.1, we assume,

$$s_0 = 0 \ , \tag{5.11}$$

(therefore $\mu_{s_0} = 0$), $\parallel \boldsymbol{\beta} \parallel = 1$, and choose an appropriate sign r to obtain,

$$\begin{aligned} \boldsymbol{\alpha} &= \boldsymbol{\mu}_a \text{ and} \\ \boldsymbol{\beta} &= \mathbf{U} \ . \end{aligned} \tag{5.12}$$

The assumption $\parallel \boldsymbol{\beta} \parallel = 1$ gives,

$$\lambda = \sigma^2_{s,s_0} = \sigma^2_s + \sigma^2_{s_0} \tag{5.13}$$

and the assumption $s_0 = 0$ gives

$$\sigma^2_{s_0} = 0 \tag{5.14}$$

From Equations (5.13) and (5.14), we obtain an estimate of σ^2_s as

$$\sigma^2_s = \lambda \tag{5.15}$$

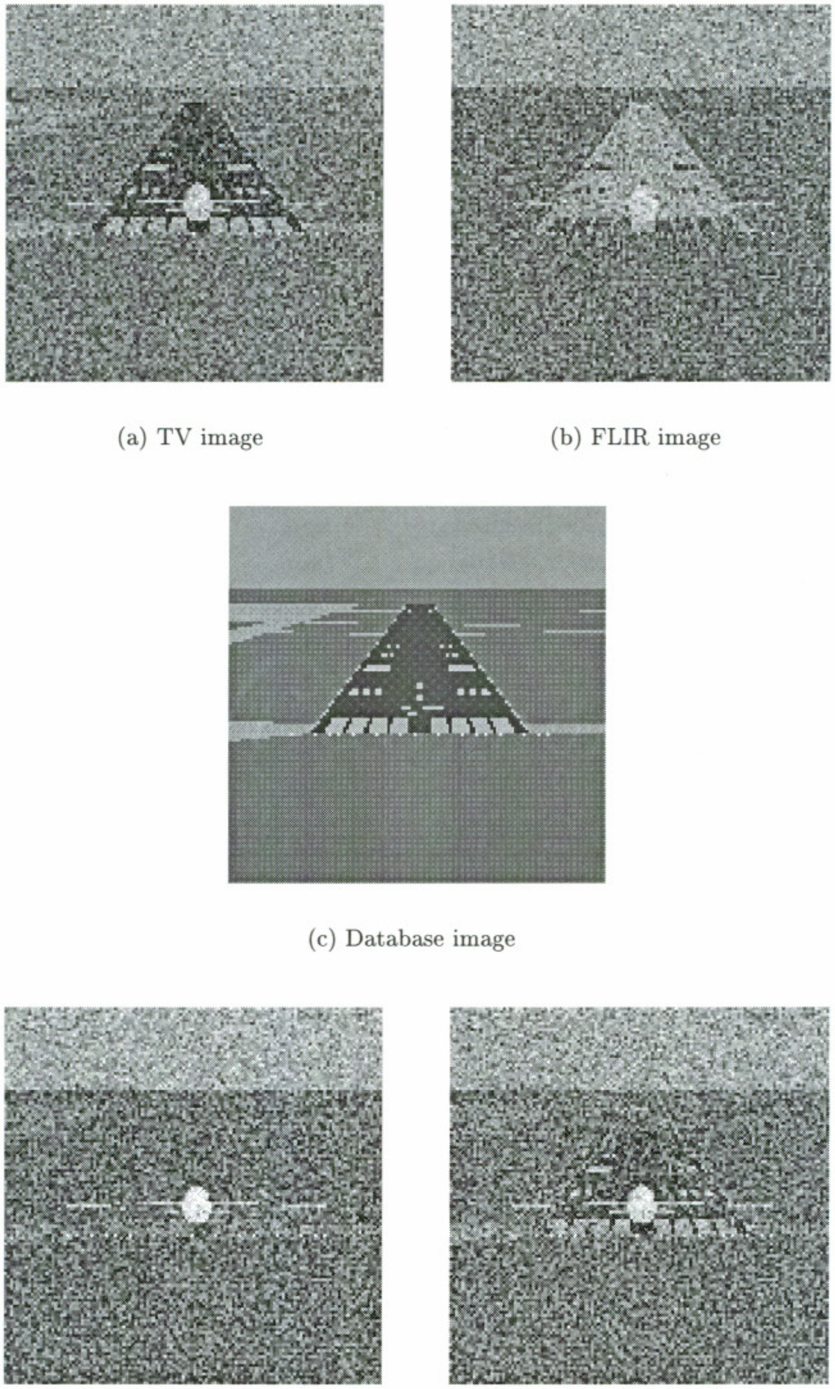
The MAP-fused image is obtained by substituting Equations (5.12), (5.11) and (5.15) in Equation (5.9). The noise variance is estimated as described in Section 4.3.1. The experimental setup used for obtaining the fused is identical to that in Table 5.4. We call this approach MAP$_1$ fusion to distinguish it from the MAP fusion approaches described in Section 5.4. Figure 5.5(f) shows the image obtained using MAP$_1$ fusion. Clearly, the MAP$_1$-fused image is less noisy compared to the ML-fused image. The estimate of σ^2_s does not contain any contribution from noise (since $\sigma^2_s = \lambda$, and λ is the principal eigenvalue of $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$). In regions of relatively low sensor image contrast such as flat regions in the scene, σ^2_s is low (since λ is low). Therefore the denominator in Equation (5.9) is large and the contribution from the sensor images is attenuated as compared to the ML fusion rule. Hence the noise in the flat regions is also attenuated. In regions of high contrast in the sensor images such as edges in the scene, σ^2_s is high (since λ is high). Therefore the denominator in Equation (5.9) is approximately equal to the denominator in Equation (5.6). The contribution from the sensor images is then similar to that in ML-fusion (since $s_0 = 0$).

Assuming $\sigma_{s0}^2 = 0$ causes the spatial variations in the scene, captured by the sensor images, to be attributed to σ_s^2. This results in an overestimation of σ_s^2 especially in regions of relatively high scene contrast in the sensor images. However, this overestimation turns out to be beneficial for fusion, as the high value attributed to σ_s^2 results in more weight to be given to the sensor images.

5.4 Fusion Using Prior Image Information From a Database

As previously discussed in Section 3.4.2, the probabilistic MAP fusion rule has a provision to include prior image information about the scene into the fused image. We demonstrate the use of this provision with the help of the ALG example. In ALG, prior knowledge about the scene may be available in the form of a terrain database of the scene. The terrain database provides an image of the scene under ideal viewing conditions (uniform illumination, unlimited visibility). However the database image is not by itself sufficient for ALG because the actual situation in the real runway scene may differ from that in the database image. The database image information must be combined with image data from the sensors to properly depict the runway scene in the fused image. We demonstrate two different methods of applying the probabilistic MAP fusion rule to combine the prior information from the terrain database with image data from the sensors.

We illustrate the use of prior information using simulated images and compare them with averaging and selection based fusion, as well as the ML fusion and MAP$_1$ techniques described in Section 5.3. We generated a sequence of noisy sensor images by adding Gaussian noise to one simulated image each from visible-band and IR (FLIR) sensors. Figure 5.6(a) shows one frame of the noisy visible-band sequence. Figure 5.6(b) shows one frame of the noisy IR sequence. These simulated sensor images depict a runway scene with an aircraft on it. The polarity of contrast of the runway surface and markings is reversed in the IR image. The taxiways that are visible on the left and right of the runway in the visible-band image have not been sensed by the IR sensor and are missing in the IR image. The variance of the added Gaussian noise in both the sensor images is equal. Figure 5.6(c) shows an image of the same scene as might be obtained from a terrain database. Although



(a) TV image

(b) FLIR image

(c) Database image

(d) Averaging

(e) Selection

Figure 5.6: Fusion of simulated TV and FLIR images

this image is clean, it does not show the actual situation on the runway — the aircraft in the sensor images is absent in the database image.

Figure 5.6(d) shows the result of fusing the sensor images of Figures 5.6(a) and 5.6(b) using averaging. Averaging was performed using 7 levels of the Laplacian pyramid. Averaging does not fuse well in regions containing polarity reversed and complementary features. The polarity reversed runway and markings have almost disappeared and the contrast of the taxiways has been reduced. Figure 5.6(e) shows the fused image obtained by selection using an area based salience measure and 7 levels of the Laplacian pyramid. Selection performs better than averaging in the polarity reversed regions, but the fused image is noisy.

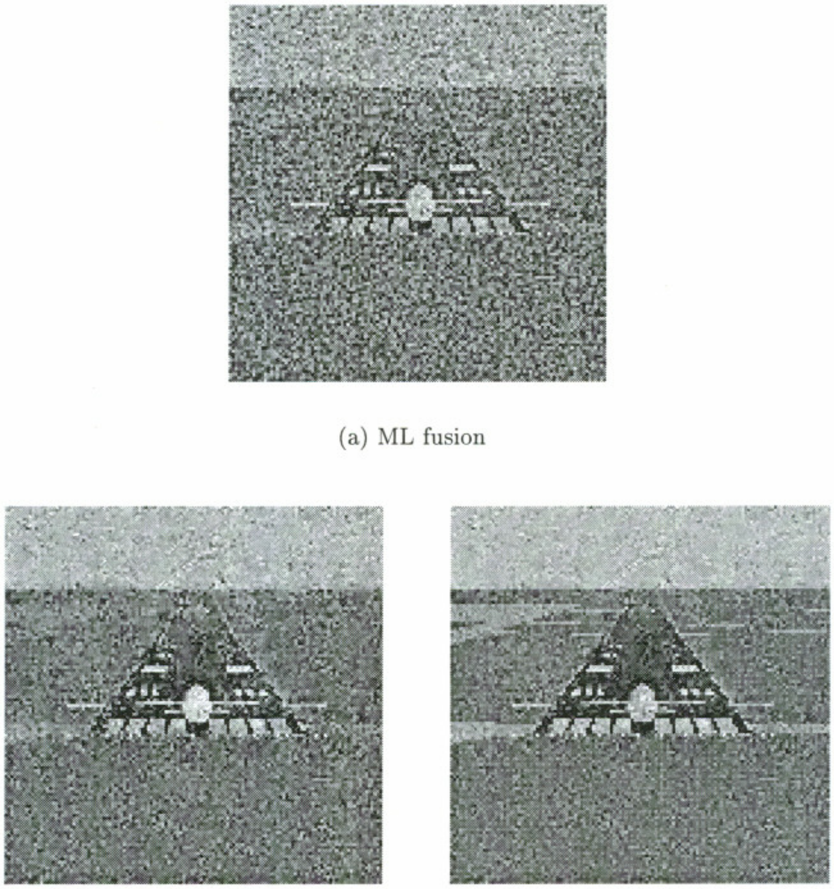
We combined the simulated noisy sensor video sequences using ML fusion and MAP_1 fusion described in Section 5.3. Every pair of video frames from the sensor video sequences was combined using probabilistic fusion. We estimated the noise variance in the sensor video sequences using the multiple frame noise estimation technique described in Section 4.3.2[4]. The experimental setup used for computing the fused image using the probabilistic fusion rules is summarized in Table 5.5. For the remainder of this section, we will focus on results of fusing the image frames shown in Figures 5.6(a) and 5.6(b).

We combined the images in Figures 5.6(a) and 5.6(b) using the ML fusion rule and the MAP_1 fusion rule described in Section 5.3. The experimental setup used for computing the fused image is summarized in Table 5.5. Figure 5.7(a) shows the result of combining the images by the ML fusion rule using Equations (5.6) and (5.8)[5]. The result of ML fusion is noisy because both the sensor images are equally noisy. However, ML fusion performs better than fusion by selection as can be seen by the higher contrast in regions containing the runway and runway markings.

Figure 5.7(b) shows the result of combining the images by MAP_1 fusion using Equations (5.9), (5.11), (5.12) and (5.15). The noise variance was estimated using multiple

[4] The simulated noisy frames for each sensor were generated by adding different realizations of Gaussian noise with the same standard deviation to a clean simulated frame. Motion compensation was not necessary since the multiple frames were generated from the same clean simulated image.

[5] Since the noise variance in the sensor images is equal, the ML-fused image in this experiment can also be computed as a local PCA projection.



(a) ML fusion

(b) MAP_1 fusion

(c) MAP_2 fusion

Figure 5.7: Fusion of simulated TV and FLIR images (continued). The two MAP methods shown, use prior image information from a terrain database

Table 5.5: Experimental setup for fusion of noisy sensor images with prior image information from a database

Size of images	128×128 pixels
Laplacian pyramid levels	7
Size of local analysis window	5×5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(h)
Noise variance	estimated from multiple frames
Processing at borders	reflected hyperpixels to extend borders

frames. The result of MAP$_1$ fusion is less noisy as compared to ML fusion. As explained in the MAP fusion experiment of Section 5.3, $\| \boldsymbol{\beta} \| = 1$ and $s_0 = 0$ implies $\sigma_s^2 = \lambda$. Although both sensor images are noisy, σ_s^2 regulates the weight given to the sensor images in different regions of the scene, in accordance with the reliability of the features in the sensor images. For example, in flat noisy regions in the sensor images, σ_s^2 is low. Hence the contribution from the sensor images is attenuated. In high contrast regions in the sensor images, σ_s^2 is high, preventing attenuation of the contribution from the sensor images. As a result, the runway surface, markings and aircraft are prominent in the MAP$_1$ fused images. The taxiways are also visible. The image is cleaner than either sensor image. But there is some loss in sharpness at the edge at the boundary caused by the horizon and the sky. Note that no database information was used to construct the images in Figures 5.7(a) and 5.7(b).

We now describe two different methods by which the database information can be included in the fused image:

MAP fusion using the database image and assumptions on the prior σ_s^2

In the first method, MAP$_2$, the database image is used as s_0. The experimental setup for this method is identical to that in Table 5.5. In addition, the database image is also decomposed into 7 levels of the Laplacian pyramid. The MAP$_2$ fused image is obtained using

$$\begin{aligned}\hat{s}_{\text{MAP}} &= \left[\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}\boldsymbol{\beta}+\frac{1}{\sigma_s^2}\right]^{-1}\left\{\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_{\epsilon}^{-1}(\boldsymbol{a}-\boldsymbol{\alpha})+\frac{s_0}{\sigma_s^2}\right\} \\ &= \frac{\dfrac{\beta_1(a_1-\alpha_1)}{\sigma_{\epsilon_1}^2}+\dfrac{\beta_2(a_2-\alpha_2)}{\sigma_{\epsilon_2}^2}+\dfrac{s_0}{\sigma_s^2}}{\dfrac{\beta_1^2}{\sigma_{\epsilon_1}^2}+\dfrac{\beta_2^2}{\sigma_{\epsilon_2}^2}+\dfrac{1}{\sigma_s^2}}\end{aligned} \tag{5.16}$$

and the least squares parameter estimates

$$\begin{aligned}\boldsymbol{\alpha}_{\text{LS}} &= \boldsymbol{\mu}_a-\boldsymbol{\beta}\mu_{s_0} \\ \boldsymbol{\beta}_{\text{LS}} &= \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r \ .\end{aligned} \tag{5.17}$$

where we assume $\| \boldsymbol{\beta} \| = 1$ to give

$$\boldsymbol{\beta}_{\text{LS}} = \mathbf{U} \ . \tag{5.18}$$

The value for s_0 in Equation (5.16) is given by the hyperpixel in the database image. The parameter μ_{s_0} given by Equation (4.19) and is estimated at each location in the pyramid by computing the mean in a 5 × 5 hyperpixel region in the database image.

$$\mu_{s_0} \approx \frac{1}{N}\sum_{n=1}^{N} s_{0n} \tag{5.19}$$

where $N = 25$ is the number of hyperpixels in the 5 × 5 region. As in MAP$_1$ fusion, we make the assumption $\sigma_s^2 = \lambda$ so that the contribution of the database image scales with the reliability of the sensor images. In flat noisy regions of the sensor images, σ_s^2 is low, hence the contribution of the database images s_0 is increased. Conversely, in high contrast regions of the sensor images, σ_s^2 is high, hence the contribution of the database image s_0 is reduced.

Figure 5.7(c) shows the result of MAP$_2$ fusion. The fused image appears better than either of the sensor images and the fused images of Figures 5.6(d), 5.6(e),5.7(a) and 5.7(b). Local polarity reversed features are preserved in the fused image and have high contrast. The taxiway is clearly visible in the fused image. Noise in the flat regions as well as the edges is reduced. The aircraft on the runway is prominent and also has high contrast. The fusion rule has given more weight to the database image in regions where the sensor images are less reliable.

Using σ_s^2 as a parameter to control the contribution of the database

We describe another method, MAP$_3$, where the database image is again used as s_0. In this method we specify the prior variance σ_s^2 and use it as a measure of confidence in the database image. The value of σ_s^2 controls the relative contribution of the sensors versus that of the database image in the fused image. A high value of σ_s^2 corresponds to low confidence in the database. Conversely, a low value of σ_s^2 corresponds to high confidence in the database.

The confidence assigned to each spatial location in the database image must be propagated to the same spatial locations in the pyramid that are at different levels of the pyramid. In the experiment described below, we achieve the propagation in the following manner. We generate a sequence of images containing white Gaussian noise with a specific standard deviation. The standard deviation of noise corresponds to the confidence in the database image. We then decompose the sequence of images into Laplacian pyramids and compute the variance at each hyperpixel location from the sequence. This variance is the value we have assigned to σ_s^2 at each hyperpixel[6].

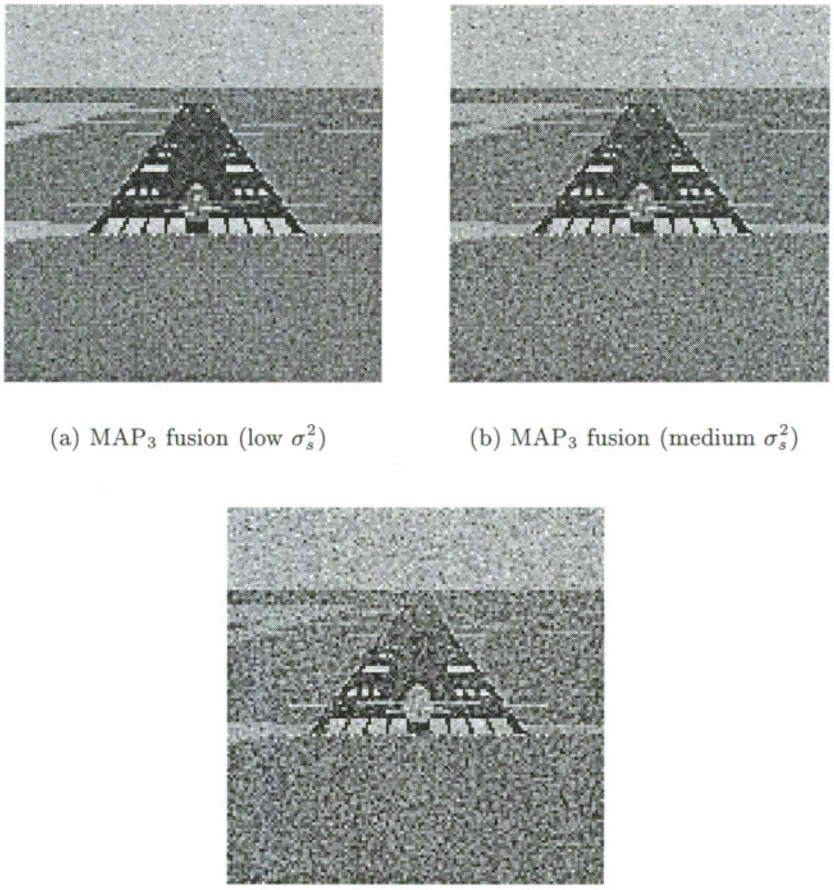
The MAP$_3$ fused image is also computed using the MAP fusion rule of Equation (5.16). The parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are computed using the least squares estimates of Equation (5.17). The spatial mean μ_{s_0} is computed as in Equation (5.19). We also compute an estimate of $\sigma_{s_0}^2$ at each hyperpixel using the database image s_0 as,

$$\sigma_{s_0}^2 \approx \frac{1}{N}\sum_{n=1}^{N}(s_{0n} - \mu_{s_0})^2 \tag{5.20}$$

where $N = 25$ is the number of hyperpixels in a 5×5 hyperpixel region. We can now compute σ_{s,s_0}^2 given by Equation (4.22), using the value of σ_s^2. Therefore, we have all the quantities needed for estimating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, and the fused image.

The experimental setup to compute a fused image using MAP$_3$ fusion is the same as that in Table 5.5. The results of fusion using three different values of σ_s^2 are shown in

[6] The confidence at each hyperpixel can be computed from the confidence assigned to each location in the database image using the mathematical expressions for the operations involved in the pyramid computation. However, the operations in the Laplacian pyramid computations involve several steps including downsampling and upsampling operations that make it difficult to propagate the confidence through the pyramid levels. Therefore, for the purpose of the MAP$_3$ experiment, we chose to use an empirical solution.



(a) MAP$_3$ fusion (low σ_s^2)

(b) MAP$_3$ fusion (medium σ_s^2)

(c) MAP$_3$ fusion (high σ_s^2)

Figure 5.8: Controlling the contribution of the prior image

Figures 5.8(a), 5.8(b) and 5.8(c). The standard deviation of the Gaussian noise images generated to compute the values of σ_s^2 for these three cases was 20, 25 and 40 respectively. From Equation (5.16), higher values of σ_s^2 reduce the weight given to the database image hyperpixels s_0 and therefore accentuate the contribution of the sensor images relative to the database image. Conversely, lower values of σ_s^2 increase the weight given to the database image hyperpixels s_0 and therefore accentuate the contribution of the database image in the fused image.

5.5 Evaluation of Fusion

In the previous sections we have demonstrated that the fused images obtained using the probabilistic fusion rules appear better than the traditional techniques of averaging and selection. The probabilistic fusion rules overcome the disadvantages of the traditional techniques outlined in Section 2.7, particularly when the sensor images are noisy. However, a quantitative evaluation of the fused images is necessary to compare fusion techniques. A typical approach to the characterization of systems that enhance signals and images is to specify signal-to-noise ratios. One problem with this approach for the purpose of evaluating fusion is that the desired fused result is not known. We overcome this problem by starting with a known desired image. We then examine the effectiveness of fusion algorithms using sensor images that are generated from the desired image.

We quantitatively evaluated the MAP fusion rule under local polarity reversals. A square wave grating, Figure 5.9(a), was used as the known desired image for evaluation[7]. Two sensor images a_1 and a_2 were constructed as transformations of the desired grating image s, perturbed by additive Gaussian noise. The image a_1 was generated as

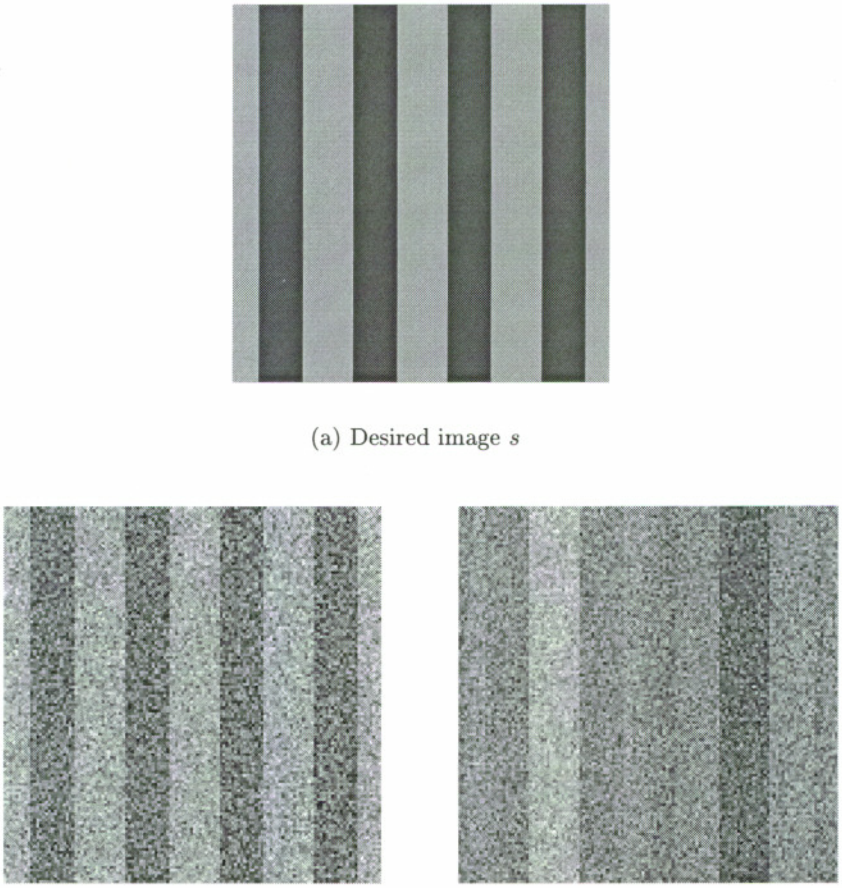
$$a_1 = s + \epsilon_1 \tag{5.21}$$

The second sensor image a_2 was generated by modulating the square wave grating by multiplying with one cycle of a sine wave. The modulated grating was then perturbed by additive Gaussian noise.

$$a_2 = ms + \epsilon_2 \tag{5.22}$$

where m represents the sine wave. Modulation by a sine wave simulates local polarity reversal. The image a_2 is shown in Figure 5.9(c). The stripes on the left side of the image have the same polarity or are in phase with the stripes in Figure 5.9(b). Whereas the stripes on the right have opposite polarity or are out of phase. We varied the noise power contained in a_1 and a_2 and compared the result of averaging, selection and MAP fusion. The fused images for a particular noise setting are shown in Figure 5.10. Each

[7] The reason for using a square wave grating was that it represents all spatial frequencies in roughly the same proportion that they are observed in natural images (i.e., $1/f$, where f is the spatial frequency).



