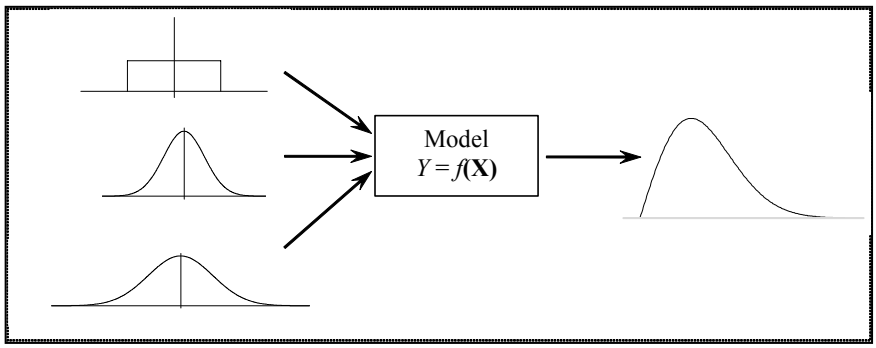
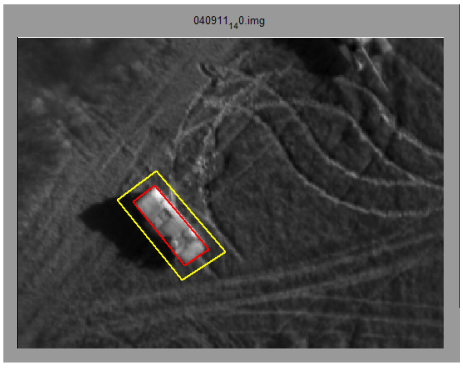
FOI-R--1421--SE
December 2004
ISSN 1650-1942
Technical report

Patrik Hermansson, Sten Nyberg, Erik Andersson and David Börjesson

Methods for validating optical signature simulations

- progress report





Sensor Technology
SE-581 11 Linköping

SWEDISH DEFENCE RESEARCH AGENCY

Sensor Technology
P.O. Box 1165
SE-581 11 Linköping

FOI-R--1421--SE

December 2004

ISSN 1650-1942

Technical report

Patrik Hermansson, Sten Nyberg, Erik Andersson and David Börjesson

Methods for validating optical signature simulations -progress report

Issuing organization	Report number, ISRN	Report type
FOI – Swedish Defence Research Agency Sensor Technology P.O. Box 1165 SE-581 11 Linköping	FOI-R--1421--SE	Technical report
	Research area code 6. Electronic Warfare and Deceptive Measures	
	Month year December 2004	**Project no.** E3071
	Customers code 5. Commissioned Research	
	Sub area code 62 Low Observables	

Author/s (editor/s)	Project manager
Patrik Hermansson Sten Nyberg David Börjesson Erik Andersson	Claes Nelsson
	Approved by Lars Bohman
	Sponsoring agency Armed forces
	Scientifically and technically responsible Patrik Hermansson

Report title

Methods for validating optical signature simulations – progress report

Abstract (not more than 200 words)

We present the progress in developing methods for validation of computer programs for prediction of optical signatures. The work is currently performed in two directions: The first is concerned with developing methods for quantifying the propagation of input data uncertainties to output parameters (i.e. radiance or temperature) in computational predictions. The objective of the second path of development is to analyze and validate the differences between simulated and measured images, for instance in the visual or IR, through image features such as statistical moments and Gabor wavelets. In particular, aspects that are important for detection, classification and identification of targets are considered.

The work on propagation of uncertainties in computational predictions has resulted in a number of proposed methods. In this report we present a first test case using the error propagation method. The results are according to expectations but it is concluded that further studies, using a more general method, is desirable to assess the results.

In the work on analyzing differences between simulated and measured images it is proposed that several textural features are used to define the discriminative capability when it comes to detect vehicles in a terrain. In a preliminary example, several feature values have been computed for a measured image and the corresponding simulated image. This work will continue during 2005.

Keywords

modelling, validation, uncertainty, error propagation, Monte Carlo, IR, signature, image features

Further bibliographic information	Language English
ISSN 1650-1942	**Pages** 42 p.
	Price acc. to pricelist

Utgivare	Rapportnummer, ISRN	Klassificering
Totalförsvarets Forskningsinstitut - FOI	FOI-R--1421--SE	Teknisk rapport
Sensorteknik	**Forskningsområde**	
Box 1165	6. Telekrig och vilseledning	
581 11 Linköping	**Månad, år**	**Projektnummer**
	December 2004	E3071
	Verksamhetsgren	
	5. Uppdragsfinansierad verksamhet	
	Delområde	
	62 Signaturanpassning	

Författare/redaktör	Projektledare
Patrik Hermansson	Claes Nelsson
Sten Nyberg	**Godkänd av**
David Börjesson	Lars Bohman
Erik Andersson	**Uppdragsgivare/kundbeteckning**
	Försvarsmakten
	Tekniskt och/eller vetenskapligt ansvarig
	Patrik Hermansson

Rapportens titel (i översättning)

Metoder för validering av optisk signatursimulering

Sammanfattning (högst 200 ord)

Den här lägesrapporten beskriver det pågående arbetet med att utveckla metoder för validering av datorprogram för simulering och prediktion av optiska signaturer. Utvecklingsarbetet har två huvudinriktningar: I den ena inriktningen utvecklas metoder för att kvantifiera fortplantning av osäkerheter i indata till osäkerheter i utdata, t.ex. radians, vid predikteringar av optiska signaturer. Den andra huvudinriktningen fokuserar på att analysera och validera skillnader mellan simulerade och uppmätta bilder, i t.ex. det visuella eller IR, genom att beräkna och jämföra ett antal texturmått såsom exempelvis bildmått och Gabormått. I synnerhet sådana egenskaper hos bilder som är viktiga för detektion, klassificering och identifiering av mål i bakgrund studeras.

Arbetet på metoder för felfortplantning har resulterat i att ett antal metoder har föreslagits. Vi presenterar här ett första testfall där vi använder en enklare felfortplantningsmetod. Resultaten kan sägas vara enligt förväntan men för att avgöra resultatens giltighet krävs ytterligare studier, med t.ex. mer avancerade felfortplantningsmetoder.

I arbetet på att validera simulerade och mätta bilder har ett antal texturmått föreslagits för att analysera skillnader mellan simulerade och mätta bilder. I ett inledande exempel har ett antal mått beräknats för en uppmätt och motsvarande simulerad bild. Detta arbete pågår och kommer att fortsätta under år 2005.

Nyckelord

modellering, validering, osäkerhet, felfortplantning, Monte Carlo, IR, signatur, texturmått

Övriga bibliografiska uppgifter	**Språk** Engelska
ISSN 1650-1942	**Antal sidor:** 42 s.
Distribution enligt missiv	**Pris: Enligt prislista**

CONTENTS

1 INTRODUCTION

Computer programs for prediction of optical signatures of targets and backgrounds are valuable tools in signature assessment and signature management. Simulations make it possible to study optical signatures from targets and backgrounds under conditions where measured signatures are missing or are incomplete. In order to obtain confidence in computational predictions of optical signatures it is important that the "error", or uncertainty, in these predictions can be credible bounded. The process of validating a model or computer program, where measurements are compared to computational predictions, provides the basis for characterising the predictive capabilities of models. This process of validations and predictions is fundamentally statistical in several respects. For instance, there are statistical uncertainties in the input data used in the simulations as well as in the measurement data to which the computational predictions are compared.

The computational results from simulations of optical signatures are often obtained as "images" (radiance maps in some wavelength band). Furthermore, measurements (using cameras or IR sensors) in a wavelength band of the optical (ultraviolet, visual and infrared) region are usually also presented as (calibrated) images. In assessing optical signatures a main interest is to find features in such images that are important when it comes to detection, classification and identification. Examples of such features can be edge concentration, low-, mid- and high-pass energy and total energy. Since it is the features that are relevant for assessing signatures, it is natural that it is these features that are studied in a validation of computer programs for prediction of optical signatures.

At FOI several commercial programs have been used for optical signature predictions (see Ref 1 and Ref 2). Two of the commercial optical signature prediction programs available at FOI are CameoSim and RadThermIR. CameoSim is an advanced IR program aiming at producing high fidelity physics based images originally applied to camouflage assessments. RadThermIR is a 3-dimensional (with some restrictions) heat transfer program that uses Finite Difference Methods to predict the temperature distribution for a target and after that also predict the IR radiance. Earlier works on validating the commercial signature prediction programs at FOI are summarized in Ref 1. The model predictions in these earlier validation efforts were performed in a deterministic manner and therefore the uncertainty in the predictions were unaccounted for. Furthermore, textural features, except the lowest order statistical moments, were not analyzed in these validation works.

In this report we will present the progress on developing methods for validation of computer programs for simulation of optical signatures. The development is based on two of the model validation aspects outlined above, namely the (statistical) assessment of uncertainty propagation in computational predictions and the validation of image features. In Sections 2-5 we will present the work on developing and implementing methods for analyzing and quantifying the propagation of input data uncertainties to output data parameters in predictions of optical signatures. Then, in Section 6, we will present the progress on validating image features and, finally, in Section 7 we will summarize and outline the continued work on these methods.

2 VALIDATION AND UNCERTAINTY QUANTIFICATION

In this section we will briefly discuss the different sources of error and uncertainty in a model computational prediction. We will also touch upon how these enter the process of validating optical signature simulation software against measurements.

2.1 SOURCES OF UNCERTAINTY AND VALIDATION OF MODELS

There are usually several sources for uncertainty in the output from a computer model prediction. In this report we will focus on the uncertainty in the result from a computer simulation that can be characterized as being due to *uncertainties in the model inputs*. *Model uncertainties and errors*, due to the structure or form of the actual model, and *implementation errors,* due to errors and approximations in the (numerical) implementation of models, of course also contribute to the uncertainty, or error, of a model simulation. However, here we will not address model uncertainties and implementation uncertainties explicitly. The different classes of input data used in a typical simulation of optical signatures are illustrated in Figure 1.

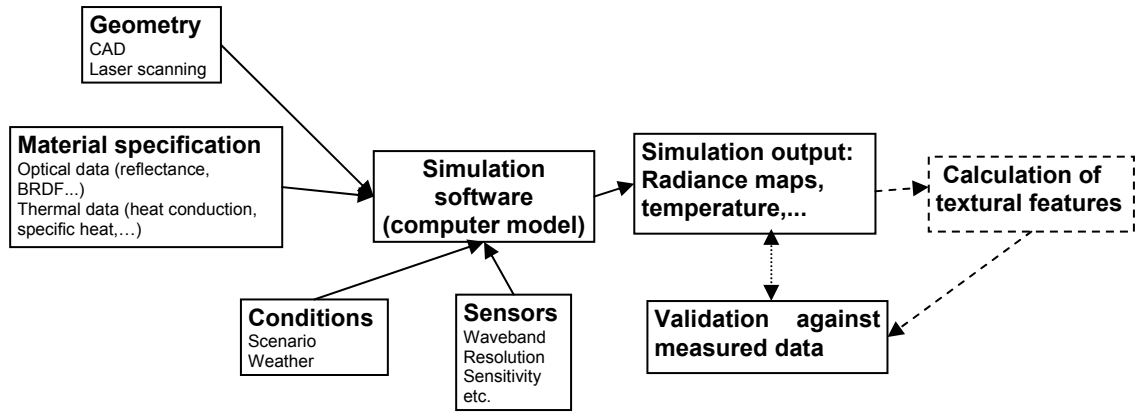


Figure 1 Computational prediction and validation of optical signatures.

Considering the schematic computer model simulation of optical signatures in Figure 1, we can symbolically view the model (simulation software) as a deterministic mapping (function), $\mathbf{y} = \mathbf{f}(\mathbf{x})$, of input parameters, $\mathbf{x} = (x_1, x_2, ..., x_n)$, to one or more output parameters, $\mathbf{y} = (y_1, y_2, ..., y_m)$. If the input data have uncertainties, these will propagate through the model to uncertainties in the output data. We therefore consider the input data as random variables $\mathbf{X} = (X_1, X_2, ..., X_n)$, and the output data as random variables $\mathbf{Y} = (Y_1, Y_2, ..., Y_m)$, related to the input variables through $\mathbf{Y} = \mathbf{f}(\mathbf{X})$, see Figure 2.

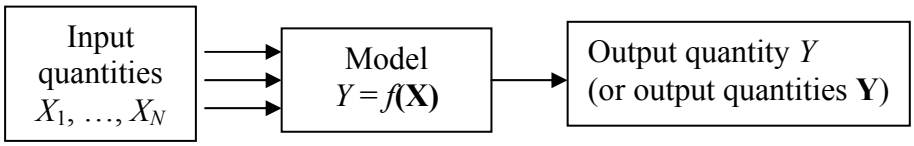


Figure 2 Mapping of input random variables to output random variables.

The schematic computational prediction procedure illustrated in Figure 2 results in output quantities being characterized as random variables. These random variables represent the uncertainty in the prediction due to uncertainties in the input variables. However, these uncertainties do not represent the entire uncertainty or error in a model prediction, as explained

above. Therefore, the (only) way to asses the total prediction error is to compare predictions to measurements, which themselves may have uncertainties. These comparisons can then form the basis for assessing to what extent the total prediction uncertainty or error is due to uncertainty propagation from input parameters and how much uncertainty is due to other sources, such as deficiencies in the form of the models themselves. We will not describe the process for validating computational predictions against measurements in any detail since our focus here will be on the uncertainty propagation aspects. However, an overview of important issues in validation of computer models is given in Ref 3.

2.2 MODEL INPUT UNCERTAINTY QUANTIFICATION

Input data can have uncertainties due to a number of reasons. Commonly, input values are uncertain because they are more or less well-founded guesses, so called guessimates, or they are estimated from measured data, or the input parameters do not actually correspond to observable quantities, e.g. in lumped-parameter models.

According to the BIPM/ISO Guide, Ref 4, uncertainties in measured quantities are either determined through statistical analysis of a series of observations, *Type A uncertainties*, or the uncertainties are evaluated by other means, including expert judgement, *Type B uncertainties*.

In the case of Type A uncertainties a set of q independent observations of input parameter X_i are available and statistical moments of these input quantities can be calculated. For our simulations it is enough to calculate estimates of the sample mean, $\bar{x}_i$, the standard deviation of the sample mean, σ_i, the sample covariance of dependent variables, $\mathrm{cov}(x_i, x_j)$, and the degrees of freedom, $\nu_i = q - 1$. These statistical quantities are calculated in the standard way as presented in for instance Ref 5 or Ref 6.

When using the Monte Carlo procedure for evaluating uncertainty propagation in simulations (to be presented in Section 4) it is also necessary to associate *probability density functions (pdf's),* with statistical moments as above, to the input quantities X_i. In the case of Type A uncertainties, Gaussian pdf's, or Poisson pdf's in some discrete cases, are in many cases the best choices but exceptions to this can be found. If some or all of the input quantities are correlated, joint (multivariate) pdf's should be assigned.

In the case of input quantities with Type B uncertainties, the BIPM/ISO Guide, Ref 4, makes several recommendations on obtaining uncertainties. All of these recommendations have in common the establishment of a range of possible true values of the quantities in question. For instance we might know that a parameter x can only obtain values in the range $[\bar{x} - d, \bar{x} + d]$. In this case we can associate for instance either a rectangular or a triangular distribution to the range of possible values. In the case of a rectangular distribution the mean is $\bar{x}$ and the standard deviation is $d/\sqrt{3}$. Similarly, for the triangular case the mean is $\bar{x}$ and the standard deviation is $d/\sqrt{6}$. It is often assumed that the Type B uncertainties are known and therefore they have infinite degrees of freedom, i.e. $\nu_i \to \infty$.

