FUSION OF SYNTHETIC AND INFRARED IMAGERY

Philippe Simard

Department of Electrical and Computer Engineering
McGill University, Montréal

March 1999

A Thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfilment of the requirements for the degree of
Master of Engineering

Abstract

This research presents the concept of an Enhanced and Synthetic Vision System (ESVS) that could help aircraft pilots see in low visibility conditions. The basic idea is that synthetic and infrared imagery would be fused to maximize image content. Three fusion algorithms were implemented and tuned for this particular purpose. Artificially generated infrared images as well as synthetic images were used to evaluate the algorithms' performance. An observer experiment, based on a typical search and rescue scenario, showed that image fusion leads to an increase in useful information. Different registration conditions were also simulated to investigate their effect on a potential ESVS.

Résumé

Cette recherche présente le concept d'un système de vision synthétique et accrue (SVSA) qui améliorerait la vision des pilotes d'aéronef dans de mauvaises conditions de visibilité. L'idée générale est de fusionner de l'imagerie synthétique et infrarouge pour maximiser le contenu des images. Trois algorithmes de fusion ont été développés dans ce but spécifique. Des images infrarouges artificielles ainsi que des images synthétiques ont été générées pour évaluer la performance de chaque algorithme. Une étude, basée sur un scénario typique de recherche et sauvetage, montre que la fusion d'images augmente l'information visuelle. Différentes conditions de correspondance des images ont aussi été simulées pour mesurer leur impact sur un potentiel SVSA.

Acknowledgements

First and foremost, I would like to thank my supervisor at CAE Electronics Ltd., Norah K. Link, for her continuous help and support throughout my research work. Thanks also to my manager, Ron V. Kruk, for his helpful suggestions. Thanks to my supervisor at McGill University, Frank P. Ferrie, for his advice.

Heart-felt thanks to my parents, whose encouragement throughout my degree was greatly appreciated. Finally, thanks to Bianca for her support and understanding during the course of this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1. Overview

Image fusion can be defined as the process of combining two or more source images into a single composite image with extended information content. It is proposed that fusion of infrared (IR) and synthetic images could be particularly useful to provide aircraft pilots with a visual reference to the ground in low visibility conditions. Both IR and synthetic images would provide visual information not normally available to the pilot in these situations, but both are subject to limitations. The IR sensor typically has small field of view or low resolution, it is subject to degradation due to weather effects, and it can be quite noisy. Synthetic images can be generated with both large field of view and high resolution. However, they will suffer real-world correlation problems due to the resolution of the polygonal representation and the resolution and accuracy of available source data. Objects may be displaced or not represented at all because of modeling compromises or because they were not present in the source data. Thus, an Enhanced and Synthetic Vision System (ESVS) which fuses synthetic and sensor images could provide a means for pilots to perceive the necessary visual information from the two different image sources, in one single image.

2. Motivation

Multisensor fusion (MSF) research was initiated in the late seventies for robotics applications where understanding of reflectance images of three-dimensional scenes was to be supported by data from a range sensor [**7**]. Sensor fusion technologies have also been driven by military applications such as target detection, recognition and tracking both in the tactical and strategic arenas [**11**]. Although the terms "multisensor fusion" and "multisensor integration" are often not distinguishable in the literature, it is useful to separate the more general issues involved in the integration of multiple sensing devices at the system architecture/control level, from the more restricted issues of the actual fusion of sensor data. Multisensor integration, as defined in [**6**], refers to the systematic use of the data provided by multiple sensors to assist in the accomplishment of a task by a system; multisensor fusion refers to any stage in the integration process where there is an actual combination (fusion) of different sources of sensor data into a single representational format. This thesis will deal primarily with the latter.

Sensor data fusion can take place at different levels of processing or representation [**1**] and generally, three levels are considered:

(i) High level fusion: sometimes called decision, report or classification level fusion, this level of fusion is performed on labels attached to objects/targets in each sensor processing stream. These labels are usually combined considering their associated confidence levels and uncertainties in sensor performance.

(ii) Intermediate level fusion: also known as feature, derived measurement, symbolic or information fusion, this level of fusion is generally preferred when very different sensors are used. Typically, features such as shapes, edge, orientation, color or range are derived from each sensor. Then, they are combined quantitatively and/or qualitatively.

(iii) Low level fusion: this level is also referred to as sensor, signal, data or pixel level fusion, and it is the lowest level at which data fusion can be performed.

Fusion at this level is usually done using schemes developed in the traditional signal and image processing fields.

Here, we are concerned with low level fusion. To our knowledge, this is the first time fusion of synthetic and IR imagery has been studied. However, prior work was done for fusing low-light visible CCD and IR imagery, which is very similar to our problem. Fusion algorithms were developed to either involve only operations on corresponding pixels or to be based on pyramid representations. Although low-light visible CCD is somewhat close to synthetic imagery, they are different in the sense that synthetic imagery is not affected by weather conditions and therefore always has high contrast. These methods also share the same goal of trying to preserve as much as possible of the two image sources. In our case, sensor imagery should be privileged when it shows good contrast because it is the modality that best represents the outside world.

One of the most interesting fusion algorithms for low light visible and infrared imagery based only on operations on pixels was developed by Waxman et al. [**13**][**15**][**14**]. Their method uses opponent processing in the form of feedforward center-surround shunting neural networks [**3**] to contrast enhance and adaptively normalize both visible and IR imagery. Both positive and negative polarity ("on" and "off") enhanced IR imagery is then combined with the enhanced visible imagery to create two single-opponent color-contrast grayscale images. The opponent processed visible and opponent-color (forming a set of three grayscale images) are then linearly combined to produce a fused image. Their algorithm was implemented for potential night driving on the road with a fusion display as a driving aid [**12**].

Toet et al. [**10**] have also developed a pixel-based fusion algorithm. It basically determines the unique component of both sources of imagery to enhance the representation of sensor-specific details in the final fused result. First, the common component of the two original input images is determined. Second, the common component is subtracted from the original images to obtain the unique component of each image. Third, the unique component of each image modality is subtracted from the image of

the other modality. This step serves to enhance the representation of sensor-specific details in the fused result. Finally, a fused image is produced by linearly combining the images resulting from the last step.

Both methods were analysed in [**9**] to verify if they in fact improve situational awareness. An observer experiment was performed to test if the increased amount of detail in the fused images could yield an improved observer performance in a task that requires situational awareness. Results have demonstrated that observers can indeed determine the relative location of a person in a scene with a significantly higher accuracy when they perform with fused images, compared to the original image modalities independently.

Other methods based on pyramid representations were also investigated. The basic strategy is to use a feature selection rule to construct a fused pyramid representation from the pyramid representations of the original data. Such an algorithm was recently developed by Li et al. [**4**] based on the wavelet transform. Basically, the wavelet transforms of the input images are appropriately combined, and the new image is obtained by taking the inverse wavelet transform of the fused wavelet coefficients. An area-based maximum selection rule and a consistency verification step are used for feature selection. Although such approaches seem to lead to extremely good results, it was decided that pyramid based algorithms would suffer from being almost non-implementable for a real-time system, which is a main concern for an ESVS.

This research makes the following contributions:

(i) It shows that fusion of synthetic and infrared imagery does in fact augment visual information in bad visibility conditions. More precisely:
   (a) A static analysis based on feature detection ranges proves that an ESVS could improve pilots' vision.
   (b) A dynamic analysis reveals that the dynamism (or flow) of the images can further increase the number of detected features.

(ii) It presents the experimental setup used for studying fusion. Simulated image sequences of a typical search and rescue scenario were generated in order to

investigate different registration problems as well as fusion algorithms' performance.

(iii) It describes three algorithms that were implemented and adapted for the fusion of infrared and synthetic imagery: the pixel averaging method which is the simplest method that can be used for image fusion, the TNO method developed by Toet et al. and the MIT method from Waxman et al.

3. Organization of the thesis

This thesis describes in detail the image fusion research that was conducted to evaluate fusion algorithms and to investigate the effect of sensor visibility and synthetic database correlation problems on fused images. It is organized as follows. Chapter 2 presents the experimental setup. It describes the way test images were generated and how the evaluation was conducted.

Chapter 3 describes the three algorithms that were selected for evaluation and how they were adapted for the fusion of synthetic and infrared imagery. The evaluation results are presented and discussed in Chapter 4 for both the static and dynamic evaluation. The algorithms' complexity is then analyzed in Chapter 5. Finally, conclusions are drawn in Chapter 6.

CHAPTER 2

Experimental Setup

1. Generation of Source Images

1.1. IG Selection. An image generator (IG), sometimes referred to as a computer graphics rendering system, generally takes objects' descriptions, information about illumination and pose and generates a life-like image of these objects. Such an IG was to be used to generate both the synthetic image and an (emulated) infrared image. The selected IG is an existing image generator emulator developed at CAE Electronics Ltd. as a tool to evaluate designs for future generations of IG hardware. Using such an image generator emulator for research had several advantages. First, one can control with great precision the content of its images. This allowed us to use the IG to represent both the real world sensor images and the synthetic world with absolute control over sensor image quality and the correlation between sensor and synthetic content. Also, these images could be generated at low cost on a standard PC compared to the costs that would be encountered with a real image generator. At the same time, however, the resources to generate in software what dedicated hardware could produce in real-time are considerable, and resource limitations required some trade-offs in the experimental design. These are described with the evaluation setup in Section 5.

CAE's IG is used to produce high quality color images that are projected in flight simulators. The IG processes a database of objects which are defined by polygons on

which textures and colors are applied and which are positioned in 3D space. Depending on the viewer position, the gaze direction and the field of view, the IG determines which objects are visible and projects them onto a 2D space. At the same time, it modifies the color at each position in the image depending on the position of light sources relative to the projected surface and on atmospheric fading. The resulting 2D projection forms the image used in flight simulation [**2**]. Figure 2.1 presents an example of a grayscale image rendered with the IG emulator.

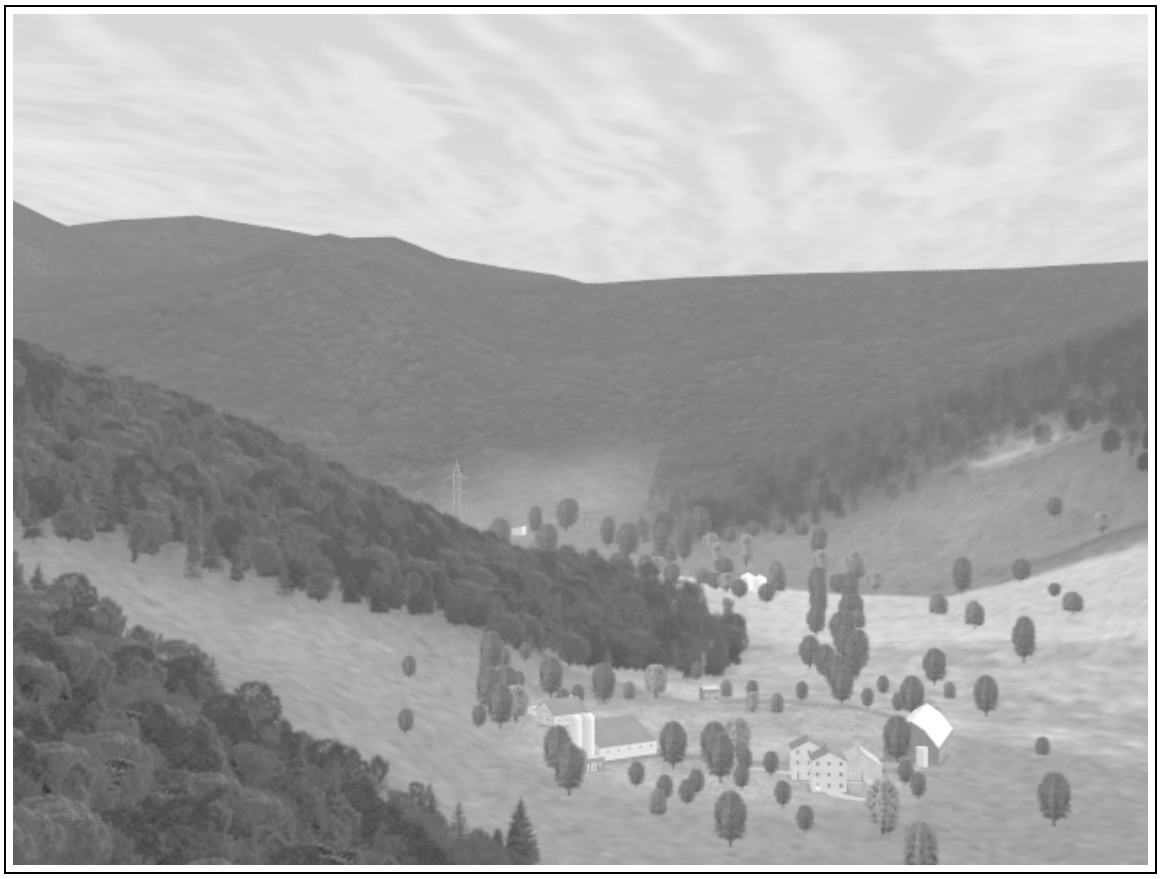

FIGURE 2.1. Example of a CAE's IG emulator image

1.2. Sensor Simulation Design. The sensor simulation design can be divided in two major parts. First, the image that would be produced in the infrared spectrum was simulated using the IG emulator. The IG emulator database colors were modified to give objects an infrared appearance. Also, the fading description

was tuned to simulate atmospheric effects. Second, the sensor electronics were emulated with a new software program. This program modifies the image produced by the emulator to simulate the limitations of a real sensor built from real hardware. The two steps involved in the sensor simulation are explained in the following paragraphs.

1.2.1. *Simulation of Infrared Objects.* Thermal imaging sensors extend human vision by making visible the light naturally emitted by warm objects. Most common imaging sensing systems operate in one or several of the visible, reflected IR, emitted IR or microwave portions of the spectrum. Note that there is an important distinction between reflected IR and emitted IR energy. Emitted IR is directly related to the sensation of heat; reflected IR is not. In this research, we are concerned with sensors that detect emitted IR.

Earth surface features emit radiation primarily in the thermal infrared wavelengths. This radiation can be expressed by

$$W = \epsilon \sigma T^4, \tag{2.1}$$

where

ϵ = spectral emissivity of the material,

σ = Stefan-Boltzmann constant,

T = absolute temperature (K) of the emitting material.

One must then know the temperature of an object as well as its emissivity to compute its emitted energy. When this emitted energy is computed, the corresponding color that will be displayed by the sensor screen can be calculated. As it is possible to modify an object's color in the emulator database, it was concluded that we could in fact just encode each object with the right color to get an "infrared look". This method had the advantage of being fast and easy, as it required no changes to the existing IG emulator.

As only one base color can be given to an object, only one temperature can be simulated at a time. However, the sensor simulation we are concerned with in this research did not have to be time variant (see the matrix of test conditions,

Section 3.1). Since object temperatures vary with the time of day and with the seasons, the presentation time was fixed at around 5 p.m. during a fall day at around 10^o Celsius.

A database with a fixed set of objects was selected. Each object (or part of an object) was assigned an appropriate temperature and an emissivity factor, and the corresponding emitted energy of each object was computed. Finally, energy across all objects was linearly mapped to the color intensity range available. Each polygon within an object was attributed this new infrared color. Note that the colors were mapped to produce a white hot image*. Also, because the emulator database has full color capability while an infrared image is grayscale, the red, green and blue components were all set to the new computed color intensity. Figure 2.2 shows the IR sensor-color tuned version of Figure 2.1.

1.2.2. *Simulation of Atmospheric Effects.* The atmosphere has a significant effect on the intensity and spectral composition of the energy recorded by a thermal system. The atmosphere intervening between a thermal sensor and the ground can increase or decrease the apparent level of radiation coming from the ground. The effect that the atmosphere has on a ground signal will depend on the degree of atmospheric absorption, scatter, and emission at the time and place of sensing.