(a) Desired image s

(b) Sensor image a_1

(c) Sensor image a_2

Figure 5.9: Images used for evaluating fusion results

fusion technique employed 7 levels of the Laplacian pyramid. The fused image obtained by averaging, Figure 5.10(a), has the right half wiped out due to pattern cancellation of local polarity reversed stripes. The fused image obtained by selection, Figure 5.10(b), is noisy. The contrast of the stripes on the right is reduced. The result of ML fusion is shown in Figure 5.10(c). ML fusion eliminates the problems caused by the polarity reversals. However, the fused image is still noisy. The MAP fused image is obtained using the MAP$_1$ method described in Section 5.3 and is shown in Figure 5.10(d). The stripes in the grating are clearly visible in the MAP-fused image. The image appears less noisy than the sensor images and the fused images obtained by averaging and selection. To quantitatively evaluate the fused images, we have developed an error measure based on root-mean-square (RMS) noise. The RMS error between the desired image s and its estimate $\hat{s}$ is given by

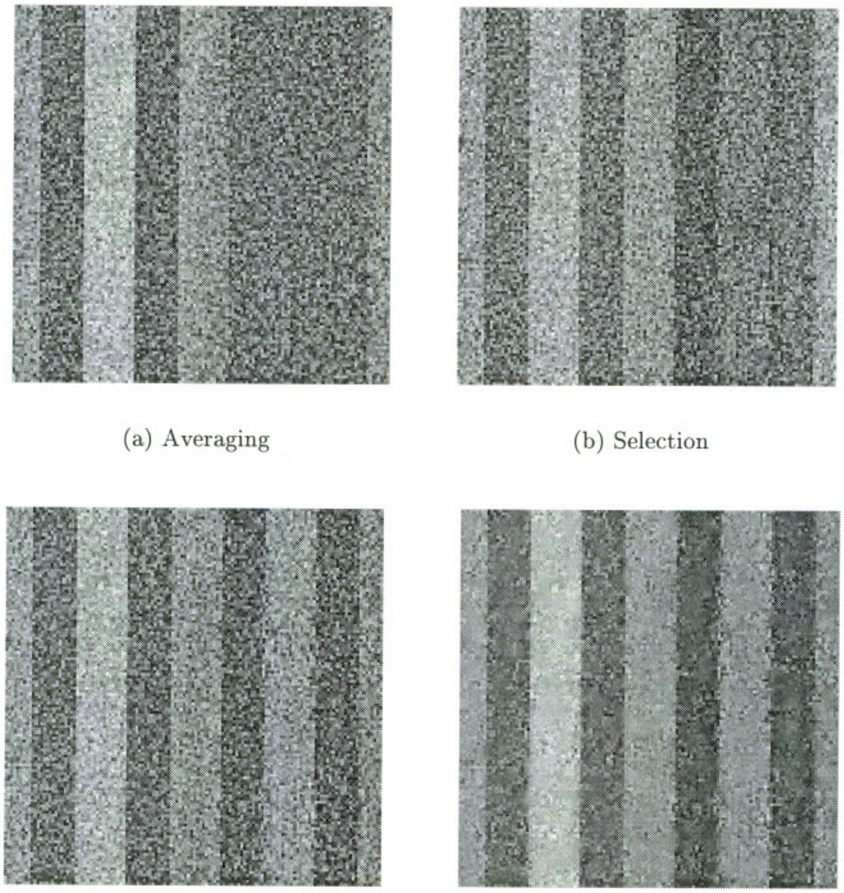
$$e_{rms} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(s_i - \hat{s}_i)^2} \tag{5.23}$$

where i refers to the i^{th} pixel in each image and N is the total number of pixels. However, this error measure penalizes the fused image for incorrect gain or a shift in the graylevels. The comparison of the fused images should be insensitive to gain differences or shift differences in the fused images. To address this problem, we have developed an error measure that is invariant to scaling and shifting of the graylevels in the fused image. We find regression parameters γ and δ that minimize the difference between the fused image $\hat{s}$ and the image

$$\tilde{s} = \gamma s + \delta \tag{5.24}$$

Note that γ and δ hold over the entire image and therefore are not local in nature. These parameters are obtained by minimizing

$$\sum_{i=1}^{N}(\hat{s}_i - \gamma s_i - \delta) \tag{5.25}$$



(a) Averaging

(b) Selection

(c) ML fusion

(d) MAP fusion

Figure 5.10: Fusion of noisy grating images

and are given by

$$\gamma = \frac{\frac{1}{N}\sum_{i=1}^{N} \hat{s}_i s_i - \frac{1}{N}\sum_{i=1}^{N} \hat{s}_i \frac{1}{N}\sum_{i=1}^{N} s_i}{\frac{1}{N}\sum_{i=1}^{N} s_i^2 - \frac{1}{N}\sum_{i=1}^{N} s_i \frac{1}{N}\sum_{i=1}^{N} s_i} \tag{5.26}$$

and

$$\delta = \frac{1}{N}\sum_{i=1}^{N} \hat{s}_i - \gamma \frac{1}{N}\sum_{i=1}^{N} s_i \ , \tag{5.27}$$

where i is the i^{th} pixel and N is the number of pixels in each image. The RMS error, invariant to scaling and shifting of the graylevels, is given by

$$\hat{e}_{rms} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (s_i - \tilde{s}_i)^2} \ . \tag{5.28}$$

The fused images were evaluated using this error measure as a function of the noise added to the sensor images. The graph in Figure 5.11 shows a plot of the RMS error of the fused images versus the signal-to-noise ratio in the first sensor image. MAP fusion has the lowest RMS error. Averaging does worse than MAP fusion. Selection has the highest error due to the noise in the sensor images. The difference between the errors due to MAP fusion and averaging is more pronounced when the input signal-to-noise ratio is low. Note that the errors produced by MAP and averaging are similar at high input signal-to-noise ratios. However, the MAP-fused image looks visually better than the fused image obtained by averaging which results in pattern cancellation. The fact that the error produced by averaging and MAP fusion is similar at high input signal-to-noise ratios may be explained by two factors. First, observe that the left half of the sensor images a_1 and a_2 have the same polarity of contrast. Therefore, the RMS error is low in the left half of the fused image obtained by averaging. However, the error in the right half is higher due to the polarity reversal. Second, note that there is a slight modulation in the left half of the MAP-fused image due to which this half looks brighter. The brightness increases from the left side to the center of the image. The sine wave in sensor image a_2 seems like a complementary feature to the probabilistic fusion rule at the level of the pyramid

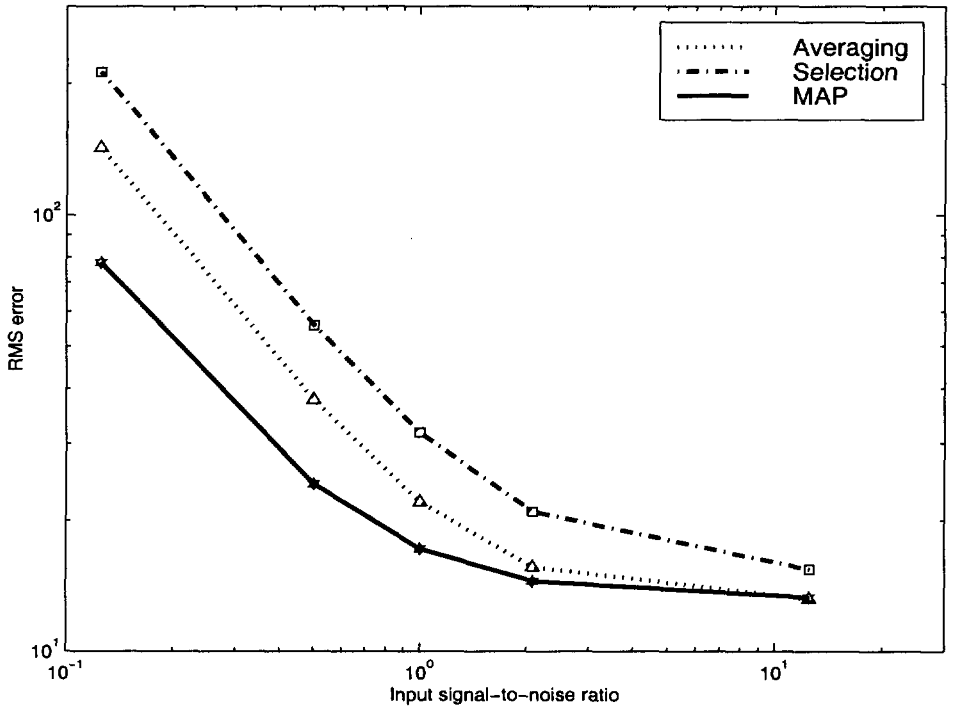


Figure 5.11: Evaluation of fusion results using a scale and shift invariant error measure

where this sine wave component is prominent. As a result, the sine wave is retained in the fused image and appears as modulation. This modulation increases the error. Now observe the right half of the MAP-fused image. There is a similar modulation in the other direction due to which this half appears darker. Again, this increases the error. Note that though the modulation due to the sine wave appears in the MAP-fused image, the effect of polarity reversal has been removed. At low input signal-to-noise ratios, the error due to the noise in the averaging fused image dominates the error caused by the modulation in the MAP-fused image and therefore, the MAP-fused image produces a lower error. The evaluation approach described above, brings out the drawback of quantitative evaluation of fused images. The quantitative evaluation results do not exactly correspond to the visual results. The results suggest that more sophisticated quantitative measures are necessary to evaluate fused images. Evaluation techniques would benefit from a knowledge of task (e.g. detection, search) specific information as well as a modeling of the characteristics of the human visual system using a vision model [50, 45, 75].

5.6 Summary

In this chapter, the probabilistic fusion rules and the estimates of model parameters developed in Chapters 3 and 4 were used to fuse multisensor images. The experiments described in this chapter demonstrate the efficacy of the probabilistic fusion approach. Probabilistic fusion preserves the contrast of local polarity reversed features and complementary features. If the variance of noise in the sensor images is equal or if the noise variance is zero, the probabilistic fusion rule is just a local PCA projection. The results of probabilistic fusion on sensor images with small noise are comparable to that of selection-based fusion as shown in Figure 5.1. When the variance of noise in the sensor images is high, but equal, the probabilistic MAP fusion rule outperforms other fusion techniques as shown in Figure 5.7(c). When the variance of noise in the sensor images is unequal, the probabilistic fusion gives a lower weighting to the sensor having larger noise. However complementary features from the noisy sensor are retained in the fused image. The results suggests that the model-based probabilistic fusion approach does overcome the drawbacks faced by traditional fusion techniques.

The probabilistic fusion approach also has the provision to include prior imagery of the scene into the fused image. The results described in Section 5.4 illustrate the usefulness of the ability to include prior information into the fused image. The experiments demonstrate that the information in the prior image can be balanced against the information in the sensor images. The probabilistic fusion approach can be adapted to either use the prior image in conjunction with confidence in the sensor images or to externally control the weightage given to the prior image in the fused image.

We have also quantitatively evaluated the fusion results. The evaluation methodology consisted of constructing a set of sensor images from a known image and computing a fused image from these sensor images. The fused image was then compared to the known image using a RMS based error measure. Although the evaluation results do not entirely agree with the visual results, they do indicate that the probabilistic fusion approach performs better than the traditional techniques.

The probabilistic fusion approach can be used for fusion of images from multiple sensors as well as video sequences from multiple sensors as demonstrated by the experiments described in this chapter. The computational complexity of the probabilistic fusion approach is discussed in Appendix I. In this appendix, we have also outlined ways for reducing the computational complexity for practical implementations of probabilistic fusion.

We illustrated our approach using runway images from the landing guidance application in aviation as an example. However, since we do not make specific assumptions about the image content, our approach is equally effective on other types of multisensor images. The fusion experiment described in Appendix H demonstrates the effectiveness of our approach in fusing hyperspectral images.

Chapter 6

Conclusions and Future Work

We have presented a probabilistic model-based approach for fusion of images from multiple sensors. The probabilistic model incorporates the process by which the scene gives rise to the sensor images. The model explicitly accounts for contrast reversals, complementary features and noise, which adversely affect existing fusion techniques. We derived our fusion rules and estimation of model parameters using a theoretical framework. The results that we have presented illustrate that our fusion approach yields improved fusion results as compared to the existing techniques.

Although the combination of the sensor images has been the central focus of this thesis, we have also presented techniques for conformal geometric representations, multisensor image registration and interpretable display of fused images. The solutions to these issues are integral to most practical applications involving multisensor image fusion. These techniques are described in Appendix A, B and C.

In this chapter, we first present our conclusions in Section 6.1. In Section 6.2 we suggest directions for future work from the point of view of extending the work presented here as well as applying our fusion framework to other problem domains.

6.1 Conclusions

In this section we present the conclusions from the theory and experimental results presented in Chapters 3, 4 and 5. We outline the salient contributions of our probabilistic model-based multisensor image fusion approach and the advantages offered by this approach. We also note the limitations of our approach in terms of the validity of our assumptions and computational considerations.

6.1.1 Salient contributions of probabilistic model-based fusion

Feasibility of a model-based approach

We have developed a principled approach to the problem of fusion of multisensor images. Under certain assumptions, we explicitly modeled the process by which the true underlying scene gives rise to the sensor images. Our model, defined within a multiresolution Laplacian pyramid representation, explicitly accounts for mismatched, noisy image features. We derived a theoretical framework for fusion based on this model. We showed how our proposed fusion solution can be used to fuse a pair of sensor images, or a sequence of images from multiple sensors (i.e., video fusion).

Efficacy of local affine functions

We have shown that our simple model consisting of noisy, locally affine functions is effective in extracting the complex local relationships between the sensor images and the scene. In the example images we analyzed, this model was able to capture the effects of local polarity reversals, complementary features and noise.

Probabilistic framework effective in characterizing noise

The probabilistic framework of our fusion approach is advantageous in adapting the fusion rules in cases of changing signal and noise conditions. We have shown that our probabilistic approach of characterizing the uncertainty due to noise in the sensor images is important for fusion. Our results indicate that incorporating this uncertainty in our model alleviated the sensitivity of fusion to noise.

Simple interpretation of fusion rules

The probabilistic model-based approach estimates the underlying scene from the sensor images. These estimates constitute the fusion rules. The fusion rules resemble PCA-like projections and are easy to interpret. The fused images obtained by these rules are locally weighted linear combinations of the sensor images. The weights scale the sensor images according to the signal and noise content in the sensor images.

Improved fusion results

The results of our fusion experiments show that our approach addresses the problems faced by existing fusion techniques based on selection and averaging methods, while retaining their advantages. The fused images produced by our probabilistic fusion have relatively lower noise, and show better contrast and feature retention than selection or averaging methods. A quantitative evaluation shows that probabilistic fusion outperforms the existing techniques in noisy conditions.

Inclusion of prior knowledge provides reliable fusion

We have shown that prior knowledge about the scene, in the form of a prior image, can be included in the fused image in a principled manner using the Bayesian approach. Simulated experiments using a prior image from a terrain database show promising results. The fused image retains features from the sensor images and the prior image and is less noisy than the sensor images. We have also shown that the contribution of the prior image in the fused image can be controlled depending upon the confidence in the prior.

6.1.2 Limitations

Our fusion approach is based on a set of assumptions as described in Section 3.3.1. The approach applies well when the assumption of the local affine functions holds. The approach would probably deteriorate under situations where the local affine mapping does not apply, or where the noise in a sensor image is correlated with the scene or correlated across sensors. The parameter estimation would be adversely affected in cases where the assumption of slow spatial change of the affine parameters does not apply.

The advantages offered by our probabilistic fusion approach come at an increased computational cost. Computation of the fused pyramid requires approximately 350 operations per location in the pyramid, not including any computations required for motion compensation for noise variance estimation. However, these operations could be performed in parallel for each location in the pyramid. Alternatively, the model could be simplified to reduce the number of estimations needed. Computational considerations are discussed in Appendix I.

Finally, our conclusions are based on experimental results obtained using a small set of real and simulated images and with respect to simulated noise conditions. To examine the extent to which our approach is applicable in real situations would require fusion to be performed on live data from multiple sensors. In addition to a quantitative evaluation, the results would have to be evaluated with respect to the specific application for which multisensor image fusion is to be used — for example, evaluation with respect to the human visual system.

6.2 Suggestions for Future Work

6.2.1 Extension of our fusion approach

Simplifying the model

Currently the image formation model consists of an affine mapping at each hyperpixel within the pyramid hierarchy. This generates an overabundance of model parameters, particularly when the image features change slowly from one hyperpixel to another. Early experiments show that the model can be simplified by using the same model parameters over regions of several square hyperpixels rather than recalculating for each hyperpixel. This would reduce the computational complexity of the approach. A further refinement could be provided by adopting a mixture model [66] to build up the image formation model.

Multiple frames for estimating the scene and model parameters

We use image data from multiple frames within a sensor video sequence for estimating the noise variance. However, the rest of the estimation uses a single image from each sensor. Our approach can be extended to use multiple motion compensated frames from each sensor to estimate the scene s as well as the parameters, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, of the local affine functions. This would be a logical extension of the use of multiple frames for estimating the noise variance. We believe that increased use of video information from multiple frames would provide further robustness to noise and improved fusion results. In a preliminary experiment we used a spatiotemporal local analysis window extending over 5×5 hyperpixels and 10 frames to compute the data covariance matrix and the data mean that are used to estimate the affine parameters. The generated fused image was less noisy than that produced by a single frame parameter estimation. The probabilistic model framework would have to be extended to derive the fusion rules and the parameter estimates from multiple frames.

Investigation of other multiresolution representations

Our approach was based on decomposing the images into multiresolution Laplacian pyramids, generating the fused pyramid by applying the fusion rules, and obtaining the fused image from this fused pyramid. The use of other multiresolution representations (for example, wavelet transforms) within our framework could be investigated to determine if they are better suited for performing fusion.

6.2.2 Application to other domains

Although different fusion problems have a different structure, their solutions can lead to similar approaches. Below we discuss a few applications to which our fusion framework can be applied.

Multifocus fusion and composite imaging

In multifocus fusion, images obtained from the same camera but with different focus settings are fused [16]. The change in focus could be modeled as a linear transformation of the scene by the camera. Therefore, our approach could be adapted to multifocus fusion. In composite imaging [16], multiple narrowband images are fused to obtain a broadband image. Here too, each narrowband image could be modeled as a linear transformation of an underlying broadband image.

Multisensor image registration

Image registration consists of aligning a pair of images such that the corresponding features coincide. Local polarity reversals and complementary features in multisensor images cause additional difficulties in traditional registration. Identification of polarity reversed and complementary features would facilitate improved registration (see Appendix B). To do this, one would require to fuse the images. However, fusion requires registered images. Our image formation model for multisensor images could be modified to account for misregistration. For example, $a_i(\vec{l}, \vec{m})$, where $\vec{m}$ is a motion vector could be an affine function of s as in Equation 3.1. The task would then be to estimate the motion vectors in addition to the other parameters.

Application to speech processing

Consider the problem of combining two noisy streams of a speech signal obtained from two different microphones. Each stream could be modeled as a linear transformation by the microphone of the underlying speech signal with additive noise. An approach similar to the one described in this work, could also be applied to enhancement of noisy speech. Consider, for example, that different bands of the short-time speech spectrum are the multiple observations. Here, the model would define the observations as noisy linear transformations of the underlying speech spectrum. Moreover, the model would be local since a different linear transformation would apply to each short-time speech segment. This model framework is very similar to our model. The observations could then be fused to obtain an enhanced estimate of the underlying speech spectrum.

Pattern recognition

In pattern recognition, the objective is to classify a feature vector of an observed signal as one of several patterns or classes [25]. Consider the case where features from multiple observations are available. These features can be considered as linear (or affine) transformations of an underlying feature. The linear transformation would model attenuation or amplification of the features, complementarity and noise. The multiple features could be fused to obtain more reliable features for classification.

Superresolution

Superresolution consists of generating a high resolution image from a sequence of low resolution images of a scene [65]. This problem is similar to the speech enhancement problem discussed above. Again, the low resolution images could be modeled as linear transformations of the high resolution image. The fusion problem would then be to estimate the high resolution image from the observed low resolution ones. Here, as in registration, the model would have to account for motion between the low resolution images.

Video frame interpolation

The task in video frame interpolation is to interpolate the missing video frame between two available frames. This problem could be expressed in terms of a model similar to the one described in this work. The available frames could be considered as noisy affine mappings of the missing frame. Uncovered backgrounds and occlusions caused by moving objects, give rise to complementary information. Similarly, lighting variations would cause a change in gain. The mappings would have to be a function of the local motion vectors between the available frames.

6.3 Final Remarks

In this dissertation we presented a probabilistic model-based solution to the problem of multisensor image fusion. The combination of information from multiple sensors will find increased application in a large variety of problems, given that the available computational power is steadily increasing. This would entail fusion of sensors as disparate as say imaging and acoustic sensors. We have shown that using simple models and dealing with uncertainty can help in obtaining reliable and improved fusion results. We hope that this work will encourage the further development of probabilistic model-based fusion approaches.

Bibliography

[1] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3):283–310, 1989.

[2] S. T. Barnard and M. A. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, December 1982.

[3] A. Basilevsky. *Statistical Factor Analysis and Related Methods.* Wiley, New York, 1994.

[4] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Heirarchical model based motion estimation. In *Second European Conference on Computer Vision (ECCV'92)*, pages 237–252, 1992.

[5] J. R. Bergen and P. J. Burt. A three-frame algorithm for estimating two-component image motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(9):886–896, September 1992.

[6] V. Bhaskaran and K. Konstantinides. *Image and Video Compression Standards: Algorithms and Architectures.* Kulwer Academic Publishers, Boston, 1996.

[7] L. G. Brown. A survey of image registration techniques. *Computing Surveys*, 24(4):325–376, 1992.

[8] L. Bui, M. Franklin, C. Taylor, and G. Neilson. Autonomous landing guidance system validation. In J. G. Verly, editor, *Enhanced and Synthetic Vision 1997*, pages 19–25. Proceedings of SPIE, volume 3088, 1997.

[9] M. A. Burgess, T. Chang, D. E. Dunford, R. H. Hoh, W. F. Home, and R. F. Tucker. Synthetic vision technology demonstration: Executive summary. Technical Report DOT/FAA/RD-93/40, I, Research and Development Service, Washington, D.C., December 1993.

[10] M. A. Burgess, T. Chang, D. E. Dunford, R. H. Hoh, W. F. Home, R. F. Tucker, and J. A. Zak. Synthetic vision technology demonstration: Flight tests. Technical

Report DOT/FAA/RD-93/40, III, Research and Development Service, Washington, D.C., December 1993.

[11] P. J. Burt. The pyramid as a structure for efficient computation. In A. Rosenfeld, editor, *Multiresolution Image Processing and Analysis.* Springer-Verlag, New York, 1984.

[12] P. J. Burt. Smart sensing within a pyramid vision. *Proceedings of the IEEE*, 76(8):1006–1015, August 1988.

[13] P. J. Burt. A gradient pyramid basis for pattern-selective image fusion. In *1992 SID International Symposium, Boston, Digest of Technical Papers*, pages 467–470. Society for Information Display, Playa del Rey, CA, 1992.

[14] P. J. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, Com-31(4):532–540, 1983.

[15] P. J. Burt and E. H. Adelson. Merging images through pattern decomposition. In A. G. Tescher, editor, *Applications of Digital Image Processing VIII*, pages 173–181. Proceedings of SPIE, volume 575, 1985.

[16] P. J. Burt and R. J. Kolczynski. Enhanced image capture through fusion. In *Fourth International Conference on Computer Vision*, pages 173–182. IEEE Computer Society, 1993.

[17] P. Buser and M. Imbert. *Vision.* MIT Press, Cambridge, Massachusetts, 1992.

[18] P. Cheeseman, B. Kanefsky, R. Kraft, and J. Stutz R. Hanson. Super-resolved surface reconstruction from multiple images. Technical Report FIA-94-12, Artificial Intelligence Research Branch, NASA Ames Research Center, December 1994.

[19] L. Chiariglione. The development of an integrated audiovisual coding standard: MPEG. *Proceedings of the IEEE*, 83(2):151–157, February 1995.

[20] J. J. Clark and A. L. Yuille. *Data Fusion for Sensory Information Processing Systems.* Kluwer, Boston, 1990.

[21] *CIE Recommendations on uniform color spaces, color differenve equations, psychometric color terms. Supplement No. 2 to CIE publication No. 15 (E-1.3.1) 1971/(TC-1.3.).* International Commission on Illumination, 1978.

[22] G. Duane. Pixel-level sensor fusion for improved object recognition. In C. B. Weaver, editor, *Sensor Fusion*, pages 180–185. Proceedings of SPIE, volume 931, 1988.

[23] T. Fechner and G. Godlewski. Optimal fusion of TV and infrared images using artificial neural networks. In S. K. Rogers and D. W. Ruck, editors, *Applications and Science of Artificial Neural Networks*, pages 919–925. Proceedings of SPIE, volume 2492, 1995.

[24] M. R. Franklin. Application of an autonomous landing guidance system for civil and military aircraft. In J. G. Verly, editor, *Synthetic Vision for Vehicle Guidance and Control*, pages 146–153. Proceedings of SPIE, volume 2463, 1995.

[25] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, San Diego, CA, 1990.

[26] R. C. Gonzalez and R. E. Woods. *Digital Image Processing.* Addison Wesley, Reading, MA, 1992.

[27] A. Goshtasby. Image registration by local approximation methods. *Image and Vision Computing*, 6(4):255–261, 1988.

[28] D. L. Hall. *Mathematical Techniques in Multisensor Data Fusion.* Artech House, Norwood, MA, 1992.

[29] H. H. Harman. *Modern Factor Analysis.* The University of Chicago Press, Chicago, 1976.

[30] Y. Hel-Or. Studies in gradient-based registration. Technical report, NASA Ames Research Center, 1994.

[31] B. K. P. Horn. *Robot Vision.* MIT Press, Cambridge, MA, 1986.

[32] B. K. P. Horn and B. G. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[33] S. A. Hovanessian. *Introduction to Sensor Systems.* Artech House, Norwood, MA, 1988.

[34] B. H. Hudson, M. J. Gary, M. A. Burgess, and J. A. Zak. Synthetic vision technology demonstration: Sensor tower testing. Technical Report DOT/FAA/RD-93/40, II, Research and Development Service, Washington, D.C., December 1993.

[35] M. Irani and S. Peleg. Improving resolution by image registration. *Computer Vision, Graphics and Image Processing*, 53:231–239, May 1991.

[36] K. Jack. *Video Demystified: A Handbook for the Digital Engineer.* HighText Publications Inc., California, 1993.

[37] K. G. Jöreskog. Some contributions to maximum likelihood factor analysis. *Psychometrika*, 32:443–482, 1967.

[38] K. G. Jöreskog. Factor analysis by least-squares and maximum-likelihood methods. In K. Enslein, A. Ralston, and H. S. Wilf, editors, *Statical Methods for Digital Computers, Volume III of Mathematical Methods for Digital Computers*, pages 125–153. John Wiley and Sons, New York, 1977.

[39] J. R. Kerr, D. P. Pond, and S. Inman. Infrared-optical multisensor for autonomous landing guidance. In J. G. Verly, editor, *Synthetic Vision for Vehicle Guidance and Control*, pages 38–45. Proceedings of SPIE, volume 2463, 1995.