3 THE ERROR PROPAGATION METHOD

In this section we will briefly present a standard technique for analyzing the propagation of uncertainty in a model $Y = f(\mathbf{X})$ of the type described in Section 2.1. This type of method usually only provide information of mean and variance of the output parameters based on approximations.

3.1 ERROR PROPAGATION AT FIRST AND SECOND ORDER

A well established method for evaluating the uncertainty in $Y = f(\mathbf{X})$ due to uncertainties in the input quantities $\mathbf{X}$ is through the (lineraized) error propagation approach. This method (first order expansion) is the standard approach for analyzing measurement uncertainty in the BIPM/ISO Guide, Ref 4.

Assuming the existence and continuity of the first and second order partial derivatives of the function f we can expand $Y = f(\mathbf{X})$ in a second-order Taylor series about the mean value $\bar{\mathbf{x}}$:

$$Y = f(\mathbf{X}) \approx f(\bar{\mathbf{x}}) + \sum_{i=1}^{n} \frac{\partial f}{\partial X_i}(X_i - \bar{x}_i) + \frac{1}{2!}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 f}{\partial X_i \partial X_j}(X_i - \bar{x}_i)(X_j - \bar{x}_j) \qquad (1)$$

If we consider Eq. (1) up to *linear terms (first order)* it follows that (see for instance Ref 6):

$$\bar{y} \approx f(\bar{\mathbf{x}}) \text{ and}$$

$$\sigma_y^2 \approx \sum_{i=1}^{n} c_i^2 \sigma_{x_i}^2 + 2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} c_i c_j \operatorname{cov}(X_i, X_j), \text{ where } c_i = \left.\frac{\partial f(\mathbf{X})}{\partial X_i}\right|_{\mathbf{X}=\bar{\mathbf{x}}} \qquad (2)$$

From Eq. (2) we can approximately calculate the mean $\bar{y}$ and the variance σ_y^2 of the output parameter Y provided we know the means, variances and covariances of the input parameters, and the *sensitivity coefficients* (partial derivatives), c_i, of the considered model.

If we keep also the second order terms in Eq. (1) we obtain a *second order* analogue to Eq. (2) given by:

$$\bar{y} \approx f(\bar{\mathbf{x}}) + \frac{1}{2!}\sum_{i=1}^{n} \frac{\partial^2 f}{\partial X_i^2}\sigma_{x_i}^2 \text{ and}$$

$$\sigma_y^2 \approx \sum_{i=1}^{n}\left(\frac{\partial f}{\partial X_i}\right)^2 \sigma_{x_i}^2 + \left(\frac{1}{2!}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 f}{\partial X_i \partial X_j}\sigma_{x_i}\sigma_{x_j}\right)^2 \qquad (3)$$

In Eq. (3) we have assumed that the variables X_i are independent so that their mutual covariances are zero.

It is clear that Eqs. (2) and (3) are only approximations of the statistical moments. The linerarized error propagation, which is the standard method in the BIPM/ISO Guide, Ref 4, is of course only exact if $Y = f(\mathbf{X})$ is a linear function of $\mathbf{X}$. A method through which the accuracy of the linearized version of error propagation can be improved has been proposed in Ref 7**.** The basic idea in this approach is to determine "the best" way to obtain additional samples

for calculating the derivative, $\partial f / \partial X_i$, of the response with respect to the *i*'th variable. In the standard linear error propagation partial derivatives are calculated at the nominal point $\bar{\mathbf{x}}$, but in Ref 7 the response with respect to a variable is the weighted sum of derivatives at three nearby points around the nominal. Therefore, the total variance of a system response is formulated as follows:

$$\sigma_y^2 \approx \sum_{i=1}^{n}\left[\sigma_{x_i}^2 \sum_{j=1}^{3}\left\{\phi_j\left(\left.\frac{\partial f(\mathbf{X})}{\partial X_i}\right|_{\mathbf{X}=\mathbf{x}^{(j)}}\right)\right\}\right] \tag{4}$$

where the weights $\phi_j, j \in \{1,2,3\}$ are normalized to $\sum_{j=1}^{3}\phi_j = 1$ and the derivatives are calculated at the three points $\mathbf{x}^{(1)} = \bar{\mathbf{x}} - \boldsymbol{\delta}$, $\mathbf{x}^{(2)} = \bar{\mathbf{x}}$ and $\mathbf{x}^{(3)} = \bar{\mathbf{x}} + \boldsymbol{\delta}$. In Ref 7 a methodology for determining the parameters $\boldsymbol{\delta}$ and $\phi_j, j \in \{1,2,3\}$ is presented for the case when $\mathbf{X}$ follow a normal distribution or a uniform distribution.

We conclude this subsection by noting that the function *f* in our model $Y = f(\mathbf{X})$ is usually not known explicitly in the applications we have in mind here. This is so because in our applications *f* represents the combined result of many physical models and numerical algorithms implemented in complex software for computational predictions of optical signatures. This means that the sensitivity coefficients (partial derivatives) cannot be calculated analytically, but a numerical method has to be applied. Here we present two finite-difference formulas for calculating the sensitivity coefficients, Ref 8:

$$\left.\frac{\partial f(\mathbf{X})}{\partial X_i}\right|_{\mathbf{X}=\bar{\mathbf{x}}} \approx \frac{f(\bar{\mathbf{x}} + \Delta x_i \hat{\mathbf{x}}_i) - f(\bar{\mathbf{x}})}{\Delta x_i} \tag{5}$$

$$\left.\frac{\partial f(\mathbf{X})}{\partial X_i}\right|_{\mathbf{X}=\bar{\mathbf{x}}} \approx \frac{f(\bar{\mathbf{x}} + \Delta x_i \hat{\mathbf{x}}_i) - f(\bar{\mathbf{x}} - \Delta x_i \hat{\mathbf{x}}_i)}{2\Delta x_i} \tag{6}$$

where $\hat{\mathbf{x}}_i$ is the unit base vector for coordinate x_i. The formula in Eq. (5) has an error on the order of Δx_i while Eq. (6) results in an error on the order of $(\Delta x_i)^2$. However, Eq. (6) requires one more evaluation of *f* than Eq. (5): in our case one more simulation run of the signature software. The increment Δx_i should in principle be chosen as small as possible but not so small that it leads to large cancellation errors. Even better approximations, than Eq. (6), of the derivatives can of course be obtained, by for instance Richardson extrapolation, but a rough "rule of thumb" is to use Eq. (6) with $\Delta x_i = \sigma_{x_i}$ and in our example in Section 5 we will use this choice.

3.2 LIMITATIONS OF ERROR PROPAGATION AND EXPANDED UNCERTAINTY

The error propagation approach has some important limitations. One such limitation was mentioned above, namely the truncation of a Taylor's series expansion until first (or second) order terms. This is an approximation that in some cases could need higher order terms to reach sufficient accuracy. Another limitation is that the error propagation method provides no information about the pdf for the output parameter *Y*. Knowledge about the pdf, or the corresponding distribution function, for *Y* is needed to determine confidence intervals for *Y*. In

practical applications of the error propagation method, the distribution of Y is often taken as Gaussian. Therefore, the *expanded uncertainty* $U(Y)$ is computed as the product of the coverage factor k and the standard deviation σ_y so that $U(Y) = k\sigma_y$, the coverage factor being assimilated to the Gaussian variate. Thus, it is very common to find reported uncertainties obtained using a coverage factor $k = 2$, which gives a level of confidence of 95.45% and the reported uncertainty is given as $\bar{y} \pm U(Y)$. Similarly, if the pdf of Y is considered to be approximated by a Student's distribution, then the coverage factor k is taken as the tabulated Student's t-value for a given significance value and effective degrees of freedom, ν_{eff}. The effective degrees of freedom is calculated from the Welch-Satterthwaite equation which is given by (Ref 9):

$$\frac{\sigma_y^4}{\nu_{eff}} = \sum_{i=1}^{n} \frac{\left(\partial f / \partial X_i\right)^4 \sigma_{x_i}^4}{\nu_i} + 2\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{\left(\partial f / \partial X_i\right)^2 \left(\partial f / \partial X_j\right)^2 \operatorname{cov}^2(x_i, x_j)}{\sqrt{\nu_i \nu_j}} \tag{7}$$

As was mentioned in Section 7, Type B uncertainties usually are assumed to contribute with infinite degrees of freedom. This rather arbitrary choice can also be seen as a limitation of the error propagation method.

In Section 4 we will consider methods which are more general than the ordinary error propagation approach and which overcome most of the limitations of the error propagation approach.

4 MONTE CARLO SIMULATION AND RELATED METHODS

In Section 3 we considered the error propagation method which in many cases, with only a small number of model evaluations, can give satisfactory estimates of the uncertainties in output parameters which are due to uncertainties in input parameters. In this section we will consider so called Monte Carlo methods, which are more general than, and can overcome most of the limitations of, the ordinary error propagation method. However, the benefits obtained with Monte Carlo methods come at a prize: they are in many cases computationally expensive.

4.1 MONTE CARLO METHODS IN UNCERTAINTY ANALYSIS

In the standard error propagation approach presented in Section 3, uncertainties in input parameters characterized by standard deviations (and means) are propagated through the model to provide approximate values on mean and standard deviations in the output quantities. In Monte-Carlo methods (see for instance Ref 5, Ref 8 and Ref 10), on the other hand, the input quantities are given as random variables with pdf's. The input pdf's are then propagated through the model to form a pdf for the output quantity. Therefore, Monte Carlo analysis is a tool for combining 'distributions' and thereby, propagating more than just statistical uncertainties. This situation is illustrated in Figure 3.

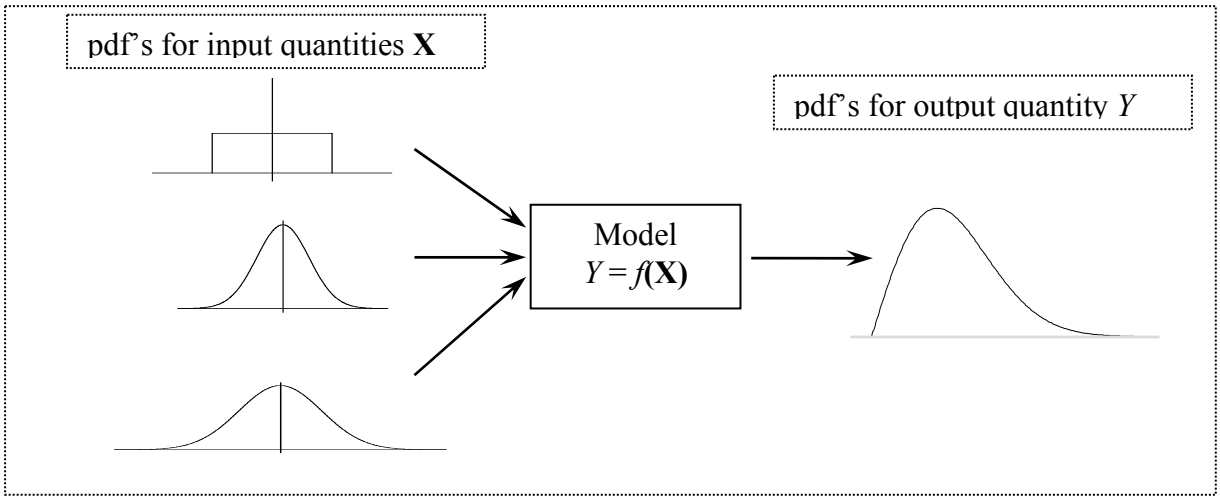


Figure 3 In Monte Carlo methods the uncertainty analysis is performed in terms of pdf's. Distributions are propagated through the model.

The Monte-Carlo method is a rather general numerical procedure which relies on the simulation of random variables. Monte Carlo simulation is for instance an efficient method for performing numerical integration of (in particular higher dimensional) integrals (see for instance Ref 8 and Ref 10). The values of any random variable with any pdf can be 'simulated' by the suitable transformation of a rectangular random variable, uniformly distributed within the interval (0,1). So called pseudo-random numbers, η, uniformly distributed within the interval (0,1) can be successfully generated by a number of computational algorithms found in the literature (see Section 4.2). From these pseudo-random numbers, the values ξ of any other random variable X with a pdf $p(x)$ within the interval (a,b) can be simulated by solving the integral equation:

$$\int_a^{\xi} p(x)dx = F(\xi) = \eta \qquad (8)$$

In Eq. (8) $F(\xi)$ is the cumulative distribution function for the random variable X. Numbers ξ distributed according to $p(x)$ are therefore given by $\xi = F^{-1}(\eta)$. These calculations can be rapidly performed thanks to the high speed of modern computers by using suitable software.

Evaluation of propagation of uncertainty in input parameters using Monte Carlo simulations can be summarized in the following steps:

1. The computer model is identified with a function $Y = f(\mathbf{X})$ where $\mathbf{X} = (X_1, X_2, ..., X_n)$ are n random input variables and Y is a random output variable. (Several output variables can also be treated.)
2. Identify pdf's $p(x_i)$ for all relevant input parameters (see Section 2.2). In the case of correlated random variables X_i and X_j, their values are sampled from a joint probability density function $p(x_i, x_j)$.
3. Select the number M of Monte Carlo trials.
4. Generate M samples $\{x_{i1}, x_{i2}, .., x_{iM}\}$ of each x_i using, for instance, Eq. (8). These x_i are then considered as (pseudo) random numbers drawn from a pdf $p(x_i)$.

5. Compute M results $\{y_1, y_2, .., y_M\}$ by evaluating the model $Y = f(\mathbf{X})$ for the M samples $\{x_{i1}, x_{i2}, .., x_{iM}\}$, i.e. in our case run the optical signature simulation software M times. The results $\{y_1, y_2, .., y_M\}$ are then considered to be distributed like the random output variable Y.

In step 2 above, the number of Monte Carlo trials, M, should be a "sufficiently" large number. What a "sufficiently large" number is depends on for instance the number of input parameters, the shape of the pdf for the output quantity, the pseudo random number generator used, and the level of probability required. However, a value of the order of M=50000 is according to the experience of some authors, Ref 5, often enough to deliver a two-figure accuracy in providing a 95% coverage interval in Y.

By choosing a bin-width for output variable Y and counting the number of simulation results $\{y_1, y_2, .., y_M\}$ in the different bins, the pdf for the output quantity Y can be estimated, see Figure 4.