Atmospheric absorption and scattering tend to make the signals from ground objects appear colder than they are, and atmospheric emission tends to make ground objects appear warmer than they are. Depending on atmospheric conditions during imaging, one of these effects will outweigh the other. This will result in a biased sensor output. Both effects are directly related to the atmospheric path length, or distance, through which the radiation is sensed. Thus, meteorological conditions have a strong influence on the form and magnitude of the atmospheric effects. Fog and clouds, depending on their water droplet size, can be essentially opaque to thermal radiation.

*Refer to Section 1.2.3 for a definition of a white hot image.

FIGURE 2.2. IR sensor color tuned image

To simulate atmospheric effects, the fade function of the emulator was used. This function provides a visibility factor for each layer of the atmosphere. We created a constant visibility range corresponding to an assumed value of percentage of humidity. While this did not necessarily represent a true computation of absorption and scatter, it served the desired purpose to reduce contrast between distant objects in a well-defined manner. An artificial ceiling with a very low visibility range was created to simulate clouds. Figure 2.3 shows a color tuned image in which atmospheric effects have been introduced.

1.2.3. *Simulation of Sensor Electronics.* At this point, a grayscale image has been produced to present the objects rendered as seen by a thermal sensor. However, the image quality must be degraded to represent the limitations of the sensor electronics. In other words, the image produced is simply too perfect to look like a real

FIGURE 2.3. Sensor image with atmospheric effects

sensor image. Figure 2.4 shows a simple schematic of one possible implementation of thermal sensor system electronics which was used for the simulation design. A thermal detector array, made from four vertically stacked detectors, is used to detect the thermal signature of the scene. These four detectors therefore scan four display lines at a time. A high gain amplifier then boosts the small signal variations that the detectors produce. A control loop, whose task is to correctly position the signal on the range of grayscale intensities that the display is able to present (by the way of signal gain and offset), is then reached (automatic gain control). Since the signal had to pass through amplification stages, some noise has been introduced and the need to reduce it is evident. For this purpose, a spatial filter, usually a pixel average filter, is applied on the entire frame to obtain a better signal-to-noise ratio. Also, on most thermal systems, one can choose to display an image that presents high thermal

energy emitters in light tones (white hot) or in dark tones (black hot). Black hot images are produced by feeding the signal through an inverter. The last features implemented are the brightness and contrast controls available to the user through the display. These controls provide better image rendering and may serve to accentuate particular aspects of the scene.

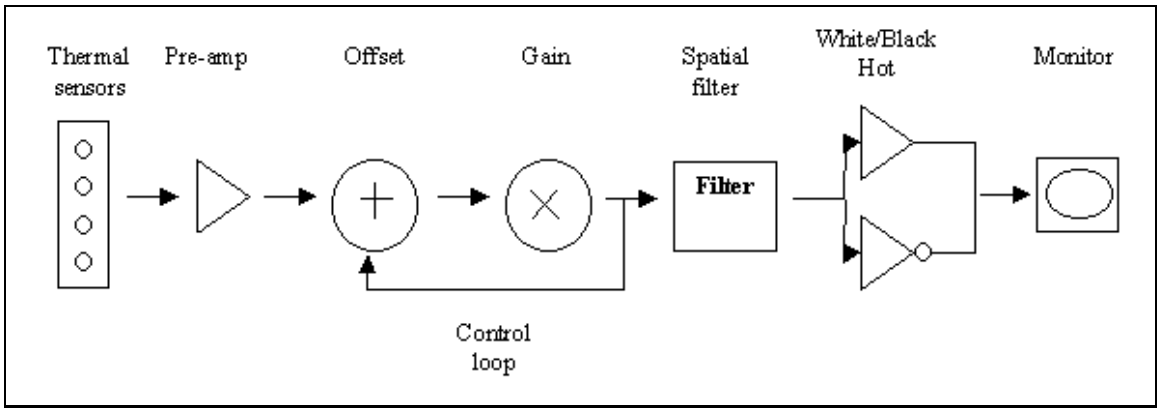

FIGURE 2.4. Simple block diagram of the implementation of a thermal sensor system

From the simple sensor schematic presented in Figure 2.4, it is evident that five sensor hardware limitations had to be simulated: noise, row mismatch, blurring, white hot/black hot and brightness/contrast. These have been implemented in order as described below. First, the noise that can be expected in the signal produced by the detectors is simulated by filling a dummy buffer with random intensity values and adding it to the image pixel by pixel. As a consequence, each pixel intensity is increased by a random intensity value in the following way:

$$I_n = I_{original} + I_{noise}, \tag{2.2}$$

where

I_n = new pixel intensity with noise,

$I_{original}$ = original pixel intensity,

I_{noise} = noise random intensity.

The noise was generated with a pseudo-random function using a uniform distribution.

Second, one can expect some differences in the signals driven by the four detectors. Since four lines of the displayed image are scanned at the same time with these detectors, this usually leads to a repeated pattern of intensity mismatch between rows in the image that is referred to as sensor mismatch. To simulate this effect, a temporary image buffer is first created with a pattern of four lines, each of the four containing a randomly selected offset. This offset buffer is then added to the original image. Thus, the new pixel intensity of a given line is

$$\mathrm{I}_{nm} = \mathrm{I}_{n} + \mathrm{I}_{offset}, \tag{2.3}$$

where

I_{nm} = new pixel intensity with sensor mismatch,

I_{n} = pixel intensity with noise,

$\mathrm{I}_{\mathrm{offset}}$ = line offset intensity.

Third, most infrared sensors integrate a noise removal technique to improve the signal to noise ratio. The most common technique employed is local pixel averaging (blurring), and in the simulation, a 3×3 kernel of equal weights is passed over the image. Each pixel is attributed the average intensity of itself and of its eight nearest neighbors. Fourth, as explained in a previous section, a sensor system can display white as either high or low level of emitted energy. In the white hot mode, high thermal energy emitters are displayed in white tones and low energy emitters in black tones. Conversely, in the black hot mode, low thermal energy emitters are rendered in white tones and high energy emitters in black tones. This mode of operation has been implemented by inverting the white hot color tuned image, so that each pixel in the black hot mode is given the intensity

$$\mathrm{I}_{blackhot} = \mathrm{I}_{maximum} - \mathrm{I}_{whitehot}, \tag{2.4}$$

where

$I_{\text{black hot}}$ = pixel intensity in black hot mode,

I_{maximum} = maximum possible pixel intensity (e.g. in 8 bit, 255),

$I_{\text{white hot}}$ = pixel intensity in white hot mode.

Finally, a thermal image can be tuned to different levels of brightness and contrast. Brightness can be seen as a global offset, positive or negative, applied to each pixel intensity of an image. On the other hand, contrast can be seen as a global gain factor applied to each pixel intensity of an image. These are implemented as follows:

$$I_{brightness} = I_{original} + \text{Offset} \tag{2.5}$$

$$I_{contrast} = I_{original} \times \text{Gain}, \tag{2.6}$$

where

$I_{\text{brightness}}$ = pixel intensity with brightness,

I_{contrast} = pixel intensity with contrast,

I_{original} = original pixel intensity.

Offset = offset value,

Gain = gain value.

An example of a simulated infrared sensor image with low visibility, sensor mismatch, noise removal, and brightness and contrast adjustments is shown in Figure 2.5.

The reader can refer to [**16**] [**5**] [**8**] for further information on infrared sensors.

2. Flight Path Description

In order to study image fusion in a real flight situation, a flight path has been modeled to simulate a typical approach in to a crash site in hilly, forested terrain. It consists of an approach descending from 500 ft[†] to 0 ft, allowing us to examine the transition from primarily SVS (geographic database) to primarily EVS (imaging sensor) assisted flight. The terrain over which it passes contains several key features.

[†]1 ft = 0.3048 m

FIGURE 2.5. Simulated sensor image

Basically, it is a valley within a region of rolling hills and small mountains. There is farmland with some individual trees and farm buildings on the valley floor, and the hillsides are forest covered. An intermediate ridge between the start point and the crash site needs to be avoided and thus detected as soon as possible. There is a clearing on a hillside at the end of the valley in which a crashed plane has been introduced in the sensor images to simulate a search and rescue situation. The clearing is on rising ground ending in a ridge with a saddle just above the crash site, and there is an alternate escape route to the left. This clearing constitutes the end point of the flight path. In addition to a crashed plane, a fire that could have be initiated by the plane occupants to aid detection by a search-and-rescue (SAR) crew has been modeled in the sensor images. The flight path and terrain profiles are depicted in Figure 2.6.

	Sensor White Hot	Sensor Black Hot
High Vis. - Sensor Undegraded	**Sensor Condition 1**	**Sensor Condition 2**
Medium Vis. - Sensor Slightly Degraded	**Sensor Condition 3**	**Sensor Condition 4**
Low Vis. - Sensor Severely Degraded	**Sensor Condition 5**	**Sensor Condition 6**

TABLE 2.1. Sensor conditions

The aircraft starts with 0^o pitch at an altitude of 500 ft above the end point and at about 6000 ft out. It follows a 5^o slope for 5500 ft, and for the last 500 ft, the inclination of its trajectory becomes 2^o. The aircraft stops at a 30 ft altitude over the end point, then pitches up 5^o to simulate the hover condition. While pitch angles throughout the flight path are not completely realistic, they were deemed to be sufficient for the purposes of the experiment.

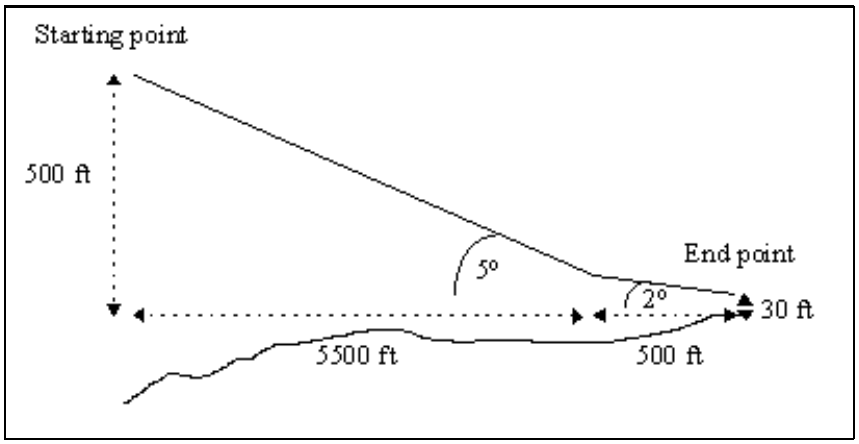

FIGURE 2.6. Flight path and terrain profiles

3. Matrix of Test Conditions

3.1. Sensor Conditions. Sensor modality and weather (visibility) conditions were varied to produce six conditions of sensor image quality to test different fusion situations. These are listed in Table 2.1.

The three visibility ranges are 3 nm[‡], 1.5 nm and 0.5 nm for the high, medium and low visibility respectively. They correspond to high humidity percentages of above 90%.

[‡]1 nm = 1 nautical mile = 1.8519 km

Registration Condition 1	Identical database and viewpoint to sensor image.
Registration Condition 2	Aircraft position sensor noise introduced on flight path.
Registration Condition 3	Missing objects, object position offsets.
Registration Condition 4	Global database offset (extreme).
Registration Condition 5	Decreased terrain resolution (intermediate).
Registration Condition 6	Local terrain elevation errors (extreme).
Registration Condition 7	Global database offset (intermediate).
Registration Condition 8	Decreased terrain resolution (extreme).
Registration Condition 9	Local terrain elevation errors (intermediate).

TABLE 2.2. Registration conditions

3.2. Registration Conditions. Inconsistencies can be expected between the content of synthetic images generated from a stored database and the IR images in an ESVS. We refer to these inconsistencies as registration problems. By affecting the relative geometry, or registration, of the sensor and synthetic images, these conditions would affect the fusion process. Nine different registration conditions were defined to study separately the different effects that misregistration between the two image sources would have on fusion algorithms. The conditions are listed in Table 2.2.

Registration conditions 1, 2, 4, 6, 7 and 9 consist of generating the synthetic images with the same database as that of the sensor except for local modifications as follows:

- In all conditions, the crash site was removed.
- In registration condition 2, some noise has been added to the viewpoints to simulate the noise that can be expected from the aircraft navigation system and attitude indicators, which will be used to drive the synthetic viewpoint in ESVS. The added error varies from -15m to +15m.
- For conditions 6 and 9, the intermediate ridge has been raised by 15m and by 45m respectively. This simulates local terrain elevation errors that can be found in digital elevation maps from which the virtual terrain is constructed.

While the database has not been changed for conditions 4 and 7, it has been rotated and translated to simulate errors in the conversion of source data to the IG

coordinate system. The values used are 1.0^o rotation and 5m translation for the extreme offset, and 0.5^o rotation and 3m translation for the intermediate offset (these would be typical in an ESVS). While other registration conditions have been simulated by generating different versions of synthetic images, conditions 4 and 7 were applied to the sensor images to create a sense of the displacement of sensor images relative to the synthetic when route planning is executed against the synthetic image.

Significant database remodeling was effected for conditions 3, 5 and 8. The forest with 3D trees has been replaced by a forest canopy since it is most likely that such 3D trees will not be rendered in real-time because of their complexity. Also, several objects have been removed from the original database, while the remaining ones have been rotated and translated to simulate errors in source cultural data[§] and synthetic database modeling compromises. Registration conditions 5 and 8 represent the databases generated with a terrain of decreased resolution, i.e. using fewer polygons (by a factor of 4 and 8 respectively). Note that our full resolution terrain is constructed from a digital elevation map (DEM) that has a grid spacing of 40m. As a consequence, conditions 5 and 8 represent a grid spacing of 160m and 320m respectively.

4. Sensor and Synthetic Image Configuration

The resolution and field of view (FOV) of the sensor and synthetic images were set to reflect the expected fields of view of an ESVS. The sensor resolution was set to 640 horizontal by 480 vertical. The FOV was defined as 40^o horizontal by 30^o vertical. Although most sensors do not offer such a wide FOV, it was suitable for the purpose of the research to have images with high content. Since in the final system the synthetic image will cover a larger field of view, it was set for the experiments to 50^o horizontal by 40^o vertical. Because we needed a perfect pixel to pixel registration, the resolution was calculated from the previous data: 816 horizontal by 656 vertical.

[§]Cultural data include objects such as trees, buildings, roads, etc. that would have been surveyed for the creation of a database.

Note that the fused portion of the output image has the same FOV and resolution as that of the sensor. Figure 2.7 presents the image configuration.

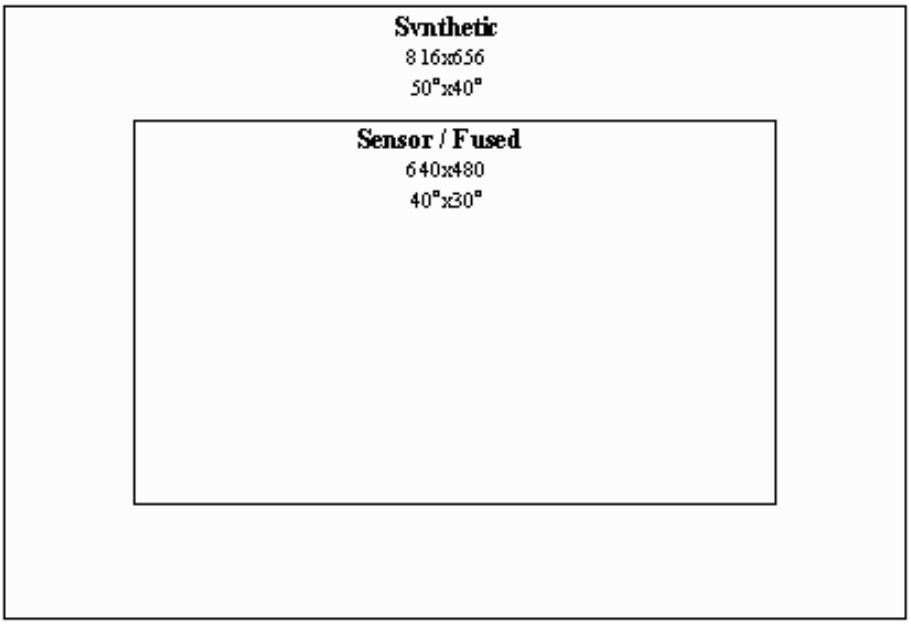

FIGURE 2.7. Image configuration

5. Evaluation

Two types of evaluation were used to study image fusion: a static and a dynamic evaluation. Both shared the same goal of determining if the fusion of synthetic and infrared imagery would improve information content. The approach taken was to show subjects images along the flight path and ask them to assess if certain features were visible or not. As it will be further explained in the following paragraphs, still images were used to estimate the impact of sensor and registration conditions on fusion. Also, a 30 Hz sequence was generated to evaluate the dynamic performance of the algorithms. We refer to these two types of evaluation as static and dynamic respectively.