[40] L. A. Klein. *Sensor and Data Fusion Concepts and Applications.* SPIE, 1993.

[41] D. Kundur, D. Hatzinakos, and H. Leung. A novel approach to multispectral blind image fusion. In B. V. Dasarathy, editor, *Sensor Fusion: Architectures, Algorithms, and Applications*, pages 83–93. Proceedings of SPIE, volume 3067, 1997.

[42] F. W. Leberl. *Radargrammetric Image Processing.* Artech House, Norwood, MA, 1990.

[43] H. Li, B. S. Manjunath, and S. K. Mitra. Multisensor image fusion using the wavelet transform. *Graphical Models and Image Processing*, 57(3):235–245, May 1995.

[44] H. Li and Y. Zhou. Automatic visual/IR image registration. *Optical Engineering*, 35(2):391–400, 1996.

[45] J. Lubin. A visual discrimination model for imaging system evaluation and design. Technical report, David Sarnoff Research Center, 1995.

[46] B. D. Lucas and T. Kanade. An iterative image registration technique with an application in stereo vision. In *Seventh International Joint Conference on Artificial Intelligence*, pages 674–679, Los Altos, CA, 1981. William Kaufmann Inc.

[47] S. Mann and R. W. Picard. On being undigital with digital cameras: extending dynamic range by combining differently exposed pictures. Technical Report TR-323, Massachussetts Institute of Technology, Media Laboratory, Perceptual Computing Section, 1995.

[48] H. H. Nagel. On the estimation of optical flow. *Artificial Intelligence*, 33(3):299–324, November 1987.

[49] A. N. Netravali and B. G. Haskell. *Digital Pictures Representation, Compression, and Standards.* Plenum Press, New York, 1995.

[50] M. Pavel and A. Ahumada. Model-based optimization of display systems. In M. Helander, T. K. Landauer, and P. V. Prabhu, editors, *Handbook of Human Computer Interaction*, pages 65–85. North Holland, 1997.

[51] M. Pavel, J. Larimer, and A. Ahumada. Sensor fusion for synthetic vision. In *1992 SID International Symposium, Boston, Digest of Technical Papers*, pages 475–478. Society for Information Display, Playa del Rey, CA, 1992.

[52] W. K. Pratt. *Digital Image Processing.* Wiley, New York, 1991.

[53] S. Ranganath. Image filtering using multiresolution representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:426–440, May 1991.

[54] P. J. Ready and P. A. Wintz. Information extraction, SNR improvement, and data compression in multispectral imagery. *IEEE Transactions on Communications*, COM-21:1123–1131, 1973.

[55] B. Roberts and P. Symosek. Image processing for flight crew situation awareness. In J. G. Verly and S. Welch, editors, *Sensing, Imaging, and Vision for Control and Guidance of Aerospace Vehicles*, pages 246–255. Proceedings of SPIE, volume 2220, 1994.

[56] M. C. Roggemann, J. P. Mills, M. Kabrisky, S. K. Rogers, and J. A. Tatman. An approach to multiple sensor target detection. In C. B. Weaver, editor, *Sensor Fusion II*, pages 42–52. Proceedings of SPIE, volume 1100, 1989.

[57] M. S. Sanders and E. J. McCormick. *Human Factors in Engineering and Design.* McGraw-Hill, New York, 1993.

[58] R. A. Schowengerdt. *Techniques for Image Processing and Classification in Remote Sensing.* Academic Press, New York, 1983.

[59] R. R. Schultz and R. L. Stevenson. Extraction of high-resolution frames from video sequences. *IEEE Transactions on Image Processing*, 5(6):996–1011, June 1996.

[60] I. K. Sethi and R. Jain. Finding trajectories of feature points in a monocular image sequence. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 9(1):56–73, January 1987.

[61] M. Shoucri, G. S. Dow, S. Fornaca, B. Hauss, L. Yujiri, J. Shannon, and L. Summers. Passive millimeter wave camera for enhanced vision systems. In J. G. Verly, editor, *Enhanced and Synthetic Vision 1996*, pages 2–8. Proceedings of SPIE, volume 2736, 1996.

[62] V. C. Smith and J. Pokorny. Spectral sensitivity of color-blind observers and the cone photopigments. *Vision Research*, 12:2059–2071, 1972.

[63] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision.* Chapman and Hall, London, 1993.

[64] B. T. Sweet and C. Tiana. Image processing and fusion for landing guidance. In J. G. Verly, editor, *Enhanced and Synthetic Vision 1996*, pages 84–95. Proceedings of SPIE, volume 2736, 1996.

[65] A. M. Tekalp. *Digital Video Processing.* Prentice-Hall, Upper Saddle River, NJ, 1995.

[66] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analysers. Technical report, NCRG/97/003, Neural Computing Research Group, Aston University, UK, 1997.

[67] M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. Technical report, NCRG/97/010, Neural Computing Research Group, Aston University, UK, 1997.

[68] A. Toet. Hierarchical image fusion. *Machine Vision and Applications*, 3:1–11, 1990.

[69] A. Toet. Multi-scale image fusion. In *1992 SID International Symposium, Boston, Digest of Technical Papers*, pages 471–474. Society for Information Display, Playa del Rey, CA, 1992.

[70] A. Toet, L. J. van Ruyven, and J. M. Valeton. Merging thermal and visual images by a contrast pyramid. *Optical Engineering*, 28(7):789–792, July 1989.

[71] A. Toet and J. Walraven. New false color mapping for image fusion. *Optical Engineering*, 35(3):650–658, 1996.

[72] J. G. Verly. Enhanced and synthetic vision. In *Enhanced and Synthetic Vision 1996*, pages ix–x. Proceedings of SPIE, volume 2736, 1996.

[73] B. A. Wandell. *Foundations of Vision.* Sinauer Associates, Sunderland, MA, 1995.

[74] Y. Wang, M. T. Freedman, J. H. Xuan, Q. Zheng, and S. K. Mun. Multimodality medical image fusion: probabilistic quantification, segmentation, and registration. In Y. Kim and S. K. Mun, editors, *Medical Imaging 1998: Image Display*, pages 239–249. Proceedings of SPIE, volume 3335, 1998.

[75] A. B. Watson. Detection and recognition of simple spatial forms. Technical report, NASA Technical Memorandum 84353, NASA Ames Research Center, Moffett Field, CA, April 1983.

[76] A. M. Waxman, D. A. Fay, A. Gove, M. Seibert, J. P. Racamato, J. E. Carrick, and E. D. Savoye. Color night vision: Fusion of intensified visible and thermal imagery. In J. G. Verly, editor, *Synthetic Vision for Vehicle Guidance and Control*, pages 58–68. Proceedings of SPIE, volume 2463, 1996.

[77] A. M. Waxman, A. N. Gove, D. A. Fay, J. P. Racamato, J. E. Carrick, M. C. Seibert, and E. D. Savoye. Color night vision: opponent processing in the fusion of visible and IR imagery. *Neural Networks*, 10(1):1–6, 1997.

[78] T. A. Wilson, S. K. Rogers, and L. R. Myers. Perceptual-based hyperspectral image fusion using multiresolution analysis. *Optical Engineering*, 34(11):3154–3164, November 1995.

[79] G. Wolberg. *Digital Image Warping.* IEEE Computer Society Press, Los Alamitos, CA, 1992.

[80] Q. Zheng and R. Chellappa. A computational vision approach to image registration. In *Proceedings of the International Conference. on Pattern Recognition*, pages 193–197. IEEE Computer Society Press, Los Alamitos, CA, 1992.

Appendix A

Conformal Geometric Representation of Multisensor Imagery

A.1 Introduction

The discussion on fusion in Chapters 2, 3, 4 and 5 was based on the assumption that the images from multiple sensors have identical geometric representations. In reality, this assumption is not valid. Visible-band and infrared sensors, and imaging radars operate in different regions of the electromagnetic spectrum. The physics underlying the mechanism of each of these sensors is different, and dictates the imaging geometry of the sensor. As a result different sensors generate images that are different geometric representations of a scene. The differences are particularly apparent between radar and other imaging sensors.

Visible-band and infrared sensors generate imagery that is conformal with — or very similar to — a view of a scene as seen through a window, i.e., a projective transformation. Figure A.1(a) shows a simulated forward looking infrared (FLIR) image. The imaging geometry of a radar is quite different. An imaging radar transmits a narrow radio beam periodically swept over a range of angles along the azimuth. Objects in the scene reflect different proportions of the impinging radio energy. The radar output (B-scope) image shows the reflected energy as a function of azimuth and range[1], i.e., distance from the radar antenna. Figure A.1(b) shows a simulated millimeter wave radar (MMWR) image.

The heterogeneity of the image representations increases the complexity of a system

[1] The exact relationship between the scene and the radar image depends on the details of the radar technology. A complete characterization of this relationship requires consideration of the radar equation, multiple reflections, polarization effects, and details of the radar design [42].

that is designed to benefit from the diversity of the sensors. The range images generated by an imaging radar cannot be directly fused with visible-band and infrared images. A pragmatic approach to fuse images from different types of sensors involves:

1. Transformation of image data to a conformal (and preferably invertible) geometric representation.
2. Fusion of images in the domain of the conformal representation.
3. Transformation of the fused images to another representation for display, if required.

An example of such an approach involves transformation of the radar image to a projective transformation, typically called the C-scope [10]. This process is called radar image rectification. This is followed by fusion in the C-scope image.

In this appendix we describe the projective and range imaging geometries and review the process of radar image rectification in Section A.2. We then review several conformal representations and discuss their advantages and drawbacks in Section A.3. In Section A.4 we describe a new conformal representation called M-scope and demonstrate its application to fusion of radar and infrared images.

A.2 Rectification of Radar Images

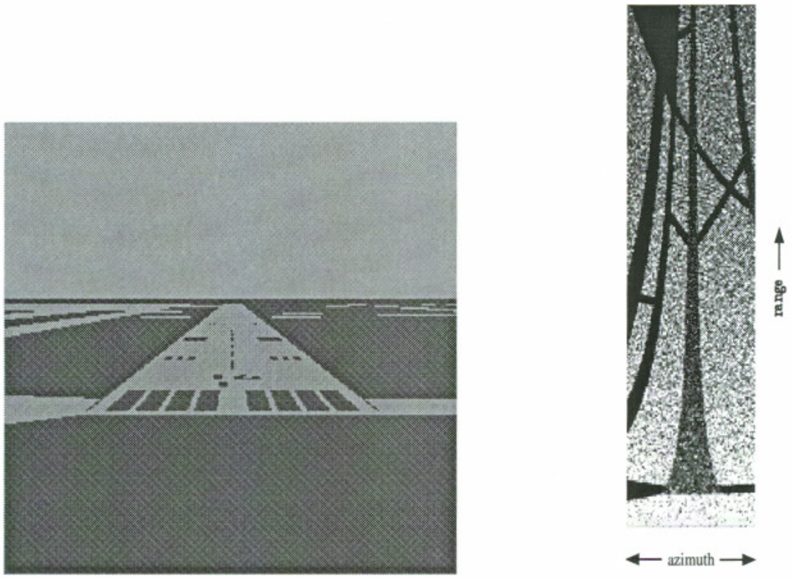
Radar image rectification consists of two steps:

1. Reconstruction of the 3-D scene geometry from a 2-D range image representation.
2. Transformation of the 3-D coordinates into a 2-D projective image representation.

Before reviewing the rectification process, we briefly describe the projective and range imaging geometries.

A.2.1 Projective imaging geometry

The projective transformation projects 3-D points in the scene onto a 2-D image plane [26]. The projective imaging geometry is illustrated in Figure A.2. The image plane is parallel



(a) FLIR projective image

(b) MMWR range image

Figure A.1: Different geometric representations

to the $X-Y$ plane and the focal point of the camera lens is at $(0, H, 0)$. The relationships between the 3-D and 2-D coordinates are given by the projection equations,

$$x = f\frac{X}{Z}, \tag{A.1}$$

and

$$y = y_0 - f\frac{H-Y}{Z}, \tag{A.2}$$

where (X, Y, Z) represents the 3-D location of an object, (x, y) are the 2-D image plane coordinates, y_0 is the location of the horizon, and the constant f is the focal length of the optics used for the projection.

A.2.2 Range imaging geometry

A pixel $s(r, \theta)$ in the radar B-scope image in Figure A.1(b) represents the intensity s of the reflected radio wave at a range given by its ordinate r (vertical coordinate), and the

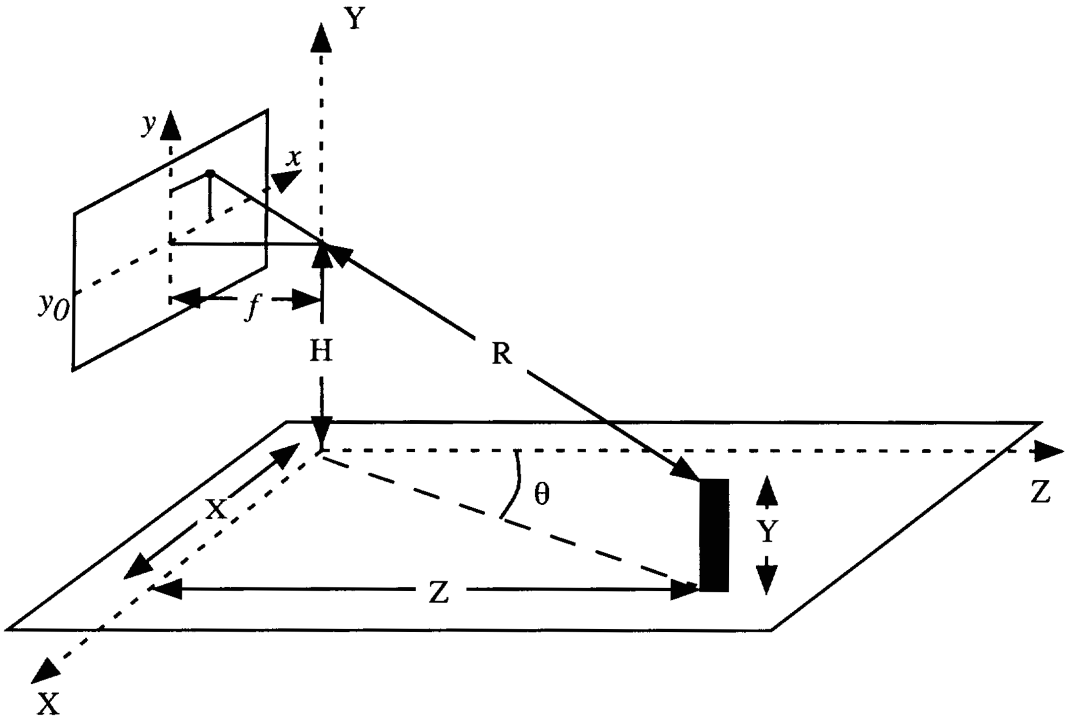


Figure A.2: Projective imaging geometry

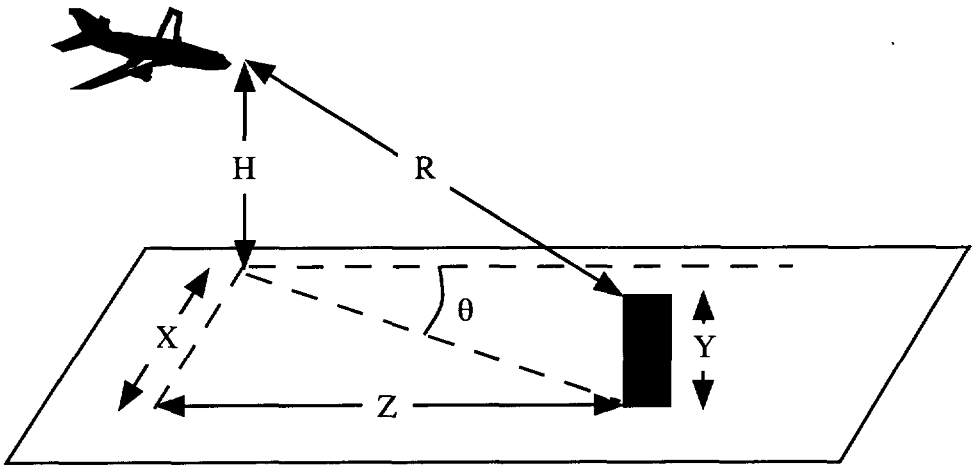


Figure A.3: Radar imaging geometry

azimuth given by the abscissa θ (horizontal coordinate). The actual world range R is related to the B-scope coordinate by a scaling constant k, $r = kR$.

For the range geometry illustrated in Figure A.3, in which the radar is located at $(0, H, 0)$, the B-scope coordinates are related to the world coordinates by,

$$\tan\theta = \frac{X}{Z} \ , \tag{A.3}$$

$$R^2 = X^2 + (H - Y)^2 + Z^2 \ , \tag{A.4}$$

where (X, Y, Z) represents the 3-D location of a reflecting object, and H is the altitude of the platform that contains the imaging radar.

The 3-D scene coordinates (X, Y, Z) are reconstructed from (r, θ) using Equations (A.3) and (A.4). The projective coordinates (x, y) are then obtained from (X, Y, Z) using Equations (A.1) and (A.2). However, reconstruction of (X, Y, Z) from (r, θ) is underdetermined. There are three unknowns X, Y, Z and only two equations, Equations (A.3) and (A.4). It is necessary to introduce additional constraints to obtain a solution.

A.2.3 Radar rectification based on flat earth assumption

The traditional approach to rectification is to assume that all the reflected radar energy is from objects located on a plane. This assumption specifies a constraint which imposes

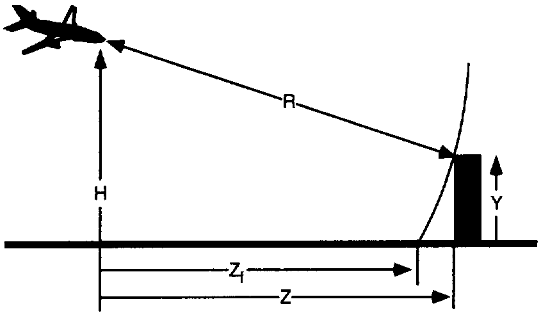


Figure A.4: Error due to flat earth assumption

$Y = 0$ or *flat earth* for all objects in the scene. Under this flat earth assumption, the horizontal image plane coordinate x is determined by the azimuth coordinate θ in the radar range image. From Equations (A.3) and (A.1),

$$x = f \tan\theta \tag{A.5}$$

To compute the vertical image plane coordinate y it is necessary to compute the distance Z of each point. An estimate of Z, denoted by Z_f, can be obtained by combining Equation (A.4) with Equation (A.1) and using the flat earth assumption $Y = 0$,

$$Z_f = \sqrt{\frac{R^2 - H^2}{1 + \tan^2(\theta)}} = f\sqrt{\frac{R^2 - H^2}{f^2 + x^2}}. \tag{A.6}$$

Using Z_f, the 2-D vertical coordinate y can be computed from Equation (A.2) as,

$$y = y_0 - H\sqrt{\frac{f^2 + x^2}{R^2 - H^2}}. \tag{A.7}$$

The flat earth assumption, results in errors for objects that are not on the plane $Y = 0$. For example, an object located at (X, Y, Z), with $Y > 0$ is incorrectly localized at Z_f as shown in Figure A.4. The estimate Z_f is smaller (i.e., closer to the radar) than the actual value Z. The error in Z is given by

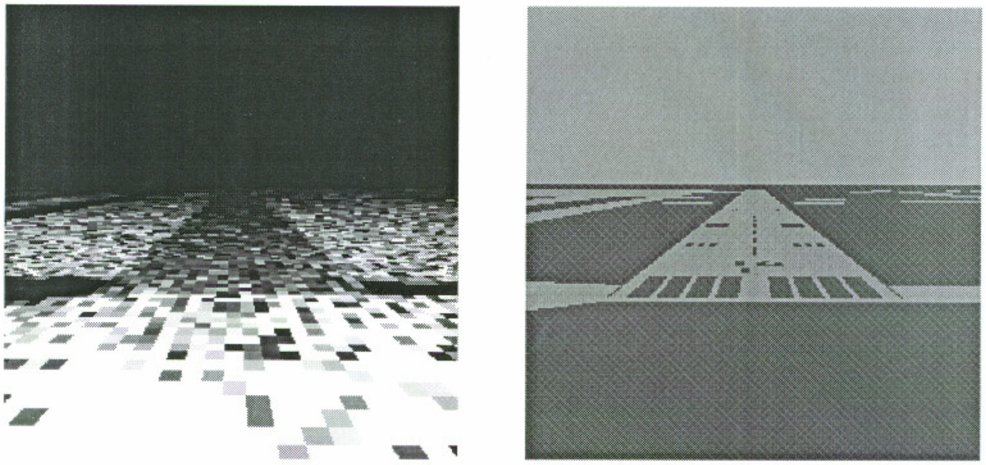
$$\Delta Z = Z - Z_f = Z\left[1 - \sqrt{1 - \frac{2YH - Y^2}{Z^2(1 + \tan^2\theta)}}\right] , \tag{A.8}$$

and the corresponding error in the image plane vertical axis is given by

$$\Delta y = f\left[\frac{H - Y}{Z} - \frac{H}{Z_f}\right] . \tag{A.9}$$

In applications such as ALG[2], the imaging radar is located on an aircraft. The flat earth assumption is therefore a reasonable approximation if the aircraft elevation is much larger than the height of objects in the scene, $H >> Y$. However, the error increases for objects that are either near (small Z), or are at the same elevation as the radar as in the final stages of a landing.

A.3 Review of Conformal Geometric Representations



(a) Radar C-scope

(b) FLIR C-scope

Figure A.5: C-scope representation

In the previous section we discussed rectification of radar images and described the conversion of a radar range image in B-scope representation to a C-scope representation using the flat earth assumption. In this section, we discuss the appropriateness of the C-scope, B-scope and plan position indicator representations for fusing radar (MMWR) images with infrared (FLIR) and visible-band images.

[2] The ALG application is discussed in Chapter 1.

A.3.1 C-scope representation

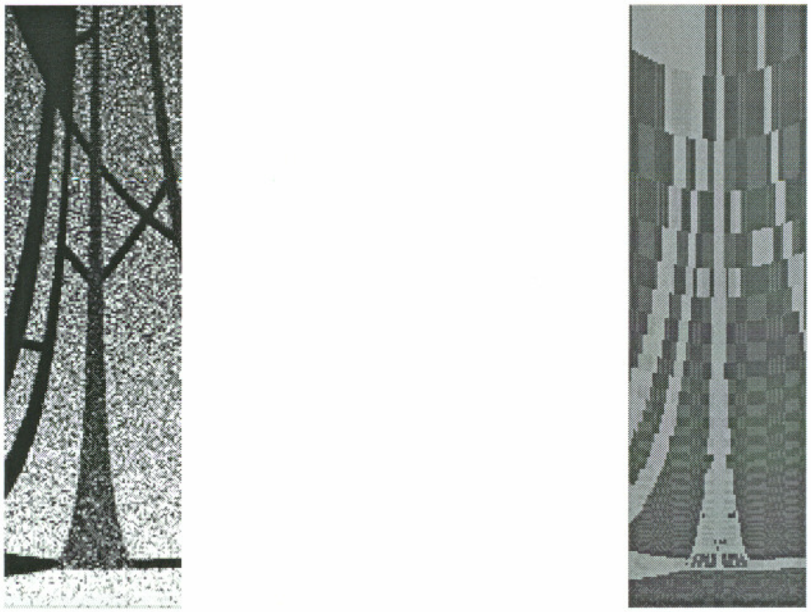
One approach to fuse a radar image with an infrared (or visible-band) image is to use the C-scope as the conformal representation. The advantage of this approach is that only the radar range image needs to be converted to C-scope (as described in Section A.2). Subsequent steps of fusion and display of the fused image are then performed in C-scope itself. Figure A.5 shows conformal C-scope representations of a simulated radar range image and a simulated infrared image from the ALG application. Figure A.5(a) is the radar range image of Figure A.1(b) after conversion to C-scope.

A drawback of this approach, however, is that useful information about distant structures in the B-scope image is lost during conversion from B-scope to C-scope. This is because the resolution[3] of a radar range image in B-scope and a projective image in C-scope is different as a function of 3-D world distance from the sensors along the Z coordinate. For an imaging radar, the resolution in range is approximately constant and nearly independent of the gaze angle. For a projective image, the resolution decreases with distance and is inversely proportional to the distance along the Z axis. At some distance along the Z axis (away from the sensing platform), the resolution of the radar becomes higher than that of the infrared (or visible-band) sensor. During conversion from B-scope to C-scope, the higher resolution data about distant structures is converted into lower resolution data. Another drawback of converting a radar image to C-scope is that objects close to the radar are significantly aliased as in the lower portion of Figure A.5(a).

A.3.2 B-Scope representation

An alternative to the above approach is to convert the projective image to B-scope representation. To convert to B-scope we use the flat earth assumption to obtain the range coordinates (R, θ) from the projective coordinates (x, y) using Equations (A.5) and (A.7). Figure A.6 shows the B-scope representation of the FLIR image of Figure A.1(a). The drawbacks of this approach are complementary to those of the C-scope representation and can be observed in Figures A.6(a) and A.6(b). The fine details of Figure A.5(b) in near

[3] Here, *resolution* refers to the area of the scene covered by one pixel.



(a) Radar B-scope

(b) FLIR B-scope

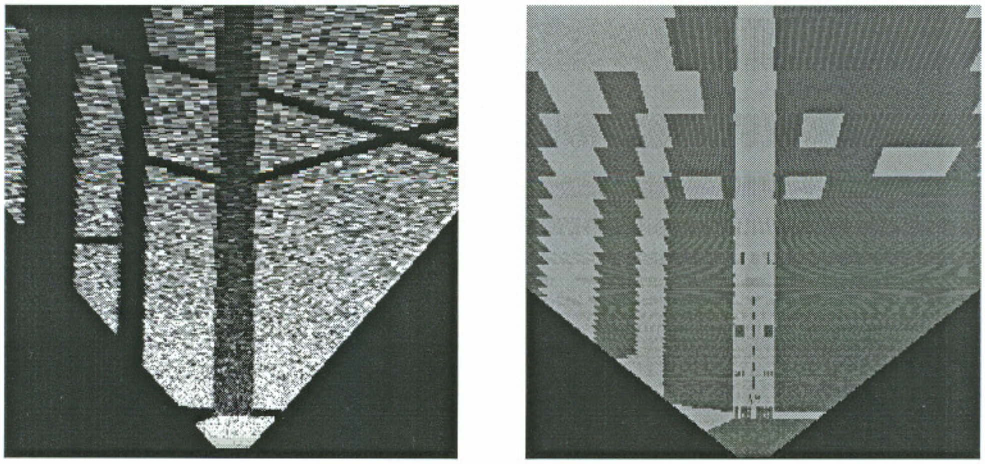
Figure A.6: B-scope representation

locations (locations in the image that are closer in distance to the imaging sensor) in the horizontal (i.e., X) direction are lost.