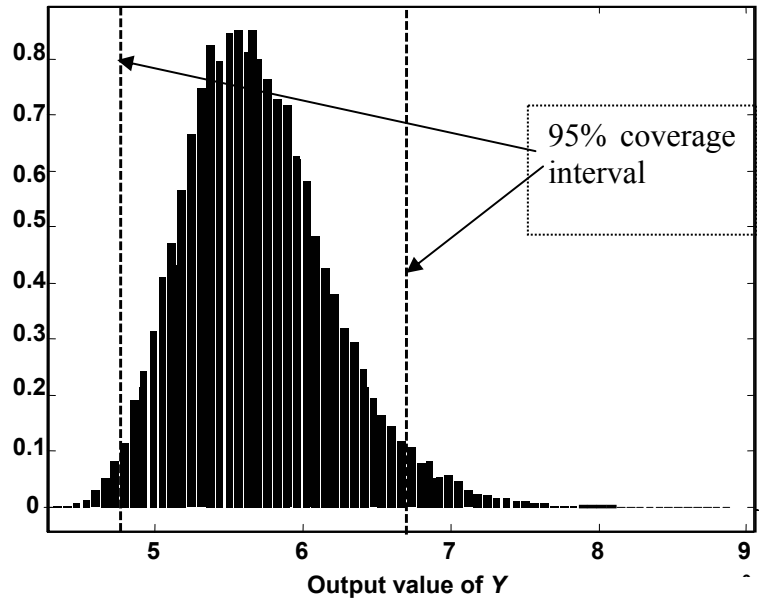


Figure 4 Example of an estimated pdf from the result of a Monte Carlo simulation.

From the estimated set of results $\{y_1, y_2, .., y_M\}$ the variance σ_y^2 of the output quantity Y can be calculated by:

$$\sigma_y^2 = \frac{\sum_{i=1}^{M} y_i^2 - \left(\sum_{i=1}^{M} y_i\right)^2 / M}{M - 1} \tag{9}$$

If a coverage probability, p, is selected, the confidence interval for the result can be evaluated as $y_{(1+p)M/2} - y_{(1-p)M/2}$, where we have **sorted** the y values into non-decreasing order. When the skewness value of the calculated discrete distribution for Y is near zero, the confidence interval becomes symmetric and the expanded uncertainty $U(Y)$ can be approximated by

$$U(Y) = \frac{y_{(1+p)M/2} - y_{(1+p)M/2}}{2} \tag{10}$$

The corresponding coverage factor can then also be calculated by $k = U(Y)/\sigma_y$.

We mentioned in the beginning of this section that the most severe limitation of the Monte Carlo method is that the model simulation runtime could be long in some complex cases. Single runs of optical signature simulation software can take from a few seconds up to hours (or even more) in run time on a standard single processor PC (Personal Computer), depending on the complexity of the problem solved. Therefore it is clear that, for instance, 50000 Monte Carlo runs can be a problem in terms of runtime. However, Monte Carlo simulations are well suited for distribution on many computers/processors. If we, for instance have 10 identical computers available and will perform M= 50000 simulations, the simulations can easily be distributed so that "only" 5000 of them are run on each computer and the computing time will then essentially decrease to one tenth of the computing time on a single computer.

Another possible approach to attack the problem with expensive (with respect to time) Monte Carlo simulations can in some cases be to use a response surface methodology, Ref 11, where the computer model (signature simulation program) is replaced by a generated response surface (in some domain of the input parameter space), which can be evaluated much faster than the original computer model. We will not discuss this approach here since it is not likely that we will make use of it in our applications.

4.2 PSEUDO RANDOM NUMBER GENERATORS AND OTHER SAMPLING TECHNIQUES

As was explained in Section 4.1 the ability to generate pseudo-random numbers from a (continuous) rectangular distribution is fundamental in Monte Carlo methods, since it is the basis for generating numbers from any distribution using a formula like Eq. (8). It is therefore especially important that the underlying rectangular pseudo random number generator is sound. A large number of pseudo number generators can be found in the literature. A good review of several pseudo number generators for use in Monte Carlo methods is given in Ref 12. The Hill-Wichmann generator, Ref 13, is a combination of three so called congruential generators. This generator is one of the pseudo random number generators recommended in Ref 12 for generating rectangular pseudo-random numbers in the interval [0, 1]. We have implemented the Hill-Wichmann pseudo random number generator and in Figure 5 we show an example of pseudo random numbers generated using this routine.

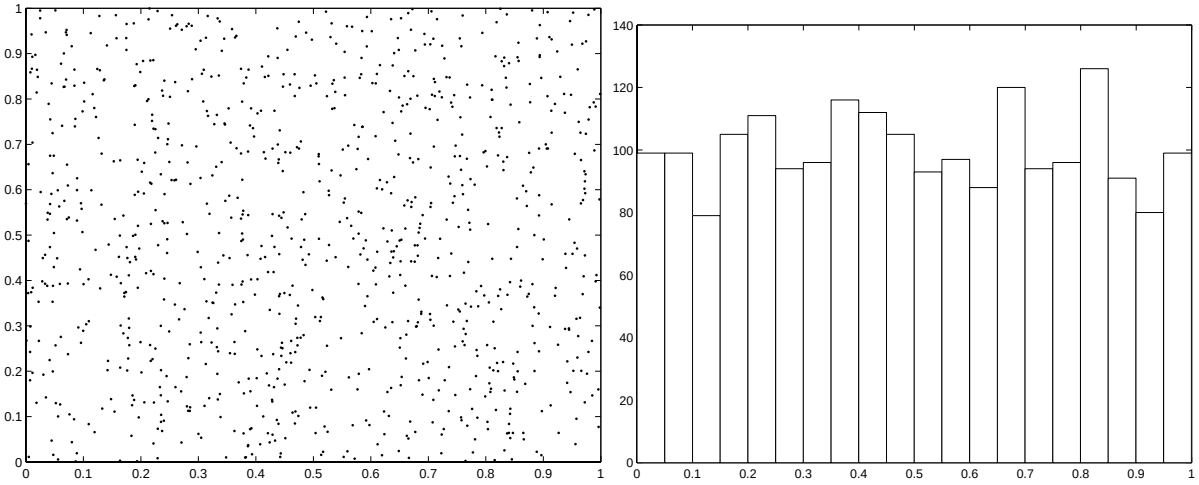


Figure 5 1000 pseudo random numbers generated in each of two dimensions with the Hill-Wichmann generator. Left figure shows a 2D scatter plot and the right figure shows a frequency histogram for the 2000 numbers.

Once we have decided on a sound generator for generation of pseudo random numbers from a rectangular distribution, Eq. (8) or other procedures based on the principle of Eq. (8) can be used to generate pseudo random numbers from other distributions. In Ref 5 and Ref 10 easy-to-use procedures for obtaining values from the most common distributions (e.g. Gussian, multivariate Gaussian and Student's-t) are summarized. For instance, the Box-Muller transform can be used to generate pairs of pseudo random numbers distributed according to $N(0,1)$ (Gaussian distribution with zero mean and standard deviation 1). In the Box-Muller transform two independent standardized Gaussian variates, z_1 and z_2, are obtained from two standardized (uniformly distributed in [0,1]) random variates x_1 and x_2 through the formulas $z_1 = \sqrt{-2\ln x_1}\cos 2\pi x_2$ and $z_2 = \sqrt{-2\ln x_1}\cos 2\pi x_2$. Samples from a Gaussian distribution $N(\mu,\sigma^2)$ are easily obtained by taking $\widetilde{X} = \mu + \sigma Z$, where $Z \in N(0,1)$.

We will finish this section by briefly presenting samplings schemes which are closely related to the ordinary Monte Carlo method using pseudo random numbers. These alternative sampling techniques are often referred to as *Quasi Monte Carlo Simulations.* What separates Quasi Monte Carlo methods from ordinary Monte Carlo, is that Monte Carlo use (pseudo) random numbers as explained above while Quasi Monte Carlo methods use number sampling schemes where the requirement of "randomness" has (partly) been abandoned. In fact, in our intended use of Monte Carlo, just like in the use of Monte Carlo for numerical integration, the true randomness of the generated numbers is not so much relevant. More important is to sample the uniform distributions as uniformly as possible under certain restrictions. For a more "strict" discussion on the condition of uniformity see for instance Ref 10. One can show, Ref 10, that Monte Carlo used for numerical integration with pseudo random numbers has an error scaling as $1/\sqrt{M}$, where M is the number of pseudo random number samples. By replacing the pseudo random numbers with carefully chosen sequences called *quasi random number sequences* it has been shown (see for instance Ref 10) that the integration error scales as $(1/M)\ln^p M$ for some p, i.e. a considerable improvement compared to ordinary Monte Carlo. In this presentation we will not present the precise mathematical arguments for the advantage, i.e. requiring less simulations, Quasi Monte Carlo has over ordinary Monte Carlo in our particular applications. However, it should be quite easy to imagine that the more uniformly distributed quasi random sequences will determine, especially the tails of, the distri-

bution of an output parameter (called Y in previous sections) with significantly fewer simulations than ordinary Monte Carlo.

An example of a quasi random number sequence is the so called Halton sequence. The Halton sequence and other quasi random number sequences can be found in Ref 10. We have implemented the (two-dimensional) Halton sequence and in Figure 6 we show the results corresponding to those for ordinary pseudo random numbers in Figure 5. From these two figures we see that the Halton sequence does indeed distribute the numbers more uniformly than pseudo random numbers.

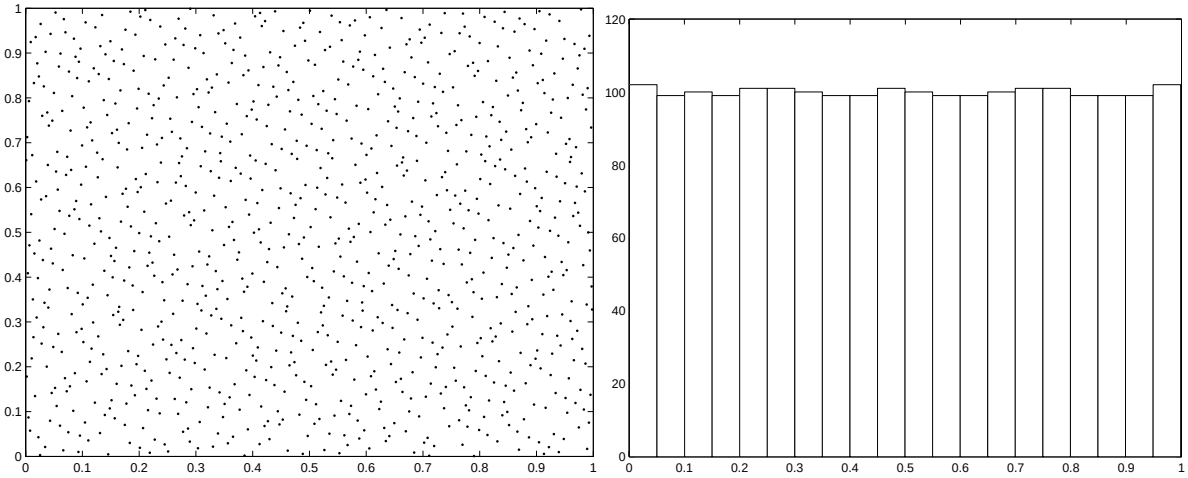


Figure 6 1000 quasi random numbers generated in each of two dimensions using the two-dimensional Halton sequence. Left figure shows a 2D scatter plot and the right figure shows a frequency histogram for the 2000 numbers.

A reasonable question to ask is: If the more uniformly distributed Quasi random numbers result in a faster convergence than random numbers, then why not use a completely regular grid (so called lattice models) in d dimensions to define sampling points? If d is small (less than about 3) a regular grid may indeed be preferable. However, if we span a regular grid in d dimensions with N points per dimension, the regular grid consist of N^d data points. This will, in general, for higher dimensions lead to a much larger number than the necessary number of samplings in Monte Carlo or Quasi Monte Carlo Methods. A method which is based on sampling nodes randomly in a regular grid is the so called Latin Hypercube sampling (LHS), see for instance Ref 9 and references therein. Latin Hypercube sampling has many of the attractive features in common with quasi Monte Carlo methods. We will however not present the Latin Hypercube approach any further here.

5 ERROR PROPAGATION IN THERMAL PREDICTIONS USING RADTHERMIR – A TEST CASE

In this section we will present a first step towards implementing some of the methods presented in Sections 3 – 4 and applying them to uncertainty and sensitivity studies of optical signature simulation software. We will use the linearized error propagation method presented in Section 3.1 to estimate the uncertainty in output from the thermal signature simulation software RadThermIR. The test case we will consider has previously been used in a deterministic validation of signature simulation software.

5.1 THE VALIDATION TEST CASE AND UNCERTAINTIES IN INPUT DATA

We have chosen to illustrate the use of error propagation in validation of signature simulation software by considering the measurements on, and simulations of, one of the two flat tilted panels with different types of coatings which were presented in Ref 2. In Ref 2 the signature simulation programs SensorVision, CameoSim and RadThermIR were validated *deterministically* against temperature and radiance measurements on the panels. In our analysis of the error propagation we will here restrict ourselves to the panel with paint coating and perform uncertainty propagation simulations of it using the program RadThermIR. The panels consist of different layers that form a flat surface of 1 x 1.2 m^2. A cross-section of the panels and a photograph of the panels at the measurement site is shown in Figure 7 (the images are from Ref 2). Radiance measurements in MWIR and LWIR using IR (Thermovision) cameras were performed during a full 24-hour period. Contact temperature measurements, using a Pt 100 temperature sensor mounted inside the front Aluminium sheet close to the front surface, were performed during the same time period and measurements of weather data begun almost 24 hours before the contact temperature and the radiance measurements. During the first day the sky was clear and the weather sunny. The second day the sky was cloudy and at about 13:00 it started to rain mixed with snow. Soon after that the experiment was stopped. Radiance data and weather data were collected with a time interval of about one minute in between individual measurements.

Figure 7 Cross section of the panels and a photo of the two panels on stands (paint panel to the left).

After the measurements were completed, calibrated radiance data were calculated and the radiance was corrected for atmospheric transmission losses, which gave the radiance at the panel surfaces.

In Ref 2 RadThermIR was used to predict surface temperatures and radiance from the panels in the two wavelength bands. In these simulations “best-estimate” (mean) values of the input data were used. In order to analyse the propagation of uncertainties in input data to the predicted output quantities we first have to quantify the uncertainties in input parameters. When applying the linear error propagation formula in Eq. (2) we need estimates of means, variances and, for dependent variables, covariances of the input parameters. Determining these parameters is a difficult task indeed, especially since some parameters have mean values which are “guessimates”. We have tried to follow the guidelines outlined in Section 2.2 but our main objective here is to illustrate the use of the error propagation method and *not* to find the best possible (most realistic) choices of input parameter uncertainties. In Table 1 we have summarized the data we use for input parameters with uncertainties.

Table 1 Input parameter uncertainties.