Again, the static evaluation was used to reduce the number of images required in the experiment. A 30 Hz sequence would require approximately 1900 images per test

run, whereas for the static evaluation 16 snapshots were used per run. This reduced the cost of the experiment by:

- Reducing the computing time and resources required to generate and store the test sequences (each image required about 1 hour to generate and was 0.7 Mb when stored as a compressed file).
- Reducing the time required to run each subject through the test sequences.

When the static evaluation was completed, one representative condition was selected for implementation at 30 Hz to evaluate the dynamic behavior of the algorithms. Twenty Pentium II - 266 MHz PCs, running 24 hours per day for 20 days were used to generate the images for this part of the experiment.

5.1. Static Evaluation Setup. 15 sample points along the flight path were chosen. For the first section at 5^o glide slope (see Figure 2.6), snapshots were taken approximately every 500 ft (ground range). For the last section, the distance in between consecutive sample points was decreased to 50 ft. For all image viewpoints, the values of the roll, pitch and yaw of the aircraft (or observer) were set to zero. Also, the final point has been duplicated with the viewpoint pitched up 5^o. The sequence is therefore composed of 16 images along the flight path. All the sensor conditions were generated as well as all the registration conditions. Together with the three fusion methods (see Chapter 3), this led to 162 different sequences of fused images (6 sensor conditions $\times$ 9 registration conditions $\times$ 3 fusion methods).

To verify whether or not image fusion leads to an increase in useful information presented to the pilot, a set of criteria was developed which consisted of assessing the visibility of given features and in verifying if there were conflicts between objects or terrain features due to registration problems. The criteria are listed in Table 2.3. Recall that the flight path traverses a box canyon, taking the aircraft over a farm and an intermediate ridge to the crash site, which is situated in a clearing on rising ground.

Criterion 1	Mid ridge visibility
Criterion 2	Ridge behind crash visibility
Criterion 3	Far peaks visibility
Criterion 4	Clearing visibility
Criterion 5	Clearing obstacles visibility
Criterion 6	Crash site visibility
Criterion 7	Terrain conflicts
Criterion 8	Object conflicts

TABLE 2.3. Static evaluation criteria

These criteria represent critical features that a pilot would need to detect as soon as possible. For our purposes, the subject was required to view the 16 images along the flight path and to note at which image the features first became visible (criteria 1 to 6) and which images contained terrain or object conflicts (criteria 7 and 8). Subjects were asked to indicate what terrain or object conflicts were observed (for example, overlapping ridges that did not match, creating confusion as to where the terrain actually was).

Three subjects were asked to evaluate each of the 162 sequences as well as each of the pure sensor conditions (non-fused). The pure sensor conditions were used to establish a baseline against which to compare the overall fusion performance. The sequences were not randomized for the evaluation process. It was judged that the randomization would not have substantially changed the results since the three subjects were already familiar with the sequences.

5.2. Dynamic Evaluation Setup. When the static evaluation was completed, one representative condition was selected to implement at 30 Hz for an evaluation of the dynamic behavior of the algorithms. Because of the resources required to generate the images, only one sensor and one synthetic sequence were produced. Sensor condition 3 (white hot, medium visibility) and synthetic condition 8 (low terrain resolution) were selected to reflect the most likely conditions under which an ESVS might be used. All three fusion methods were used to generate fused sequences and to further test the algorithms in real time.

To represent an aircraft's approach, the speed was set to 75 knots[¶] at the starting point and decreased to 0 knots at the end point. While the speed profile was not completely realistic, the speeds generated were typical of an approach, and this provided a reasonable length of time to view the approach and the dynamic behavior (the resulting approach run was 65 seconds). Roll, pitch and yaw values were adjusted to produce typical small aircraft movements. As in the static evaluation, the pitch value was smoothly incremented during the end of the sequence so that the viewpoint for the final scene presents is pitched up by 5^o.

[¶]1 knot = 1.8519 km/h

CHAPTER 3

Fusion Methods

Three fusion methods have been investigated and modified to fit the ESVS requirements. In this particular application, a good fusion method should include as much sensor information as possible because it more closely represents the reality. The synthetic image should therefore be used when the sensor content is very low, and to represent the scene outside of sensor coverage. The pixel averaging method was selected for study as the simplest way to fuse two images. The two other algorithms implemented for evaluation were the TNO method (developed in The Netherlands at the TNO Human Factors Institute) and the MIT method (developed at MIT Lincoln Labs). The latter two algorithms were originally developed for real-time (or near real-time) fusion of low-light visible images with IR images.

The three methods are further described in the following paragraphs. Examples center around the sample sensor image in Figure 3.1 (white hot, low visibility) and a corresponding synthetic image in Figure 3.3 (fully registered terrain, polygonal forest canopy, missing or displaced objects).

1. Pixel Averaging Method

The pixel averaging method assigns the average of each corresponding pixel from the sensor frame and the synthetic frame to the fused image. However, pure pixel

averaging presents major problems when fusing sensor and synthetic images. An equal contribution of both images can lead to one of the two following situations:

(i) A poor quality (low content) sensor image obscures synthetic data without adding any value to the image.

(ii) A good quality sensor image is obscured by uncorrelated synthetic data.

The algorithm was therefore modified to take into account the quality of the sensor image. The simple average has been transformed to a weighted average that can provide more weight to high-quality sensor (i.e. high content/contrast) images and less weight to low content images. The problem then becomes one of being able to measure the quality of the sensor image. Metrics to estimate the quality of an image can be formulated in either the spatial or frequency domains. Operations in the frequency domain were not investigated because their complexity represents an obstacle to real-time performance. The content of the sensor image was therefore evaluated by calculating the intensity standard deviation, under the assumption that greater deviations result from more scene content present in the sensor image while lower deviations would result from sensor images obscured by atmospheric effects or noise. Therefore, the relative weight applied to a sensor image in the average would be directly proportional to its intensity standard deviation. In order to compute the sensor weight corresponding to a given standard deviation, we apply a linear function

$$\text{Weight}_{sensor} = \text{Offset} + \text{Gain} \times \text{Standard deviation}, \tag{3.1}$$

where the offset and gain values are tuned to the overall performance of the sensor. The offset and gain values used for the experiment were tuned empirically by taking the minimum and maximum standard deviations from several sample images to create a linear ramp of effective sensor weights. In our experimental setup, a very low contrast image measure was given a sensor weight of 0.5 while a very high contrast image was weighted at 1.0. Images with intermediate contrast measures were assigned weights using the linear function presented in Equation 3.1. While this scheme produces satisfactory results, sensor images can have interesting regions, i.e.

regions of high scene content, while other regions are obscured. Such a sensor image is shown in Figure 3.1.

FIGURE 3.1. Sample sensor image with both high and low content portions

Note that the top part of the image contains only noise while the bottom part has interesting features such as individual trees and a ridge line. Using the same relative weights throughout the image brings out the interesting features of the sensor image in the foreground but brings out too much of the obscured background and therefore decreases the contrast of the synthetic features in the background where they are most needed. Superior performance was obtained by dividing the image into subparts. The contrast measured for these subparts was used to calculate sensor weights for the center of each region, and the weight at each pixel was calculated as a bilinear interpolation of the region-center weights. This interpolation eliminated boundary problems where adjacent tiles had significantly different weights. In tuning the algorithms prior to the experiment, a six by six grid seemed to offer the best compromise between large regions that could discard important features of the sensor

image and small regions for which the standard deviation would not be meaningful. This particular sensor weights setup is shown in Figure 3.2.

W_{11}	W_{12}	W_{13}	W_{14}	W_{15}	W_{16}
W_{21}	W_{22}	W_{23}	W_{24}	W_{25}	W_{26}
W_{31}	W_{32}	W_{33}	W_{34}	W_{35}	W_{36}
W_{41}	W_{42}	W_{43}	W_{44}	W_{45}	W_{46}
W_{51}	W_{52}	W_{53}	W_{54}	W_{55}	W_{56}
W_{61}	W_{62}	W_{63}	W_{64}	W_{65}	W_{66}

FIGURE 3.2. Sensor weights setup

Finally, in order to further improve the contrast of the sensor-synthetic fused frame, the fused image was remapped to use the full range of available display intensities. The minimum and maximum intensities of the image are used to compute a gain and offset to linearly remap all pixel intensity values. This improves the contrast of the fused image without changing the intensity distribution, resulting in a more natural appearance than that achieved with other contrast enhancement techniques such as histogram equalization.

The pixel averaging method can be summarized as follows:

1) Find the intensity standard deviation of each of the sensor image subparts.
2) Compute the sensor weight of each tile based on the deviation value and using the linear function presented in Equation 3.1.
3) Compute the sensor image weight W_{ij} for a given pixel (i,j) as a bilinear interpolation of the four nearest neighbor region weights.

4) Perform a weighted average on each pixel in the two source images to produce the fused image:

$$\text{Fused}(i,j) = \text{W}_{ij} \times \text{Sensor}(i,j) + (1 - \text{W}_{ij}) \times \text{Synthetic}(i,j) \tag{3.2}$$

5) Map the resulting image onto the whole range of available intensities to maximize contrast.

It is important to emphasize that the IR sensor would have to be set in manual gain mode so that when the image contains for example, only noise, the standard deviation remains low. This would allow the sensor contribution to the fused image to be low. The autogain mode should therefore not be used on the IR sensor for the algorithm to work properly.

To illustrate the performance of the algorithm, a representative synthetic image from registration condition 3 (Figure 3.3) has been fused with a corresponding sensor image from sensor condition 5 (Figure 3.1). The result is presented in Figure 3.4. Notice that the fused image contains the visible features of the sensor image while not obscuring the synthetic objects where only sensor noise was present.

2. TNO Method

The TNO method was developed in The Netherlands at the TNO Human Factors Institute by Toet et al. to fuse low visible CCD camera images with infrared sensor images. The algorithm as developed by the TNO Institute was adapted and tuned for the purpose of sensor-synthetic fusion to account for sensor image quality and synthetic scene content. The method can be described as follows. (Note that the algorithm operates strictly on the intensities of individual pixels.)

First, the common component of the two original input images is determined. This is simply implemented as a local minimum operator. It is therefore given by:

$$\text{Common}(i,j) = \text{Sensor}(i,j) \cap \text{Synthetic}(i,j) = \text{Min}\ \{\text{Sensor}(i,j), \text{Synthetic}(i,j)\} \tag{3.3}$$

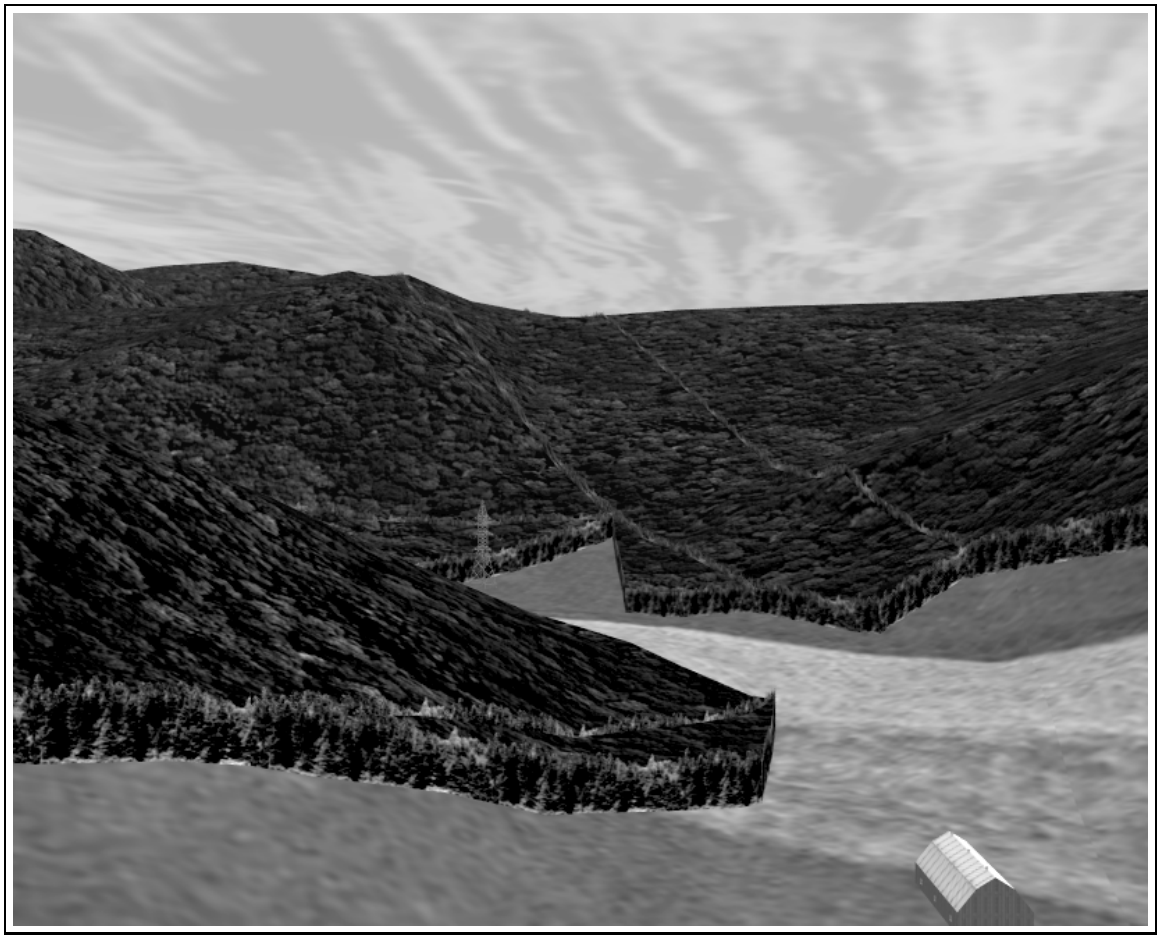

FIGURE 3.3. Synthetic image

Next, the common component is subtracted from the original images to obtain the unique component (characteristic) of each image. This unique component represents the details that are unique to each image. This operation is then:

$$\text{Sensor}(i,j)^* = \text{Sensor}(i,j) \text{ - } \text{Common}(i,j) \tag{3.4}$$

$$\text{Synthetic}(i,j)^* = \text{Synthetic}(i,j) \text{ - } \text{Common}(i,j) \tag{3.5}$$

The unique component of the synthetic image is then subtracted from the sensor image to enhance the representation of sensor specific details in the final fused result. Note that the subtraction result is bounded to zero in the case it is negative. The