A.3.3 Plan position indicator representation

The plan position indicator (PPI) representation [42] represents a 3-D scene by an orthogonal projection onto a plane parallel to the earth's surface. The PPI view is a bird's eye view, i.e., an image of the earth from a viewpoint located high above the surface of the earth. The PPI image represents the scene as a function of the world coordinates X and Z (the 3-D scene geometry is illustrated in Figure A.2).

Figure A.7 shows the simulated FLIR and MMWR images of Figure A.1 converted to PPI. The advantage of this approach is that the losses in resolution due to differences in sampling variabilities are equalized since images from all the sensors have to be transformed to PPI for fusion. The drawback of this approach is that some information from both sensors is lost. The loss is particularly significant for short distances. Besides, this



(a) Radar PPI

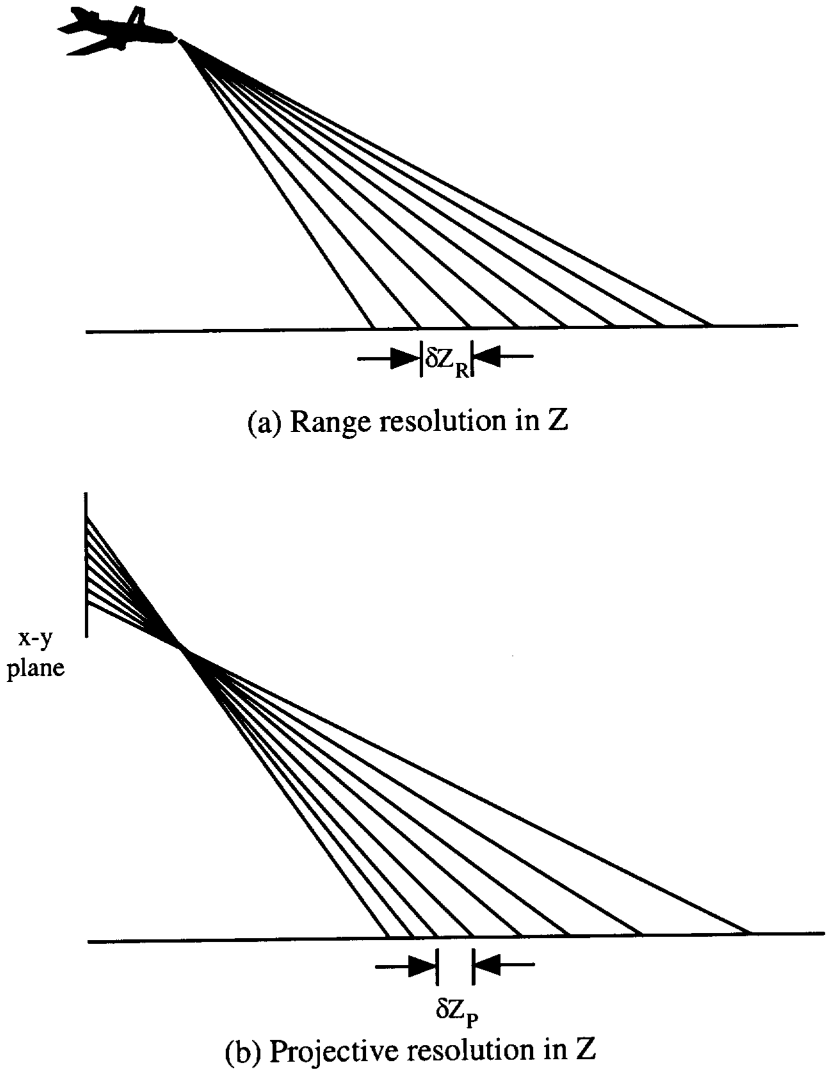
(b) FLIR PPI

Figure A.7: PPI representation of simulated radar and FLIR images

approach is computationally expensive compared to the above approaches because images from all sensors have to be converted to PPI.

A.4 M-Scope Representation

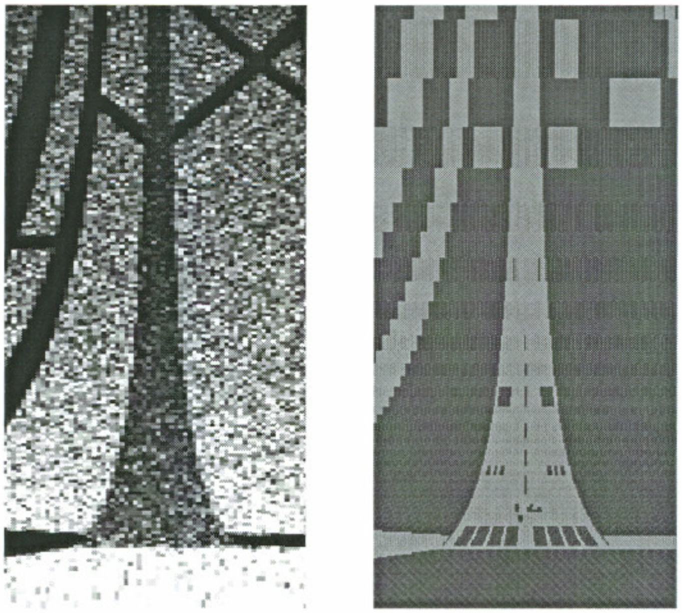
We have developed a conformal mixed scope (M-scope) representation that takes advantage of the differences in resolution to retain the best resolution of each of the sensors. The M-scope is similar to the B-scope in that it represents the distance along the Z coordinate versus azimuth angle. However, the Z dimension is represented by a nonlinear scale. This nonlinear scaling reduces the effects of the differences in resolution between the sensors, which are more pronounced along the Z direction. The nonlinear scale is defined such that the step size in Z at any location in the M-scope image is dictated by the sensor which has the highest resolution along Z at that location. Now consider a range image and a projective image to be converted to M-scope. For a vertical increment of one pixel in the range image, let the corresponding increment in Z be δZ_R as shown in Figure A.8. Similarly, for a vertical increment of one pixel in the projective image, let the corresponding increment in the Z direction be δZ_P. Then the Z increment in the M-scope image δZ_M



(a) Range resolution in Z

(b) Projective resolution in Z

Figure A.8: Z increment for M-scope representation



(a) Radar M-scope (b) FLIR M-scope

Figure A.9: M-scope representation

is given by

$$\delta Z_M = \begin{cases} \delta Z_P & \text{if} \qquad \delta Z_P < \delta Z_R \\ \delta Z_R & \text{otherwise} \end{cases} \tag{A.10}$$

For small distances along the Z direction (near locations), the increment in Z in the M-scope image is dictated by the resolution of the projective image. For large distances (far locations) the Z increment in M-scope is governed by the radar range resolution.

Figure A.9 shows the M-scope representations of the simulated FLIR and MMWR images of Figure A.1. Comparing these images with the corresponding C-scope and B-scope images in Figures A.5 and A.6, one can observe that the M-scope images resemble the C-scope representation at the near locations whereas they resemble the B-scope images at the far locations. The M-scope resembles the B-scope with enlarged near features.

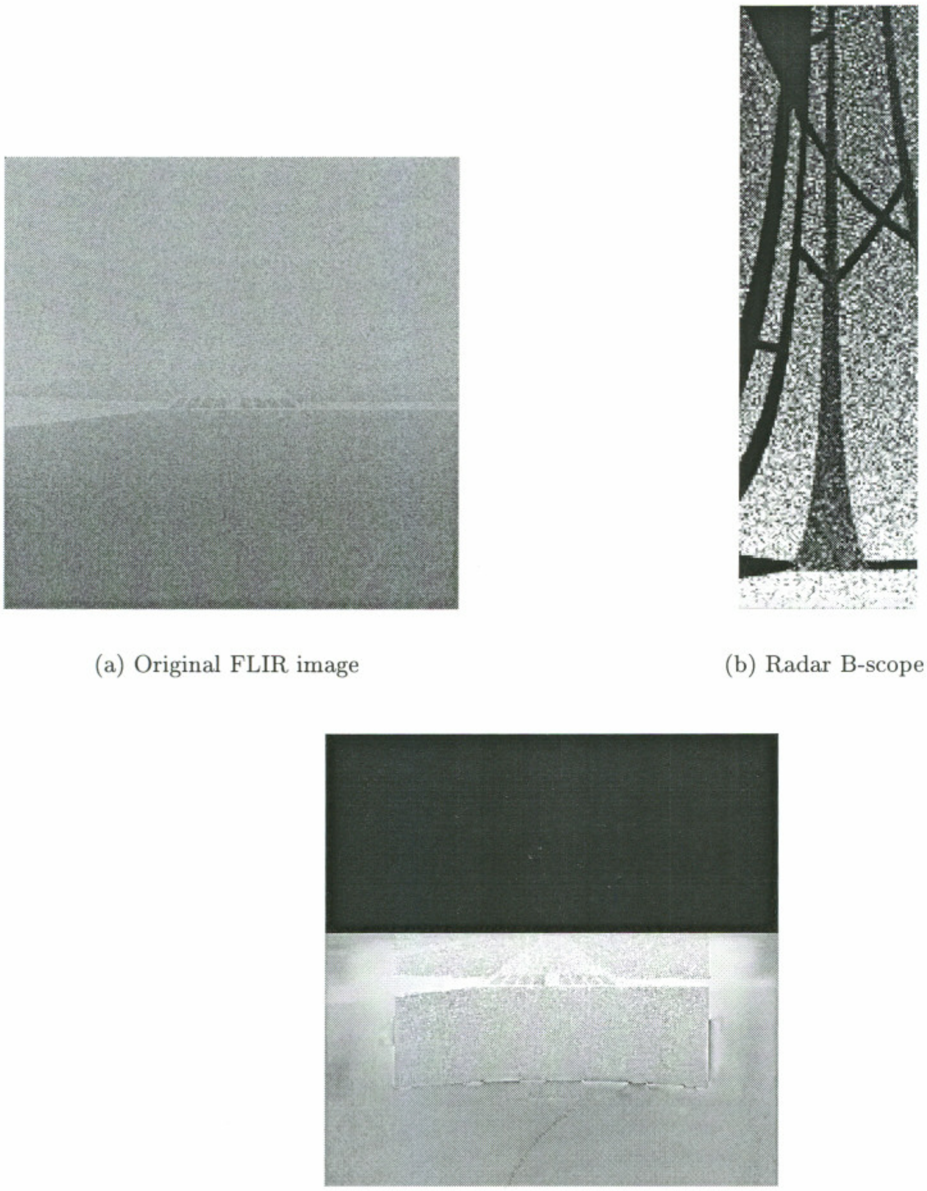
The advantage of M-scope is that the loss of information in the conversion process is reduced. The M-scope representation retains the image information present in the original representation since it preserves the best sensor resolution at each location. The drawback of this approach is that all the sensor images must be transformed into this representation. This extra computation can be justified for applications where it is crucial to retain information from the sensors before further processing. For example consider an application where images from MMWR and FLIR are fused for target recognition. The M-scope provides an advantage over other representations because MMWR image data in far locations and the FLIR image data in near locations are better preserved in M-scope.

A.4.1 Fusion in the M-scope domain

Figure A.10 illustrates the use of the M-scope representation for fusion of simulated MMWR and FLIR images. The simulated FLIR image in Figure A.10(a) was generated by simulating atmospheric attenuation (to simulate the effects of fog) in the noiseless FLIR image and perturbing with additive white Gaussian noise. The simulated radar range image in Figure A.10(b) was obtained by generating a range image and perturbing it by multiplicative log-normally distributed white noise. These images were converted to M-scope representation and fused using the ML fusion rule in Section 3.4.1. The parameters of the image formation model were estimated as described in Section 4.4 using the FLIR image as a reference image. The resulting fused image is shown in Figure A.10(c). The fused image contains salient image features from both the images The near points of the scene are dominated by the FLIR image (runway markings), whereas the distant areas (edges of the runway) are from the radar image. The fused image does not contain image features above the horizon since they do not appear in the M-scope representation. In practice, the portion of the FLIR image above the horizon could be retained.

A.5 Summary

We reviewed the C-scope, B-scope and PPI conformal representations and described the new M-scope representation. The M-scope transformation preserves the best resolution of either of the sensor images that are to be converted to M-scope. The M-scope geometry reduces the loss of image information during conversion to and from M-scope, and is well-suited for fusion of MMWR imagery with imagery from FLIR or visible-band sensors.



(a) Original FLIR image

(b) Radar B-scope

(c) Fused image

Figure A.10: Fusion of radar and FLIR

Appendix B

Registration of Multisensor Images

B.1 Introduction

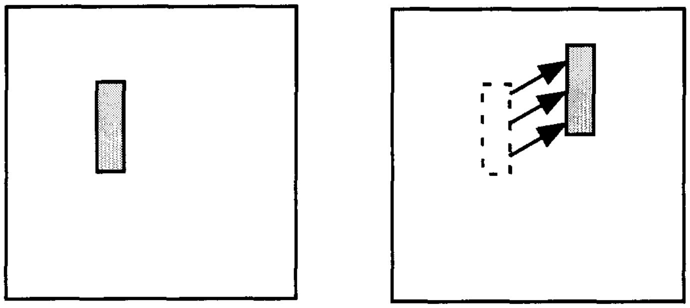
Image registration is an important component of multisensor image fusion. It is necessary to ensure that images to be fused are registered so that they can be compared and combined. The goal of registration is to establish a spatial correspondence between two images of the same scene[1] and determine a geometric transformation that aligns one image with another. Automatic registration of images obtained from multiple sensors is a difficult task because of the mismatch in the images caused by local polarity reversals and complementary features.

We describe traditional approaches to registration in Section B.2 and discuss why these approaches may not work when the images to be registered are from different types of sensors. In Section B.3 we describe modifications to the traditional approaches to apply in the case of different types of sensors. Finally, we demonstrate the results of registration using our technique.

B.2 Same-sensor Registration

Before describing our approach to multisensor image registration, we briefly discuss techniques for registration of images from the same sensor or from similar types of sensors. Although several different approaches exist for registration of same-sensor images [7], they

[1] A misalignment between image features in two images of the same scene can be caused by a variety of factors including different positions of sensors imaging the scene and movement of the imaging platform.



(a) Image 1 (b) Image 2

Figure B.1: Optical flow

all depend upon similarities between the images. To estimate the geometric differences between the images, one can either exploit similarities that exist between the graylevels in the two images or exploit similarities that exist between matching points in the two images [31]. These two approaches give rise to two broad categories in image registration — optical flow approaches and feature mapping approaches.

B.2.1 Optical flow approaches

Optical flow techniques [32, 46, 1, 30] take advantage of the similarity in graylevels between the two images to be registered. Optical flow is the apparent motion or spatial shift of brightness patterns (graylevels) between two images. For example, in Figure B.1 the arrows represent the motion of the shaded object from image 1 to image 2. These techniques assume that the observed brightness of objects in the images to be registered is constant. This is called the brightness constancy assumption. This assumption enables one to relate the motion to the shift of brightness patterns. In addition, a smoothness constraint is assumed to ensure that the motion of nearby points is similar.

Gradient based registration techniques

The gradient based techniques [5, 4, 30] estimate the optical flow field (i.e., the motion) between the images by relating the motion to the gradient of image brightness. Consider two images I_1 and I_2 that have to be registered. The motion between the images can be represented in two ways:

1. Associate a motion vector (p_x, p_y) with each pixel (x, y) in image I_1 so that it aligns with image I_2. This is a local motion model since a different motion vector is assigned for each pixel. In this case the motion description is not compact and usually the problem is underdetermined.

2. Formulate the motion as $(p_x, p_y) = f(x, y; \mathbf{p})$ where $\mathbf{p}$ is a parameter vector that is same for the entire image and the function f determines the structure of the motion at each pixel. This is a global parametric motion model. This model imposes a smoothness constraint since the motion of adjacent pixels is constrained due to f.

Some techniques also use a combination of local and global models. For example, block motion compensation techniques used in video compression algorithms [6, 19].

For planar scenes, it is advantageous to use the global parametric motion model [30]. If I_1 and I_2 are the two images to be registered, then $I_2(x, y)$ can be expressed in terms of $I_1(x, y)$ as

$$I_2(x, y) = I_1(x + p_x, y + p_y) \tag{B.1}$$

where $p_x(x, y, \mathbf{p})$ and $p_y(x, y, \mathbf{p})$ depend upon (x, y), the parameters $\mathbf{p}$ and the model used to describe the global motion. If the captured scene is assumed to be a planar surface, the motion of the scene can be approximated by a projective transformation [5], given by

$$\begin{aligned} p_x(x, y, \mathbf{p}) &= p_1 + p_3 x + p_5 y + p_7 x^2 + p_8 xy \\ p_y(x, y, \mathbf{p}) &= p_2 + p_4 x + p_6 y + p_7 xy + p_8 y^2 \ , \end{aligned} \tag{B.2}$$

where $\mathbf{p} \equiv [p_1, .., p_8]^{\mathrm{T}}$ are the parameters of the global motion model. When the distance between the scene and the camera is large, the motion can be approximated by an affine model, with $p_7 = p_8 = 0$ in Equation B.2. Translation is a special case of the projective model when $p_3 = p_6 = 1$ and $p_4 = p_5 = p_7 = p_8 = 0$ in Equation B.2.

The registration parameters are estimated by minimizing the sum squared error,

$$E = \sum_{x,y} (I_2(x, y) - I_1(x + p_x, y + p_y))^2 \tag{B.3}$$

If the displacements (p_x, p_y) are small, the above equation can be simplified by a Taylor series approximation of $I_1(x+p_x, y+p_y)$,

$$I_1(x+p_x, y+p_y) = I_1(x,y) + p_x I_{1x}(x,y) + p_y I_{1y}(x,y) \tag{B.4}$$

where

$$I_{1x} = \frac{\partial I_1(x,y)}{\partial x}, \tag{B.5}$$

$$I_{1y} = \frac{\partial I_1(x,y)}{\partial y} \tag{B.6}$$

The sum squared error can now be expressed as

$$E = \sum_{x,y} (\Delta I - p_x I_{1x} - p_y I_{1y})^2 \tag{B.7}$$

where,

$$\Delta I = I_2(x,y) - I_1(x,y) \tag{B.8}$$

The motion between the images is obtained by setting the derivatives of the error measure in Equation B.7 with respect to the parameters of the motion model to zero and solving the resulting set of equations. Such an estimation is accurate only when the displacements between the images are a fraction of a pixel, so that the Taylor series approximation holds. The estimation is improved by using an iterative alignment procedure that begins with an initial guess of the parameters. At each iteration, one of the images is warped[2] according to the initial estimates and then the estimation is repeated to obtain residual displacements.

To capture large displacements, a hierarchical coarse-to-fine refinement of the registration parameters is performed in a multi-resolution framework such as a Gaussian or Laplacian pyramid scheme [14]. The registration parameters are first iteratively estimated at a lower resolution to obtain a coarse solution. This solution is then used to initialize the parameter estimation at a higher resolution to obtain more accurate parameters. Bergen et al. [4, 5] and Hel-Or [30] provide detailed descriptions of gradient-based registration techniques.

[2] Warping consists of applying a geometric transformation followed by resampling [79].

The performance of gradient-based registration techniques is adequate only in the cases where the illumination conditions in the images to be registered are identical. In other words, the brightness constancy assumption should be a good approximation for the techniques to provide accurate registration.

B.2.2 Feature-mapping approaches

Feature-mapping techniques [44, 31, 80, 79], register images by finding a correspondence between points in the two images. Features such as edges, corners or contour lines in one image are matched to those in the other image. The locations of matching features in the two images are then used as control points. Registration is performed by finding a geometric mapping that aligns these control points. For example, a set of four matching points, (x_i, y_i) and $(x_i + p_{x_i}, y_i + p_{y_i})$ where $i = 0 \ldots 3$, in the images I_1 and I_2 provide a set of eight simultaneous equations in terms of the unknown parameters $p_1 \ldots p_8$. The equations are solved to obtain estimates of the parameters. The estimated parameters are used to warp and register the images.

The main difficulty in feature-mapping techniques is the search for consistent matching features in both the images. Consistency checking is essential to avoid the possibility of false matches. One also needs to ensure that the ambiguities that exist in matching points along edges do not result in incorrectly matched features.

B.3 Multisensor Registration

The registration techniques discussed above are suitable for registration of images from the same or similar sensors. For such images the brightness constancy is a reasonable assumption, and ambiguities between features in the images to be registered are less likely as long as the misalignment is small. However, registration of images from sensors such as visible-band and infrared pose additional problems. Different types of sensors capturing the same scene produce images that have quite different graylevels and features as described in Section 3.2. We now discuss the difficulties faced in multisensor registration and describe our approach to solve these problems.

B.3.1 Difficulties in registration caused by multiple sensors

Images captured through different types of sensors pose additional difficulties to traditional registration techniques. The graylevel characteristics of the images are often different because the sensors capture the ambient lighting conditions differently. For example, there are noticeable graylevel disparities between the visible-band and infrared images shown in Figure 3.1. Local polarity reversed features (Figure 3.1) cause further mis-match in graylevels. As a result, the brightness constancy assumption is not reasonable and the performance of optical flow techniques degrades.

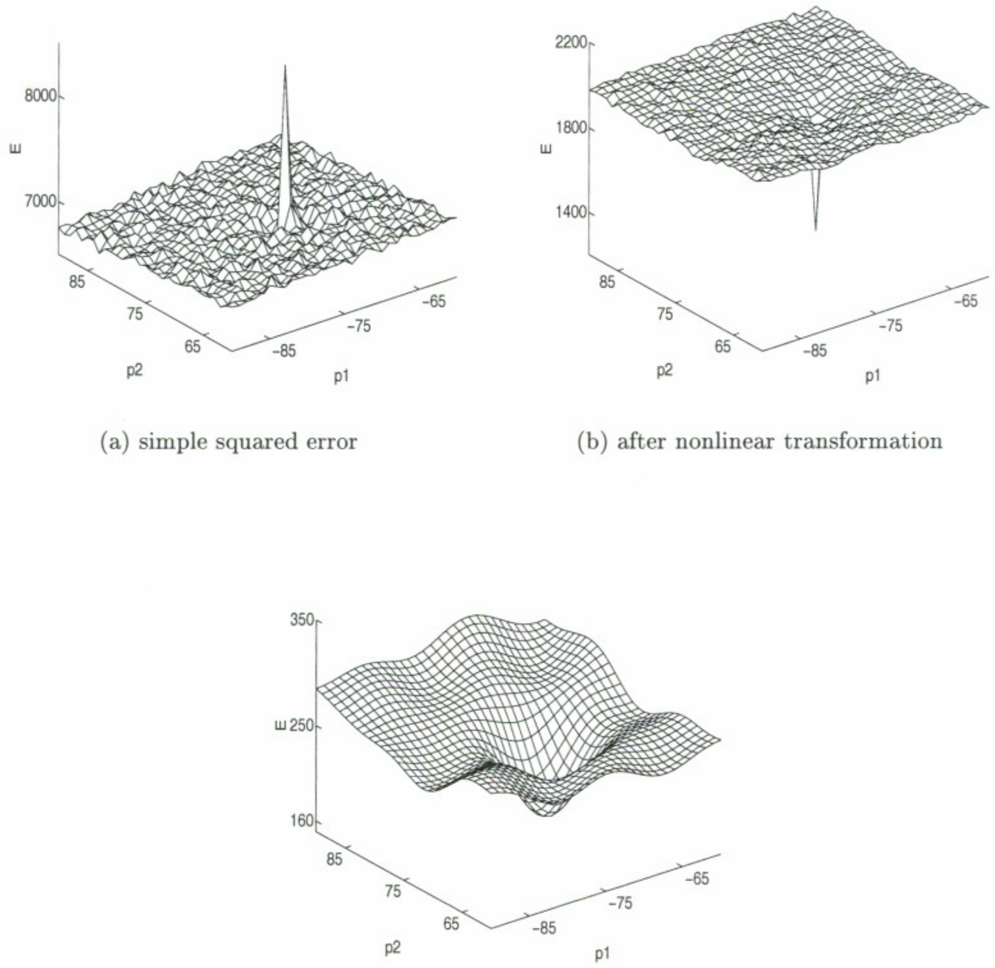
The presence of complementary features in multisensor images causes problems for feature-mapping techniques. Since features in one image may be absent in the other image (Figure 3.1), the search for corresponding matched features becomes difficult. There is a high possibility of obtaining inconsistently matched features which can impair registration.

B.3.2 Invariant representation for registration

As discussed above, the assumption of brightness constancy is not suitable for registration of multisensor images because the relationships between graylevels in multisensor images cannot be correctly described by optical flow. The gradient-based registration technique described in Section B.2.1 cannot be directly applied if I_1 and I_2 are multisensor images. For instance, for images containing graylevel disparities and local polarity reversals, the sum squared error of Equation (B.3) is not necessarily *minimum* even when the images are perfectly registered. Figure B.2(a) is a plot of the error (sum squared difference) as a function of translation for a pair of two misaligned images. The images contain local polarity reversed features and therefore, the translation corresponding to the correct registration gives the maximum error (shown by the peak of the error surface).

To address the problems caused by local polarity reversals and graylevel disparities, we transform the images into a representation that is invariant to local polarity reversals and insensitive to global changes in brightness. The particular transformation that we have chosen is the absolute value of hyperpixels in a Laplacian pyramid representation[3]. The

[3] The Laplacian pyramid representation is described in Section 2.4.1



(a) simple squared error

(b) after nonlinear transformation

(c) after smoothing

Figure B.2: Error surfaces for highest resolution of Laplacian pyramid.