Input parameter class	Input parameter	Type of uncertainty (A/ B)	Assumed distribution	Estimated mean	Estimated standard deviation
Geometric	Aluminium thickness	B	rectangular	3 mm	0.5/√3=0.29
	Heat foil thickness	B	rectangular	0.5 mm	0.1/√3=0.058
	Divinycell thickness	B	rectangular	40 mm	1/√3=0.58
Environmental	surface normal azimuth	B	rectangular	256°	1/√3=0.58
Thermal	Al conductivity	B	rectangular	201 W/m K	(237-201)/√3=20.7846
	Al density	B	rectangular	2770 kg/m^3	(2770-2700)/√3=40.41
	Al specific heat	B	rectangular	884 J/Kg K	(938-884)/√3=31.18
	Heat foil conductivity	B	rectangular	5 W/m K	0.5/√3=0.29
	Heat foil density	B	rectangular	2000 kg/m^3	10/√3=5.8
	Heat foil specific heat	B	rectangular	1000 J/Kg K	10/√3=5.8
	Divinycell conductivity	B	rectangular	0.024 W/m K	0.002/√3=0.0012
	Divinycell density	B	rectangular	48 kg/m3	(55-48)/√3=4.04
	Divinycell specific heat	B	rectangular	1900 J/Kg K	50/√3=28.8675
	Solar absorptivity of paint	B	rectangular	0.805	0.02/√3=0.012
	Thermal emissivity of paint	B	rectangular	0.8	0.02/√3=0.012
	Thermal conductance of paint	B	rectangular	1x10^{-7} W/cm^2 K	1x10^{-7}/√3=5.8x10^{-8}
	convection coefficient 1	B	rectangular	5.7	0.49/√3=0.2829
	convection coefficient 2	B	rectangular	3.8	0.87/√3=0.5023
Weather	air temperature	B	rectangular	each time step	0.3/√3= 0.173
	solar irradiance	B	rectangular	each time step	5.0%
	Humidity	B	rectangular	each time step	2/√3=1.15
	Wind speed			each time step	0.17

The mean values for all parameters, including those input parameters not listed in Table 1, have been chosen to be the same as those used in the deterministic predictions in Ref 2. One could in principle include other input parameters with uncertainties in this list but in the present analysis we restrict ourselves to these. Rectangular distributions have been assigned to most input parameters. The basis for this is that we have classified the uncertainties as being of type B. For some parameters this choice could be questioned but we leave those considerations for later studies. We have furthermore assumed that all input parameters are *independent* random variables. This is surely not so for some of the parameters, for instance the weather parameters, but for simplicity (and lack of data for evaluating covariances) we work with independent random variables here. Finally, the standard deviations listed in Table 1 have all been estimated from some kind of, more or less reliable, data. We will not present the arguments and data underlying these choices of standard deviations here. However, we can note that the uncertainties in the two convection coefficients in the McAdams convection model, Ref 14, are probably somewhat "under conservative" in the way that the estimates of the uncertainties used here are only based on McAdams data for smooth versus rough surfaces and no other sources of uncertainty in these parameters have been included.

Another complication in determining uncertainties for input parameters in our applications is that some of them really should be characterized as *stochastic processes* and not just as random variables. A stochastic process, $X(t)$, can be described as a parameter dependent random variable. For a particular fixed value of t, the stochastic process is an ordinary random variable. The stochastic process can have an autocovariance (and higher order auto- correlations). Due to thermal inertia, temperatures and radiances predicted with RadThermIR will depend on the weather data histories and not just on the weather at the actual time. The weather input quantities should therefore be viewed as stochastic processes. However, RadThermIR per-

forms its calculations based on discrete time steps, $t = t_0, t_1, ..., t_n$, and therefore we actually have n (in reality n should be chosen as dependent on the thermal time constant) random variables for each weather parameter. However, again it is difficult to assess the dependency of these n random variables. We have therefore taken the "conservative", but perhaps not so appealing, approach that the uncertainties in the n time steps are 100% correlated (in a "worst case" combination). This means that we can calculate sensitivity coefficients with respect to the weather parameters by applying the same variation in the finite difference formula (Eq. (6)) to all time steps.

5.2 RESULTS FOR TEMPERATURE PREDICTIONS AND COMPARISON WITH MEASUREMENTS

By using the input data uncertainties given in Table 1 we have used the linearized error propagation formula in Eq. (2) to calculate the, in RadThermIR, predicted uncertainties in the surface temperature and the radiance for the panel with paint coating. In Eq. (2) we identify the mapping f with the combined algorithms and models in RathermIR. In this evaluation the sensitivity coefficients (partial derivatives) have been calculated using the finite difference formula in Eq. (6) for every (5 minute) time step performed in the RadThermIR simulation. In Appendix A we show plots of all calculated sensitivity coefficients versus time when surface temperature is the output parameter. In Appendix B we show the calculated standard uncertainty in temperature versus time.

In figures Figure 8 to Figure 10 we summarize the results of the error propagation analysis for predicted paint panel surface temperature. In Figure 8 we show the predicted mean temperature and the measured contact temperature versus time. This result is the same as that obtained in Ref 2. In Figure 9 we show the predicted mean temperature plus/minus one standard uncertainty, and the measured temperature, versus time. In Figure 10 we show the predicted mean temperature plus/minus two standard uncertainties, and the measured temperature, versus time.

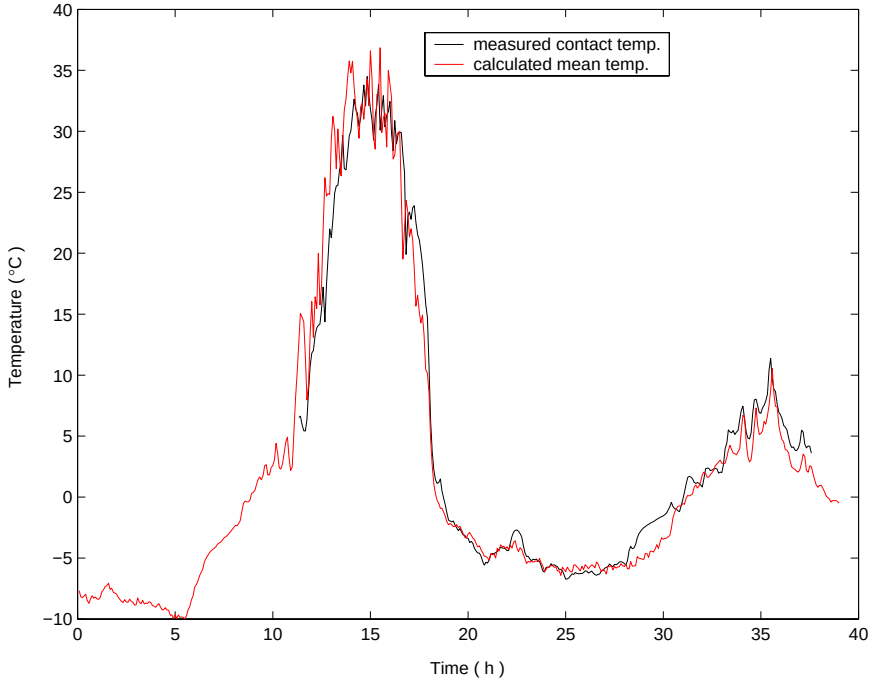


Figure 8 predicted mean temperature and the measured contact temperature versus time.

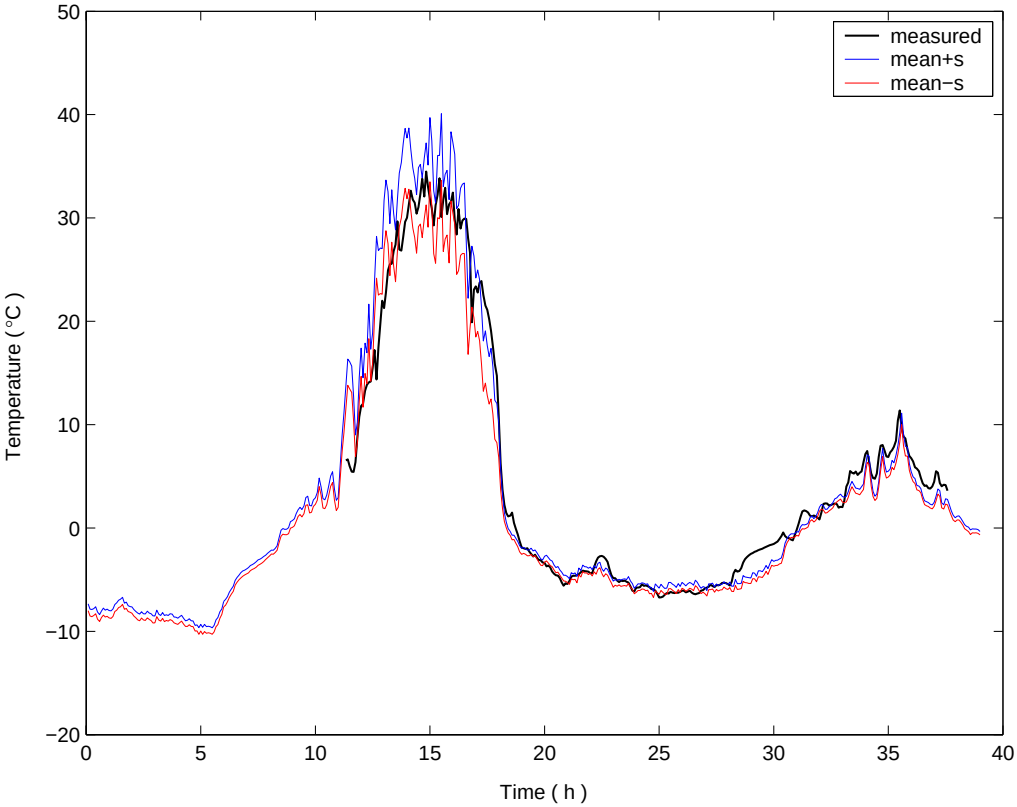


Figure 9 predicted mean temperature plus/minus one standard deviation, and the measured temperature, versus time.

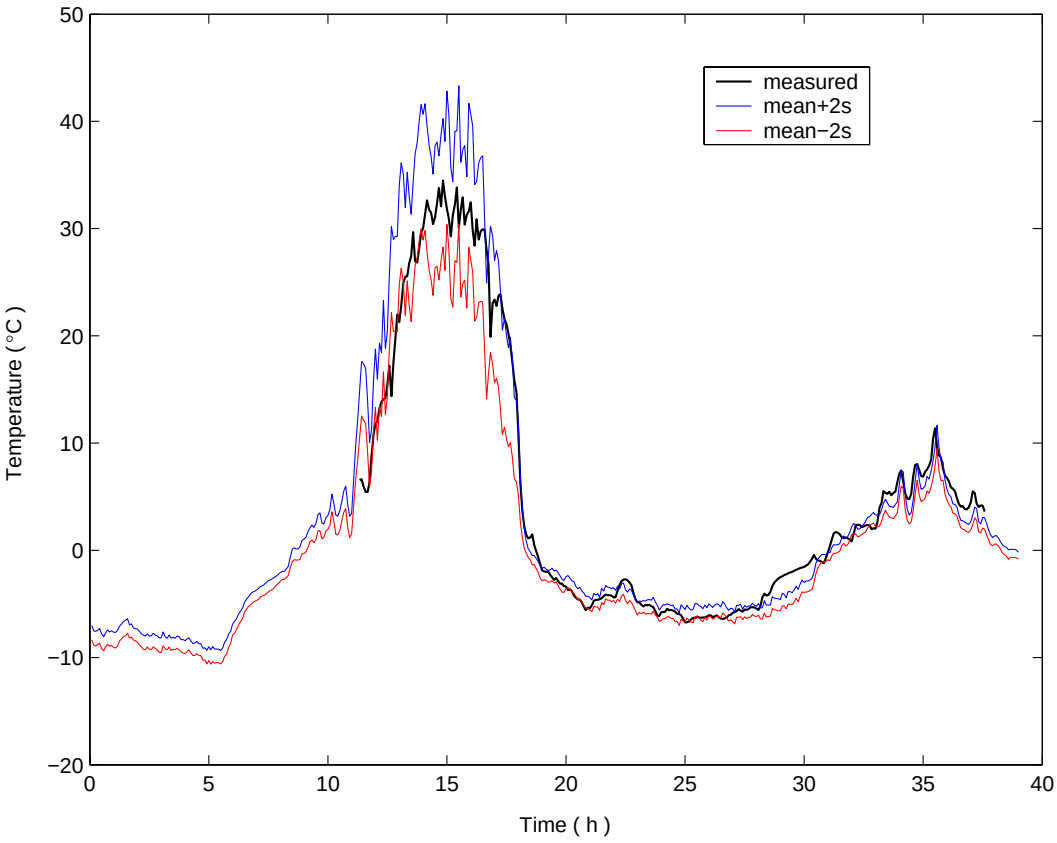


Figure 10 Predicted mean temperature plus/minus two standard deviations and the measured temperature versus time.

In the same way as for panel surface temperatures above, we summarize the results of the error propagation analysis for predicted paint panel radiance in Figure 11 and Figure 12. Furthermore, the calculated standard uncertainty in radiance versus time is displayed in Appendix B. In Figure 11 we show the predicted mean radiance and the measured radiance versus time. In this plot the predicted radiance has been compensated for the lack of sensor response function modelling in RadTherm (referred to as LPL equivalent radiance in Ref 2). The result in Figure 11 is the same as that obtained in Ref 2. Figure 12 shows predicted mean radiance plus/minus two standard uncertainties, compensated to yield a LPL equivalent radiance, and the measured radiance versus time. When considering the results in Figure 11 and Figure 12, one should also bear in mind that there most probably was a constant offset in all radiance (Thermovision) measurement results, as explained in Ref 2. The size of this error was estimated to about 0.97 W/(sr m^2) in Ref 2.

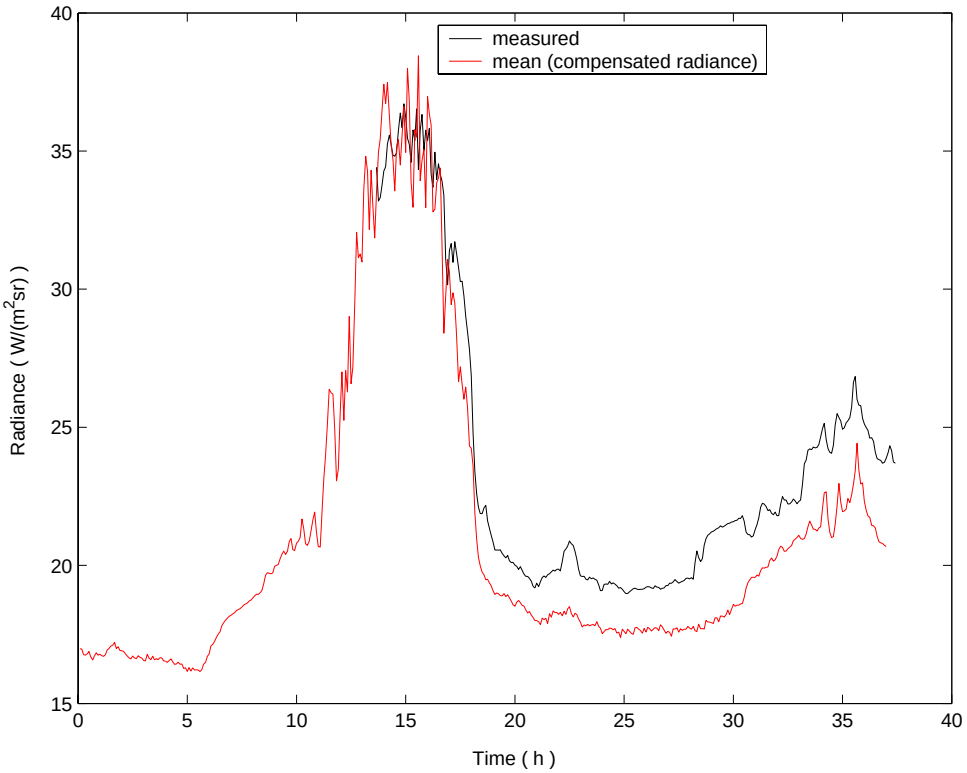


Figure 11 Predicted mean radiance, compensated to yield a LPL equivalent radiance, and the measured radiance versus time.