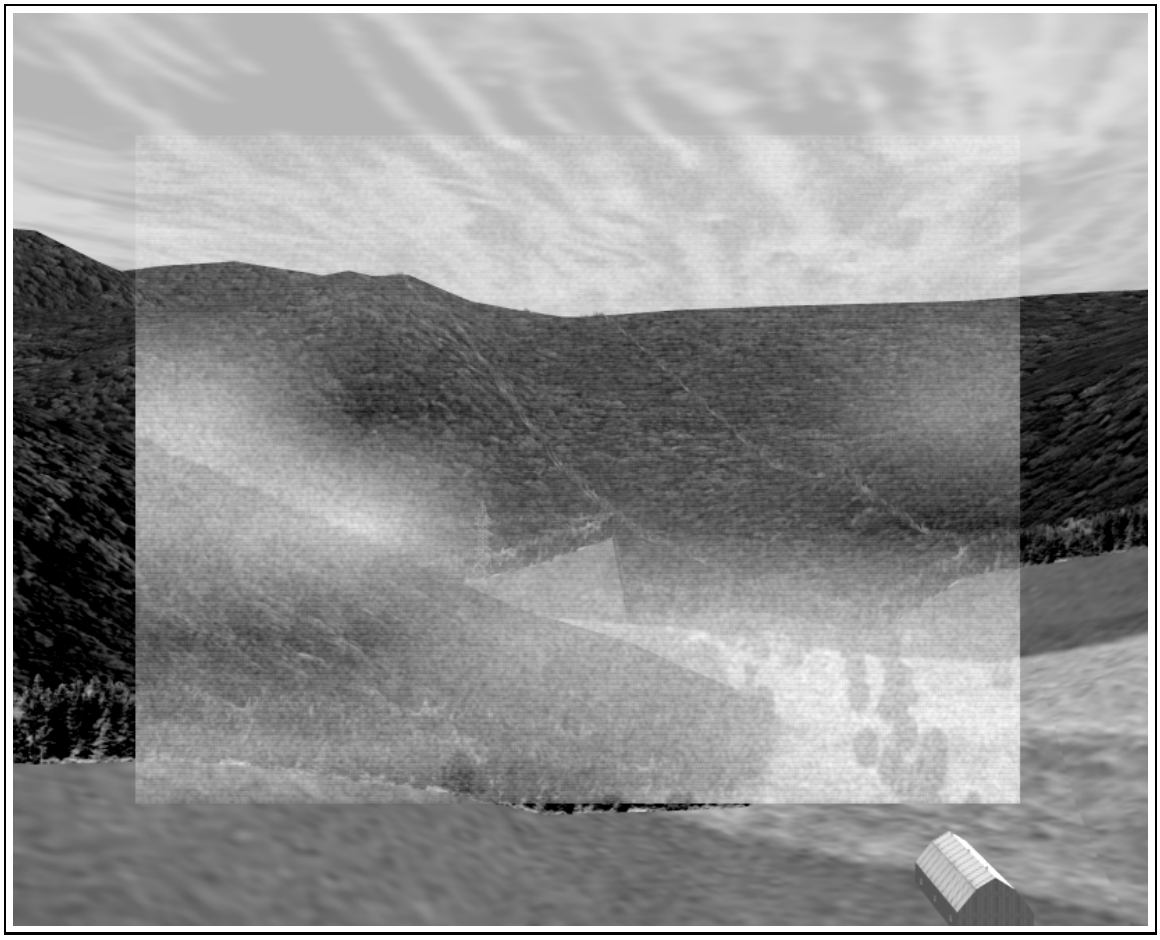

FIGURE 3.4. Fused image using pixel averaging method

equation used is:

$$\text{Sensor}_{enhanced}(i,j) = \text{Max } (0, \text{Sensor}(i,j) \text{ - Synthetic}(i,j)^*) \tag{3.6}$$

Finally, a weighted average is computed with the original synthetic image and the enhanced sensor image. The characteristic components of the individual images can be further underscored by including into the weighted average the absolute value of the difference between the two characteristic images. This difference channel is computed with:

$$\text{Difference}(i,j) = \text{Abs}(\text{Sensor}(i,j)^* \text{ - Synthetic}(i,j)^*) \tag{3.7}$$

The final image is thus calculated with:

$$\begin{aligned} \text{Fused}(i,j) &= \text{Weight}_{sensor} \times \text{Sensor}_{enhanced}\,(i,j) \\ &+ (1.0 - \text{Weight}_{sensor} - \text{Weight}_{diff}) \times \text{Synthetic}(i,j) \\ &+ \text{Weight}_{diff} \times \text{Difference}(i,j) \end{aligned} \tag{3.8}$$

As in the pixel averaging method, both the sensor and the difference channel weights applied in the average were determined from the standard deviation of intensities in the original sensor image. More weight was given to the sensor enhanced image and to the difference channel image for higher deviations. Table 3.1 shows the values that were used in the experiments to calculate the sensor and difference channel weights. The weights were calculated for sub-windows and interpolated in the same manner as in the pixel averaging method.

	Sensor weight	**Difference weight**
Very high std. deviation	0.75	0.15
Very low std. deviation	0.50	0.00
Intermediate std. deviation	linear function	linear function

TABLE 3.1. Sensor and difference weights applied

The modified TNO method can be summarized as follows:

1) Find the intensity standard deviation of each sensor image subparts.
2) Compute sensor and difference weights of each tile based on the deviation value using the linear function presented in Equation 3.1.
3) Compute the sensor and difference weights at an individual pixel as a bilinear interpolation of the four nearest neighbor region weights.
4) Find the common component of the two images (Equation 3.3)
5) Find the characteristic component of the sensor image (Equation 3.4).
6) Find the characteristic component of the synthetic image (Equation 3.5).
7) Subtract the synthetic characteristic from the original sensor image to produce an enhanced sensor image (Equation 3.6).

8) Subtract the sensor characteristic from the synthetic characteristic to produce a difference channel (Equation 3.7).
9) Produce the fused image with the weighted average of the synthetic, enhanced sensor and difference images (Equation 3.8).
10) Map the resulting image onto the whole range of available intensities.

An example of an image fused with the TNO method is illustrated in Figure 3.5. Note that the input images are the same as before.

This algorithm presents similar results to the averaging method. However, one major advantage of this method is that white objects in the synthetic image can survive the fusion process. This would allow certain important objects (e.g. objects modeled after range sensor returns) to be modeled specifically to remain in the fused image even when the sensor image quality is good. In this particular case, the mechanism of the TNO method would be as follows:

- The common component found in the area of the image covered by the object would be the sensor image intensities (step 3 of the method).
- The common component would be subtracted from the original sensor image (step 4 of method) to produce a zero value for the sensor characteristic component.
- The difference channel would also be zero (step 7 of the method).
- Because of the low intensity value of the sensor characteristic component and the difference channel, they would also bring the weighted average (step 9 of the method) toward a dark value in the fused image. Thus, the object would be preserved in reverse contrast.

3. MIT Method

The MIT method was developed at the Massachusetts Institute of Technology Lincoln Laboratory by Waxman et al. Its purpose is similar to the TNO method: fusion of low visible CCD camera images with infrared sensor images. The method

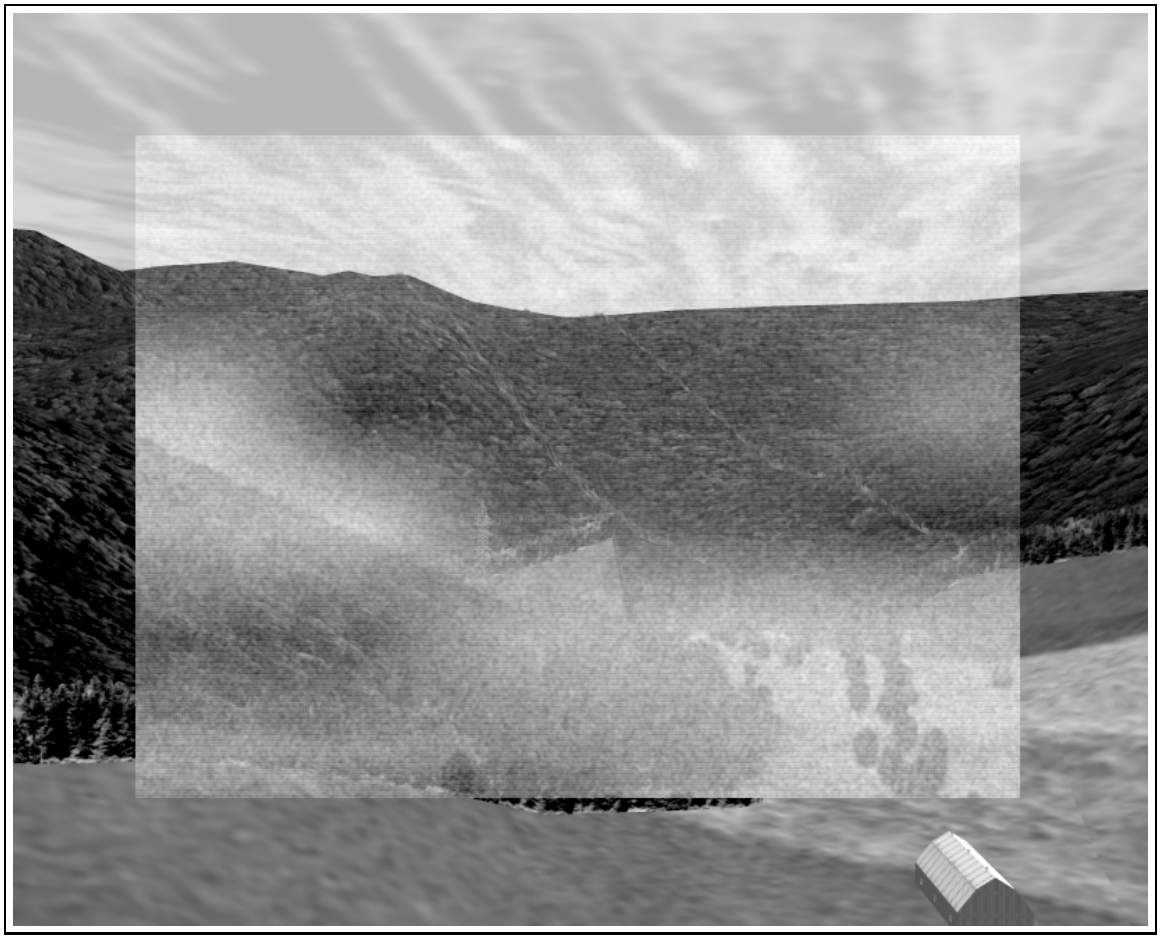

FIGURE 3.5. Fused image with TNO method

is based on opponent processing in the form of feed-forward center surround shunting neural networks [**3**] In other words, center surround cells (differential filters) are connected in parallel to form a neural network which performs fusion.

Although the method is much more effective in fusing low light visible images with IR images than any other method, it leads to some anomalous results for IR-synthetic fusion. The center-surround color opponent cells act as a selection process to preserve the best contrast of the two source images to produce the fused image. As a consequence, because of the high contrast of the synthetic images compared to that of the sensor images, very little of the sensor image comes through in the fused images even for relatively high quality sensor images. Therefore, we modified

the algorithm to take this into account by incorporating parts of the TNO and pixel averaging methods to create a pseudo-MIT method. The following is an overview of the method.

First, a center-surround cell which represents a spatial differential filter is applied to obtain an initial fusion of the synthetic and sensor original images. The center of the cell is applied to the sensor image and the surround to the inverse synthetic image. This results in a high-contrast image that combines the highest contrast features of both images. Second, the sensor characteristic contribution is calculated by subtracting the synthetic image from the initial fused image. In the same fashion, the synthetic characteristic contribution is calculated by subtracting the sensor image from fused image. The subtraction results for steps 2 and 3 are bounded at the minimum to zero. The final fused image is then produced using a weighted average of the synthetic characteristic contribution image, the sensor characteristic contribution image, the original sensor image and the original synthetic image. Note that this method uses the global contrast of the entire image to compute the relative weights in the final average. Local contrast is effectively taken into account in the initial MIT-style center-surround fusion.

The following steps are thus required:

1) Compute the standard deviation of the whole sensor image.
2) Invert the synthetic image (i.e. map black to white and white to black).
3) Filter the images using a sensor center, synthetic surround to produce a high-contrast fused image using the following equation:

$$\text{High contrast}_{\text{fused}}(i,j) = \frac{\sum_{p,q\epsilon[-1,1]} (\text{C}_{pq} \times \text{ Sensor}(i+p, j+q) \text{ - } \text{E}_{pq} \times \text{ Synthetic}_{inverted}(i+p, j+q))}{1 + \sum_{p,q\epsilon[-1,1]} (\text{C}_{pq} \times \text{ Sensor}(i+p, j+q) + \text{E}_{pq} \times \text{ Synthetic}_{inverted}(i+p, j+q))} \tag{3.9}$$

where,

$$C_{pq} = e^{-0.5^{-2}log(2)\times(p^2+q^2)} \tag{3.10}$$

and

$$E_{pq} = e^{-0.75^{-2}log(2)\times(p^2+q^2)} \tag{3.11}$$

Note that (p,q) vary to perform a 3×3 filtering operation centered on pixel (i,j).

4) Map the high contrast fused image and the sensor image onto the whole range of intensities.

5) Find the sensor characteristic contribution by subtracting synthetic image from the fused image (bound to zero if negative):

$$\text{Sensor}_{characteristic}\,(i,j) = \text{Max}(0,\text{High contrast}_{\text{fused}}(i,j) \text{ - Synthetic}(i,j)) \tag{3.12}$$

6) Find the synthetic characteristic contribution by subtracting the sensor image from the fused image (bound to zero if negative):

$$\text{Synthetic}_{characteristic}\,(i,j) = \text{Max}(0, \text{ High contrast}_{\text{fused}}(i,j) \text{ - Sensor}(i,j)) \tag{3.13}$$

7) Find the sensor weight from the total intensity variance in the sensor image using the same linear function as in the pixel averaging method.

8) Produce a fused image using a weighted average of the synthetic characteristic contribution, sensor characteristic contribution, original sensor and original synthetic.

$$\begin{aligned} \text{Fused}_{ij} \quad = \quad & \text{Weight}_{sensor} \times \text{Sensor}(i,j) + \\ & (0.50 \quad \text{- Weight}_{sensor}) \times \text{Synthetic}(i,j) + \\ & 0.25 \quad \times \text{Sensor}_{characteristic}\,(i,j) \; + \\ & 0.25 \quad \times \text{Synthetic}_{characteristic}\,(i,j) \end{aligned} \tag{3.14}$$

9) Map the resulting image onto the entire range of possible pixel intensities.

The sensor weights used in the experiments were tuned to vary linearly between 0.0 and 0.5 with higher weights applied for high sensor contrast measures images. As with the TNO and pixel averaging methods, the infrared sensor autogain mode should not be used. And similarly to these methods, this algorithm could be implemented in hardware to be performed directly on the pixel stream. Since the center-surround operation involves a 3×3 filter, three lines of an image must be read prior to processing. An example of an image fused using the MIT method is shown in Figure 3.6.

Observe that the forest canopy on the far hills has reversed contrast. This is due to the particular response of the center-surround cell. Notice also that the mountain ridges at the top of the image are highlighted. This effect can also be explained by the particular response of the center-surround cell. Because it is a neighborhood operation and the surround of the cell is fed with the synthetic image, the original high contrast fused image contains the features of the synthetic frames, but they are blurred. When the synthetic frame is subtracted from this fused image, an edge highlighting is performed on the synthetic image. This can be seen as an advantage for highlighting coarse features such as ridge lines and forest canopy when the synthetic image begins to be obscured by sensor noise.