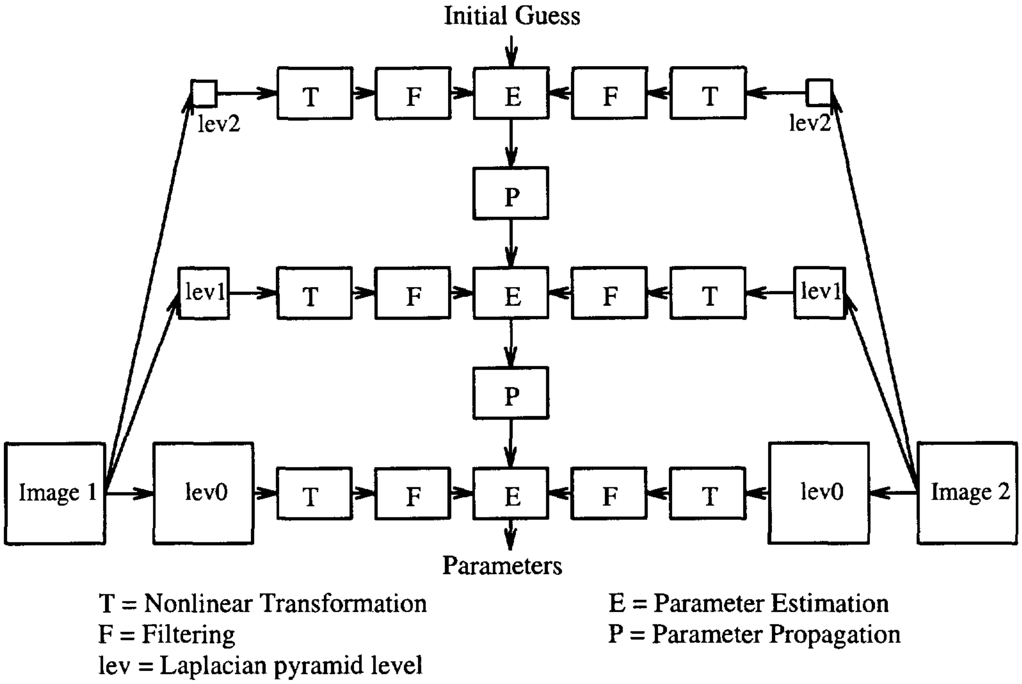


Figure B.3: Block diagram of registration process.

absolute value operation is a nonlinear operation on the Laplacian pyramid intensities. The resulting intensity representation is identical for signals which are exact opposites of each other. The nonlinear transformation (consisting of the Laplacian pyramid operations followed by the absolute value operation) results in invariant representations of local polarity reversed image features. For example, the nonlinear transformation produces identical representations of signals that are exact opposites of each other. An additional characteristic of this representation is that the levels of the Laplacian pyramid are bandpassed versions of the original image. Note that graylevel differences between the sensor images are accounted for by the low spatial frequencies in the images. The bandpass operation ensures that these low spatial frequencies are removed. After applying the nonlinear transformation, the error measure of Equation (B.3) is given by,

$$E = \sum_{x,y} (G'_2[I_2](x,y) - G'_1[I_1](x + p_x, y + p_y))^2 \ , \tag{B.9}$$

where G' denotes the nonlinear operation of taking the absolute value. Figure B.2(b) shows the error surface obtained using the nonlinear representation. The minimum point on the error surface now corresponds to the translation parameters that give the correct registration. The registration parameters, $p_1, .., p_8$, are estimated by minimizing the modified error measure in Equation (B.9).

B.3.3 Smoothing the error surface to reduce registration errors

Experiments using Equation (B.9) for estimating the registration parameters showed that the error measure does not always lead to accurate solutions and sometimes the parameter estimation does not converge. The accuracy and convergence of the minimization depends upon the error surface. Figure B.2(b) shows the error surface, computed using Equation (B.9), as a function of translation for a pair of multisensor images containing additive noise. The noise in the sensor images caused the error surface to be noisy, especially at higher resolutions. A noisy error surface can lead to erroneous registration because the minimization procedure may converge to a local minimum instead of the global minimum.

Smoothing of the error surface reduces local minima, thereby improving parameter estimation. We smoothed the error surface by applying a low-pass filter to the absolute of Laplacian pyramid representation. With smoothing, the error measure of Equation (B.9) becomes,

$$E = \sum_{x,y} (G_2[I_2](x,y) - G_1[I_1](x+p_x, y+p_y))^2 \ , \tag{B.10}$$

where G is an operator that denotes both the absolute value operation described above as well as the low-pass filtering operation. The low-pass filter operation removes the high spatial frequency noise and smears or spreads the features in the absolute of Laplacian pyramid. Figure B.2(c) shows the error surface after low-pass filtering the absolute of Laplacian pyramid. The error surface is smoother and the basin containing the global minimum is widened. A wider error surface leads to robust parameter estimation and problems caused by local minima are reduced. The trade-off is that the precision of the estimated registration parameters is reduced due to the wider basin in the smoothed error surface.

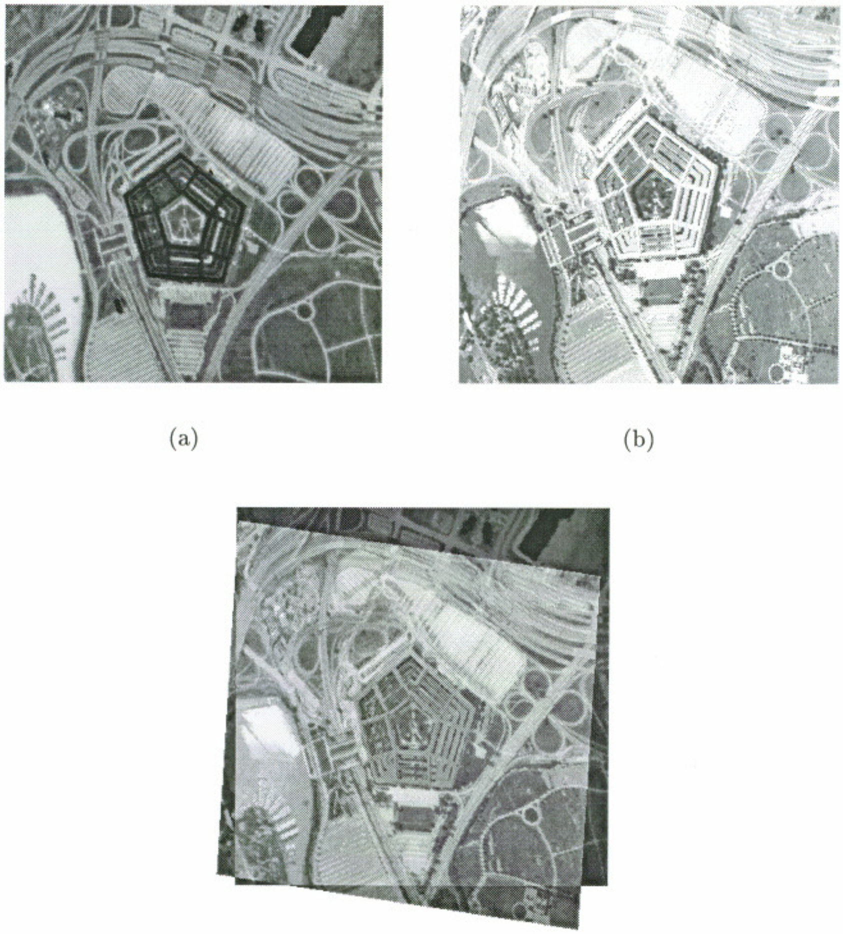
The frequency response of the low-pass filter used for filtering the levels of the absolute Laplacian pyramid affects the extent to which the error surface is smoothed. The lower the cut-off frequency of the low-pass filter, the smoother is the resulting error surface. The low-pass filter that we have used is the five tap Gaussian kernel given in Section 2.4.1. Although, this filter is not optimal for all images, we were able to register several sets of multisensor images with this filter as shown in Section B.4.

The steps in the modified gradient-based registration technique are illustrated in Figure B.3. The two images to be registered are transformed into levels of a Laplacian pyramid. The levels of the Laplacian pyramid are then subjected to the nonlinear transformation and passed through a low-pass filter as described above. The resulting representation is fed into the parameter estimation step. Registration proceeds hierarchically from lower resolutions (higher levels of the pyramid) to higher resolutions (lower levels of the pyramid). At each level, parameters are iteratively estimated by minimizing the error measure. Parameters estimated at a lower resolution are used as an initial guess at a higher resolution. The parameters are propagated from one level to another by a change in coordinates. For example to propagate parameters from a lower resolution to a higher resolution twice in size, the parameters p_1 and p_2 are multiplied by 2 whereas the parameters p_7 and p_8 are divided by 2. All the other parameters remain the same.

B.4 Experiments and Results

We have applied our registration technique to register several sets of multisensor images. We have used both simulated and real images from different types of sensors. Figures B.4(a) and B.4(b) are images of the Pentagon obtained from sensors operating in different portions of the infrared spectrum. Local polarity reversed features are present in the center and the lower left portion of these images. These images were registered using the affine model. Figure B.4(c) shows the superimposed images after registration. Note that the edges of structures and the roads in the two images are properly aligned.

Figures B.5(a) and B.5(b) are visible-band and FLIR images, respectively, and show a runway scene as an aircraft is about to land. These images contain local polarity reversed features (runway markings) as well as overall graylevel disparities. These images were registered using the projective model. Figure B.5(c) shows the two images superimposed


(a)

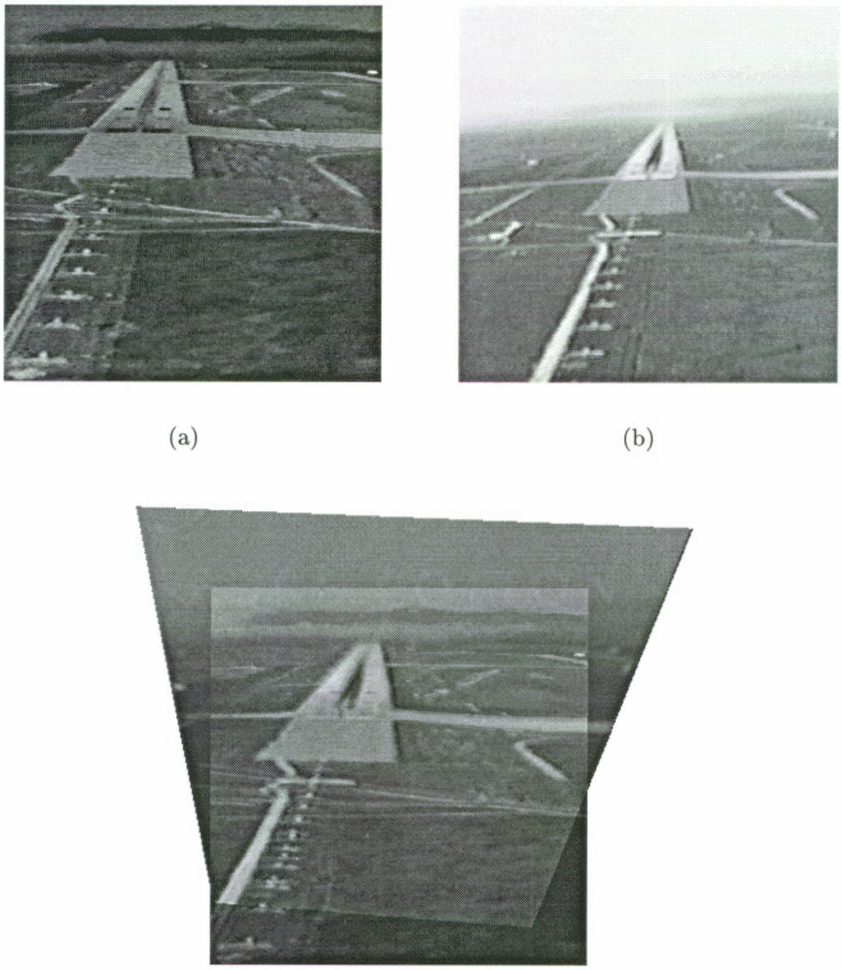
(b)

(c)

Figure B.4: Registration using affine model (Data from AMPS).

after registration. The edges of the runways and the taxiways are now aligned. These are the very images that we used to demonstrate our fusion algorithms in Chapter 5. We have observed that the residual misalignment towards the bottom of the images is difficult to remove. This misalignment may have been caused due to the high speed with which the aircraft (on which the sensors were placed) was approaching the runway. The high speed coupled with the different scanning rates of the sensors likely resulted in a nonlinear distortion between the images in regions closer to the sensors.

Figure B.6 shows the registration of FLIR and MMWR images. The FLIR image in Figure B.6(a) contains features which are absent in the MMWR image in Figure B.6(b).

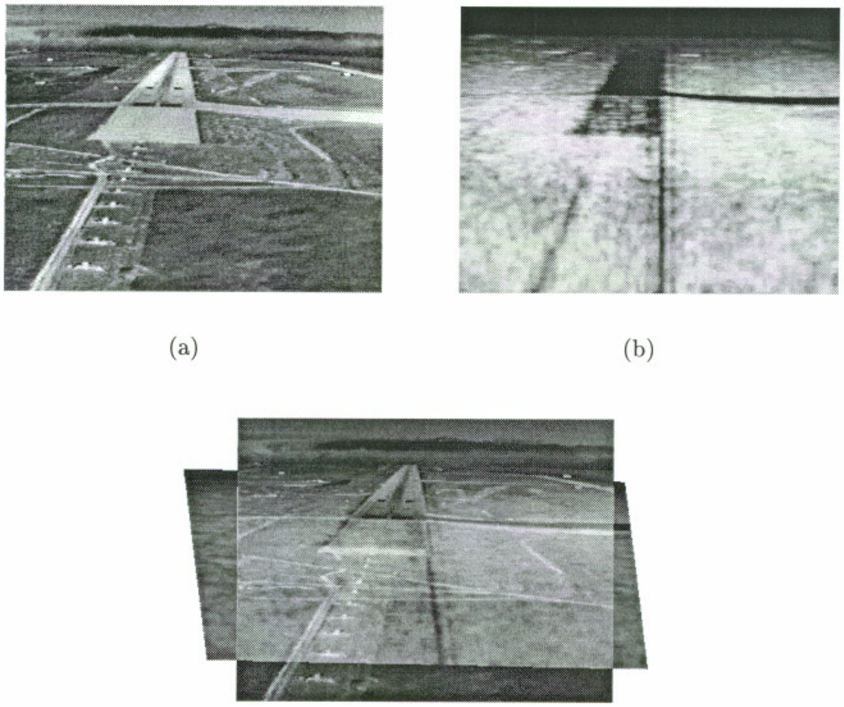


(a)

(b)

(c)

Figure B.5: Registration of FLIR and visible-band images (Data from SVTD [9]).



(a)

(b)

(c)

Figure B.6: Registration of FLIR and radar images (Data from SVTD [9]).

In spite of the differences caused by complementary features and local polarity reversals, our technique registered the images using the affine model as shown in Figure B.6(c).

A quantitative evaluation of registration is a difficult task, especially with multisensor images, unless the correct registration is known a priori. We evaluated our approach using simulated multisensor images where the correct registration is known. Our experiments indicate that the registration is accurate to within a pixel.

B.5 Discussion and Conclusions

We described an approach for registration of multisensor images. This approach circumvents the difficulties due to multiple sensors in traditional registration. We used the absolute of Laplacian pyramid representation to produce invariant image representations under local polarity reversals and graylevel disparities. The technique then modifies these representations by low-pass filtering them. This results in smoothing of the error surface, and facilitates robust estimation of registration parameters.

It is important to note that knowledge of local polarity reversed and complementary features can further aid the task of registration. Registration algorithms can incorporate the knowledge of location of such features to improve the registration. We performed some preliminary experiments based on this approach. After a first pass at registration, we estimated the location of complementary features using the probabilistic fusion rules of Chapter 3 on the registered images. We then used this knowledge to suppress the contribution to the error from these locations and performed a second pass at registration. Our experiments indicate that there is an improvement in registration accuracy especially for images in which complementary features are significant. This implies that fusion of the images prior to registration could facilitate improved registration. However, it is necessary to register the images before performing fusion. Therefore registration and fusion need to carried out in an iterative manner. Registration in itself can be considered a form of fusion, since it involves comparison and combination of image information.

Our discussion on registration focussed on spatial misalignments in image features. The temporal sampling rate (frame-rate or the number of images generated per second) of the sensors causes yet another misalignment between the images. Temporal misalignments between images can be corrected using frame interpolation techniques [49, 65]. These techniques interpolate an image frame from adjacent frames, within a sensor video sequence.

Appendix C

Display of Multisensor Fused Images

C.1 Introduction

The probabilistic fusion rules retain features from the sensor images. However, the fused image does not indicate the source of the image features (sensor-specific information). Knowledge of the source of the image features is likely to be helpful in fusion applications. For example, the knowledge that a bright patch in the fused image originated from the IR sensor is critical in interpreting the patch as a hot object. Similarly, knowledge that a dark patch is from a visible-band sensor could help disambiguate between a shadow and an actual object. Representation of sensor-specific information could also enable identification of image features that have been masked by features from another sensor. In addition, the fusion rules do not differentiate between relevant and irrelevant complementary features (see Section 3.2.1). Ideally, one would want to identify and suppress irrelevant complementary features. Identifying irrelevant complementary features is a difficult machine vision task. An alternative would be to display the sensor-specific information and let the observer decide whether the features are relevant.

One way to do that is to use additional dimensions as in color representation. In this appendix we explore several methods for representing the source of complementary information in the fused image using pseudocolor representations. Section C.2 briefly reviews prior work in pseudocolor fusion. In Section C.3 we propose a set of principles for mapping fused data onto additional dimensions and discuss possible mapping approaches. In Section C.4 we describe two approaches to map the fused image and the sensor images onto color channels, and illustrate each approach with an example. In Section C.5, we describe an approach for representing complementary features using color. The results of experiments described here demonstrate the feasibility and potential of mapping approaches.

C.2 Prior Work

Waxman et al. [76, 77] have described a technique for generating pseudocolor fused images from low-light visible-band and IR sensors. The visible-band image is contrast enhanced and two contrast enhanced images are derived from the IR image using on and off center-surround processing. These images are then assigned to the red, green and blue channels of a color display. The pseudocolor renderings of night scenes produced by this technique have a natural-like appearance. Although the technique improves the appearance of the fused image, it does not necessarily deliver the sensor-specific information.

Toet and Walraven [71] have proposed another approach for generating a pseudocolor fused display to represent the source of the image information. The common component of two sensor images is estimated as the minimum graylevel value at each pixel location. Then the unique component of each sensor image is computed by subtracting the common component from the original. The unique component of each sensor image is then subtracted from the other sensor image. The resulting images are applied to the red and green channels of a color display. The fused display retains sensor-specific image information. However, the shortcoming of this approach is that it does not distinguish between reliable signal and noise.

C.3 Mapping Fused Data to Convey Sensor-specific Details

The fusion techniques described in this dissertation generate a graylevel fused image from graylevel sensor images. We now outline a set of principles for mapping the fused images and the sensor images onto additional dimensions to represent sensor-specific information. We also discuss different approaches to mapping.

C.3.1 Principles of mapping

For a mapping approach to be effective, the mapping of the sensor data and the fused data onto the perceptual dimensions cannot be arbitrary. We propose that the following set of principles should underly any choice of mapping:

1. Matched spatial resolution — the spatial resolution of the sensors should match the spatial resolution of the corresponding perceptual dimension. A sensor that has low spatial resolution should be mapped onto a dimension that has low spatial resolution.

2. Matched temporal resolution — the temporal resolution of the sensors should match the temporal resolution of the corresponding perceptual dimension. A sensor with slower response or frame rate would be mapped onto a sluggish perceptual dimension.

3. Separability of dimensions — the perceptual dimensions used for complementary components should be perceptually separable and identifiable.

4. Integrability of dimensions — the perceptual dimensions for the common components should be integrable.

5. Cross-dimension masking — the inter-dimension masking should not impair the perceptibility of the complementary features.

C.3.2 Different approaches to mapping

The approaches to represent the source of the image features in the fused display can be divided into two broad categories. In the first category each sensor is mapped onto a particular dimension and the human visual system is used to fuse the images. Examples of this category are the approaches of Waxman et al. [76] and Toet and Walraven [71].

An alternative category of approaches is to fuse the sensor images to obtain a graylevel fused image and then use additional dimensions to convey the source of the image features. One approach could be to use color to convey the sensor-specific information. Another approach could be to use shading or texture to render the image features arising from each sensor differently. Use of blinking to denote certain features as complementary features could be yet another approach. Similarly, translucent overlays could be used to convey additional information.

C.4 Color Mapped Fused Display

Human color vision is believed to be trichromatic[1] or three-dimensional [73] (i.e. consisting of three channels). Because of the three-dimensional nature of color perception, the color of any light can be represented by a projection of the intensity versus wavelength distribution onto three *primary* color (chrominance) channels. One such trichromatic model is the RGB (red-green-blue) model. An alternative model of color perception is the *opponent color* perception model. According to this model, the trichromatic channels are recoded after the initial stage of vision into one luminance (achromatic) channel and two chrominance channels. Using the notation from vision literature — S, M, and L are responses of cones in the human retina that correspond to short, medium and long wavelengths respectively. The peak sensitivities of the L, M and S cones roughly correspond to the R, G and B wavelengths. The luminance signal, P_Y, and the two chromatic components, P_{RG} and P_{YB}, are computed in terms of the cone absorptions as

$$\begin{aligned} P_Y &= L + M, \\ P_{RG} &= L - M, \\ P_{YB} &= L + M - S. \end{aligned} \tag{C.1}$$

The chromatic component P_{RG} represents the "red-green" opponent channel, whereas the chromatic component P_{YB} represents the "yellow-blue" opponent channel. Based on the three dimensional nature of color vision, we have explored two possible mappings of the fused image and the sensor data onto transformations of the color dimensions.

[1] The trichromatic theory of human color vision states that the color of light entering the human eye may be specified by three numbers rather than a complete function of wavelengths over the visible range [49].

C.4.1 Fusion using red-green color map

The human visual system performs fusion by combining the signals from the three types of cones. This motivates our first technique where we directly map data from a sensor to a particular cone type. The additive representation of the luminance component shown in Equation (C.1), in combination with the red-green opponency, suggests that a simple mapping of one sensor onto the M channel and the other onto the L channel would yield the desired fusion effects and ensure the integrability of common components as well as separability of complementary components. Due to the additive representation of the luminance component in terms of L and M, the common features in the sensor images would appear along the luminance channel. Due to the subtractive representation of the red-green opponent channel in terms of L and M, the complementary features or polarity reversed features would appear along the red-green opponent channel.

The mapping from the sensor images a_1 and a_2 to the L (red) and M (green) channels is implemented as

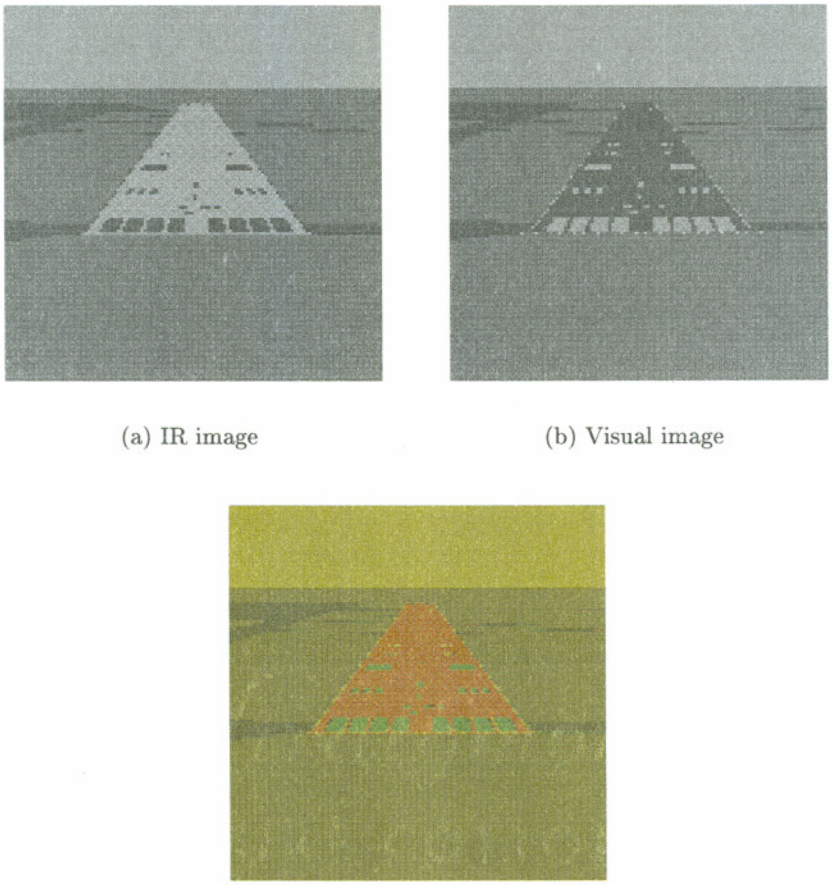
$$L = w_1 a_1$$

$$M = w_2 a_2$$

where the weights w_1 and w_2 determine the gain of each channel. For the R, G and B values applied to the monitor excitation to translate into the L, M and S cone absorptions, the L, M and S absorptions need to be converted to R, G, and B monitor excitation values. In order to do this, one needs to know the cone sensitivities and the monitor spectral power distribution (SPD) for R, G and B. The cone sensitivities can be assumed to be well described by linear functions as given by Smith and Pokorny [62]. In addition, the transformation between the cone absorptions and the standard RGB representation can be assumed to be linear. Then the LMS values can be obtained as [73]

$$[L, M, S] = [\text{cone sensitivity matrix}][\text{monitor SPD matrix}][R, G, B]. \tag{C.2}$$

The R, G, and B values are obtained by the inverse relation. However, in our experiments we assumed that the L and M absorptions correspond directly to the R and G excitations and therefore $w_1 a_1$ and $w_2 a_2$ are the excitations applied to R and G respectively. An



(a) IR image

(b) Visual image

(c) Fused image

Figure C.1: Simple color scheme for fusion (simulated images)

example of the fusion process is illustrated using two synthetic images in Figure C.1. In this example we applied the IR image in Figure C.1(a) to the "red" channel (L) and the visible-band image in Figure C.1(b) to the "green" channel (M). In this example the gain for each channel was equal (unity). However, when the sensor images are noisy, the gain for each channel can be selected in accordance with the ML fusion rule of Equation 3.13. The fused image formed as a result of the red-green color map is shown in Figure C.1(c).

The direct mapping approach shown in Figure C.1 has two drawbacks. First, polarity reversals can result in incorrect conclusion about the identification of the source of the information. The sensor images shown in Figures C.1(a) and C.1(b) contain the same image features, although these features are polarity reversed in some regions. Specifically, the runway surface and the markings on the runway have opposite polarity of contrast. The combination of the dark runway from the visible-band sensor with the bright runway from the IR sensor results in the runway appearing red in the fused image. The red appearance of the runway implies incorrectly that the runway in the fused image originates from the IR sensor alone. This is misleading since the same information is present in both images, albeit polarity reversed. Second, polarity reversals can result in reduction in detectability of edges or their motion because the resulting chromatic edges appear less salient than those defined by luminance. In addition, since the red and green channels form an opponent color direction, lights that stimulate a mixture of L and M cones are harder to perceive than lights that stimulate just one of these two types of cones [73]. To rectify these problems we examined alternate mappings that explicitly address local polarity reversals and map the sensors on to color directions that are mutually independent and orthogonal.