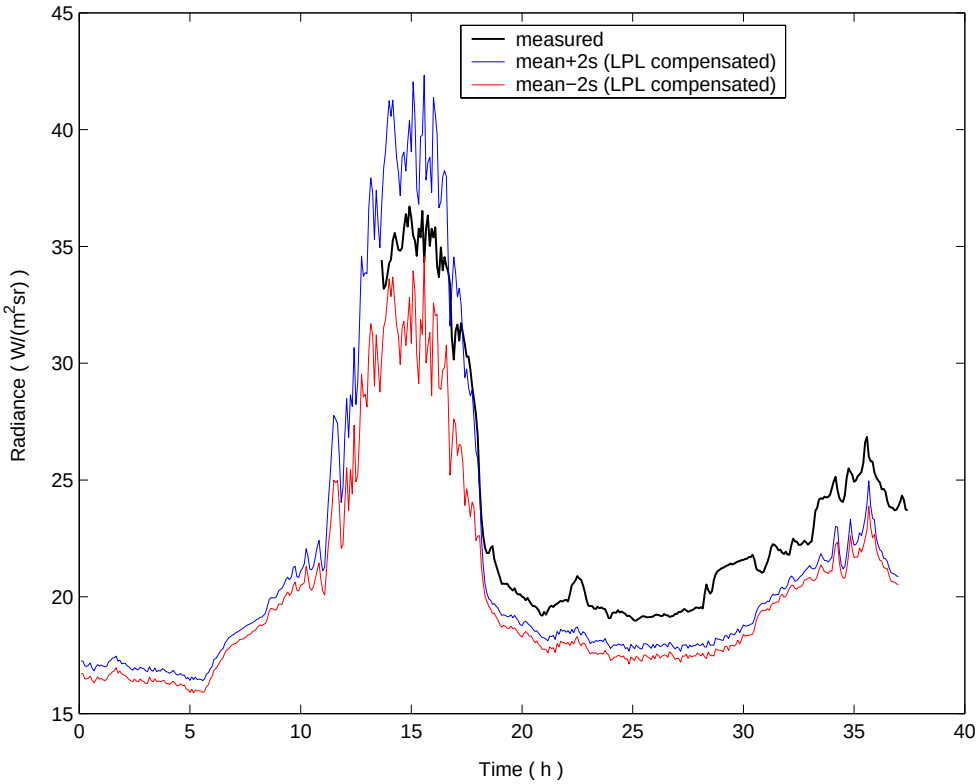


Figure 12 Predicted mean radiance plus/minus two standard deviations, compensated to yield a LPL equivalent radiance, and the measured radiance versus time.

From the results shown in Figure 8 to Figure 12 we see that the majority of measured data are covered by the plus/minus two standard deviations (coverage factor 2) predictions. The fact that not all measurements lie within these limits can be attributed to, for instance:

1. The coverage factor 2 corresponds to the 95% coverage interval for a Gaussian distribution and need therefore not cover all data.
2. The limitations of the error propagation method discussed in Section 3.2. And also numerical errors in calculating the sensitivity coefficients.
3. "Under-conservative" choices of uncertainties in input parameters and systematic "bias" errors in the input parameters. If possible, covariances for dependent parameters should also be used.
4. Model errors and model uncertainties.
5. The measured quantities have an uncertainty. (According to Ref 2 the contact temperature probes have an estimated uncertainty of approximately 0.1 ºC at k=2 and the measured radiance has an uncertainty in the order of 0.1 W/(sr m^2)). Also, as was mentioned earlier there was probably a constant offset in the Thermovision measurement results on the order of 0.97 W/(sr m^2).
6. In the comparison of predicted and measured radiances, the compensation to yield the LPL equivalent radiance imposes an uncertainty (errors).

Of these contributions, item 2 could be studied through performing an uncertainty analysis of RadThermIR using a MonteCarlo type of method (see also Section 7). The presumed problems mentioned in item 3 could be treated by a more careful study of the input parameter values and their uncertainties. For instance, as was mentioned above the uncertainties used for the convection model parameters are probably under-estimated in our analysis.

6 COMPARISON OF MODELED AND SENSOR IMAGES OF VEHICLES

Images of vehicles generated from modeling software, e.g. CAMEO-SIM, and from field registrations differ in certain aspects. The main interest here is to find differences in those aspects that are important when it comes to detection, classification and identification. Here we will mostly deal with detection of vehicles. In this case detection means to find an object, some military vehicle, in the terrain.

6.1 METHODOLOGY

In order to characterize the vehicle, several features are computed from the image. A small local region, usually 16 * 16 pixels, is defined for each pixel. Some features use the image directly, other use the magnitude of the Fourier transform of the image. Image features include mean value, standard deviation and edge concentration. For each region the Fourier transform is computed. Several measures are then computed from this transform. They are formulated to be orientation invariant. The translation invariance comes directly from the Fourier transform. Features include low-, mid- and highpass energy, total energy, and the elongeness is also characterized by computing moments on the Fourier transform. In the latest implementation Gabor wavelets are used. From the local responses invariant features are calculated, similar to the Fourier features. A description of the features is given in Ref 15. An interesting observation in that paper was that a single feature, the deviation of the edge concentration distribution, correlated with perceptual detection experiments with a correlation coefficient of 0.88.

For a more complete description of the vehicle, several features should be used. A demand when comparing with perception experiments has always been that the features should have a simple and direct connection with a physical property. However, this is not the only goal in the current context. We may also want to characterize the differences between modeled and real image when this imagery is used in automatic target recognition.

Previously a lot of work was done on finding a set of good features with high discriminative capability when it comes to detect vehicles in the terrain, see for instance Ref 1. However, images from different wavelength bands may have quite dissimilar feature properties. Therefore several features are used here. Earlier studies, Ref 16, have shown that about 3 to 4 dimensions will be enough if we can choose these dimensions in a good way. This include almost always that the feature space is transformed in some way. To find the importance of each feature, linear discrimination analysis is used Ref 17. We know the position of the target and background features in the feature space. Then it is easy to see how each feature axis is oriented with respect to the discriminant line (linear case). The angle between the axes and this line is an indication of the importance of a feature. The distance measure applied to the target and the background distributions is the Bhattacharrya measure defined as

$$\mathcal{B}(X) = -\ln \int \left[p\left(X|\omega_1\right) p\left(X|\omega_2\right)\right]^{\frac{1}{2}} dX \qquad (11)$$

where ω_1 and ω_j represent the two objects to be compared.
The connection of this measure with detection theory is given in Ref 18 and Ref 19.

If the distributions are identical, we obtain the integral over a probability distribution function, which by definition is 1. After applying the logarithm, the distance will be 0. If the distributions are totally disjunct, we obtain the logarithm of 0 which symbolically can be defined as minus infinity. Even if the distribution in Eq. (11) is parameter free, it is difficult to estimate a continuous distribution from a sampled image.

To be useful let us assume that we have a Gaussian multivariate distribution

$$p(X|\omega_i) = \left[(2\pi)^n |\Sigma_i|\right]^{-\frac{1}{2}} \times \exp\left\{-\tfrac{1}{2}(X-\mu_i)^T \Sigma_i^{-1}(X-\mu_i)\right\} \quad (12)$$

where μ_i och Σ_i are mean value and covariance matrix for the distribution p. This gives

$$B = \frac{1}{4}(\mu_2-\mu_1)^T[\Sigma_1+\Sigma_2]^{-1}(\mu_2-\mu_1) + \frac{1}{2}\ln\left[\frac{\left|\frac{1}{2}(\Sigma_1+\Sigma_2)\right|}{\sqrt{|\Sigma_1||\Sigma_2|}}\right] \quad (13)$$

Since the covariance matrix is used, many features can be used. A nice feature is the inherent normalization which, for example, allows measurements in different wavelength bands to be combined in a simple way.

If the features are independent the distance can be further simplified. We obtain

$$B = \frac{1}{4}\frac{(\mu_{ik}-\mu_{il})^2}{(\sigma_{ik}^2+\sigma_{il}^2)} + \frac{1}{2}\ln\left[\frac{\frac{1}{2}(\sigma_{ik}^2+\sigma_{il}^2)}{\sqrt{\sigma_{ik}^2\,\sigma_{il}^2}}\right] \quad (14)$$

where only mean values and standard deviations are used. For each feature we get a distance. The total distance is simply the sum of the individual Bhattacharrya distances for each feature. This makes it much easier to estimate the distributions.

Since B is a generalized signal to noise ratio, the name GSNR is proposed which simply is defined as

$$GSNR = 4*B. \quad (15)$$

The constant 4 makes GSNR more or less equal to the common snr definition.

According to reference Ref 19 the detection probability may be written as

$$P(d) \approx 1 - exp\{-kB\}, \quad (16)$$

where *d* is the distance and the factor k is the number of independent resolution elements the sensor sees on the target. It can be expressed as

$$k = A/F. \quad (17)$$

where A is the targets area and F is the area of the sensor footprint on the target.

A second measure is called GSNRA, where A indicates that the target and the sensor footprint area are taken into account. The relation is

$$GSNRA = 4*(A/F)*GSNR. \quad (18)$$

Here the factor 4 is reintroduced to make the expression for the approximate detection probability similar to equation (16)

$$P(d) \approx 1 - exp\{-GSNRA\}. \quad (19)$$

The proposed measure GSNRA used to estimate the similarity between the target and the background has a relatively good theoretical foundation and is similar to the classical signal-to-noise ratio. The area of the target and the sensor resolution enter in an intuitive way.

6.2 EXAMPLE

In a simple preliminary example, feature values have been computed for several gabor features. Two vehicles imaged and modeled from above are shown in Figure 13. The image is a montage of a real registration and a CAMEO-SIM model. No effort has been put into modeling the background in this example.

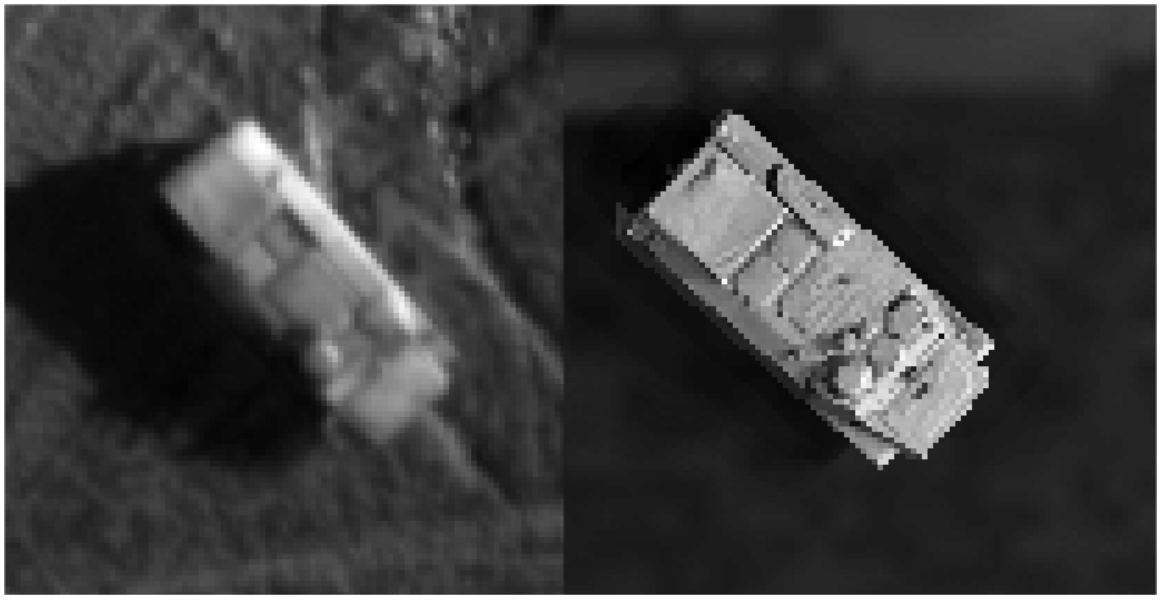


Figure 13 A real (left) and modeled (right) vehicle used for signature comparison.

Table 2 shows some Gabor feature measures. For each feature, the mean value and the standard deviation are computed. In addition some other measures are computed. The contrast is defined as

$$Contrast = (Image_{Mean} - Model_{Mean})/(Image_{Mean} + Model_{Mean}). \quad (20)$$

LDA is an estimation of the weight given to each feature using linear discrimination analysis. The generalized signal-to-noise ration is here calculated for each feature. Gmean is simply the fist term in Eq. (14), while Gdev is the second term. Gtotal is the sum of the two terms. Using all features simulaneosly gives GSNR = 24.2.

Table 2 Feature values for image and modeled vehicle.

Feature	Image		Model		Contrast	LDA	GSNR		
	Mean	Dev	Mean	Dev	(diff/sum)	Weights	Gmean	Gdev	Gtotal
Low band	0.198	0.127	0.640	0.456	0.528	0.159	0.872	1.323	2.195
Med band	0.108	0.073	0.274	0.198	0.436	0.045	0.621	0.876	1.497
High band	0.071	0.051	0.162	0.130	0.388	0.089	0.424	0.773	1.197
Frac dim	2.263	0.163	2.554	0.251	0.061	0.076	0.949	0.183	1.132
Frac err	-8.927	1.845	-3.112	3.738	0.483	0.334	1.946	0.462	2.408
Corr len	2.296	0.214	2.014	0.205	0.065	0.059	0.908	0.002	0.910
Shape	0.242	0.120	0.439	0.239	0.289	0.093	0.543	0.446	0.989
Edge conc	0.073	0.043	0.162	0.090	0.379	0.111	0.808	0.499	1.307
Max	0.099	0.063	0.241	0.169	0.418	0.034	0.620	0.842	1.462

Obviously there is a substantial difference for the fractal error feature. However, it is too early to draw some more definite conclusion. More experiments have to be done.

For the images in Figure 13 some first order statistics have been measured. In Table 3 several measures from the image histogram are shown. Table 4 shows corresponding values for the model.

Table 3 Feature values for the image of the vehicle.

Feature	Min	Max	Mean	Dev	Skewness	Kurtosis
Low band	0.198	1.117	0.492	0.176	4.002	1.534
Med band	0.030	0.364	0.156	0.074	0.237	1.819
High band	0.010	0.117	0.042	0.018	0.004	1.767
Energy	0.275	1.500	0.690	0.250	8.942	1.520
Frac dim	2.118	2.486	2.265	0.087	0.335	1.006
Frac err	-6.542	0.594	-3.59	1.482	1714	1.419
Corr len	1.600	2.300	1.869	0.153	2.517	1.028
Shape	0.057	0.380	0.180	0.060	0.059	1.404
Edge conc	0.022	0.083	0.042	0.013	0.002	1.393
Max	0.058	0.653	0.209	0.107	1.592	2.249

Table 4 Feature values for the model of the vehicle.

Feature	Min	Max	Mean	Dev	Skewness	Kurtosis
Low_band	0.172	1.901	0.849	0.341	17.709	1.598
Med_band	0.069	1.483	0.519	0.244	9.305	1.849
High_band	0.042	0.909	0.343	0.172	2.752	1.897
Energy	0.319	3.795	1.711	0.702	129.1	1.604
Frac_dim	2.391	3.290	2.916	0.190	-2.234	1.016
Frac_err	-6.418	3.473	-0.794	2.181	-2713	2.769
Corr_len	1.600	3.500	2.448	0.423	9.887	1.116
Shape	0.189	1.327	0.783	0.245	-4.447	1.307
Edgeconc	0.095	0.609	0.341	0.113	0.248	1.389
Max	0.051	0.786	0.339	0.145	1.561	1.685

Feature images for the registered image and the model are given in the following Figure 14. The images used for the feature generation are given in Appendix C.