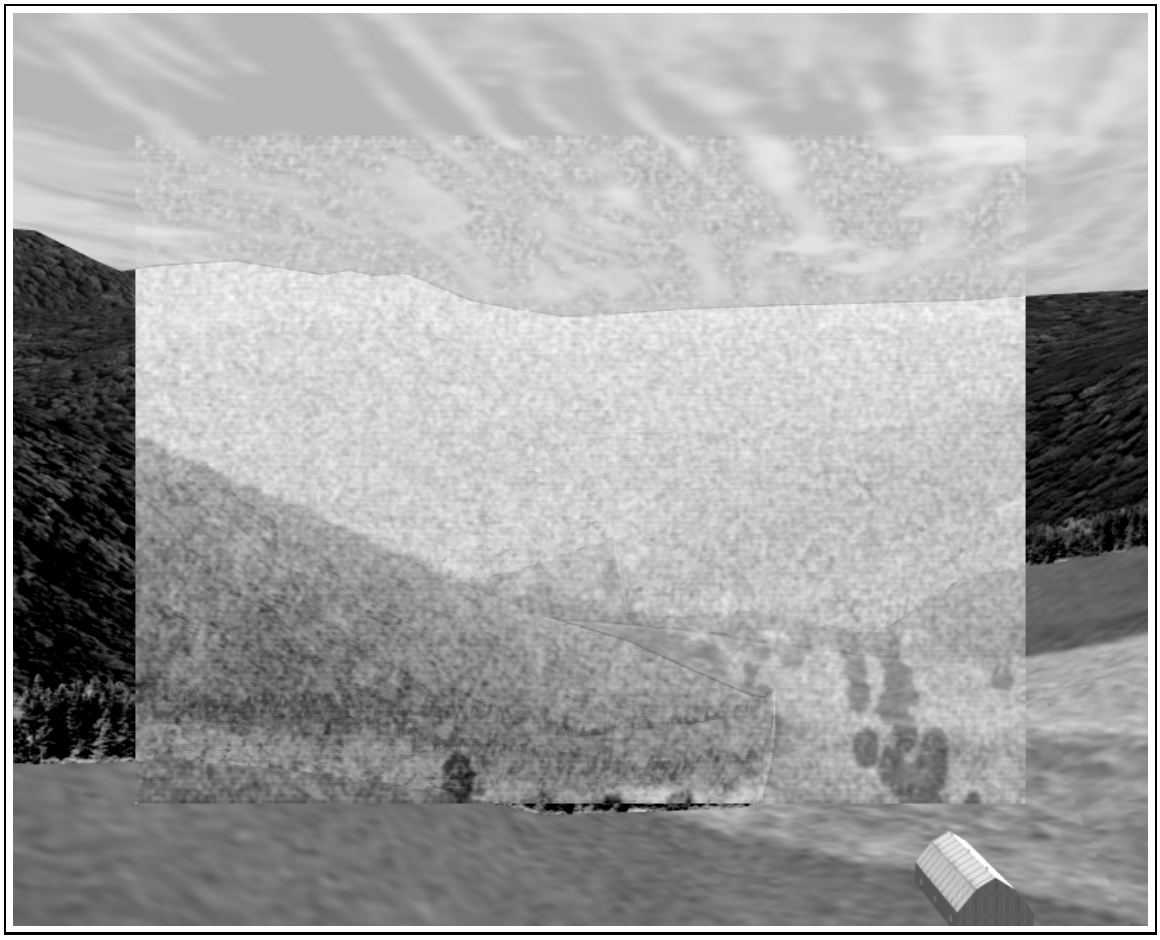

FIGURE 3.6. Fused image with MIT method

CHAPTER 4

Evaluation Results

1. Overview

The evaluation results are presented here first for the static evaluation and then for the dynamic evaluation.

The static evaluation results are broken down by sensor baseline, sensor polarity, and synthetic registration group (full registration, missing or displaced objects, terrain resolution and terrain errors). In general, all three fusion algorithms allowed information from the synthetic database to enhance the information that was available from the sensor alone in the various visibility conditions.

The dynamic evaluation revealed some problems with the computation of parameters, engendering a further tuning exercise. With these problems solved, the 30 Hz sequences showed an additional improvement in information content, and an amelioration of some of the anomalies in the static results, due to motion in the scene that enabled enhancement functions in the human (visual) information processing system.

2. Static Evaluation

The static evaluation verified that fusion dramatically improves image content. In general, the pixel averaging and TNO methods gave better results than the MIT method. Also, although the two sensor polarities exhibited similar performance in

isolation, the white hot polarity was found to be more effective overall than the black hot polarity in the fused images.

Different observations can be made concerning the database registration problems that were modeled. First, the overall performance of the fusion algorithms was not really affected by these conditions. The results are consistent: the pixel averaging and TNO methods provided similar results while the MIT performed somewhat inconsistently and less well overall. Object displacements of the magnitude modeled for the experiments (taken from typical errors found on maps) as well as the lower terrain resolutions could be tolerable in an ESVS. However, the terrain elevation errors presented a strong negative impact upon the usability of the fused image and may require some special processing.

A complete description along with detailed explanations are provided in the next paragraphs.

2.1. Results Graphs. Because many sequences had to be analyzed, results across several sequences are displayed graphically. An example of a result graph is presented in Figure 4.1.

The sensor and database conditions are explicitly written in the title line at the top of the graph. The evaluation criteria are represented on the x-axis of the graph. They are arranged in the order in which they would be seen in good visibility conditions (N.B. far peaks would be detected before the mid ridge because the relative contrast of forest against forest is less than forest against sky). The plot associates a distance (y-axis) with each criterion. This distance (in feet) is computed from the frame number at which the subject was able to see the feature, and corresponds to the distance from the observer to the endpoint of the flight path directly over the crash site. The average of the detection distances recorded by the three subjects is shown. To compare the results between the three fusion algorithms, three curves were plotted on the same graph for a given combination of sensor condition and database condition. The solid line represents the averaging method, the dashed line the TNO method, and the dash dotted line the MIT method. The graphs that follow were

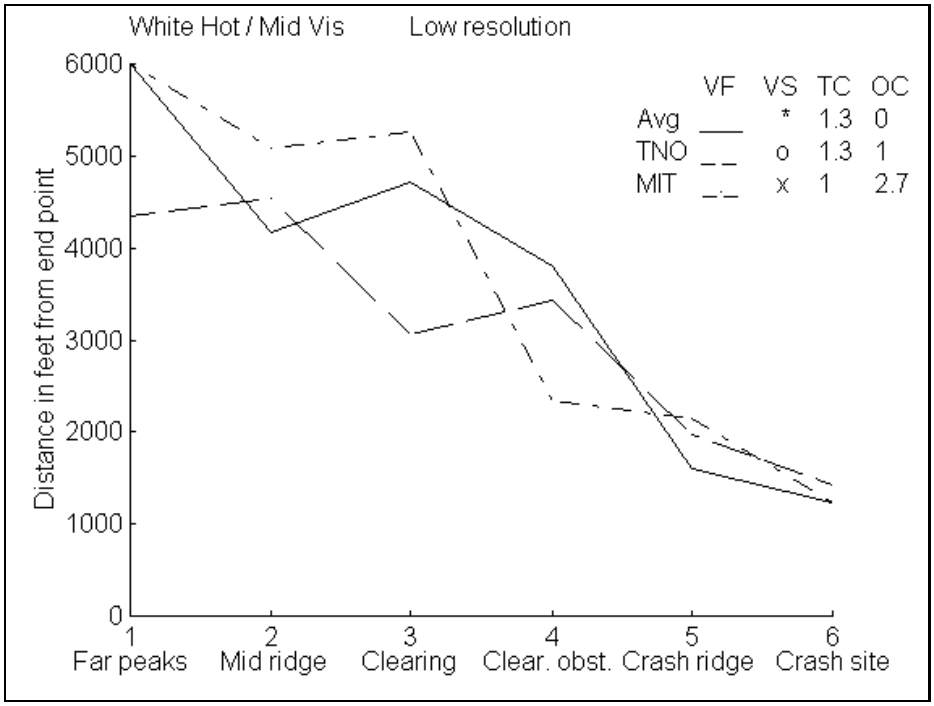

FIGURE 4.1. Example of a result graph

scaled down to fit in one page, causing the legend to be small. As a result, it might be difficult to distinguish between the TNO and MIT symbols. Note that although a curve connects the observations, it does not imply continuous results. Rather, it facilitates the comparison of the algorithms by clearly identifying crossover points in performance.

Additional statistics are also displayed in the legend for each method. The associated VS symbol is flagged when a feature appears, disappears and reappears again in the fused image. This is generally representative of an object seen in the synthetic image, then obscured by sensor noise and finally seen again in the fused image. In this case, the displaced curve shows the distance from which the feature remained in the images consistently, effectively lowering the curve. The TC and OC indicate the average number of terrain conflicts and object conflicts observed in the fused sequence.

2.2. Baseline. The sensor sequences were evaluated first to create a baseline for comparison with the fused images to assess if fusion improved the image content. The results for the two sensor polarities and the three visibility conditions are shown in Figure 4.2, with the white hot results on the left and the black hot results on the right. The high, medium and low visibility conditions are arranged from top to bottom respectively.

Observe that the far peaks were never visible in the sensor images. This is due to the fact the ceiling is low and therefore obscures long range features. The curves are progressively lower from the high visibility condition to the low visibility condition, consistent with the fact that we see objects from greater distances in better visibility conditions. Note that in the three visibility conditions, there are only minor differences in the results for the two polarities, and these can be attributed to inherent experimental variations. It can be concluded that sensor image polarity has little or no effect on feature detection.

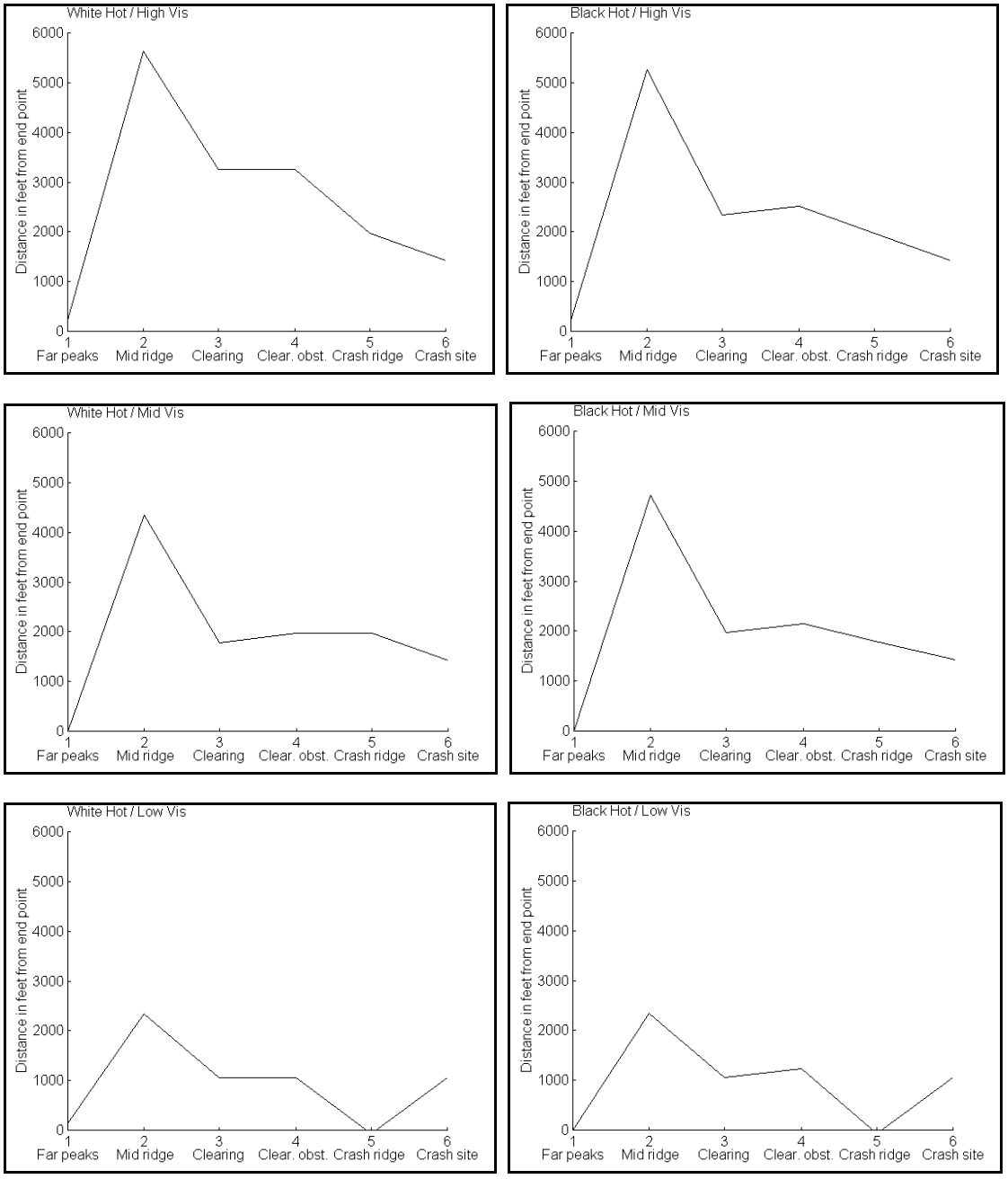

FIGURE 4.2. Sensor baseline results

2.3. One-to-one Registration Database to Real World. Database condition 1 (fully registered) was intended to specifically study how the sensor properties degrade the synthetic image as the sensor is blended in. The results for this condition are shown in Figure 4.3 in the same fashion as before (sensor polarities, left to right - visibility conditions, top to bottom).

Observe that there is significant improvement in the results compared to the sensor baseline. The far peaks that were never seen in the sensor sequences are now observed in the fused sequences. Also, the mid ridge and the clearing and its objects are detected from a much greater distance in the medium and low visibility conditions. Significantly, the crash ridge and the crash site ranges of detection have not changed. This means that the fusion has preserved close range features of the original sensor images. Thus, the fusion process seems to have the predicted characteristic of augmenting the sensor image content and providing important information to the pilot.

Overall, the MIT algorithm was less consistent than the other two algorithms. The pixel averaging and TNO methods show similar results in white hot, but the TNO algorithm has superior performance with the black hot sensor. The slight decrease in performance of the TNO algorithm for the medium visibility white hot sensor was probably due to the particular tuning of the algorithm parameters.

Note that with image fusion, there is now a significant difference in performance between the two sensor polarities. The MIT performance for the high and medium visibility conditions is much poorer with the black hot sensor: the far peaks are never observed. This algorithm was tuned to perform optimally with sensor images that were mostly white, i.e. with high pixel intensities. It tends to suppress large bright regions and enhance dark objects in the sensor image. In the black hot mode, the white objects (which tend to be significant) do not survive the fusion process as well. All three subjects agreed that white hot images looked more natural and were easier to interpret when fused with the synthetic images.