C.4.2 Fusion by color mapping in the YIQ color space

A convenient color space for an alternate mapping that satisfies the mapping principles of Section C.3.1 is the YIQ color space [36], that is used in commercial television broadcast. The YIQ color space is a transformation of the RGB color space such that the luminance (Y) information is decoupled from the chrominance (I and Q) information. The YIQ color space is derived from the YUV color space also used in broadcast television. The Y component consists of the luminance (graylevel) information. The U and V components are obtained from the R and B components by subtracting the luminance component. The chrominance components I and Q are obtained by rotating the U and V components by 33 degrees. The I and Q components have reduced bandwidth than the U and V components for comparable visual quality [52]. Also, the I component has higher spatial resolution and requires higher bandwidth than the Q component [36].

We now describe a mapping approach that uses the YIQ color space in conjunction with the probabilistic fusion rule and the mapping principles. Consider two sensor images a_1 and a_2. The zero-mean sensor images are obtained as,

$$\begin{aligned}\hat{a}_1 &= a_1 - \overline{a}_1, \\ \hat{a}_2 &= a_2 - \overline{a}_2, \end{aligned} \tag{C.3}$$

where $\overline{a}_1$ and $\overline{a}_2$ are the averages computed over the entire image of each sensor. We then compute the luminance Y and the chromatic components I and Q by

$$\begin{aligned} Y &= \hat{s} \\ I &= \hat{a}_1 \\ Q &= \hat{a}_2, \end{aligned} \tag{C.4}$$

where $\hat{s}$ is the estimate of the scene computed in Equation (3.15). The luminance channel Y has the highest spatial resolution and therefore the graylevel fused image is applied to this channel. The choice of which sensor image to apply to the I and Q channels depends on the spatial resolution of the sensors. The sensor image with higher spatial resolution is applied to I to satisfy the mapping principles.

The Y, I and Q components have the ranges $0 \leq Y \leq 255$, $-152 \leq I \leq 152$ and $-134 \leq Q \leq 134$. The fused image $\hat{s}$ and the sensor images a_1 and a_2 are appropriately scaled such that the graylevels are between 0 and 255. The zero mean sensor images then have intensity values between -128 and 128. The YIQ representation is then converted to the RGB color space by the transformation

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0000 & 0.9562 & 0.6214 \\ 1.0000 & -0.2727 & -0.6468 \\ 1.0000 & -1.1037 & 1.7006 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \tag{C.5}$$

The transformation from YIQ to RGB is linear. However, the R, G and B values have restricted ranges, $0 \leq R, G, B \leq 255$. Even if the Y, I and Q values lie within the allowed ranges, not all combinations of Y, I and Q values that lie within the allowed ranges produce R, G, and B values that lie between 0 and 255. This is illustrated in Figure C.2. The volume covered by the cube spans all the combination of Y, I and Q values that lie within the ranges given above. However, the shaded volume represents the permissible range of

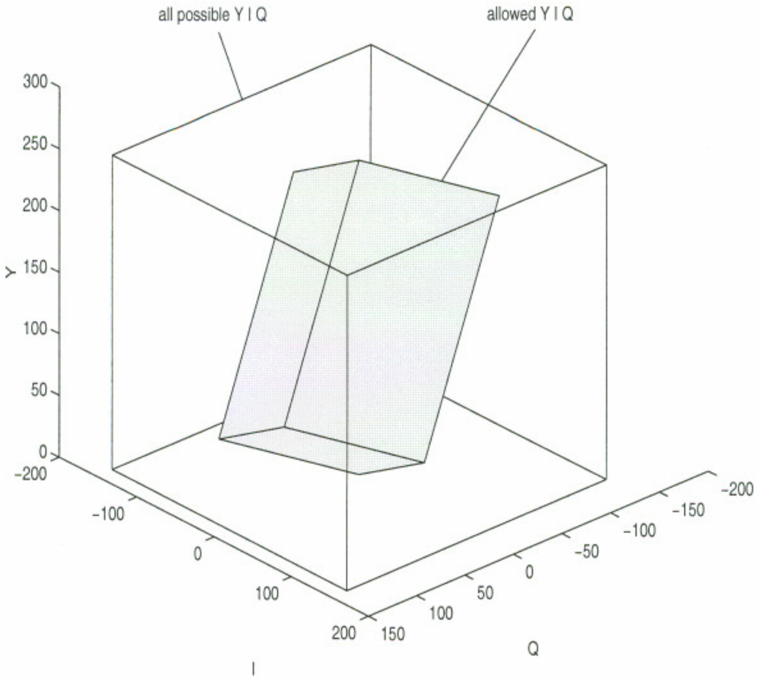


Figure C.2: Permissible values in the YIQ color space

Y, I and Q values that produce R, G, and B values that lie between 0 and 255. As a result, the mapping given in Equation C.4 must be modified in practice.

In the modified representation (Y, I', Q'), we fix the luminance component Y to that in Equation (C.4), and find the largest possible I' and Q' such that

$$\frac{I'}{Q'} = \frac{I}{Q} \tag{C.6}$$

and the R, G, B values are within range. This modification retains the proportion of the sensor intensities in the color space whereas the luminance component is unchanged. The point (Y, I', Q') lies on the surface of the shaded volume and on the line joining the point (Y, I, Q) and the origin $(0, 0, 0)$ in YIQ space.

The I' and Q' values are computed in the following manner. The restricted range of R, G, B values along with the transformation of Equation C.5 gives the following 6 constraint

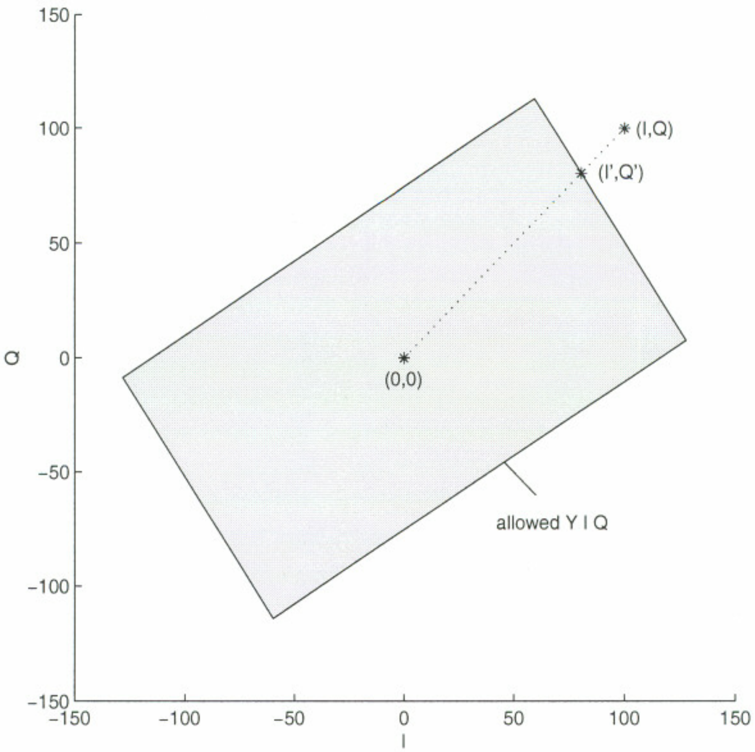


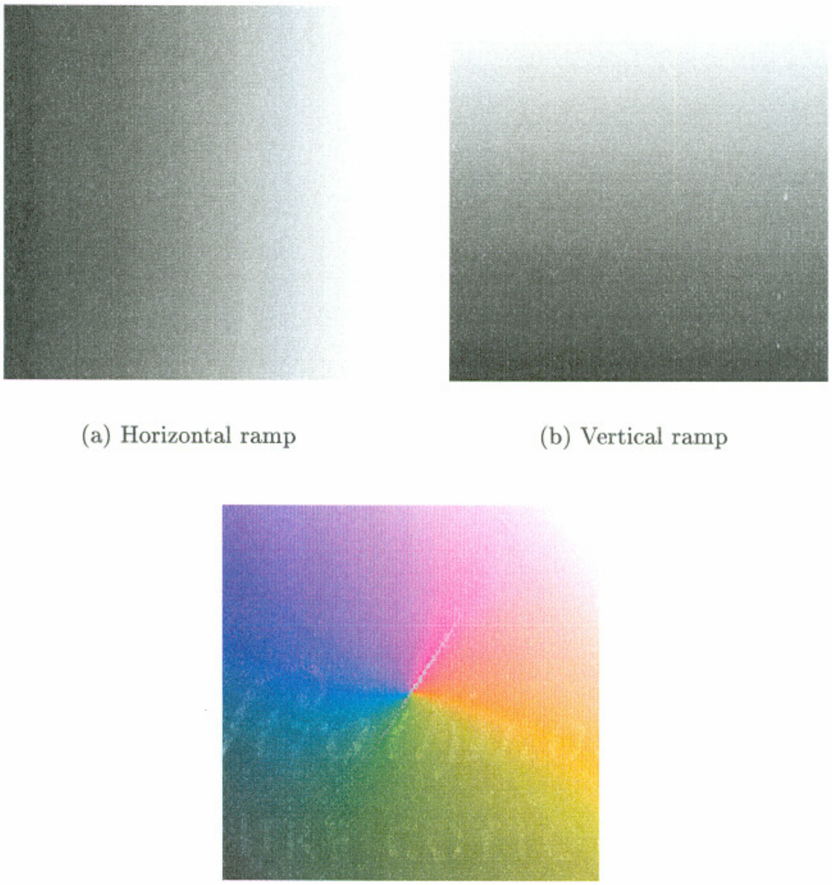
Figure C.3: Computation of modified I and Q values

lines in the YIQ space

$$\begin{aligned} 0 &\leq Y + 0.9562I + 0.6214Q \leq 255 \\ 0 &\leq Y - 0.2727I - 0.6468Q \leq 255 \\ 0 &\leq Y - 1.1037I + 1.7006Q \leq 255 \end{aligned} \tag{C.7}$$

With the value of Y fixed, we take a slice parallel to the I-Q plane through the shaded volume shown in Figure C.2. This slice is illustrated in Figure C.3 for $Y = 128$. We then find the intersection of the line passing through the point (I, Q) and the origin $(0, 0)$ in the I-Q plane as shown in Figure C.3. The intersection point closest to (I, Q) gives the desired point (I', Q'). The (Y, I', Q') values are converted to RGB space using the transformation of Equation C.5. Note that in practice the transformation is followed by another transformation using the calibration matrix [73, 36] corresponding to the particular monitor to be used for the display of the fused images.

To provide an intuitive insight we demonstrate this mapping using two test images



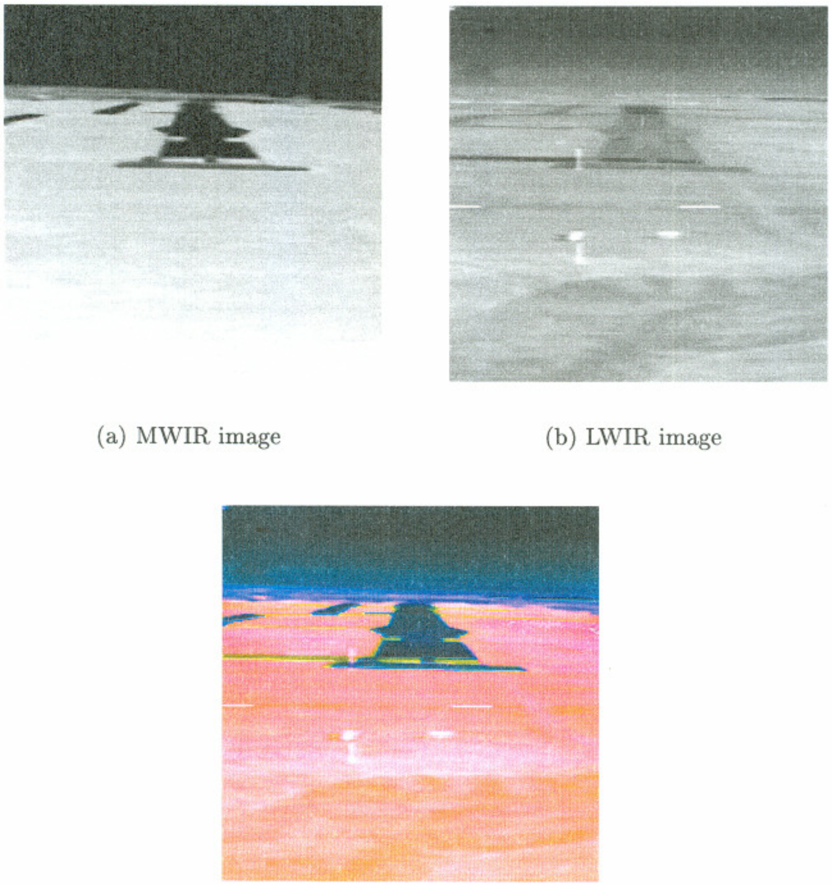
(a) Horizontal ramp

(b) Vertical ramp

(c) Fused image

Figure C.4: YIQ scheme for fusion

composed of horizontal and vertical graylevel ramps shown in Figure C.4. The image in Figure C.4(a) is a horizontal graylevel ramp with the pixel values linearly increasing from 0 to 255 from left to right representing the first sensor, a_1. The image in Figure C.4(b) is a vertical ramp with the values increasing linearly from 0 to 255 from bottom to top representing the second sensor, a_2. Figure C.4(c) is the corresponding fused image obtained by applying the YIQ scheme to the ramps. The source of the image features in the fused image can be identified by associating the different colored regions of the fused image with particular combinations of sensors. For instance, when the intensity of a_1 is high and that of a_2 is low, the resulting color is green. Similarly when both a_1 and a_2 are high, the resulting color is bright pink.



(a) MWIR image

(b) LWIR image

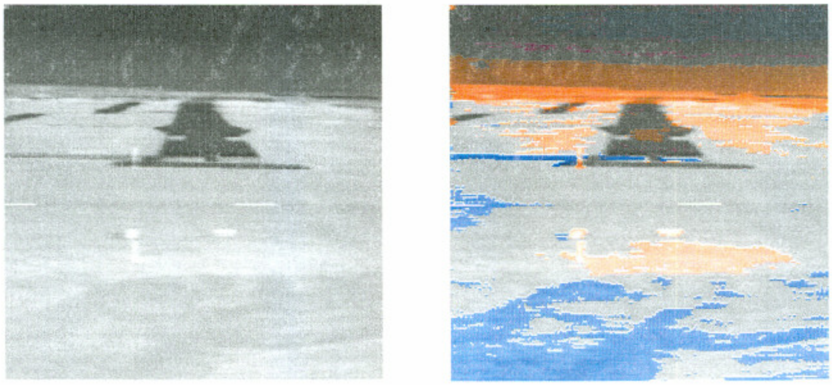
(c) Fused image

Figure C.5: YIQ scheme applied to real images (Original data from FLIR Systems Inc.)

Figure C.5 shows an example of the YIQ scheme applied to runway images from medium wave infrared (MWIR) and long wave infrared (LWIR) sensors. The bluish hue around the horizon in the fused image can be attributed to the LWIR image. The cross marks in the LWIR image that are absent in the MWIR image appear with a yellow hue. The bright pattern in the lower portion of the LWIR image appears bright pink in the fused image because of the high intensity at the corresponding locations in the MWIR image. The polarity reversed markings on the runway appear green in the fused image. The colors help identify which sensor(s) a particular image feature originated from.

The advantage of this approach is that the luminance component is identical to the fused image obtained by graylevel fusion of the sensor images. As a result, in an actual application, the color information can be turned on or off whenever necessary. The problem of local polarity reversed features is explicitly addressed by first performing the graylevel fusion. The mapping provides separability of complementary components as well as integrability of common components. The visibility of complementary features is not impaired by cross-dimension masking. Mapping in the YIQ color space provides better discrimination between the image features compared to the mapping using the red-green color space. Although this mapping allows identification of the source of sensor-specific features, the observer (for example, a pilot) has to learn the relationship between the displayed color and the interaction among the image features.

C.5 Color Mapping of Complementary Features



(a) Graylevel fusion

(b) Color display

Figure C.6: Representing complementary signal using color

In this section we explore the possibility of rendering just the complementary features using color in the fused image, and maintaining the rest of the fused image as a graylevel image. The advantage of this scheme is that the relationship between the displayed colors and the interaction among the image features is simple. This scheme ensures separability of complementary components and can help distinguish between relevant and irrelevant complementary features. The integrability of the common components is preserved since the common components appear in graylevel. As previously described in Chapter 3, complementary features are image features that are visible in one sensor image but not the others. Assume that we have two sensor images a_1 and a_2. To identify the complementary features in a_2, we describe the image a_2 in terms of the image a_1 using a locally affine transformation at each hyperpixel of the Laplacian pyramids of the sensor images,

$$a_2(\vec{l}\,) = \beta(\vec{l}\,)a_1(\vec{l}\,) + \alpha(\vec{l}\,) + \epsilon(\vec{l}\,), \tag{C.8}$$

where $\vec{l} \equiv (x, y, k)$ is the hyperpixel location, with x, y the pixel coordinates and k the level of the pyramid. Note that this mapping is similar to the image formation model of Equation (3.1). If the parameters β and α are estimated by regression along the lines of Equations (4.8) and (4.7) respectively, the parameter α at the different hyperpixels represents the complementary features from the second sensor. The image α representing the complementary features in sensor a_2 is obtained by applying the inverse pyramid transform to the pyramid of hyperpixels $\alpha(\vec{l}\,)$. We then let the luminance channel to be equal to the estimate $\hat{s}$ computed using the probabilistic fusion rules as in Chapter 5.

$$Y = \hat{s} \ , \tag{C.9}$$

and assign the complementary information to the I channel of the YIQ color space,

$$I = w\alpha \ , \tag{C.10}$$

where the weight w determines the gain of the I channel. No signal is assigned to the Q channel in this method. To generate the fused display we first convert YIQ to $YI'Q'$ and then to RGB as described in Section C.4.2. Figure C.6 shows the result of applying this mapping method to the MWIR and LWIR image of Figures C.5(a) and C.5(b) using a unit gain. The graylevel fused image is shown in Figure C.6(a). The color mapped fused image of Figure C.6(b) resembles the graylevel fused image in most regions. However, complementary features in the LWIR image are now rendered with color (the luminance component of these features remains the same as before). Regions containing complementary features in a_2 that are brighter than the corresponding regions in a_1 appear with an orange hue. Whereas regions containing complementary features in a_2 that are darker than the corresponding regions in a_1 appear with a bluish hue.

C.6 Summary and Conclusions

Graylevel fusion hides the source of the image features in the fused image (i.e., it does not indicate which sensor image a particular feature originated from). The information about the source can be inserted into the fused image using additional perceptual dimensions such as color, shading, texture and overlays. We explored the possibility of providing this hidden information to an observer using pseudocolor mapping. We generated images using color mappings in which the graylevel fused image was represented by luminance and the sensor-specific information by different color directions. The main difficulty of this approach is to find a mapping that enables an observer to easily identify and separate the contribution of each sensor in the fused image (i.e., the source of the image features in the fused image). We achieved separation of the source information using the YIQ color space for color mapping, but the approach requires the observer to learn the mapping. In another approach we rendered just the complementary features using color. This approach shows promise in terms of aiding an observer to distinguish between relevant and irrelevant complementary features. Note that the techniques described here can be refined by introducing transformations for the uniform color spaces[2]. However, finding an optimal mapping would require extensive perceptual experiments on identifying color directions. The results of our techniques, though promising, are preliminary. Empirical studies of observers' performance would be necessary to evaluate any of these and similar techniques.

[2] In a uniform color space, equal measured distances correspond to equal perceptual distances between colors. CIElab and CIEluv [21] are two standard uniform color spaces.

Appendix D

Least Squares Factor Analysis Estimation of Model Parameter $\boldsymbol{\beta}$

From Section 4.5.1, the least squares estimate of $\boldsymbol{\beta}$ at each hyperpixel is obtained by finding the $\boldsymbol{\beta}$ that minimizes the squared norm of the difference between the data covariance matrix $\boldsymbol{\Sigma}_a$ and the model covariance $\mathbf{C}(a|\mathcal{R})$ in the local analysis window,[1]

$$\begin{aligned} E_\beta &= \operatorname{tr}\left\{(\boldsymbol{\Sigma}_a - \mathbf{C})^2\right\} \\ &= \|\boldsymbol{\Sigma}_a - \mathbf{C}\|^2 \\ &= \sum_{i,j}\left\{(\boldsymbol{\Sigma}_a - \mathbf{C})^2\right\}_{ij} \end{aligned} \tag{D.1}$$

where,

$$\mathbf{C} = \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \tag{D.2}$$

Varying $\mathbf{C}$,

$$\begin{aligned} \delta E_\beta &= -2\sum_{i,j}(\boldsymbol{\Sigma}_a - \mathbf{C})_{ij}\,\delta\mathbf{C}_{ij} \\ &= -2\mathrm{tr}\left\{(\boldsymbol{\Sigma}_a - \mathbf{C})\,\delta\mathbf{C}\right\} \end{aligned} \tag{D.3}$$

Varying $\boldsymbol{\beta}$,

$$\delta\mathbf{C} = \sigma^2_{s,s_0}\boldsymbol{\beta}\delta\boldsymbol{\beta}^{\mathrm{T}} + \sigma^2_{s,s_0}\delta\boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} \tag{D.4}$$

[1] Henceforth, we drop the notation referring to the region $\mathcal{R}$ and refer to $\mathbf{C}(a|\mathcal{R})$ defined in Section 4.5 as $\mathbf{C}$.

Substituting Equation (D.4) in Equation (D.3),

$$\begin{aligned} \delta E_\beta &= -2\mathrm{tr}\left\{\sigma^2_{s,s_0}\left(\boldsymbol{\Sigma}_a - \mathbf{C}\right)\left(\boldsymbol{\beta}\delta\boldsymbol{\beta}^{\mathrm{T}} + \delta\boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}}\right)\right\} \\ &= -4\sigma^2_{s,s_0}\mathrm{tr}\left\{\left(\boldsymbol{\Sigma}_a - \mathbf{C}\right)\boldsymbol{\beta}\delta\boldsymbol{\beta}^{\mathrm{T}}\right\} \end{aligned} \tag{D.5}$$

Therefore,

$$\frac{\partial E_\beta}{\partial\boldsymbol{\beta}} = -4\sigma^2_{s,s_0}\left(\boldsymbol{\Sigma}_a - \mathbf{C}\right)\boldsymbol{\beta} \tag{D.6}$$

To obtain the $\boldsymbol{\beta}$ that minimizes E_β, we set

$$\frac{\partial E_\beta}{\partial\boldsymbol{\beta}} = 0$$

and recover,

$$\left(\boldsymbol{\Sigma}_a - \mathbf{C}\right)\boldsymbol{\beta} = 0 \tag{D.7}$$

Substituting for $\mathbf{C}$ from Equation (D.2),

$$\left(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}\right)\boldsymbol{\beta} = \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}$$

This equation imposes two constraints on $\boldsymbol{\beta}$ —

1. $\boldsymbol{\beta}$ is an eigenvector of $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$, and
2. $\sigma^2_{s,s_0}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}$ is the corresponding eigenvalue.

The solution to $\boldsymbol{\beta}$ that satisfies both these constraints is,

$$\boldsymbol{\beta}_{LS} = \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r \tag{D.8}$$

where $\mathbf{U}$ is an eigenvector, and λ is an eigenvalue of the noise-corrected covariance matrix $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$. The variable $r = \pm 1$ is the sign of $\boldsymbol{\beta}$, and determines the polarity of contrast.

Minimum Value of E_β

Equation (D.8) gives the solution for $\boldsymbol{\beta}$ in terms of the eigenvalue and eigenvector of $(\boldsymbol{\Sigma}_a - \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ but does not specify the conditions under which E_β is minimum. We now consider a case where there are two sensors, so that the noise-corrected covariance matrix $(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon)$ has two eigenvalues λ_1 and λ_2, with $\lambda_1 > \lambda_2$, and corresponding eigenvectors $\mathbf{U}_1$ and $\mathbf{U}_2$ respectively. $(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon)$ can be expressed in terms of λ_1, $\lambda_2, \mathbf{U}_1$ and $\mathbf{U}_2$ as

$$(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon) = \lambda_1 \mathbf{U}_1 \mathbf{U}_1{}^{\mathrm{T}} + \lambda_2 \mathbf{U}_2 \mathbf{U}_2{}^{\mathrm{T}} \tag{D.9}$$

Assume that

$$\boldsymbol{\beta} = \frac{\lambda_1^{\frac{1}{2}}}{\sigma_{s,s_0}} \mathbf{U}_1 r \tag{D.10}$$

From Equation (D.1),

$$\begin{aligned} E_\beta &= \operatorname{tr}\left\{(\mathbf{\Sigma}_a - \mathbf{C})^2\right\} \\ &= \operatorname{tr}\left\{\left(\mathbf{\Sigma}_a - \sigma_{s,s_0}^2 \boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}} - \mathbf{\Sigma}_\epsilon\right)^2\right\} \\ &= \operatorname{tr}\left\{\left(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon - \sigma_{s,s_0}^2 \boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}}\right)^2\right\} \end{aligned} \tag{D.11}$$

Substituting Equation (D.10) and Equation (D.9) in Equation (D.11),

$$\begin{aligned} E_\beta &= \operatorname{tr}\left\{\left(\lambda_1 \mathbf{U}_1 \mathbf{U}_1{}^{\mathrm{T}} + \lambda_2 \mathbf{U}_2 \mathbf{U}_2{}^{\mathrm{T}} - \sigma_{s,s_0}^2 \frac{\lambda_1}{\sigma_{s,s_0}^2} \mathbf{U}_1 \mathbf{U}_1{}^{\mathrm{T}}\right)^2\right\} \\ &= \operatorname{tr}\left\{(\lambda_2 \mathbf{U}_2 \mathbf{U}_2{}^{\mathrm{T}})^2\right\} \\ &= \lambda_2^2 \end{aligned} \tag{D.12}$$

If we had chosen

$$\boldsymbol{\beta} = \frac{\lambda_2^{\frac{1}{2}}}{\sigma_{s,s_0}} \mathbf{U}_2 r \tag{D.13}$$

then E_β in Equation (D.12) would have been λ_1^2. Therefore, the value of $\boldsymbol{\beta}$ that minimizes E_β is

$$\boldsymbol{\beta} = \frac{\lambda_1^{\frac{1}{2}}}{\sigma_{s,s_0}} \mathbf{U}_1 r \tag{D.14}$$

where $\mathbf{U}_1$ is the principal eigenvector and λ_1 is the principal eigenvalue of $(\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon)$.