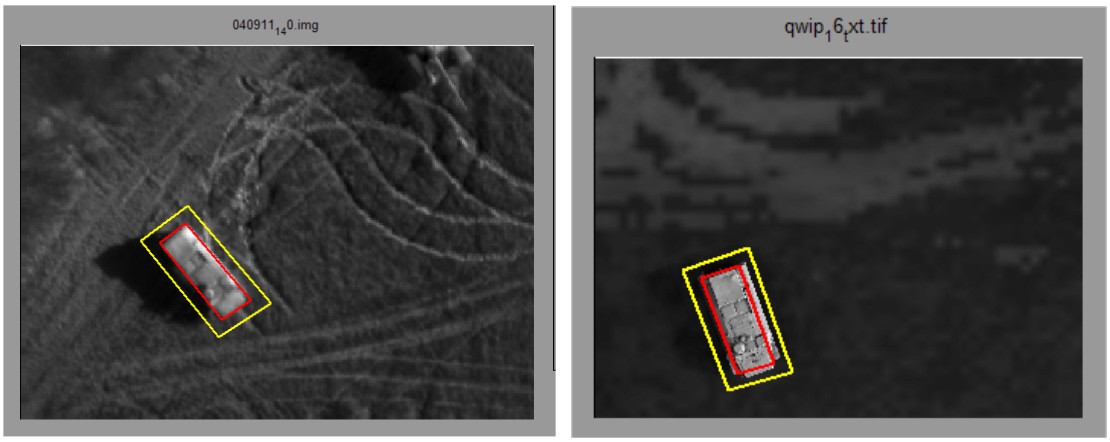


Figure 14 The registered image and the corresponding model.

7 SUMMARY AND FURTHER DEVELOPMENT

In this report we have presented the progress on developing methods for validation of computer programs for simulation of optical signatures. The development is taking two paths to asses different, but related and dependent, aspects of validating the simulation programs. The first path is concerned with developing and implementing methods for analyzing and quantifying the propagation of input data uncertainties to output data parameters (i.e. radiance or temperature) in computational predictions of optical signature simulations. The objective of the second path is to study, analyze and validate the differences between simulated images (simulated visual or infrared scenes) and measured images. In particular those aspects that are important when it comes to detection, classification and identification are considered. In the present report the focus is on detection of vehicles. In this case detection means to find an object in the terrain that is some military vehicle.

The work on propagation of uncertainties in optical signature simulation programs has resulted in a number of proposed methods. The proposed methods can be divided into two classes: Error propagation methods and Monte Carlo methods. The error propagation methods are easier to implement than the Monte Carlo methods and also require (much) less computer power. The Monte Carlo methods, however, have fewer limitations than the error propagation methods and therefore they can give more reliable results. In this report we have presented a first test case using the error propagation method in validation of the thermal signature simulation software RadThermIR. The results are according to expectations but further studies, using for instance the Monte Carlo type of methods, is needed (or at least desirable) to asses the results. Such further studies will be the objective in the continued work on these methods.

In the work on analyzing differences between simulated and measured images it is proposed that several textural features are used to define the discriminative capability when it comes to detect vehicles in a terrain. These features for measured and simulated images can then be compared. So far, in a preliminary example, several feature values have been computed for a measured image and the corresponding simulated image. Further work is needed to draw any conclusions on the results.

8 REFERENCES

Ref 1 C. Nelsson, G. Forssell, P. Hermansson, A. Hågård, S. Nyberg and R. Persson, "Optiska Signaturer: Slutrapport", FOI-R—1008-SE, 2003.

Ref 2 P. Hermansson, A. Hjelm, R. Lindell, C. Nelsson, A. Persson, S. Sjökvist and T. Winzell, "Benchmarking and validation of IR signature programs: SensorVision, CameoSim and RadThermIR", FOI-R--0952--SE, 2003.

Ref 3 https://www.dmso.mil/public/library/projects/vva/found_02/sess_papers/a2_combo.pdf

Ref 4 BIPM, IEC, IFCC, ISO, IUPAC, IUPAP, and OIML. "Guide to the Expression of Uncertainty in Measurement", 1995. ISBN 92-67-10188-9, Second Edition

Ref 5 M. G. Cox, M. P. Dainton, M. and P. M. Harris, "Software Specifications for Uncertainty Calculation and Associated Statistical Analaysis", Report to the National Measurement System Policy Unit, Department of Trade and Industry, NPL Report CMSC 10/01, 2001.

Ref 6 G. Blom, "Sannolikhetsteori och Statistikteori med Tillämpningar", Studentlitteratur, Lund, 1980.

Ref 7 T. Buranthiti, J. Cao and W. Chen, "A Weighted Three-Point-Based Strategy for Variance Estimation", Proceedings of DETC'2004, Salt Lake City, DETC2004-57363, 2004.

Ref 8 W. Cheney and D. Kincaid, "Numerical Mathematics and Computing", Brooks/Cole Publishing Company, Monteray, California, ISBN 0-8185-0357-2, 1980.

Ref 9 M. D. McKay, "Latin Hypercube Sampling as a Tool in Uncertainty Analysis of Computer Models", Los Alamos National Laboratory report, LAUR 92-2338, 1992.

Ref 10 S. Weinzierl, "Introduction to Monte Carlo Methods", NIKHEF preprint, NIHHEF-00-012, 2000**.**

Ref 11 W. Chen, L. Baghdasaryan, T. Buranathiti and J. Cao, Model Validation via Uncertainty Propagation and Data Transformations, Revision 2", Submitted to the AIAA Journal, 2003. (http://ideal.mech.northwestern.edu/pdf/AIAAvalidation03.pdf)

Ref 12 M. McCoy, "A review on random number generation and methods for generating normal distributions", Technical Report 113/2000, NationalEngineering Laboratory, UK, 2000.

Ref 13 I. D. Hill and B. A. Wichmann, "Algorithm AS183. An efficient and portable pseudo-random number generator", Appl. Statist., 31:188–190, 1982.

Ref 14 W. H. McAdams, "Heat Transmission", New York: McGraw-Hill, 1954.

Ref 15 S. Nyberg, and L. Bohman, " Assessing Camouflage Methods Using Textural Features", Opt. Eng. 40(9), 1869-1876 (2001).

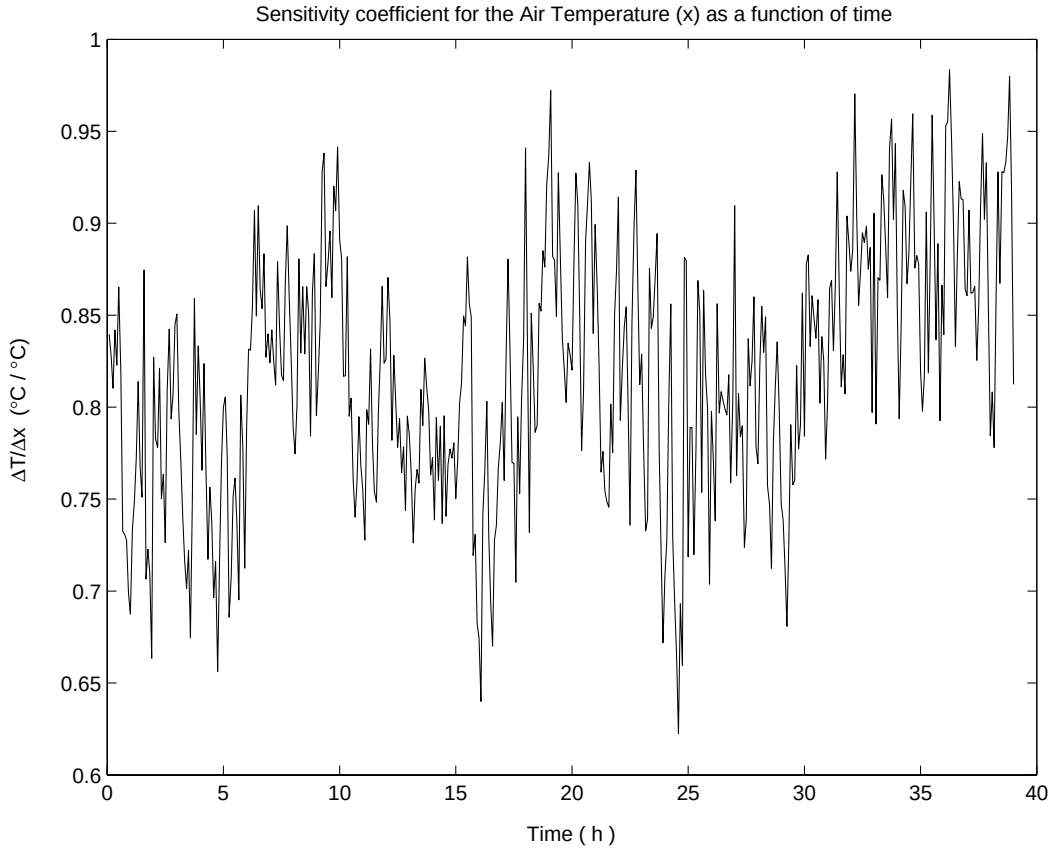
Ref 16 H. Tamura, S. Mori and T. Yamawaki, ”Textural features corresponding to visual perception”, IEEE Transaction on systems, man, and cybernetics, VOL. SMC-8, No. 6, 460-473 (1978).

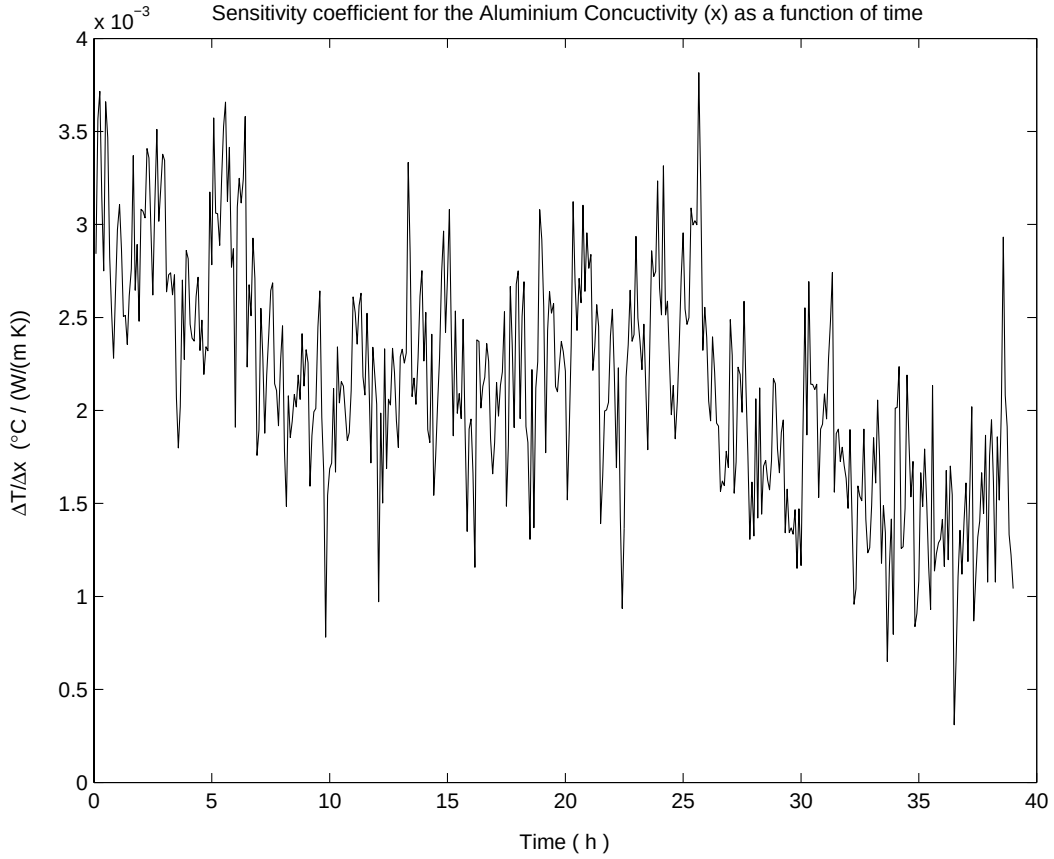
Ref 17 S. Nyberg, and L. Bohman, "Characterizing low-signature targets in background using spatial and spectral features" in Remote sensing sounders and imagers: instruments, subsystems, and algorithms, Proc. SPIE 5152, 139-149 (2003).

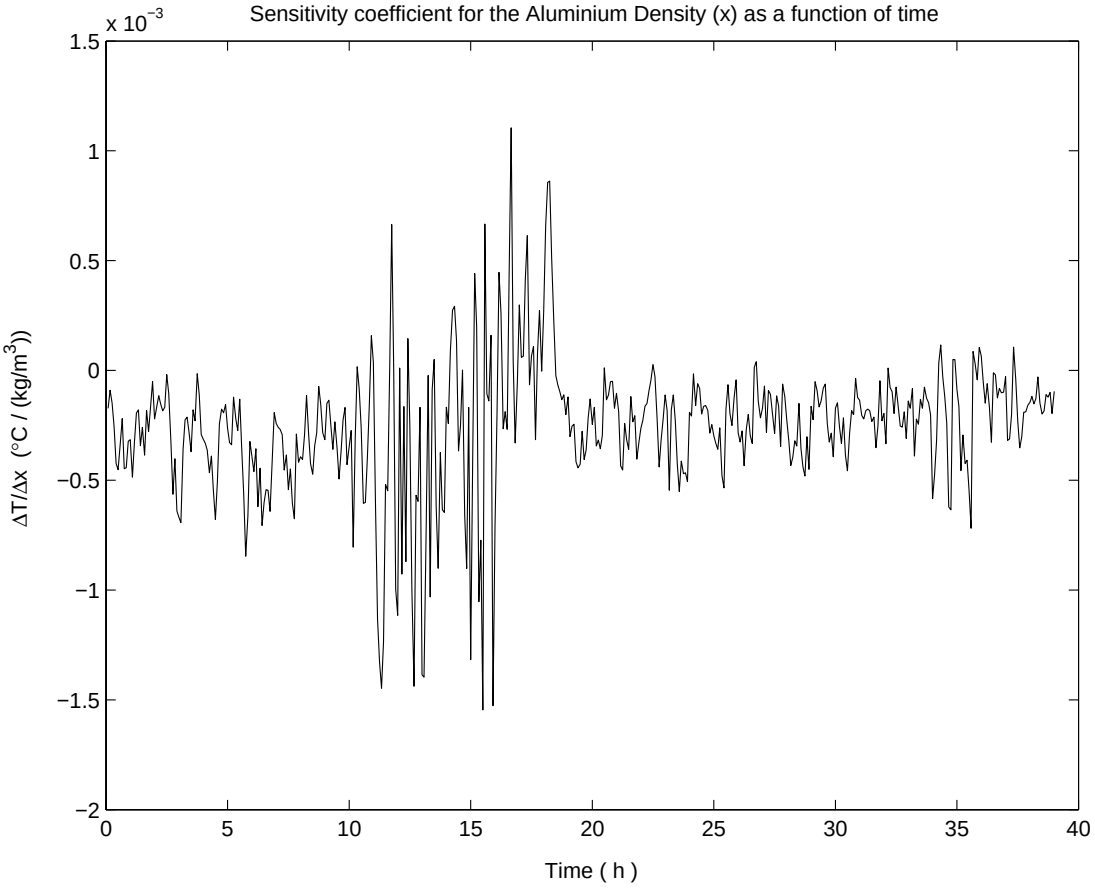
Ref 18 K. Fukunaga, “Introduction to statistical pattern recognition (2^{nd} edition)”, Academic Press, New York, 1990.