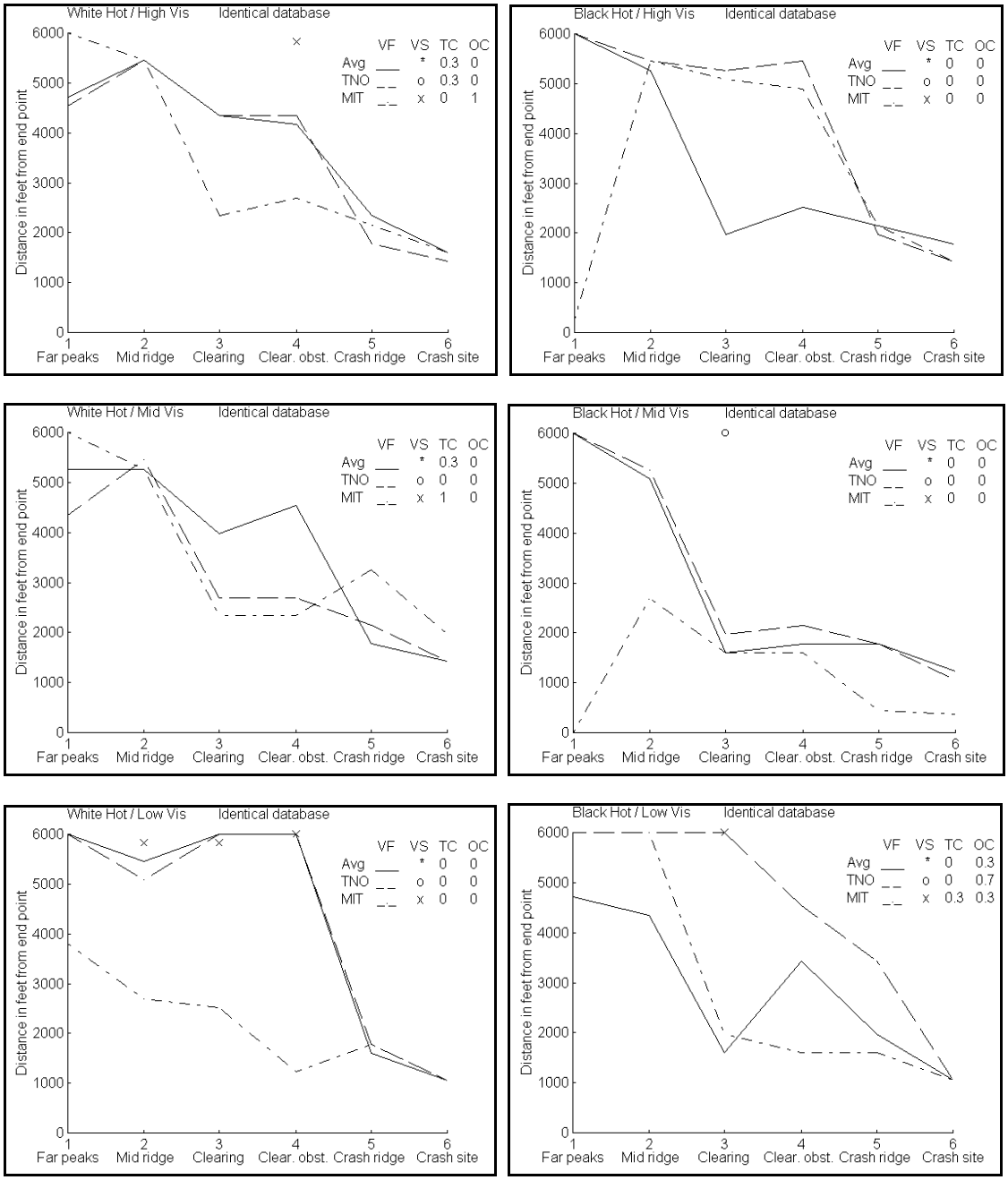

FIGURE 4.3. Fully registered database results

2.4. Object Displacements. As registration condition 2 (aircraft navigation system and attitude indicator noise), 4 and 7 (high and medium global database offset) all resulted in viewpoint displacements, they are combined here for analysis. Figure 4.4, Figure 4.5 and Figure 4.6 present the fusion results for these three conditions. As for the general fully registered condition, the pixel averaging and TNO methods have superior performance to the MIT method, and the white hot sensor mode was preferred. In addition, certain comments made by the evaluation subjects are of note. All agreed that it was relatively easy to detect displacements between synthetic and sensor objects that were not aligned. They also stated that separate objects such as buildings were more distracting than unaligned terrain features. Therefore, it could be advantageous to exclude such objects in the synthetic database in order to commit resources to better terrain definition.

Notice also that the number of object and terrain conflicts displayed on the graphs were always higher for the MIT method. This could be caused by the general property of the MIT method to keep more of the synthetic image even when the sensor image quality is good. The general conclusion with object displacements is that such registration problems would probably be tolerable for an ESVS.

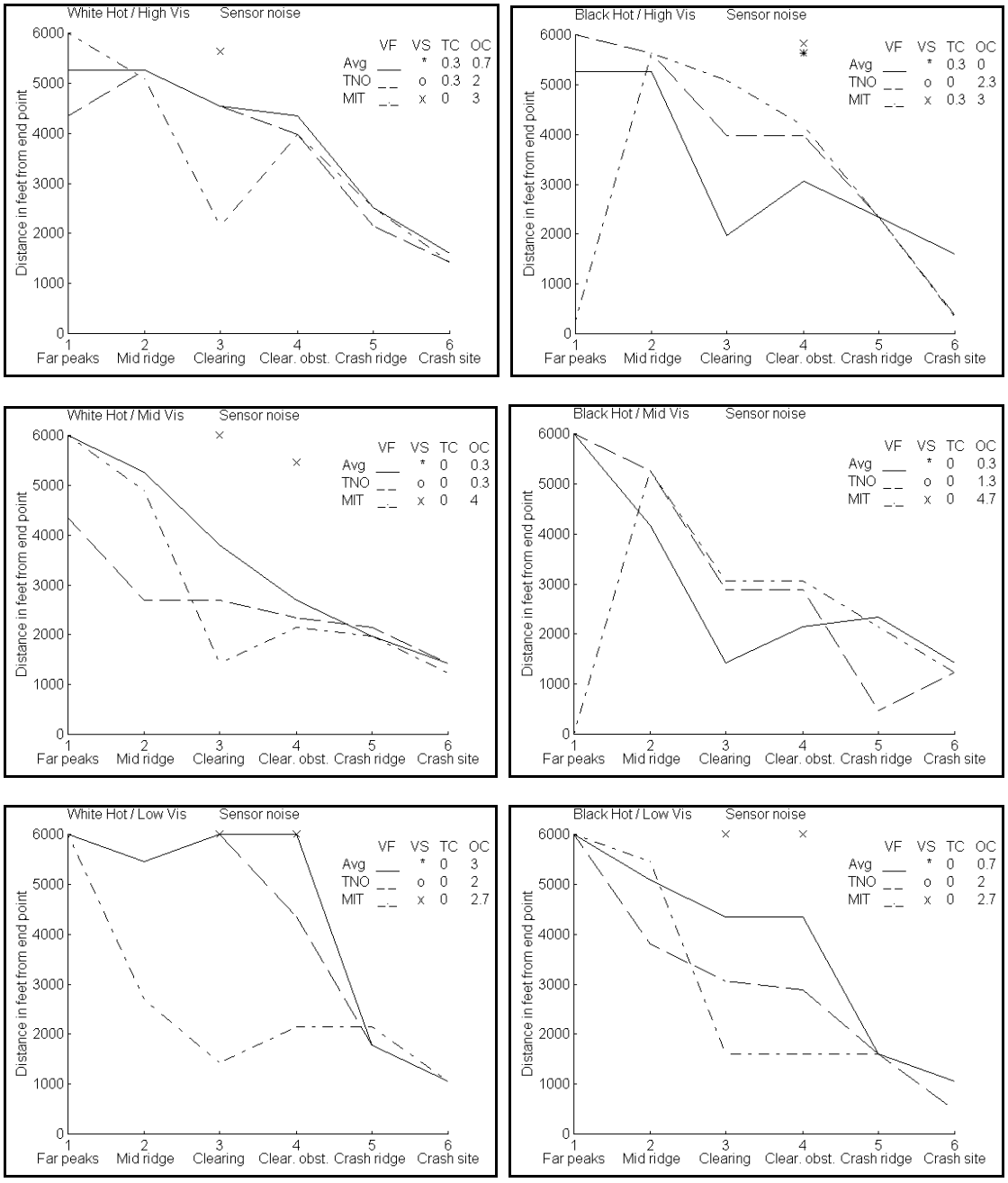

FIGURE 4.4. Position noise results

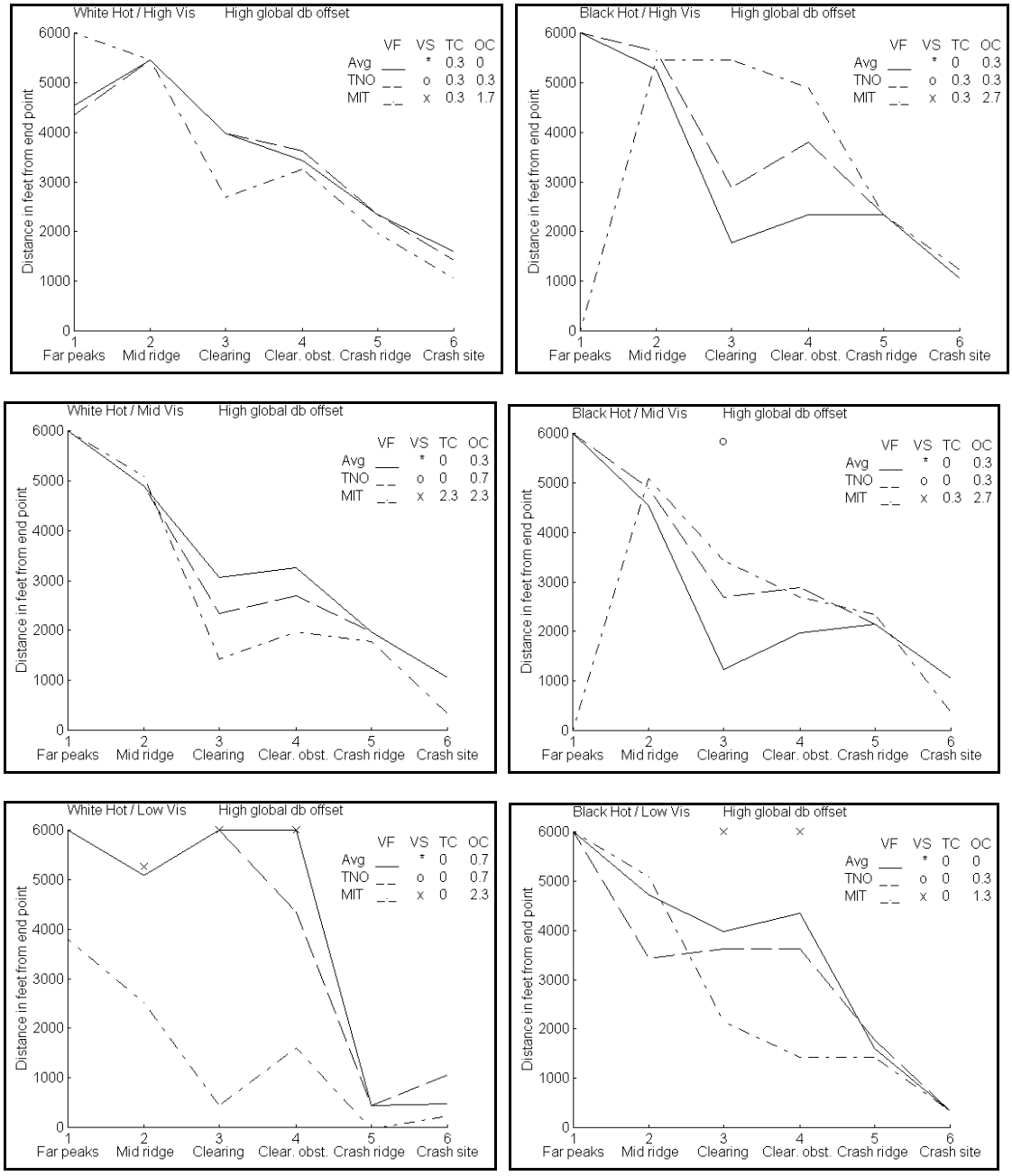

FIGURE 4.5. High database offset results

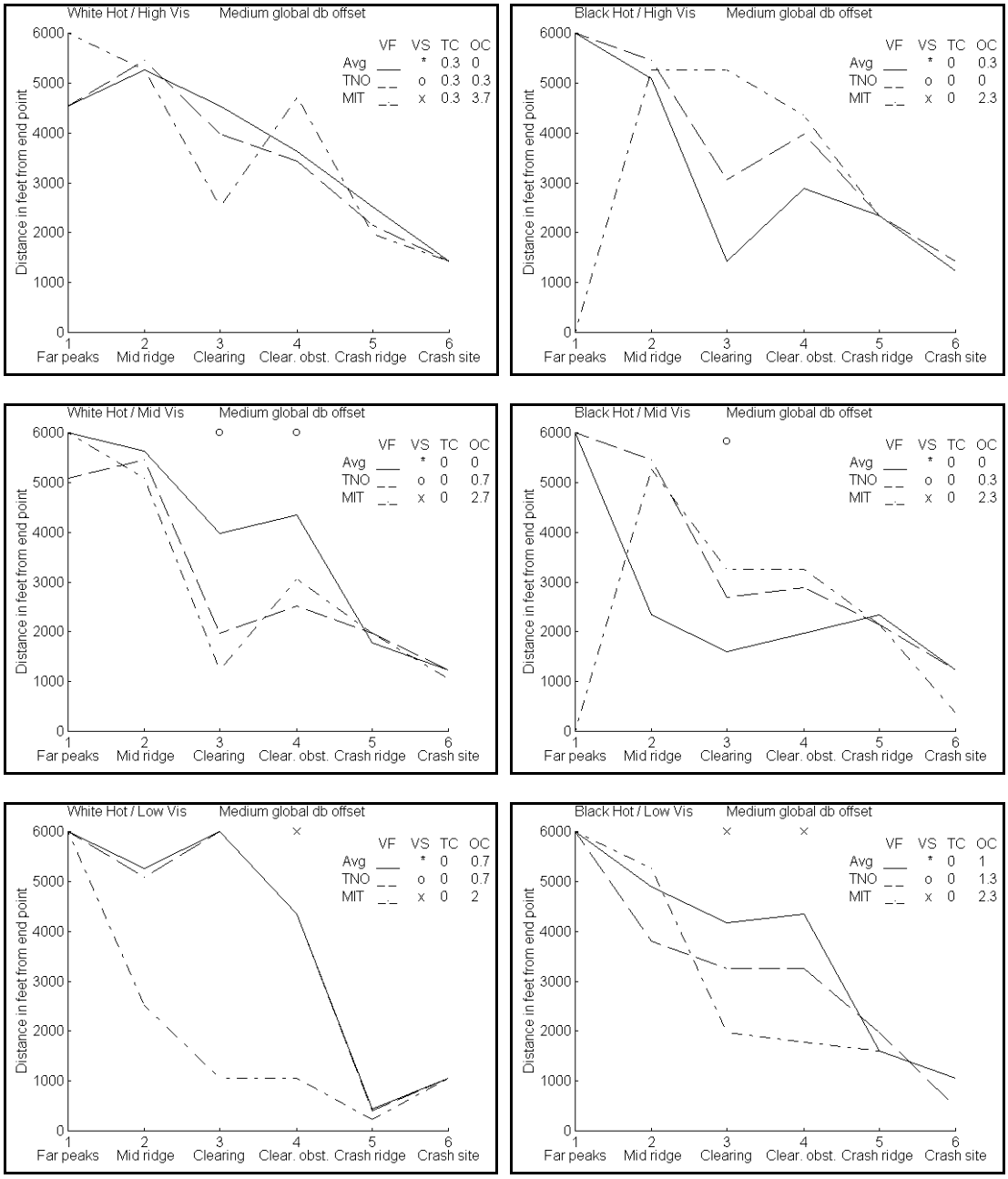

FIGURE 4.6. Medium database offset results

2.5. Terrain Resolution. The following graphs (Figure 4.7, Figure 4.8 and Figure 4.9) present the evaluation results for the three terrain resolutions (database conditions 3, 5 and 8). The different terrain resolutions were a source of particular interest as low-cost image generation technologies will not yet support the very high terrain resolutions it was originally thought might be necessary at the required 60 Hz update rate. In fact, a comparison of the results show (in accordance with subjects comments) that the lowest resolution terrain had sufficient detail to permit accurate identification of key terrain features to support route planning. As this, together with providing a stable horizon and peripheral view in low altitude conditions, is the main purpose of the synthetic portion of an ESVS, these results would seem to indicate that the low resolution terrain used in the experiment would be suitable for implementation in an ESVS.

The same conclusions can again be drawn concerning the difference in performance for a given sensor polarity and between the three algorithms, with the white hot polarity giving the best results and with the pixel averaging and TNO methods performing more consistently and better than the MIT method.