Appendix E

Maximum Likelihood Factor Analysis Estimation of Model Parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

Maximum likelihood approach to estimate $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

An alternate way to approach the problem of estimating the affine parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ is to use maximum likelihood factor analysis methods [3, 37, 38, 67]. In Section 4.5 we derived the model distribution over the local analysis window $\mathcal{R}_{\mathcal{L}}$. Each hyperpixel in the local analysis window can be assumed to be independent and identically distributed according to the model distribution over $\mathcal{R}_{\mathcal{L}}$ (which is a Gaussian, with mean and covariance given by Equations (4.18) and (4.21) respectively). The log-likelihood of observing $N = 25$ data points in the local analysis window $\mathcal{R}_{\mathcal{L}}$ of 5×5 hyperpixels can be computed as,

$$\begin{aligned}
\mathcal{L} &= \sum_{n=1}^{N} \ln[\mathcal{P}(\boldsymbol{a}_n|\mathcal{R})] \\
&= -\frac{N}{2}\ln(2\pi) - \frac{N}{2}\ln|\mathbf{C}(\boldsymbol{a}|\mathcal{R})| \\
&\quad -\frac{1}{2}\sum_{n=1}^{N} \operatorname{tr}\left\{(\mathbf{C}(\boldsymbol{a}|\mathcal{R}))^{-1}(\boldsymbol{a}_n - \boldsymbol{\beta}\mu_{s_0} - \boldsymbol{\alpha})(\boldsymbol{a}_n - \boldsymbol{\beta}\mu_{s_0} - \boldsymbol{\alpha})^{\mathrm{T}}\right\} \,, \qquad \text{(E.1)}
\end{aligned}$$

where n corresponds to the n^{th} hyperpixel in the local analysis window. To form a maximum likelihood (ML) estimate of $\boldsymbol{\alpha}$ we differentiate the log-likelihood $\mathcal{L}$ with respect to $\boldsymbol{\alpha}$, and set the result to zero to obtain

$$\boldsymbol{\alpha}_{\mathrm{ML}} = \boldsymbol{\mu}_a - \boldsymbol{\beta}\mu_{s_0} \,, \qquad \text{(E.2)}$$

where $\boldsymbol{\mu}_a$ is the data mean defined in Equation (4.24). Substituting this estimate of $\boldsymbol{\alpha}_{\mathrm{ML}}$ in Equation (E.1), the log-likelihood can be written as,

$$\begin{aligned}\mathcal{L} &= -\frac{N}{2}\ln(2\pi) - \frac{N}{2}\ln|\mathbf{C}(\boldsymbol{a}|\mathcal{R})| - \frac{1}{2}\mathrm{tr}\left\{\sum_{n=1}^{N}(\mathbf{C}(\boldsymbol{a}|\mathcal{R}))^{-1}(\boldsymbol{a}_n - \boldsymbol{\mu}_a)(\boldsymbol{a}_n - \boldsymbol{\mu}_a)^{\mathrm{T}}\right\} \\ &= -\frac{N}{2}\ln(2\pi) - \frac{N}{2}\ln|\mathbf{C}(\boldsymbol{a}|\mathcal{R})| - \frac{N}{2}\mathrm{tr}\left\{(\mathbf{C}(\boldsymbol{a}|\mathcal{R}))^{-1}\boldsymbol{\Sigma}_a\right\}, \qquad \text{(E.3)}\end{aligned}$$

where $\boldsymbol{\Sigma}_a$ is the data covariance matrix as defined in Equation (4.25). To obtain the $\boldsymbol{\beta}$ that maximizes the log-likelihood, we take the derivative of L with respect to $\boldsymbol{\beta}$, equate it to zero, and recover (the derivation is given in the following section)

$$\{\mathbf{C}(\boldsymbol{a}|\mathcal{R}) - \boldsymbol{\Sigma}_a\}\{\mathbf{C}(\boldsymbol{a}|\mathcal{R})\}^{-1}\boldsymbol{\beta} = 0 \qquad \text{(E.4)}$$

Equation (E.4) simplifies to

$$(\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}\boldsymbol{\Sigma}_a\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}} - \mathbf{I})\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}\boldsymbol{\beta} = \sigma_{s,s_0}^2\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}\boldsymbol{\beta} \qquad \text{(E.5)}$$

The solution to Equation (E.5) is

$$\boldsymbol{\beta}_{\mathrm{ML}} = \frac{(\widetilde{\lambda} - 1)^{\frac{1}{2}}}{\sigma_{s,s_0}}\boldsymbol{\Sigma}_\epsilon^{\frac{1}{2}}\widetilde{\mathbf{U}}r \qquad \text{(E.6)}$$

where $\widetilde{\mathbf{U}}$ is an eigenvector and $\widetilde{\lambda}$ the corresponding eigenvalue, of the weighted data covariance matrix, $\widetilde{\boldsymbol{\Sigma}}_a \equiv \boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}\boldsymbol{\Sigma}_a\boldsymbol{\Sigma}_\epsilon^{-\frac{1}{2}}$, and $r = \pm 1$. The maximum likelihood occurs when $\widetilde{\mathbf{U}}$ is the principal eigenvector of $\widetilde{\boldsymbol{\Sigma}}_a$. The maximum likelihood technique, like the least squares technique, does not provide an estimate of the sign of r or for σ_{s,s_0}^2 and μ_{s_0}. The sign of r can be chosen in the same way as in the least squares technique. The value of σ_{s,s_0}^2 can be chosen such that $\|\boldsymbol{\beta}_{\mathrm{ML}}\|^2 = 1$ in Equation (E.6) and the value of μ_{s_0} in Equation (E.2) can be chosen as zero in the absence of prior knowledge about the scene.

The maximum likelihood and least squares solutions for the estimation of the affine parameters are related under certain assumptions. The relationship between these two solutions is explained in detail in Appendix F.

Derivation of the maximum likelihood estimate of $\boldsymbol{\beta}$

From Equation (E.3) the log-likelihood of observing N data points in the local analysis window of 5 × 5 hyperpixels can be computed as[1]

$$L = -\frac{N}{2}\ln(2\pi) \; - \frac{N}{2}\ln|\mathbf{C}| - \frac{N}{2}\text{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\} \tag{E.7}$$

where,

$$\mathbf{C} = \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^T + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \tag{E.8}$$

where,

$$\sigma^2_{s,s_0} = \sigma^2_s + \sigma^2_{s_0} \tag{E.9}$$

and

$$\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} = \begin{bmatrix} \sigma^2_{\epsilon_1} & 0 & \cdots & 0 \\ 0 & \sigma^2_{\epsilon_1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2_{\epsilon_q} \end{bmatrix} \tag{E.10}$$

To form a maximum likelihood estimate of $\boldsymbol{\beta}$ we take the derivative of L with respect

[1] For simplicity of notation, in the rest of this appendix, we refer to $\mathbf{C}(\mathbf{a}|\mathcal{R})$ the model covariance over the local analysis window defined by region $\mathcal{R}$ simply as $\mathbf{C}$.

to $\boldsymbol{\beta}$. The derivatives are computed as follows:

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\beta}_{kl}} \ln|\mathbf{C}| &= \sum_{i,j} \frac{\partial \mathbf{C}_{ij}}{\partial \boldsymbol{\beta}_{kl}} \frac{\partial}{\partial \mathbf{C}_{ij}} \ln|\mathbf{C}| \\
&= \sum_{i,j} \sigma^2_{s,s_0} \sum_p \frac{\partial}{\partial \boldsymbol{\beta}_{kl}} \left(\boldsymbol{\beta}_{ip}\boldsymbol{\beta}^T_{pj}\right) \left(\mathbf{C}^{-1}\right)_{ij} \\
&= \sigma^2_{s,s_0} \sum_{i,j} \sum_p \left(\boldsymbol{\beta}_{ip} \frac{\partial}{\partial \boldsymbol{\beta}_{kl}} \boldsymbol{\beta}^T_{pj} + \boldsymbol{\beta}^T_{pj} \frac{\partial}{\partial \boldsymbol{\beta}_{kl}} \boldsymbol{\beta}_{ip}\right) \left(\mathbf{C}^{-1}\right)_{ji} \\
&= \sigma^2_{s,s_0} \sum_{i,j} \sum_p \left(\boldsymbol{\beta}_{ip} \frac{\partial}{\partial \boldsymbol{\beta}_{kl}} \boldsymbol{\beta}_{jp} + \boldsymbol{\beta}_{jp}\delta_{ik}\delta_{pl}\right) \left(\mathbf{C}^{-1}\right)_{ji} \\
&= \sigma^2_{s,s_0} \sum_{i,j} \sum_p \left(\boldsymbol{\beta}_{ip}\delta_{jk}\delta_{pl} + \boldsymbol{\beta}_{jp}\delta_{ik}\delta_{pl}\right) \left(\mathbf{C}^{-1}\right)_{ji} \\
&= \sigma^2_{s,s_0} \left\{ \sum_{i,j} \sum_p \boldsymbol{\beta}_{ip}\delta_{jk}\delta_{pl} \left(\mathbf{C}^{-1}\right)_{ji} + \sum_{i,j} \sum_p \boldsymbol{\beta}_{jp}\delta_{ik}\delta_{pl} \left(\mathbf{C}^{-1}\right)_{ji} \right\} \\
&= \sigma^2_{s,s_0} \left\{ \sum_i \boldsymbol{\beta}_{il} \left(\mathbf{C}^{-1}\right)_{ki} + \sum_j \boldsymbol{\beta}_{jl} \left(\mathbf{C}^{-1}\right)_{jk} \right\} \\
&= 2\sigma^2_{s,s_0} \left(\mathbf{C}^{-1}\boldsymbol{\beta}\right)_{kl} \qquad \text{(E.11)}
\end{aligned}$$

where we made use of the fact that $\mathbf{C}$ is a symmetric matrix. From Equation (E.11),

$$\frac{\partial}{\partial \boldsymbol{\beta}} \ln|\mathbf{C}| = 2\sigma^2_{s,s_0} \mathbf{C}^{-1}\boldsymbol{\beta} \qquad \text{(E.12)}$$

To obtain the partial derivative of $\mathrm{tr}\left\{\mathbf{C}^{-1}\Sigma_a\right\}$ with respect to $\boldsymbol{\beta}$ we first derive the following results:

$$\begin{aligned}
\sum_k \mathbf{C}_{ik}\mathbf{C}^{-1}_{kj} &= \delta_{ij} \\
\frac{\partial}{\partial \mathbf{C}_{mn}} \sum_k \mathbf{C}_{ik}\mathbf{C}^{-1}_{kj} &= 0 \\
\sum_k \left\{ \frac{\partial \mathbf{C}_{ik}}{\partial \mathbf{C}_{mn}} \mathbf{C}^{-1}_{kj} + \mathbf{C}_{ik} \frac{\partial \mathbf{C}^{-1}_{kj}}{\partial \mathbf{C}_{mn}} \right\} &= 0 \\
\sum_k \delta_{im}\delta_{kn}\mathbf{C}^{-1}_{kj} + \sum_k \mathbf{C}_{ik} \frac{\partial \mathbf{C}^{-1}_{kj}}{\partial \mathbf{C}_{mn}} &= 0 \\
\sum_k \mathbf{C}_{ik} \frac{\partial \mathbf{C}^{-1}_{kj}}{\partial \mathbf{C}_{mn}} + \delta_{im}\mathbf{C}^{-1}_{nj} &= 0 \\
\sum_i \mathbf{C}^{-1}_{ri} \left\{ \sum_k \mathbf{C}_{ik} \frac{\partial \mathbf{C}^{-1}_{kj}}{\partial \mathbf{C}_{mn}} + \delta_{im}\mathbf{C}^{-1}_{nj} \right\} &= 0 \\
\sum_k \sum_i \mathbf{C}^{-1}_{ri}\mathbf{C}_{ik} \frac{\partial \mathbf{C}^{-1}_{kj}}{\partial \mathbf{C}_{mn}} + \sum_i \mathbf{C}^{-1}_{ri}\delta_{im}\mathbf{C}^{-1}_{nj} &= 0 \\
\sum_k \delta_{rk} \frac{\partial \mathbf{C}^{-1}_{kj}}{\partial \mathbf{C}_{mn}} + \mathbf{C}^{-1}_{rm}\mathbf{C}^{-1}_{nj} &= 0 \\
\frac{\partial \mathbf{C}^{-1}_{rj}}{\partial \mathbf{C}_{mn}} &= -\mathbf{C}^{-1}_{rm}\mathbf{C}^{-1}_{nj} \qquad \text{(E.13)}
\end{aligned}$$

Using Equation (E.13), the partial derivative of $\mathrm{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\}$ with respect to $\mathbf{C}$ is

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{C}_{ij}} \left(\mathrm{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\}\right) &= \frac{\partial}{\partial \mathbf{C}_{ij}} \left\{ \sum_m \sum_n \mathbf{C}^{-1}_{mn}\Sigma_{a_{nm}} \right\} \\
&= \sum_m \sum_n \Sigma_{a_{nm}} \frac{\partial \mathbf{C}^{-1}_{mn}}{\partial \mathbf{C}_{ij}} \\
&= -\sum_m \sum_n \Sigma_{a_{nm}} \mathbf{C}^{-1}_{mi}\mathbf{C}^{-1}_{jn} \\
&= -\sum_m \sum_n \mathbf{C}^{-1}_{jn}\Sigma_{a_{nm}}\mathbf{C}^{-1}_{mi} \\
&= -\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{ji} \qquad \text{(E.14)}
\end{aligned}$$

Using Equation (E.14) and given that $\mathbf{C}$ and $\boldsymbol{\Sigma}_a$ are symmetric matrices, we obtain the partial derivative of $\mathrm{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\}$ with respect to $\boldsymbol{\beta}$ as

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\beta}_{kl}}\mathrm{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\} &= \sum_{i,j}\frac{\partial\mathbf{C}_{ij}}{\partial\boldsymbol{\beta}_{kl}}\frac{\partial}{\partial\mathbf{C}_{ij}}\left(\mathrm{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\}\right) \\
&= -\sum_{i,j}\sigma^2_{s,s_0}\sum_{p}\frac{\partial}{\partial\boldsymbol{\beta}_{kl}}\left(\boldsymbol{\beta}_{ip}\boldsymbol{\beta}^T_{pj}\right)\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{ji} \\
&= -\sigma^2_{s,s_0}\sum_{i,j}\sum_{p}\left(\boldsymbol{\beta}_{ip}\frac{\partial}{\partial\boldsymbol{\beta}_{kl}}\boldsymbol{\beta}^T_{pj}+\boldsymbol{\beta}^T_{pj}\frac{\partial}{\partial\boldsymbol{\beta}_{kl}}\boldsymbol{\beta}_{ip}\right)\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{ji} \\
&= -\sigma^2_{s,s_0}\sum_{i,j}\sum_{p}\left(\boldsymbol{\beta}_{ip}\delta_{jk}\delta_{pl}+\boldsymbol{\beta}_{jp}\delta_{ik}\delta_{pl}\right)\left(\mathbf{C}^{-1}\Sigma_a\mathbf{C}^{-1}\right)_{ji} \\
&= -\sigma^2_{s,s_0}\left\{\sum_{i,j}\sum_{p}\boldsymbol{\beta}_{ip}\delta_{jk}\delta_{pl}\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{ji}\right. \\
&\qquad\left.+\sum_{i,j}\sum_{p}\boldsymbol{\beta}_{jp}\delta_{ik}\delta_{pl}\left(\mathbf{C}^{-1}\Sigma_a\mathbf{C}^{-1}\right)_{ji}\right\} \\
&= -\sigma^2_{s,s_0}\left\{\sum_{i}\boldsymbol{\beta}_{il}\left(\mathbf{C}^{-1}\Sigma_a\mathbf{C}^{-1}\right)_{ki}+\sum_{j}\boldsymbol{\beta}_{jl}\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{jk}\right\} \\
&= -\sigma^2_{s,s_0}\left\{\sum_{i}\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{ki}\boldsymbol{\beta}_{il}+\sum_{j}\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\right)_{kj}\boldsymbol{\beta}_{jl}\right\} \\
&= -2\sigma^2_{s,s_0}\left(\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\boldsymbol{\beta}\right)_{kl} \qquad \text{(E.15)}
\end{aligned}$$

From Equation (E.15),

$$\frac{\partial}{\partial\boldsymbol{\beta}}\mathrm{tr}\left\{\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\right\} = -2\sigma^2_{s,s_0}\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\boldsymbol{\beta} \qquad \text{(E.16)}$$

From Equations (E.12) and (E.16)

$$\frac{\partial L}{\partial\boldsymbol{\beta}} = -2\sigma^2_{s,s_0}\mathbf{C}^{-1}\boldsymbol{\beta}+2\sigma^2_{s,s_0}\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\boldsymbol{\beta} \qquad \text{(E.17)}$$

To obtain the value of $\boldsymbol{\beta}$ that maximizes the likelihood, we equate the above derivative to zero to obtain

$$-\mathbf{C}^{-1}\boldsymbol{\beta}+\mathbf{C}^{-1}\boldsymbol{\Sigma}_a\mathbf{C}^{-1}\boldsymbol{\beta}=0 \qquad \text{(E.18)}$$

If $\sigma^2_{\epsilon_1}, \sigma^2_{\epsilon_2}, \ldots, \sigma^2_{\epsilon_q} > 0$, then $\mathbf{C}^{-1}$exists, and Equation (E.18) can be written as

$$(\mathbf{C}-\boldsymbol{\Sigma}_a)\mathbf{C}^{-1}\boldsymbol{\beta}=0 \qquad \text{(E.19)}$$

But

$$\mathbf{C}^{-1}\boldsymbol{\beta} = \boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta}(1 + \sigma^2_{s,s_0}\boldsymbol{\beta}^T\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta})^{-1} \tag{E.20}$$

From Equations (E.19) and (E.20),

$$(\mathbf{C} - \boldsymbol{\Sigma}_a)\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta}(1 + \sigma^2_{s,s_0}\boldsymbol{\beta}^T\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta})^{-1} = 0$$

Post-multiplying by $(1 + \sigma^2_{s,s_0}\boldsymbol{\beta}^T\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta})$,

$$(\mathbf{C} - \boldsymbol{\Sigma}_a)\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} = 0$$

Using Equation (E.8),

$$(\boldsymbol{\Sigma_\epsilon} + \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^T - \boldsymbol{\Sigma}_a)\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} = 0$$

$$\boldsymbol{\beta} + \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} - \boldsymbol{\Sigma}_a\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} = 0 \tag{E.21}$$

Pre-multiply by $\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}$

$$\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\beta} + \sigma^2_{s,s_0}\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} - \boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\Sigma}_a\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} = 0$$

$$(\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\Sigma}_a\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}} - I)\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\beta} = \sigma^2_{s,s_0}\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{\Sigma_\epsilon}^{-1}\boldsymbol{\beta} \tag{E.22}$$

Let

$$\widetilde{\boldsymbol{\Sigma}}_a = \boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\Sigma}_a\boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}} \tag{E.23}$$

and

$$\widetilde{\boldsymbol{\beta}} = \boldsymbol{\Sigma_\epsilon}^{-\frac{1}{2}}\boldsymbol{\beta} \tag{E.24}$$

Then,

$$(\widetilde{\boldsymbol{\Sigma}}_a - I)\widetilde{\boldsymbol{\beta}} = \sigma^2_{s,s_0}\widetilde{\boldsymbol{\beta}}\widetilde{\boldsymbol{\beta}}^T\widetilde{\boldsymbol{\beta}} \tag{E.25}$$

$$(\widetilde{\boldsymbol{\Sigma}}_a - I)\widetilde{\boldsymbol{\beta}} = \eta\widetilde{\boldsymbol{\beta}} \tag{E.26}$$

where,

$$\eta = \sigma_{s,s_0}^2 \widetilde{\boldsymbol{\beta}}^T \widetilde{\boldsymbol{\beta}} \tag{E.27}$$

The $\widetilde{\boldsymbol{\beta}}$ that satisfies Equation (E.26) as well as Equation (E.27) is given by

$$\widetilde{\boldsymbol{\beta}} = \frac{\eta^{\frac{1}{2}}}{\sigma_{s,s_0}} \widetilde{\mathbf{U}} r \tag{E.28}$$

where $\widetilde{\mathbf{U}}$ is the normalized principal eigenvector of $\widetilde{\boldsymbol{\Sigma}}_a$, $r = \pm 1$ is the sign, and

$$\eta = \widetilde{\lambda} - 1 \tag{E.29}$$

where $\widetilde{\lambda}$ is the principal eigenvalue of $\widetilde{\boldsymbol{\Sigma}}_a$. Combining Equations (E.24), (E.28) and (E.29), we obtain the maximum likelihood estimate of the factor loadings $\boldsymbol{\beta}$.

$$\boldsymbol{\beta}_{\mathrm{ML}} = \frac{(\widetilde{\lambda} - 1)^{\frac{1}{2}}}{\sigma_{s,s_0}} \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}^{\frac{1}{2}} \widetilde{\mathbf{U}} r \tag{E.30}$$

where $\widetilde{\mathbf{U}}$ and $\widetilde{\lambda}$ are the principal eigenvector and the principal eigenvalue of the noise weighted data covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}^{-\frac{1}{2}} \boldsymbol{\Sigma}_a \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}^{-\frac{1}{2}}$.

Appendix F

Relationship Between the Least Squares and the Maximum Likelihood Factor Analysis Estimation of Model Parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

The maximum likelihood factor analysis and the least squares factor analysis solutions for the estimate of the sensor bias parameter $\boldsymbol{\alpha}$ are identical. Here we derive the relationship between the maximum likelihood factor analysis solution for $\boldsymbol{\beta}$ and the least squares factor analysis solution for $\boldsymbol{\beta}$. The solutions for the estimate of the sensor gain parameter $\boldsymbol{\beta}$ are identical under two conditions – (i) when the model is exact, and (ii) the noise variance is equal in all sensors.

Exact model

From Section 4.5, the model covariance over the local analysis window defined by the region $\mathcal{R}$ is given by[1]

$$\mathbf{C} = \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^T + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \tag{F.1}$$

where R is the region specified by the local analysis window. An exact model means that the data covariance matrix is exactly of the form specified by the model covaraince, that

[1] In this appendix, we have dropped the notation referring to the region $\mathcal{R}$ and refer to $\mathbf{C}(a|\mathcal{R})$ defined in Section 4.5 as $\mathbf{C}$.

is

$$\mathbf{\Sigma}_a \equiv \mathbf{C} \tag{F.2}$$

From Section 4.5.1, the least squares solution for $\boldsymbol{\beta}$ is obtained from the relation

$$(\mathbf{\Sigma}_a - \mathbf{\Sigma}_{\boldsymbol{\epsilon}})\,\boldsymbol{\beta} = \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{\beta} \tag{F.3}$$

The least squares solution is given by

$$\boldsymbol{\beta}_{LS} = \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r \tag{F.4}$$

where $\mathbf{U}$ and λ are the principal eigenvector and the principal eigenvalue respectively of the noise corrected data covariance matrix $(\mathbf{\Sigma}_a - \mathbf{\Sigma}_{\boldsymbol{\epsilon}})$, and $r = \pm 1$ is the sign of the eigenvector $\mathbf{U}$.

From Section E, the maximum likelihood solution for $\boldsymbol{\beta}$ is obtained from

$$(\widetilde{\mathbf{\Sigma}}_a - \mathbf{I})\widetilde{\boldsymbol{\beta}} = \sigma^2_{s,s_0}\widetilde{\boldsymbol{\beta}}\widetilde{\boldsymbol{\beta}}^T\widetilde{\boldsymbol{\beta}} \tag{F.5}$$

where

$$\widetilde{\mathbf{\Sigma}}_a = \mathbf{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\mathbf{\Sigma}_a\mathbf{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}} \tag{F.6}$$

and

$$\widetilde{\boldsymbol{\beta}} = \mathbf{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\boldsymbol{\beta} \tag{F.7}$$

The maximum likelihood solution is given by

$$\boldsymbol{\beta}_{\mathrm{ML}} = \frac{(\widetilde{\lambda}-1)^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{\Sigma}_{\boldsymbol{\epsilon}}{}^{\frac{1}{2}}\widetilde{\mathbf{U}}r \tag{F.8}$$

where $\widetilde{\mathbf{U}}$ and $\widetilde{\lambda}$ are the principal eigenvector and the principal eigenvalue respectively of the noise weighted data covariance matrix $\mathbf{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\mathbf{\Sigma}_a\mathbf{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}$ and $r = \pm 1$ is the sign of the eigenvector $\widetilde{\mathbf{U}}$.

To derive the relationship between the maximum likelihood and least squares solutions for $\boldsymbol{\beta}$, we begin with the above definition of an exact model,

$$\begin{aligned} \mathbf{\Sigma}_a &= \mathbf{C} \\ &= \sigma^2_{s,s_0}\boldsymbol{\beta}\boldsymbol{\beta}^T + \mathbf{\Sigma}_{\boldsymbol{\epsilon}} \end{aligned} \tag{F.9}$$

Substituting the least squares solution for $\boldsymbol{\beta}$ from Equation (F.4) into Equation (F.9),

$$\mathbf{\Sigma}_a = \lambda \mathbf{U}\mathbf{U}^T + \mathbf{\Sigma}_\epsilon$$

Pre-multiply by $\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}$

$$\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{\Sigma}_a = \lambda\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}\mathbf{U}^T + \mathbf{\Sigma}_\epsilon{}^{\frac{1}{2}}$$

Post-multiply by $\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}$

$$\begin{aligned} \mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{\Sigma}_a\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}} &= \lambda\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}} + \mathbf{I} \\ (\widetilde{\mathbf{\Sigma}}_a - \mathbf{I}) &= \lambda\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}} \qquad \text{(F.10)} \end{aligned}$$

using Equation (F.6). Post-multiply by $\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}$

$$\begin{aligned} (\widetilde{\mathbf{\Sigma}}_a - \mathbf{I})\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U} &= \lambda\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U} \\ &= \lambda\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U}\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U} \qquad \text{(F.11)} \end{aligned}$$

Divide both sides of Equation (F.11) by $\left(\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U}\right)^{\frac{1}{2}}$

$$(\widetilde{\mathbf{\Sigma}}_a - \mathbf{I})\frac{\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}}{\left(\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U}\right)^{\frac{1}{2}}} = \lambda\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U}\frac{\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}}{\left(\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U}\right)^{\frac{1}{2}}} \qquad \text{(F.12)}$$

Let

$$\mathbf{Z} = \frac{\mathbf{\Sigma}_\epsilon{}^{-\frac{1}{2}}\mathbf{U}}{\left(\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U}\right)^{\frac{1}{2}}} \qquad \text{(F.13)}$$

Therefore

$$(\widetilde{\mathbf{\Sigma}}_a - \mathbf{I})\mathbf{Z} = (\lambda\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U})\mathbf{Z} \qquad \text{(F.14)}$$

which means that $\mathbf{Z}$ is an unit norm eigenvector of $(\widetilde{\mathbf{\Sigma}}_a - \mathbf{I})$ and $(\lambda\mathbf{U}^T\mathbf{\Sigma}_\epsilon{}^{-1}\mathbf{U})$ is the corresponding eigenvalue.