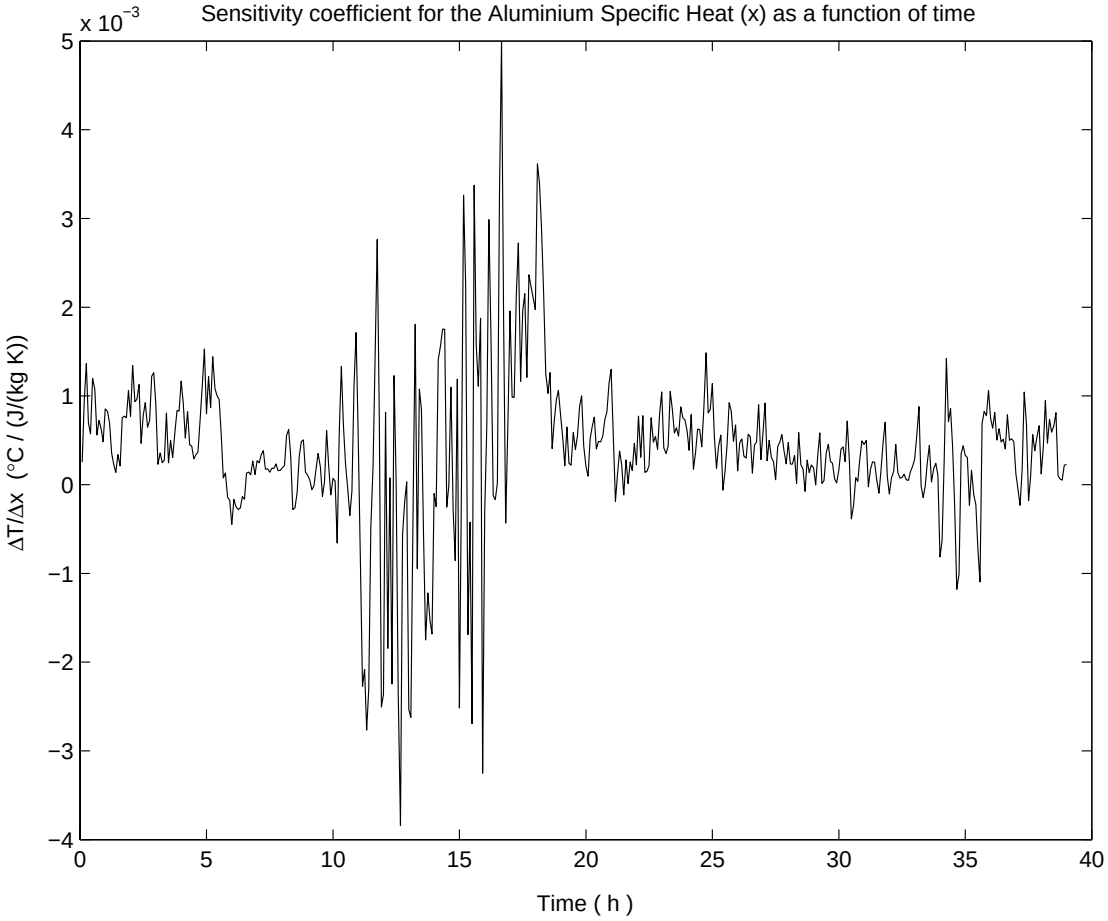
Ref 19 C. M. Birkemark, "CAMEVA, a methodology for estimation of target detectability", Opt. Eng. 40(9), 1835-1843 (2001).

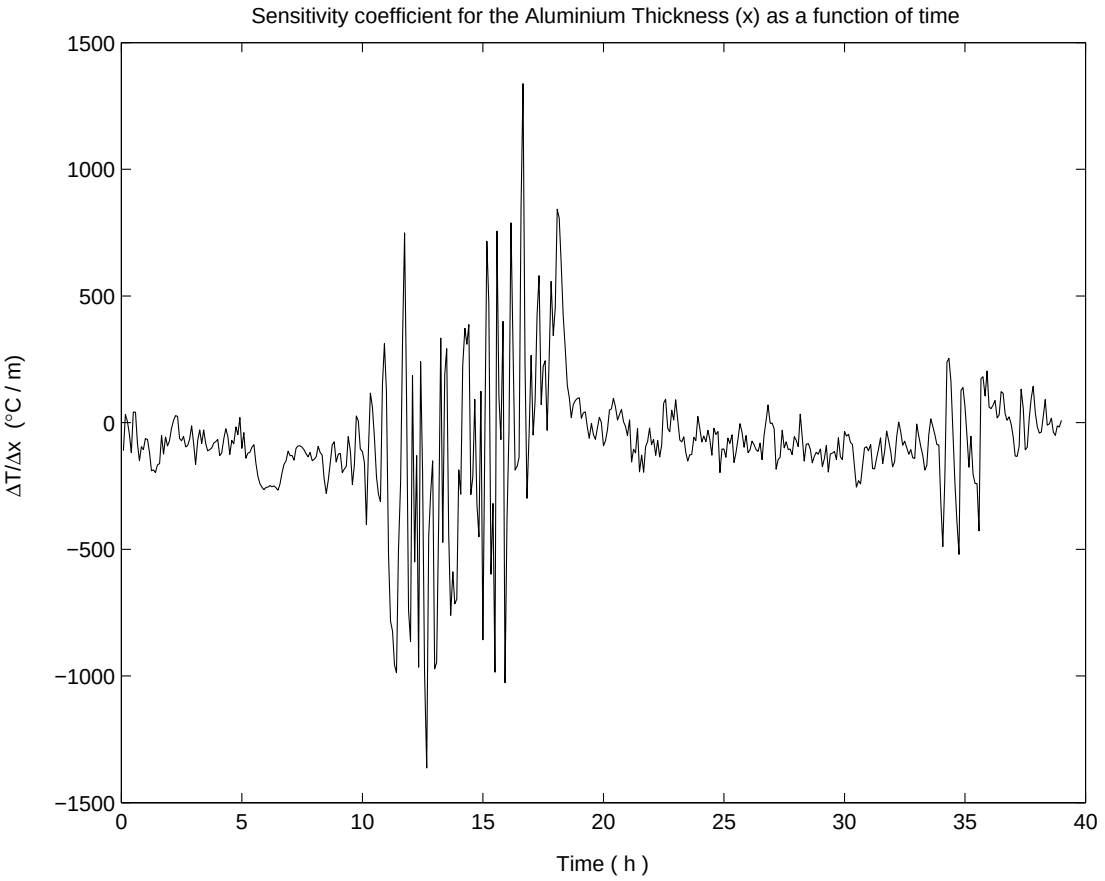
Appendix A Sensitivity coefficients for temperature

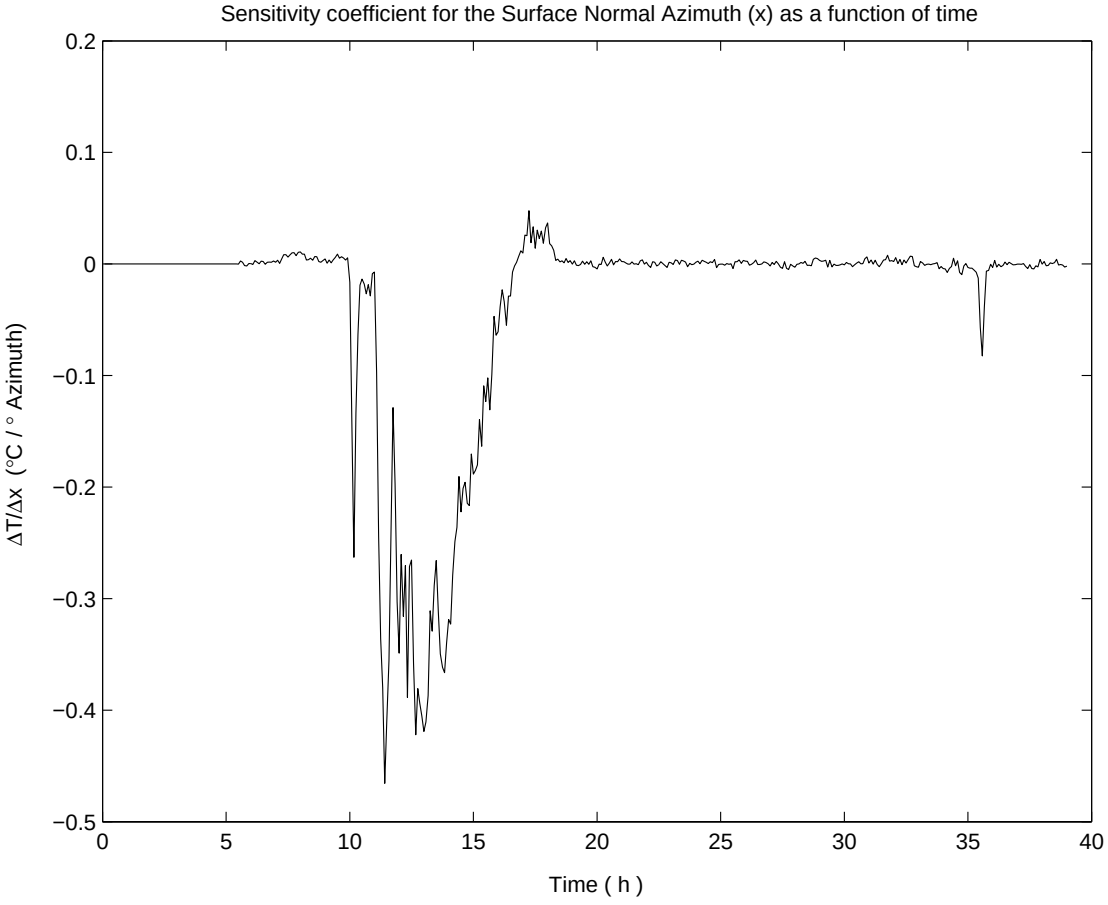


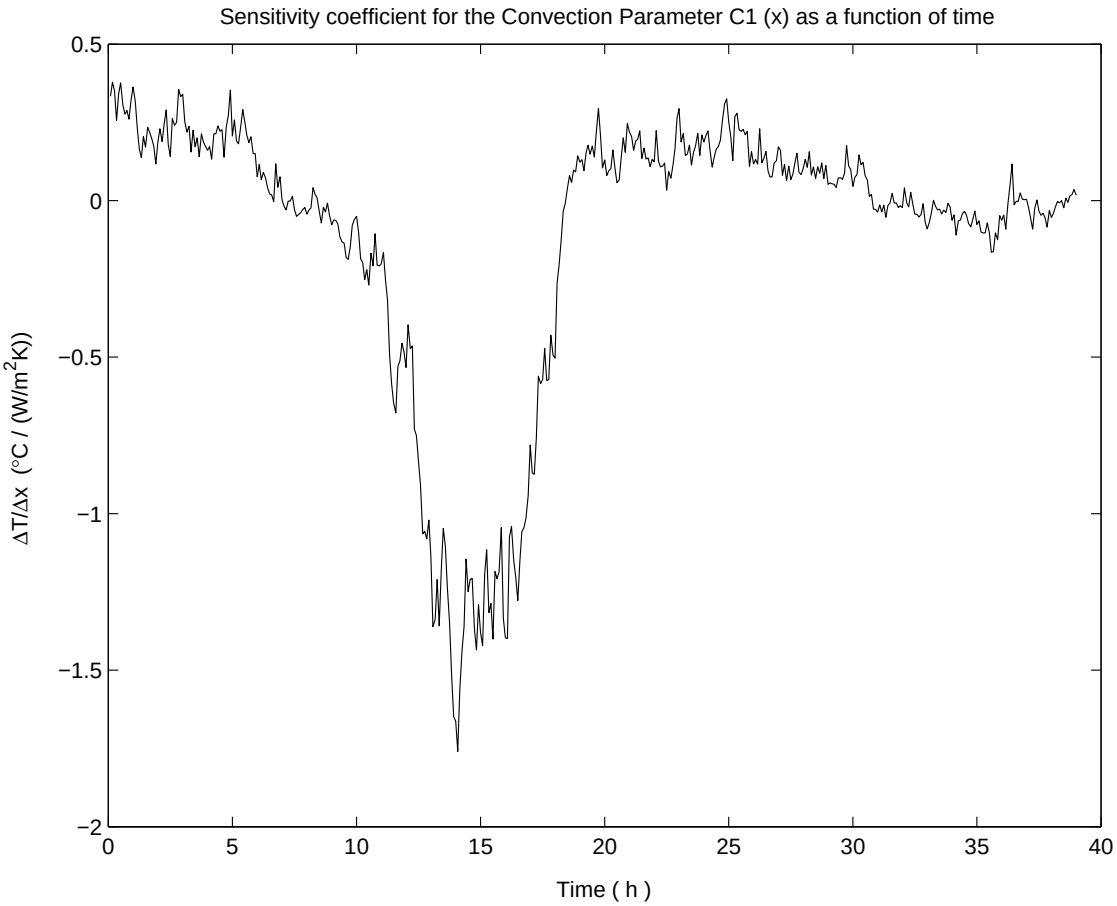


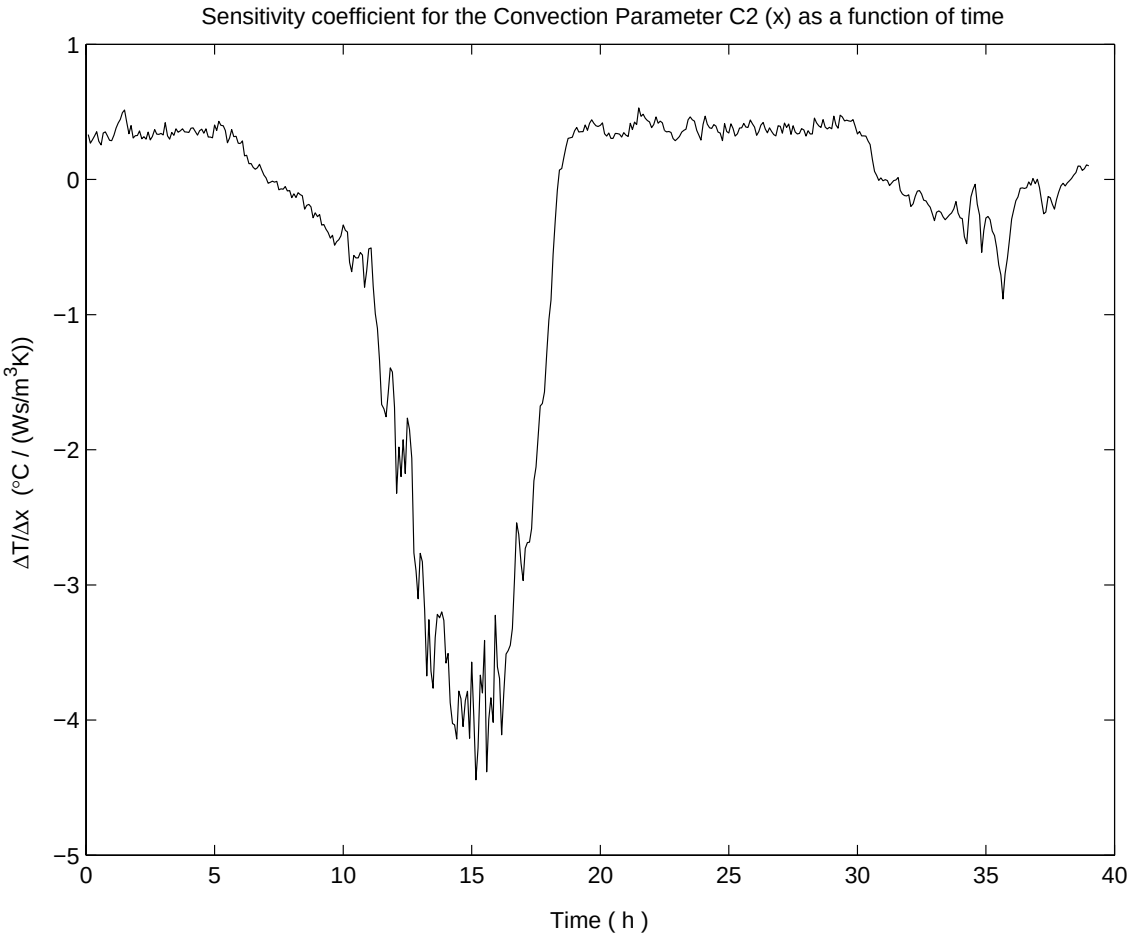

Sensitivity coefficient for the Aluminium Density (x) as a function of time
x 10^−3
ΔT/Δx (°C / (kg/m^3))
Time (h)