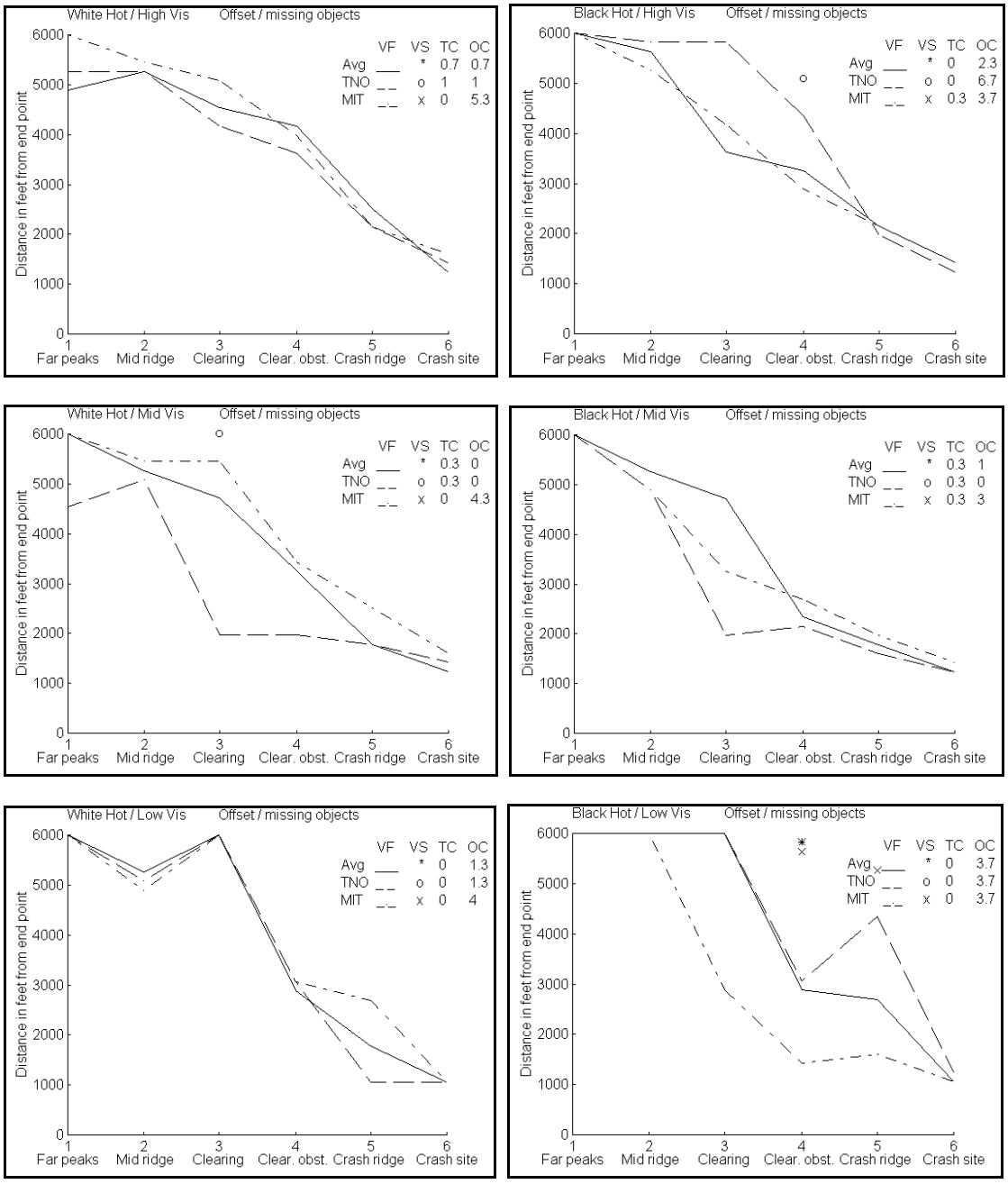

FIGURE 4.7. Full resolution (missing/displaced) results

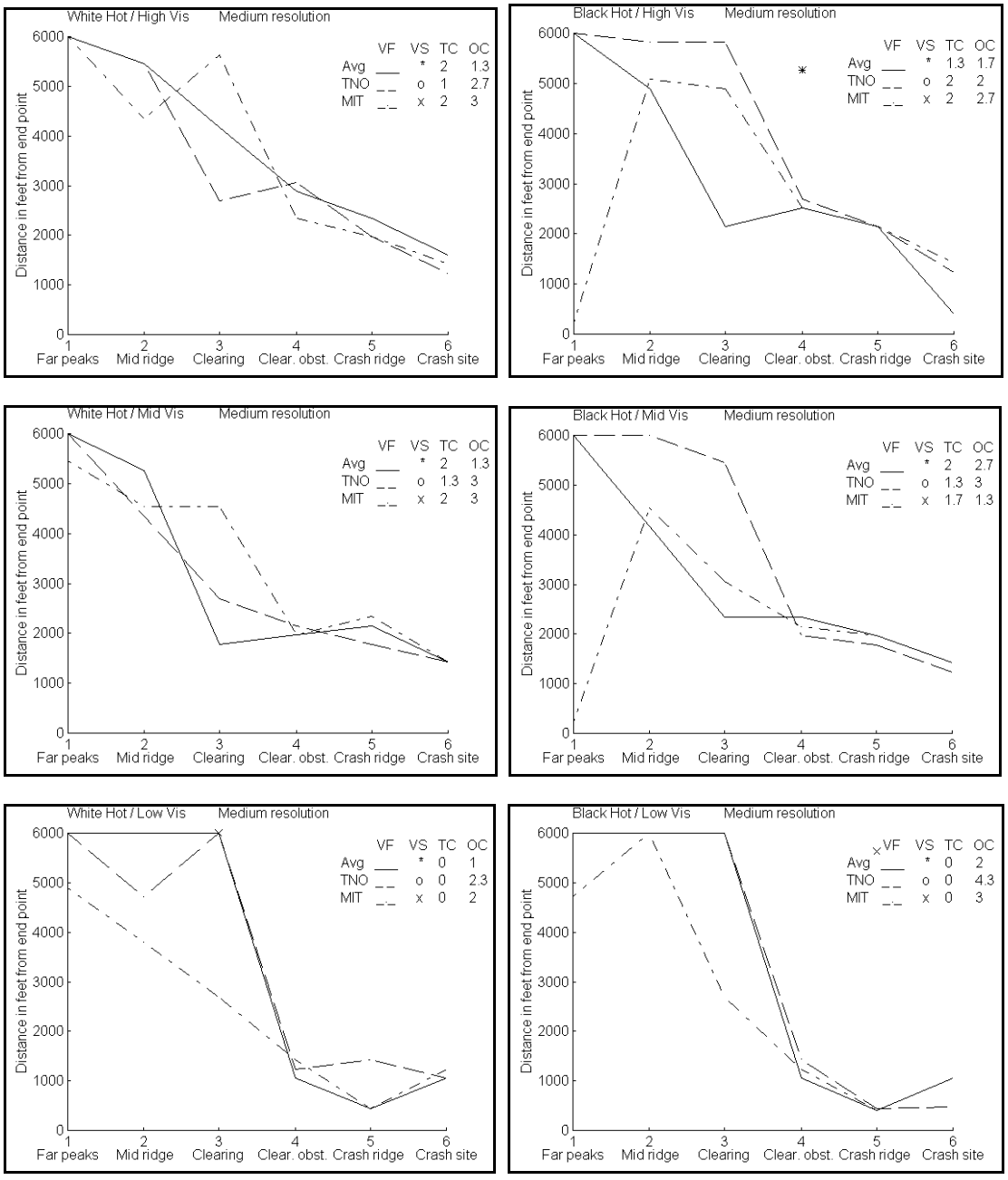

FIGURE 4.8. Medium terrain resolution results

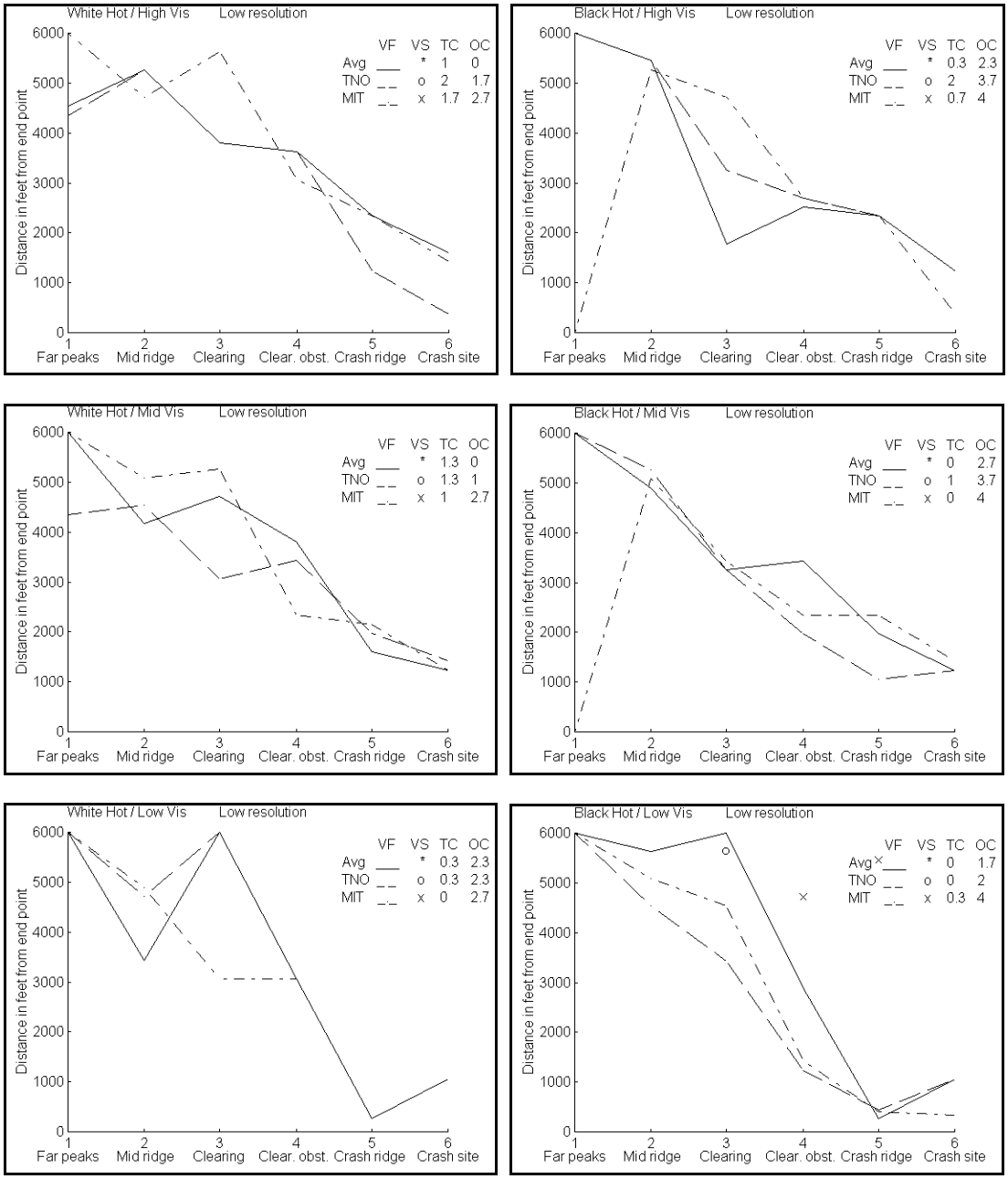

FIGURE 4.9. Low terrain resolution results

2.6. Terrain Errors. The final set of database conditions (6 and 9) represent medium and large errors in local terrain elevation (from source data). The result graphs for these conditions are shown in Figure 4.10 and Figure 4.11. Again, similar conclusions can be drawn concerning the performance of the fusion algorithms and sensor polarity.

Note the extremely high number of terrain and object conflicts on all of the graphs, especially for the high terrain elevation error condition. Initial subject reports were that the mid ridge separated into two ridges (one behind the other), and because the databases were otherwise identical it was difficult to determine which ridge was the real one. However, terrain objects that were vertically displaced made it more obvious as the ridge was approached, and in general, the terrain errors were then detected by the subjects as being an offset in the z direction.

A second problem ensued for the high elevation error condition, when the viewpoint path that had been calculated against the true terrain intersected the false terrain as it crossed the mid ridge. Terrain intersections result in highly unpredictable behavior of the image generator, as features that would normally be occulted by the terrain are rendered. The resulting images take on a surreal quality, are disconcerting to view, and would be unacceptable both in flight simulators and in an ESVS. Since it may be necessary to approach unaligned synthetic terrain in order to descend into real terrain using the E portion of the system, techniques must be developed to suppress the S image in situations where the viewpoint intersects the synthetic terrain. Such techniques would use the height above terrain feature or a range-to-polygon feature supported by most image generators to detect the condition, in addition to incorporation of registration correction from a range sensor/radar altimeter combination.