Substituting Equation (F.8) in Equation (F.8) we obtain,

$$(\widetilde{\mathbf{\Sigma}}_a - \mathbf{I})\widetilde{\mathbf{U}} = (\widetilde{\lambda} - 1)\widetilde{\mathbf{U}} \qquad \text{(F.15)}$$

Comparing Equation (F.14) with Equation (F.15),

$$\widetilde{\mathbf{U}} = \mathbf{Z} \qquad \text{(F.16)}$$

or

$$\widetilde{\mathbf{U}} = \left(\mathbf{U}^T\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\mathbf{U}\right)^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\mathbf{U} \tag{F.17}$$

and

$$(\widetilde{\lambda} - 1) = (\lambda\mathbf{U}^T\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\mathbf{U}) \tag{F.18}$$

Equations (F.17) and (F.18) give the relationship between the maximum likelihood and least squares solutions for $\boldsymbol{\beta}$.

We now verify that the maximum likelihood and least squares solutions are indeed identical using the relationships given by Equations (F.17) and (F.18). Substituting these equations into the maximum likelihood solution given by Equation (F.8),

$$\begin{aligned}
\boldsymbol{\beta}_{\mathrm{ML}} &= \frac{(\widetilde{\lambda}-1)^{\frac{1}{2}}}{\sigma_{s,s_0}}\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{\frac{1}{2}}\widetilde{\mathbf{U}}r \\
&= \frac{(\lambda\mathbf{U}^T\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\mathbf{U})^{\frac{1}{2}}}{\sigma_{s,s_0}}\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{\frac{1}{2}}\left(\mathbf{U}^T\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\mathbf{U}\right)^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\mathbf{U}r \\
&= \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s_0}}\mathbf{U}r \\
&= \boldsymbol{\beta}_{LS}.
\end{aligned} \tag{F.19}$$

Equal noise variance

The maximum likelihood and least squares solutions for $\boldsymbol{\beta}$ are also identical if the noise variance for each sensor is equal (homoscedastic noise variance), that is

$$\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} = \sigma_{\epsilon}^2\mathbf{I} \tag{F.20}$$

Under this condition, the two solutions to $\boldsymbol{\beta}$ are identical even if the model is not exact (i.e., $\boldsymbol{\Sigma}_a \neq \mathbf{C}$).

From Section E, the maximum likelihood solution for $\boldsymbol{\beta}$ is obtained by solving the equation

$$(\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\boldsymbol{\Sigma}_a\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}} - \mathbf{I})\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\boldsymbol{\beta} = \sigma_{s,s_0}^2\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-\frac{1}{2}}\boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}{}^{-1}\boldsymbol{\beta} \tag{F.21}$$

Substituting Equation (F.20) into Equation (F.21) gives

$$\begin{aligned}(\frac{\mathbf{\Sigma}_a}{\sigma_\epsilon^2} - \mathbf{I})\frac{\boldsymbol{\beta}}{\sigma_\epsilon} &= \sigma_{s,s0}^2 \frac{\boldsymbol{\beta}}{\sigma_\epsilon}\frac{\boldsymbol{\beta}^T\boldsymbol{\beta}}{\sigma_\epsilon^2} \\ (\mathbf{\Sigma}_a - \mathbf{\Sigma}_\epsilon)\boldsymbol{\beta} &= \sigma_{s,s0}^2 \boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{\beta} \qquad \text{(F.22)}\end{aligned}$$

Equation (F.22) is identical to Equation (4.30) in Section 4.5.1 (also see Equation (F.3) above) used for obtaining the least squares solution for $\boldsymbol{\beta}$. And the solution for $\boldsymbol{\beta}$ is given by

$$\boldsymbol{\beta} = \frac{\lambda^{\frac{1}{2}}}{\sigma_{s,s0}}\mathbf{U}r \qquad \text{(F.23)}$$

Hence the maximum likelihood solution to $\boldsymbol{\beta}$ obtained from Equation (F.22) is identical to the least squares solution given by Equation (F.4), where $\mathbf{U}$ and λ are the principal eigenvector and principal eigenvalue of $(\mathbf{\Sigma}_a - \sigma_\epsilon^2\mathbf{I})$. Let $\mathbf{U}_a$ and λ_a be the principal eigenvector and principal eigenvalue, respectively, of the data covariance matrix $\mathbf{\Sigma}_a$. Then

$$\mathbf{U} = \mathbf{U}_a \qquad \text{(F.24)}$$

and

$$\lambda = \lambda_a - \sigma_\epsilon^2 \qquad \text{(F.25)}$$

Substituting Equations (F.24) and (F.25) into Equation (F.23), both the least squares and maximum likelihood solutions simplify to

$$\boldsymbol{\beta} = \frac{(\lambda_a - \sigma_\epsilon^2)^{\frac{1}{2}}}{\sigma_{s,s0}}\mathbf{U}_a r. \qquad \text{(F.26)}$$

Appendix G

Conditional Covariance

Here we derive a general result concerning conditional covariance. Suppose we have

$$P(b|c)P(c) = P(b,c) \tag{G.1}$$

Then,

$$E\left[b\right] = E_c\left[E\left[b|c\right]\right] \tag{G.2}$$

where $E\left[.\right]$ denotes the expectation operator. However, for the covariance of b we have,

$$\begin{aligned}
\mathrm{cov}(b) &\equiv \int db \int dcP(b|c)P(c) \left\{ \left(b - E\left[b\right]\right)\left(b - E\left[b\right]\right)^T \right\} \\
&= \int db \int dcP(b|c)P(c) \left\{ \left[\left(b - E\left[b|c\right]\right) + \left(E\left[b|c\right] - E\left[b\right]\right)\right] \right. \\
&\quad \left. \left[\left(b - E\left[b|c\right]\left(E\left[b|c\right] - E\left[b\right]\right)\right)\right]^T \right\} \\
&= \int db \int dcP(b|c)P(c) \left\{ \left(b - E\left[b|c\right]\right)\left(b - E\left[b|c\right]\right)^T \right. \\
&\quad + \left(E\left[b|c\right] - E\left[b\right]\right)\left(E\left[b|c\right] - E\left[b\right]\right)^T \\
&\quad \left. + \left[\left(b - E\left[b|c\right]\right)\left(E\left[b|c\right] - E\left[b\right]\right)^T + \left(E\left[b|c\right] - E\left[b\right]\right)\left(b - E\left[b|c\right]\right)^T\right] \right\}
\end{aligned} \tag{G.3}$$

The last two terms of Equation (G.3) vanish using the result of Equation (G.2), to give

$$\begin{aligned}
\mathrm{cov}(b) &= \int dcP(c) \left\{ \int dbP(b|c)\left(b - E\left[b|c\right]\right)\left(b - E\left[b|c\right]\right)^T \right\} \\
&\quad + \int dcP(c) \left\{ \int dbP(b|c)\left(E\left[b|c\right] - E\left[b\right]\right)\left(E\left[b|c\right] - E\left[b\right]\right)^T \right\} \\
&= \int dcP(c) \left\{ \int dbP(b|c)\left(b - E\left[b|c\right]\right)\left(b - E\left[b|c\right]\right)^T \right\} \\
&\quad + \int dcP(c) \left\{ \left(E\left[b|c\right] - E\left[b\right]\right)\left(E\left[b|c\right] - E\left[b\right]\right)^T \right\} \\
&= E_c\left[\mathrm{cov}(b|c)\right] + \mathrm{cov}_c\left(E\left[b|c\right]\right)
\end{aligned} \tag{G.4}$$

where

$$E_c\left[\operatorname{cov}(b|c)\right] \equiv \int dcP(c)\left\{\int dbP(b|c)\left(b - E\left[b|c\right]\right)\left(b - E\left[b|c\right]\right)^T\right\} \tag{G.5}$$

and

$$\operatorname{cov}_c\left(E\left[b|c\right]\right) \equiv \int dcP(c)\left\{\left(E\left[b|c\right] - E\left[b\right]\right)\left(E\left[b|c\right] - E\left[b\right]\right)^T\right\}.$$

Appendix H

Fusion of Hyperspectral Images

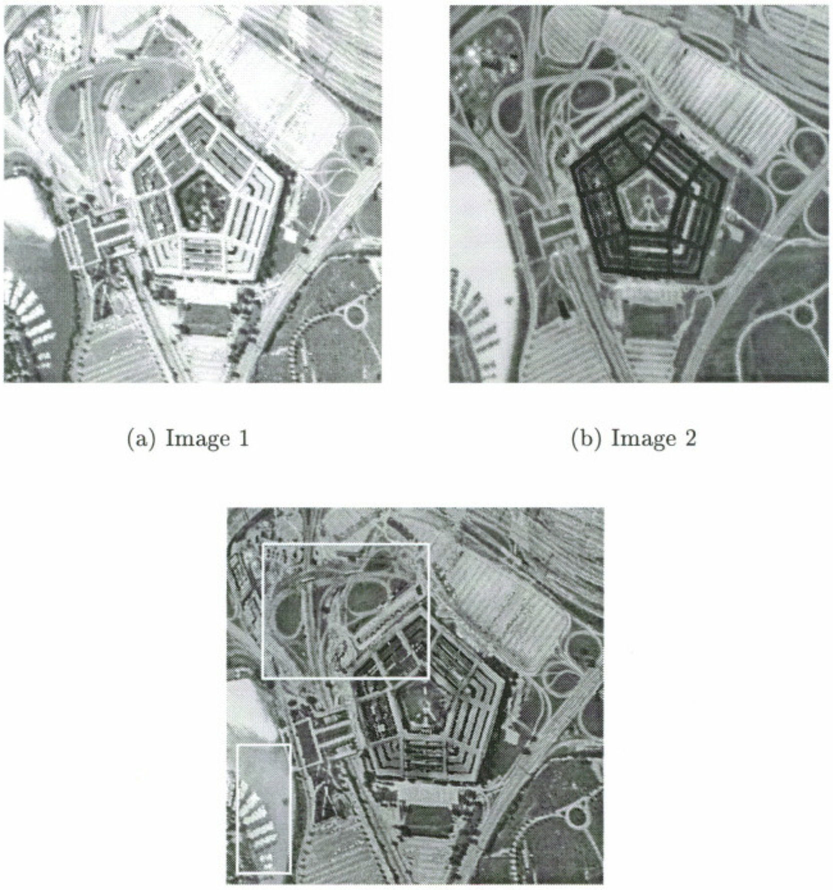
The probabilistic fusion approach can be applied to combine any kind of multisensor images. Figures H.1(a) and H.1(b) show two hyperspectral images[1] of the Pentagon. The images are captured from two different bands at different times of the day. These sensor images were fused using ML-fusion as described in Section 5.2.1. The experimental setup is given in Table H.1. The ML-fused image, shown in Figure H.1(c), has retained the complementary features from both sensor images. For example, see the marked box in the top half of the image. The roads from image 2 are clearly visible. The fused image has also preserved the contrast of local polarity reversed features. For example, see the lower right portion of the top box, and the box on the bottom left.

Figures H.2(a) and H.2(b) are another pair of hyperspectral images (also from AMPS), showing a land-mass interspersed with water. The sensor images have several common

[1] Images are from the Airborne Multisensor Pod System (AMPS). See http://www.amps.gov

Table H.1: Experimental setup to obtain the ML-fused image of Figure H.1

Size of images	300 × 300 pixels
Laplacian pyramid levels	9
Size of local analysis window	5 × 5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(h)
Noise variance	assumed equal in both sensors
Processing at borders	reflected hyperpixels to extend borders



(a) Image 1

(b) Image 2

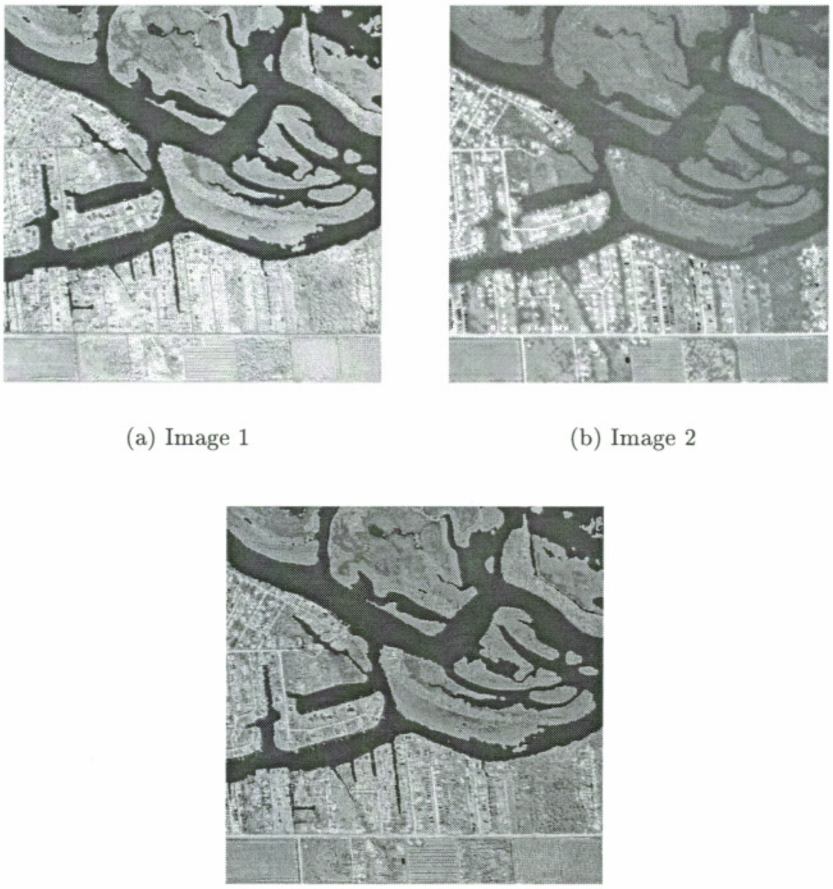
(c) ML-fusion

Figure H.1: Fusion of hyperspectral images

Table H.2: Experimental setup to obtain the ML-fused image of Figure H.2

Size of images	450 × 450 pixels
Laplacian pyramid levels	9
Size of local analysis window	5 × 5 hyperpixels
$\boldsymbol{\beta}$, $\boldsymbol{\alpha}$ computed at	each hyperpixel location
$\hat{s}$ computed at	each hyperpixel location
Constraint on sign r	shaded region in Figure 4.8(h)
Noise variance	assumed equal in both sensors
Processing at borders	reflected hyperpixels to extend borders

features, with the top half of image 1 showing higher contrast than that in image 2, and parts of the left and bottom portions of image 2 showing higher contrast than that in imagel. There are some complementary features due to water inlets and local polarity reversed features caused by buildings and roads. The images were combined using ML-fusion as in Section 5.2.1. The experimental setup is given in Table H.2. The ML-fused image in Figure H.2(c) has retained the image features with higher contrast from each sensor image. Local polarity reversed features have retained their contrast and complementary features have been preserved.



(a) Image 1

(b) Image 2

(c) ML-fusion

Figure H.2: Fusion of hyperspectral images

Appendix I

Computational Complexity

In this appendix we compare the computational complexity of our proposed fusion approach with that of existing techniques such as averaging and feature-selection in the pyramid domain. We point out the compromises that can be made in our fusion approach to reduce the computational burden.

Fusion by Averaging

Fusion by averaging (Section 2.3.1) operates on pixels of the images to be fused, requiring one operation (averaging) per pixel of the sensor images. For two sensor images of size $M \times N$, averaging would require $M \times N$ operations to obtain the fused image, making it the least expensive technique in terms of computations.

Fusion by Selection

Fusion by feature selection is computationally more expensive than averaging. To compare the computational complexity of fusion by selection, we assume that selection is performed in the Laplacian pyramid transform domain (Section 2.4). This technique has three distinct stages — pyramid construction, selection of pyramid coefficients based on the salience metric, and construction of the fused image from the fused pyramid. As in the case of averaging, we assume that there are two sensor images to be fused. Pyramid construction requires approximately 10 operations per pixel [12]. For two sensor images of size $M \times N$, this requires approximately $20 \times M \times N$ operations. The selection stage itself is computationally negligible but obtaining the salience metric can involve several computations. Let us assume that the salience metric is the energy (squared sum) of the pyramid coefficients in an area of 5 × 5 hyperpixels. This salience measure computation requires 50 operations per hyperpixel, and has to be performed at each hyperpixel location for each sensor pyramid. For an image of size $M \times N$ pixels, the Laplacian pyramid contains $(4/3) \times M \times N$ hyperpixels. Therefore, the total number of operations for the salience measure computation are approximately $100 \times (4/3) \times M \times N$. The last stage of applying the inverse transform to obtain the fused image requires approximately 10 operations per pixel, or a total of $10 \times M \times N$ operations.

Probabilistic Fusion

The probabilistic fusion approach described in this thesis also consists of three distinct stages — pyramid construction, estimation of the hyperpixels of the fused pyramid, and constructing the fused image from the fused pyramid. Again we assume that there are two sensor images to be fused and the Laplacian pyramid transform is used for fusion as described in Chapter 5. The first stage, pyramid construction, and the last stage, obtaining the fused image, require the same number of operations as in fusion by selection above — $20 \times M \times N$ and $10 \times M \times N$ operations respectively. In probabilistic fusion, the second stage consists of estimating the model parameters β and α, estimating the noise variances $\mathbf{\Sigma}_{\epsilon}$ and computing $\hat{s}$. For the maximum likelihood fusion rule, least squares estimation of β and α, and adaptive estimation of noise variance the combined computations amount to approximately 350 operations (multiplications and additions) per hyperpixel of the fused pyramid. That is a total of $350 \times (4/3) \times M \times N$ operations. This calculation does not include the computations that may be needed for motion compensation required for the adaptive estimation of the noise variance.

The above computational requirements pertain to the computation of one fused frame. For, video fusion, the number of computations required would depend upon the frame rate of the imaging sensors. For example, for sensors imaging at 30 frames per second, the number of computations would increase 30 fold. Although the number of computations required is large, it should be noted that most of these computations are independent of each other. For example the sensor pyramids can be constructed in parallel. Construction of each hyperpixel of the fused pyramid (i.e., $\hat{s}$) and the related estimation of parameters can be performed in parallel. It may be possible to achieve real time operation using multiple processors for performing these tasks in parallel.

Reducing the Computational Complexity of Probabilistic Fusion

The computational complexity of probabilistic fusion can be reduced in the following manner:

- Reducing the number of Laplacian pyramid levels of the sensor pyramids. However, using fewer Laplacian pyramid levels may adversely affect the resulting fused image. The tradeoff in using fewer Laplacian levels is explained in Section 5.2.2.

- Combining the sensor images by a direct application of local PCA as in Equation (5.5) of Section 5.2.1. Application of local PCA is computationally cheaper than the ML or MAP fusion rules, since it does not require the computation of the noise variance. This approach can work well when the noise variance in the sensor images is almost equal (note that it is exactly the same as ML and MAP fusion when the noise variances are equal). However, as illustrated in Appendix J, this approach may result in noisy fused images when the noise in one sensor image is higher than the other.

- Using the estimated affine parameters β and α over regions of several square hyperpixels rather than recomputing them for each hyperpixel location. Preliminary experiments using this simplification have yielded fusion results comparable to results using the full estimation. However, a thorough evaluation of this approach is essential to understand the differences.

- Using single frame noise estimation instead of multi-frame noise estimation. We believe that the multi-frame noise estimation technique described in Section 4.3.2 should provide better estimates than the single frame noise estimation described in Section 4.3.1. However, the single frame noise estimation is computationally cheaper since it does not need motion compensation. In our experiments the single frame noise estimation worked well when the assumptions made by the technique are valid. Results using single frame noise estimation are shown in Section 5.3.

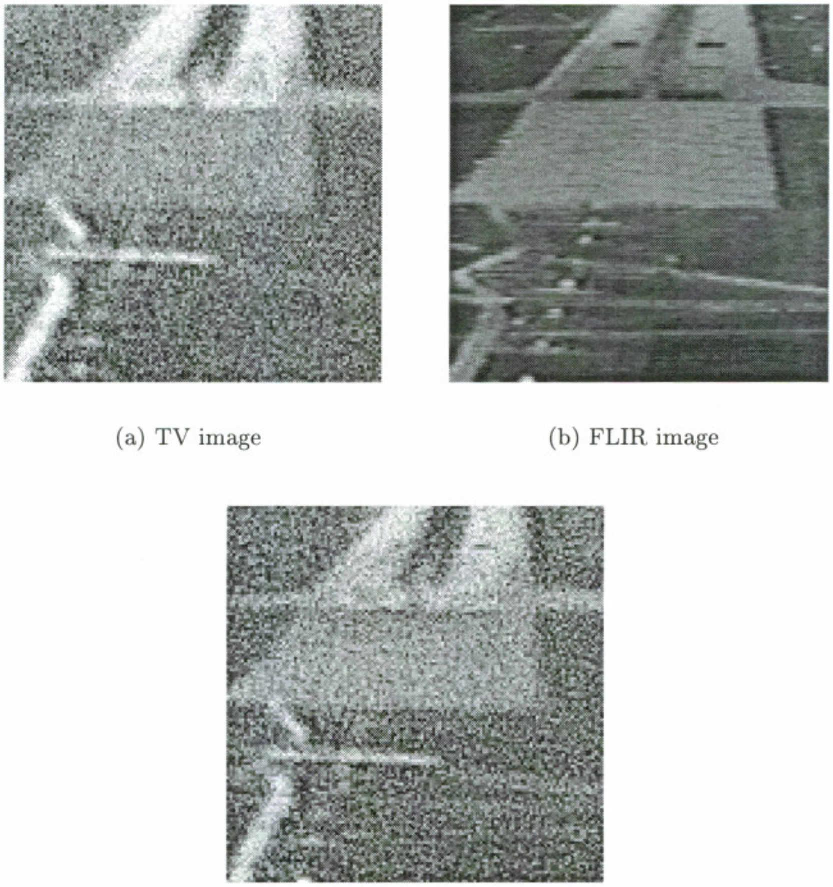
Discussion

The averaging, selection and probabilistic fusion techniques have increasing computational complexity. However, the results described in Chapter 5 indicate that our probabilistic fusion technique overcomes the drawbacks faced by averaging and selection methods. In this appendix we have outlined above, a set of approaches to reduce the computational complexity of the probabilistic fusion techniques. However, the tradeoff between reduction in computational complexity and the degradation of fusion results must be carefully evaluated in any practical implementation.

Appendix J

Local PCA on Noisy Sensor Images

Section 5.3 described the application of the probabilistic fusion rules for fusion of noisy sensor images. We now describe an experiment in which the noisy sensor images of Figure 5.5 are by the direct application of local PCA as in Section 5.2.1. Application of local PCA is a comparatively cheap substitute for the probabilistic fusion rules, since it does not require the computation of the noise variance. The sensor images are shown again in Figures J.1(a) and J.1(b). The result of fusion using local PCA is shown in Figure J.1(c). Each hyperpixel of the fused pyramid was obtained by a local PCA projection of the sensor hyperpixels in a 5×5 hyperpixel analysis window. The experimental setup is same as that in Table 5.4. Compare the fusion results of Figures 5.5(e) and 5.5(f) obtained by using the probabilistic fusion rules with the result of local PCA in Figure J.1(c). It is clear that the parameters corresponding to the noise variance in the probabilistic fusion rules play a significant role in fusion. They determine the appropriate weighting to be applied to each sensor image in order to obtain a reliable fused image, particularly when the sensor images are noisy.



(a) TV image

(b) FLIR image

(c) Local PCA

Figure J.1: Fusion of noisy images using local PCA (Original data from SVTD [9] project)

Biographical Note

Ravi Sharma was born on July 18, 1971 in Pune, India. After attending the prestigious D. G. Ruparel College of Science in Bombay, India, he went on to study at the Sardar Patel College of Engineering in 1988. Here, he obtained the Bachelor of Electrical Engineering degree with honors from the University of Bombay, in 1992. His undergraduate project in speech recognition at CMS Computers, Bombay, spurred his interest in speech processing.

On completion of his undergraduate studies, Ravi joined Digital Equipment Corporation, India, as a software engineer. However, his interest in speech processing led him to pursue higher studies. He joined the Oregon Graduate Institute (OGI) in fall 1994, where Dr. Misha Pavel got him interested in the fields of image and video processing. He then started pursuing his Ph.D. as a graduate research assistant at OGI.

He obtained his M.S. in Electrical Engineering from the Oregon Graduate Institute in 1997. During the summer of 1997, he worked at Intel Corporation researching and developing techniques for video frame interpolation. Since late 1998, Ravi has been working on the research and development of digital watermarking technology at Digimarc Corporation, Lake Oswego, Oregon.

Publications

1. R. K. Sharma, T. K. Leen, and M. Pavel, Probabilistic Image Sensor Fusion. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems 11*, pages 824-830. The MIT Press, Cambridge, MA, 1999.

2. R. K. Sharma, M. Pavel, and T. K. Leen, Multistream video fusion using local principal components analysis. In B. J. Andresen and M. Scholl, editors, *Infrared Technology and Applications XXIV*, pages 717-725. Proceedings of SPIE, volume 3436, October 1998.

3. R. K. Sharma and M. Pavel, Registration of video sequences from multiple sensors. *Proceedings of Image Registration Workshop*, Publication CP-1998-2068 53, NASA GSFC, November 1997.

4. M. Pavel and R. K. Sharma, Model-based sensor fusion for aviation. In J. G. Verly, editor, *Enhanced and Synthetic Vision 1997*, pages 169-176. Proceedings of SPIE, volume 3088, June 1997.

5. R. K. Sharma and M. Pavel, Multisensor image registration. In *1997 SID International Symposium, Boston, Digest of Technical Papers*, pages 951-954. Society for Information Display, Playa del Rey, CA, May 1997.

6. R. K. Sharma and M. Pavel, Adaptive and statistical image fusion. In *1996 SID International Symposium, San Diego, Digest of Technical Papers*, pages 969-972. Society for Information Display, Playa del Rey, CA, May 1996.

7. M. Pavel and R. K. Sharma, Fusion of radar images: rectification without flat earth assumption. In J. G. Verly, editor, *Enhanced and Synthetic Vision 1996*, pages 108-118. Proceedings of SPIE, volume 2736, May 1996.