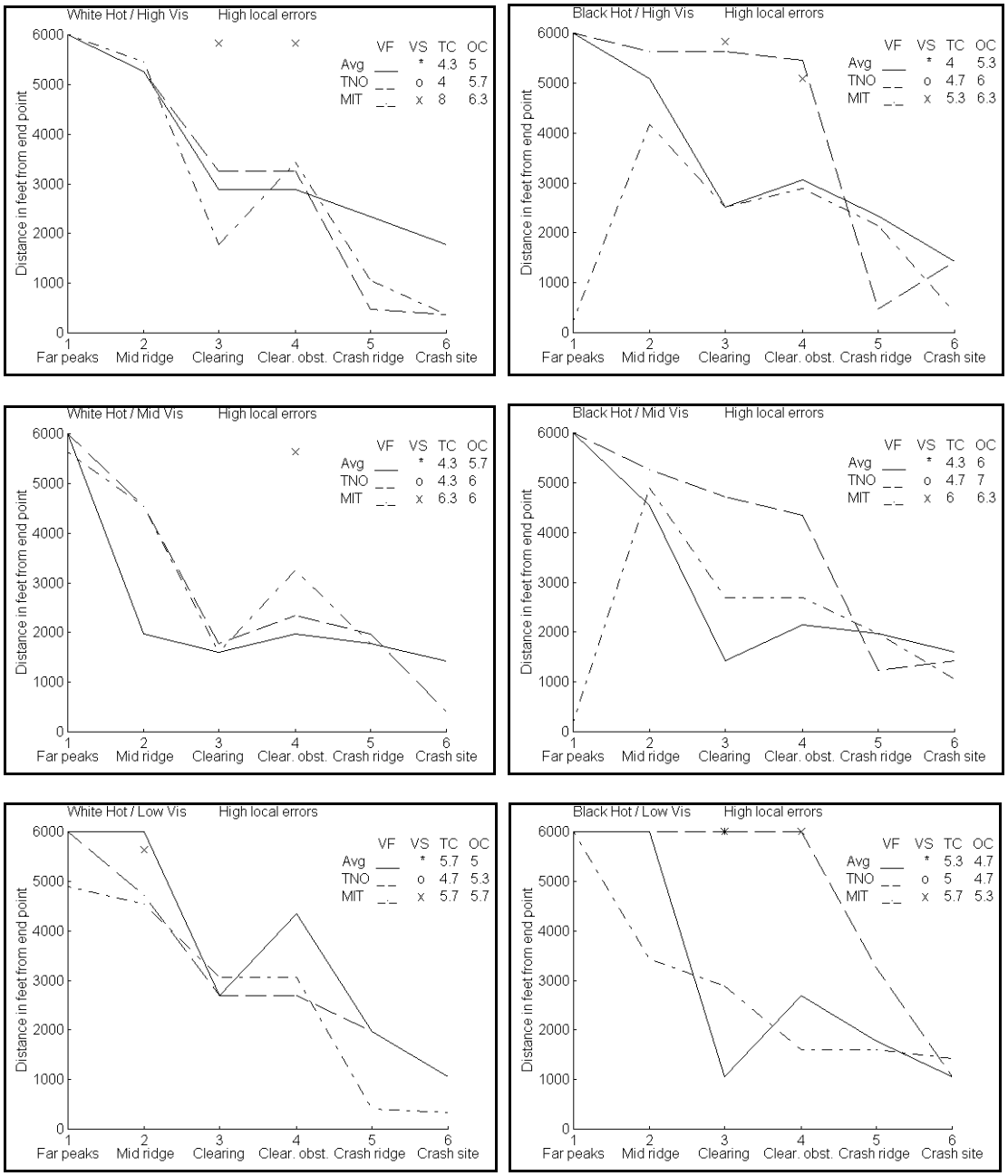

FIGURE 4.10. High terrain elevation errors results

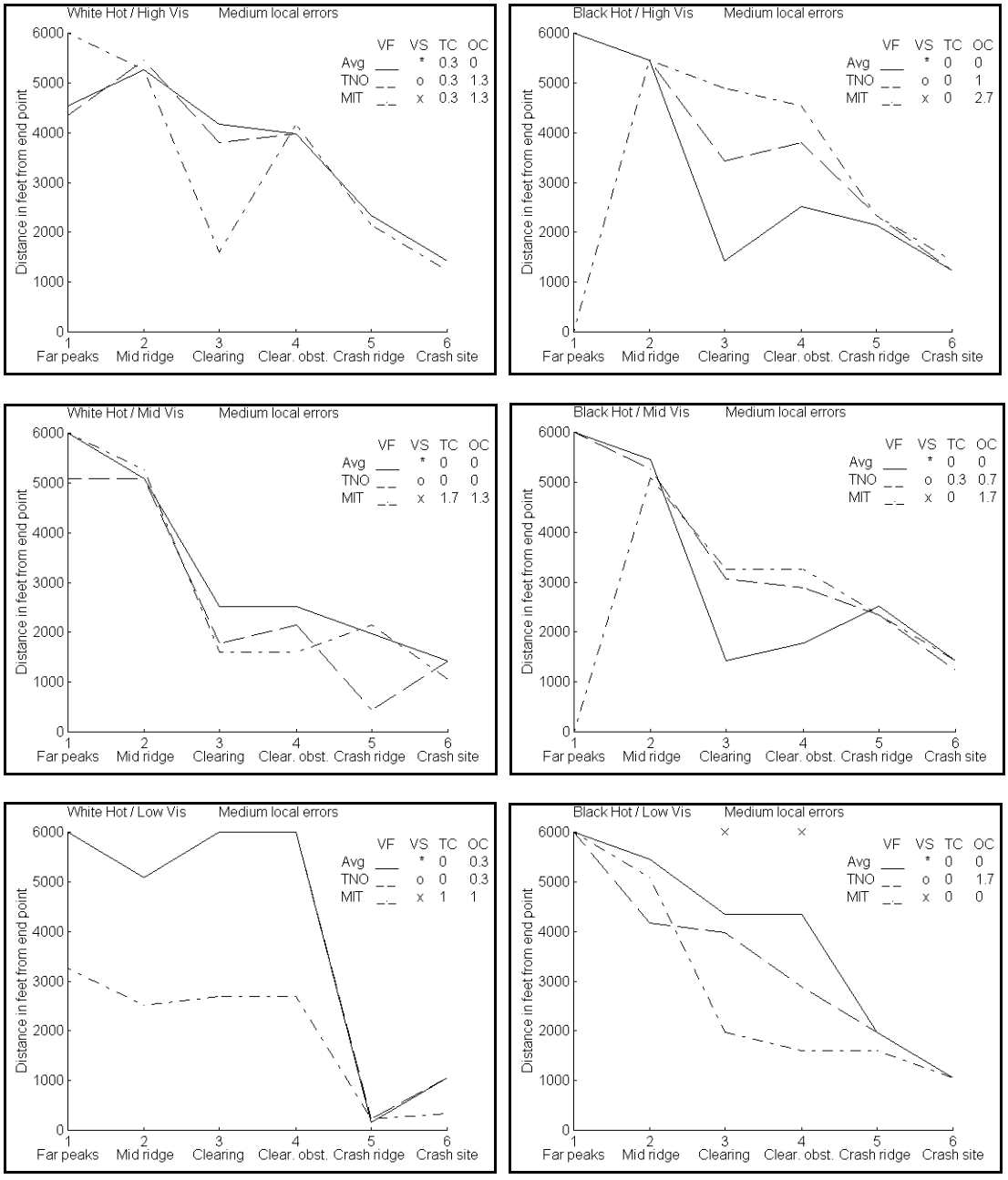

FIGURE 4.11. Medium terrain elevation errors results

3. Dynamic Evaluation

3.1. Results. Initial dynamic sequences were subject to severe flickering, which was traced to two problems:

- Variation of the relationship between the computed contrast in adjacent sub-windows (used to calculate the weights in the pixel averaging and TNO methods) due to small shifts in the direction of regard.
- Small changes from one frame to the next in the intensity distribution of the fused images.

These problems and their solutions are discussed below. Once these problems were solved, the dynamic sequences showed a marked improvement in the overall performance of all three algorithms, with an increased visibility of significant features. The MIT algorithm in particular performed significantly better, as many of the inconsistencies in the static image sequences were resolved, although anomalies such as contrast reversal of the forest canopy remained. It is apparent that, on being presented with dynamic information, the human visual system will integrate over several frames to permit correlation of scene elements from one frame to the next and to reduce noise and clean up persistence effects in the display of noisy images. This permitted objects to be separated from noisy backgrounds and to be detected at lower thresholds. It also permitted some separation of synthetic and sensor content where conflicts had previously been reported, especially for objects which at some point crossed the border between the fused inset and the purely synthetic background.

3.2. Dynamic Noise Solutions. The first flickering effect that was observed for the pixel averaging and TNO methods was manifested as a flashing at the borders between the regions of the weight grid. The cause was traced to changes in the noise from frame to frame, and more particularly to non-uniform contrast changes in the sub-windows which led to disproportionate changes in the averaging weights. The function used to translate contrast measures into weights has a relatively high gain which results in a large variation of weights even for small variations in contrast.

Time	Weight value
$t+1$	w_1
t	w_0
$t-1$	w_{-1}
$t-2$	w_{-2}
$t-3$	w_{-3}
$t-4$	w_{-4}
$t-5$	w_{-5}

TABLE 4.1. Example of weights at different time

The second problem, apparent in the MIT fused images as well as in the TNO and pixel averaging images, was a brightness flicker of the whole image. Such flickering is extremely distracting and could lead to eye strain and/or headaches. The cause was traced to the final remapping of the fused images to the full intensity range. It uses the minimum and maximum pixel intensities found in the image and because of the noise, these intensities vary between two successive frames. As a consequence, the remapping function varies too much between two images.

The obvious solution was to stabilize the weights and the remapping values so that they would vary more slowly. This was achieved by maintaining a history buffer of the values computed in previous frames. At each frame, the weights and parameters were stored in a FIFO buffer and an average of the history buffer computed. As an example, consider the time profile of the individual computed sensor weights or remapping weights shown in Table 4.1.

The original fusion methods used weight w_0 at time t and weight w_1 at time $t+1$. However, by using the average of the previously computed frames this variation could be reduced. For this particular example, the weight values would be:

$$w_t = \frac{w_0 + w_{-1} + w_{-2} + w_{-3} + w_{-4} + w_{-5}}{6} \tag{4.1}$$

$$w_{t+1} = \frac{w_{-1} + w_0 + w_{-1} + w_{-2} + w_{-3} + w_{-4}}{6} \tag{4.2}$$

The change from time t to time $t+1$ is now (w_1 - w_{-5}) / 6 instead of (w_1 - w_0). If we can assume that the change in weight over the number of frames averaged, i.e. from w_{-5} to w_0 is sufficiently small, then the change in weight used from time t to time $t+1$ will be reduced significantly without radically changing the overall quality of an individual frame. This particular scheme has the advantage of being simple and easy to implement. Tests showed that a half-second history of weights and remapping parameters removes enough of the undesired effects while not significantly delaying changes in contrast that could occur in the sensor images. Thus, because our refresh rate was 30 Hz, the last 15 values were stored and used for this purpose.

CHAPTER 5

Computational Complexity

An operations count has been used to compare the computational complexities of the three fusion algorithms. Three tables were built to facilitate the comparison. The step number as presented in the algorithm description (see Chapter 3) and an abbreviated step description are given with the corresponding operations count for the pixel averaging, TNO and MIT methods in Table 5.1, Table 5.2 and Table 5.3 respectively. The number of additions, subtractions, multiplications and divisions required to compute a single pixel in the fused image are shown for each step of the algorithm.

The averaging method has the lowest operations count. The number of operations is increased by almost 50% with the TNO method. The extra computation is required to compute the difference channel, and to compute the extra set of averaging weights for this channel. The most complex algorithm is the MIT. This is attributed to the convolution operation with the center-surround cell which, even using a small operator size, is very computationally intensive.

Step Number	Step Description	Operations per pixel				
		+	−	×	/	**Total**
1	Standard deviation calculation	1	0	0	0	**1**
2-3	Bilinear interpolation	3	3	3	0	**9**
4	Weighted average	1	1	2	0	**4**
5	Remapping	1	0	1	0	**2**
	Total	**6**	**4**	**6**	**0**	**16**

TABLE 5.1. Operations count for pixel averaging method

Step Number	Step Description	Operations per pixel				
		+	−	×	/	**Total**
1	Standard deviation calculation	1	0	0	0	**1**
2-3	Bilinear interpolation	3	3	3	0	**9**
4-5-6-7-8	Channels computation	0	4	0	0	**4**
9	Weighted average	2	2	3	0	**7**
10	Remapping	1	0	1	0	**2**
	Total	**7**	**9**	**7**	**0**	**23**

TABLE 5.2. Operations count for TNO method

Step Number	Step Description	Operations per pixel				
		+	−	×	/	**Total**
1-2-3-4-5	Center-surround operation	20	1	8	1	**30**
6-7-8	Channels computation	0	2	0	0	**2**
9	Weighted average	3	1	4	0	**8**
10	Remapping	1	0	1	0	**2**
	Total	**24**	**4**	**13**	**1**	**42**

TABLE 5.3. Operations count for MIT method

CHAPTER 6

Conclusions

From the results presented here, it is evident that fusion improves the useful content of independent synthetic and sensor images. All three algorithms examined in this study provided observers the capability to detect important features from greater distances than sensor alone. This enforces the hypothesis that sensor-synthetic fused images would be beneficial for pilots flying in poor visibility conditions. Certain key conclusions can be made regarding the characteristics of a potential proof-of-concept system.

First, although white hot and black hot sensor modes seemed to be equivalent, the white hot polarity demonstrated better results in the fused images. The three subjects also found the white hot polarity to have a more natural appearance. For these reasons, the black hot mode should not be used in an ESVS. Also, as all the algorithms use pixel intensity deviations for contrast measurements, the autogain mode should not be turned on. Note that an autogain function is performed at the output of the fusion.

As it was determined that viewpoint noise (due to inaccuracies in the aircraft navigation system and attitude indicators) and global database offset could be a source of registration problems, the overall effect on image fusion was analyzed. Experimental results as well as individual observations have shown that these particular errors would not have a severe impact on the system. Rather, such errors could be

both tolerable and detected (and compensated for) by the observer. Similarly, lower terrain resolutions did not have a great impact on the fused images.

Terrain elevation errors from source data had the most severe effects on registration and fusion. Subjects experienced confusion and had difficulty distinguishing real terrain from synthetic terrain. This effect may be less severe when the synthetic image has a lower resolution content (e.g. polygonal forest canopies). More significantly, it was discovered that the viewpoint may drop below synthetic terrain that is misaligned due to large elevation errors in source data. In this situation, the synthetic system would provide unpredictable results, generating confusing images with surreal scene content that should be obscured. Therefore, techniques must be developed to deal with these problems.

From the results, it has been shown that all three fusion methods presented relatively good results. The MIT method displayed a good overall performance but had some undesirable effects such as contrast reversals. It was also very difficult to tune and is computationally intensive. The pixel averaging method was found to be the simplest in terms of required operations per pixel. The TNO algorithm, although slightly more complex, demonstrated results similar to the pixel averaging and has the advantage of preserving certain synthetic features in a predictable way that would facilitate modeling and display of range sensor features. The TNO method could also be tuned to act as a simple pixel averaging by setting the blue channel to zero and sensor enhanced channel equal to the sensor channel.

Finally, the dynamic noise effects should be compensated for. This involves time-filtering both the relative weights used for the pixel averaging and TNO methods and the autogain function that is applied after the fusion. We found that a simple average applied over a half-second history is sufficient. (Note that this does not affect system update rate per se.)

REFERENCES

[1] G. Duane, *Pixel-level sensor fusion for improved object recognition*, Proc. SPIE **931** (1988), 180–185.

[2] J.D. Foley, A. Van Dam, S.K. Feiner, and J.F. Hughes, *Computer graphics: principles and practice*, Prentice-Hall Inc., 1996.

[3] S. Grossberg, *Neural networks and natural intelligence*, MA: MIT Press, Cambridge, 1988.

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

[5] J.M. Lloyd, *Thermal imaging systems*, Optical Physics and Engineering, 1975.

[6] R.C. Luo and M.G. Kay, *Multisensor integration and fusion: issues and approaches*, Proc. SPIE **931** (1988), 42–49.

[7] D. Nitzan, A.E. Brain, and R.O. Duda, *The measurement and use of registered reflectance and range data in scene analysis*, Proceedings of the IEEE **65** (1977), no. 2, 206–220.

[8] American Society of Photogrammetry, *Manual of remote sensing*, vol. 1-2, Falls Church, 1975.

[9] A. Toet, J.K. IJspeert, A.M. Waxman, and M. Aguilar, *Fusion of visible and thermal imagery improves situational awareness*, Proceedings of the SPIE Conference on Enhanced and Synthetic Vision **SPIE-3088** (1997a), 177–188.

[10] A. Toet and J. Walraven, *New false colour mapping for image fusion*, Optical Engineering **35** (1996), no. 3, 650–658.

[11] E.L. Waltz and D.M. Buede, *Data fusion and decision support for command and control*, IEEE transactions on systems, man, and cybernetics **SMC-16** (1986), 865–879.

[12] A.M. Waxman, J.E. Carrick, D.A. Fay, J.P. Racamoto, M. Aguilar, and E.D. Savoye, *Electronic imaging aids for night driving: low-light ccd, thermal ir, and color fused visible/ir*, Proceedings of the SPIE Conference on Transportation Sensors and Controls **SPIE-2902** (1996c), 62–73.

[13] A.M. Waxman, D.A. Fay, A.N. Gove, M. Seibert, J.P. Racamoto, J.E. Carrick, and E.D. Savoye, *Color night vision: fusion of intensified visible and thermal ir imagery*, Proceedings of the SPIE Conference on Synthetic Vision for Vehicle Guidance and Control **SPIE-2463** (1995), 58–68.

[14] A.M. Waxman, A.N. Gove, D.A. Fay, J.P. Racamoto, J.E. Carrick, M. Seibert, and E.D. Savoye, *Color night vision: opponent processing in the fusion of visible and ir imagery*, Neural Networks **10** (1997a), no. 1, 1–6.

[15] A.M. Waxman, A.N. Gove, M. Seibert, D.A. Fay, J.E. Carrick, J.P. Racamoto, E.D. Savoye, B.E. Burk, R.K. Reich, W.H. McGonagle, and D.M. Craig, *Progress on color night vision: visible/ir fusion, perception & search, and low-light ccd imaging*, Proceedings of the SPIE Conference on Enhanced and Synthetic Vision **SPIE-2736** (1996a), 96–107.

[16] W. Wolfe, *The infrared handbook*, Infrared Information Analysis Center, 1989.

Document Log:

Manuscript Version 0

Typeset by $\mathcal{AMS}$-LaTeX — 4 April 2000

Philippe Simard

E-mail address: psimard@cim.mcgill.ca

Centre for Intelligent Machines, McGill University, 3480 University St., Montreal (Quebec) H3A 2A7, Canada, *Tel.* : (514) 398-2185

Typeset by $\mathcal{AMS}$-LaTeX