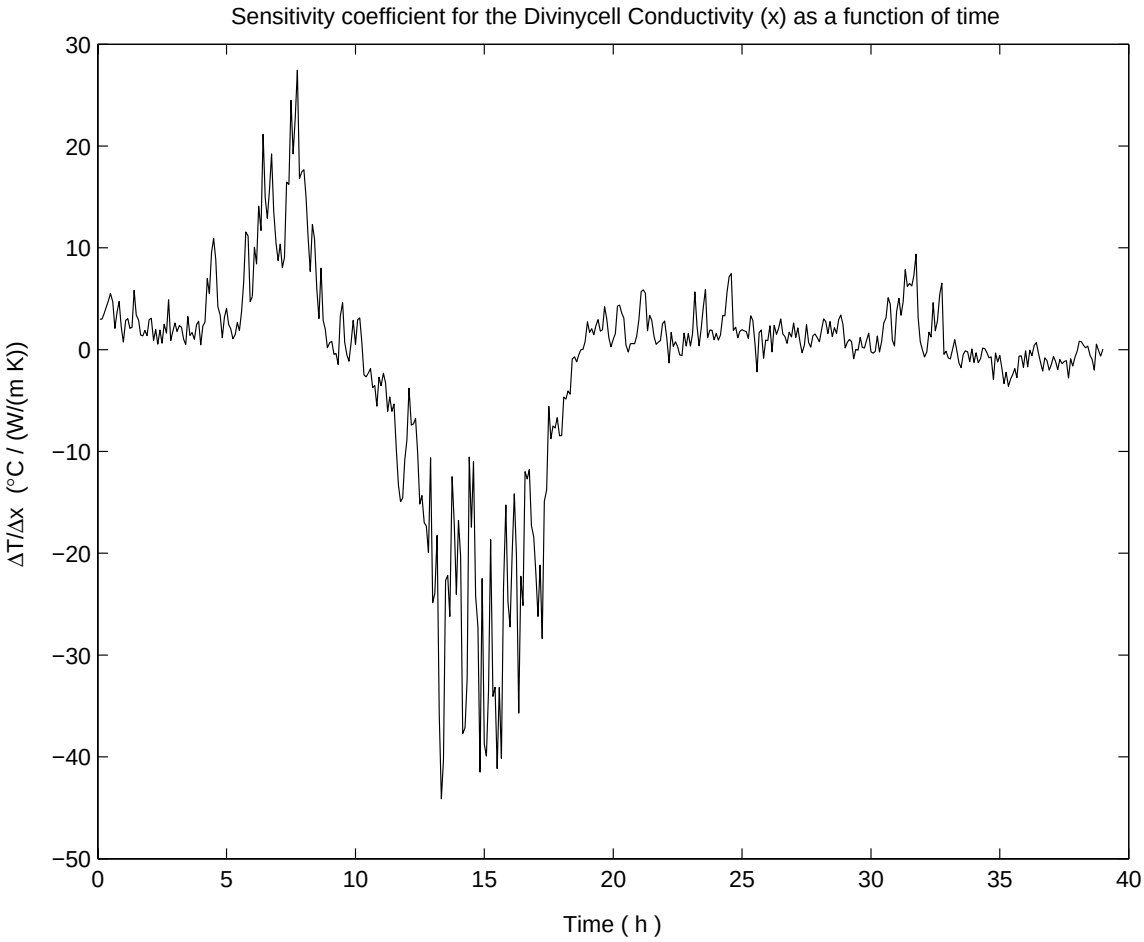

Sensitivity coefficient for the Aluminium Specific Heat (x) as a function of time
x 10^−3
ΔT/Δx (°C / (J/(kg K))
Time (h)

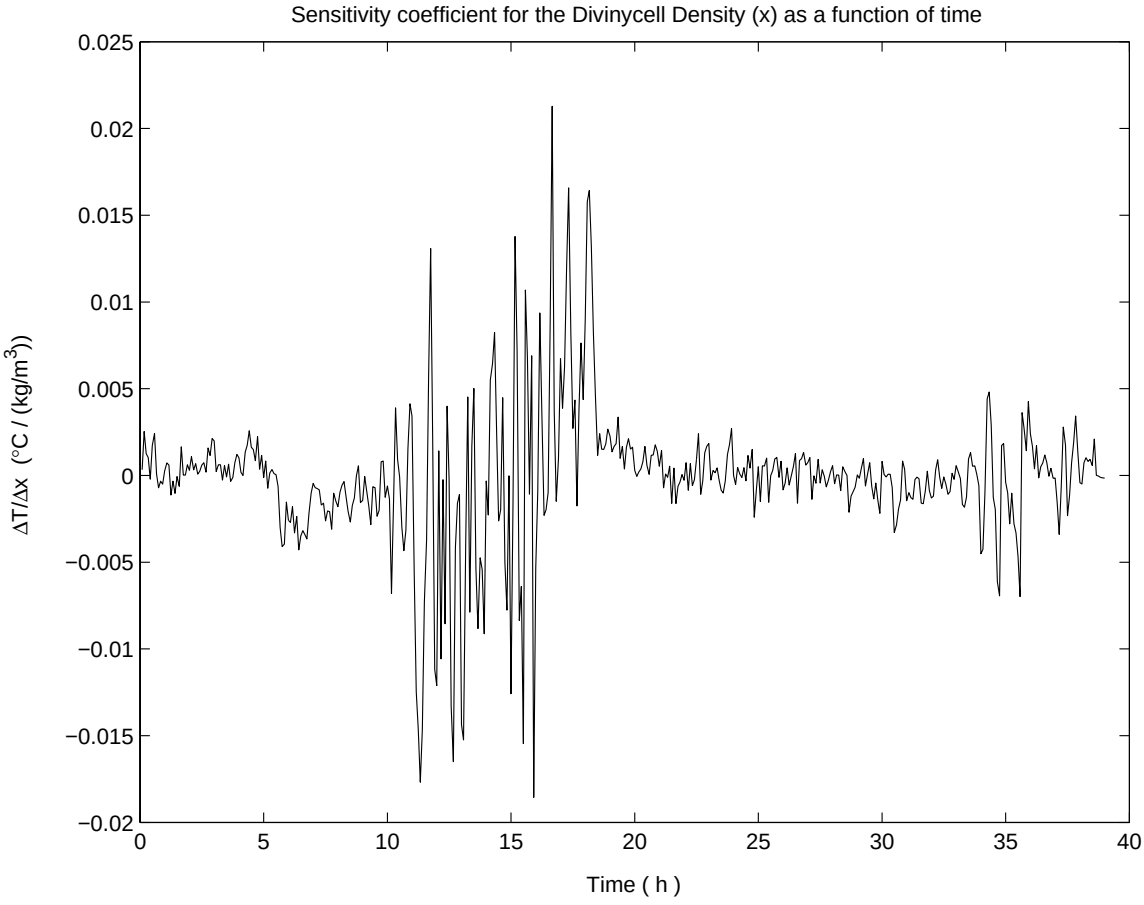

Sensitivity coefficient for the Aluminium Thickness (x) as a function of time
ΔT/Δx (°C / m)
Time (h)
1500
1000
500
0
−500
−1000
−1500
0
5
10
15
20
25
30
35
40

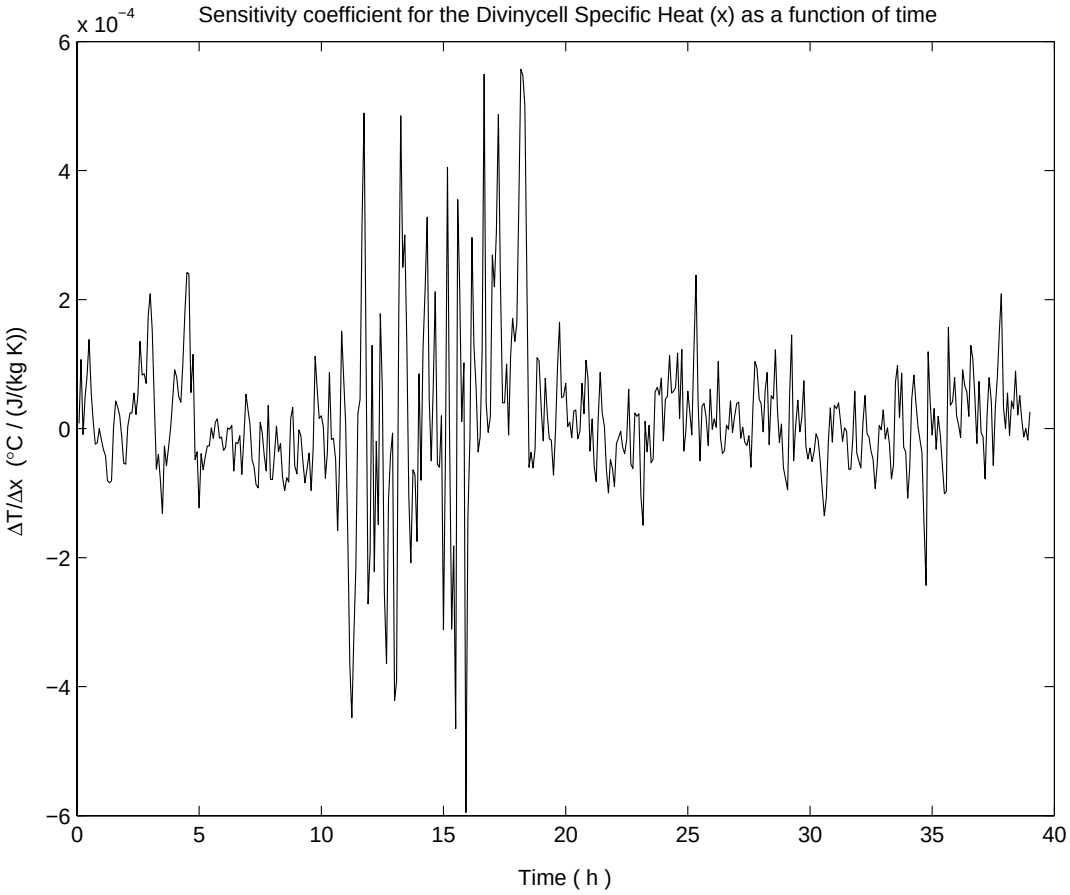

Sensitivity coefficient for the Surface Normal Azimuth (x) as a function of time
ΔT/Δx (°C / ° Azimuth)
Time (h)
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
0
5
10
15
20
25
30
35
40

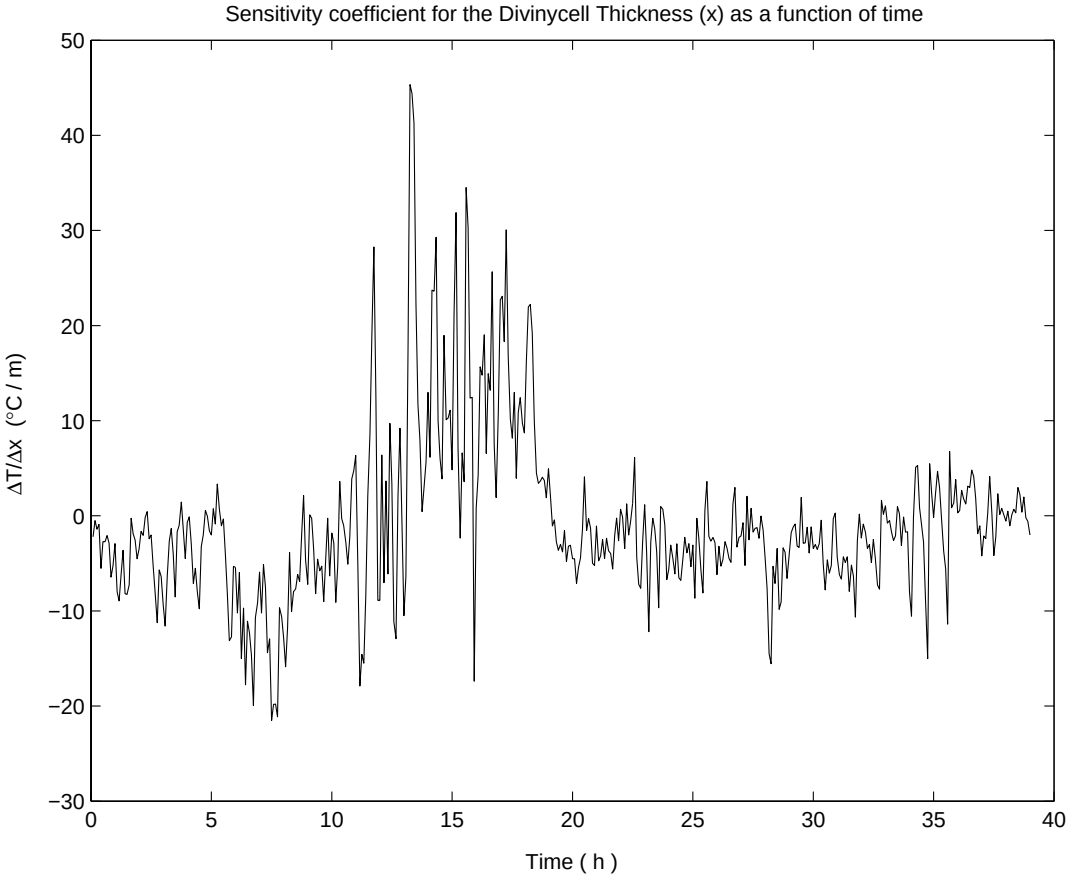

Sensitivity coefficient for the Convection Parameter C1 (x) as a function of time
ΔT/Δx (°C / (W/m²K))
Time (h)
0.5
0
−0.5
−1
−1.5
−2
0
5
10
15
20
25
30
35
40

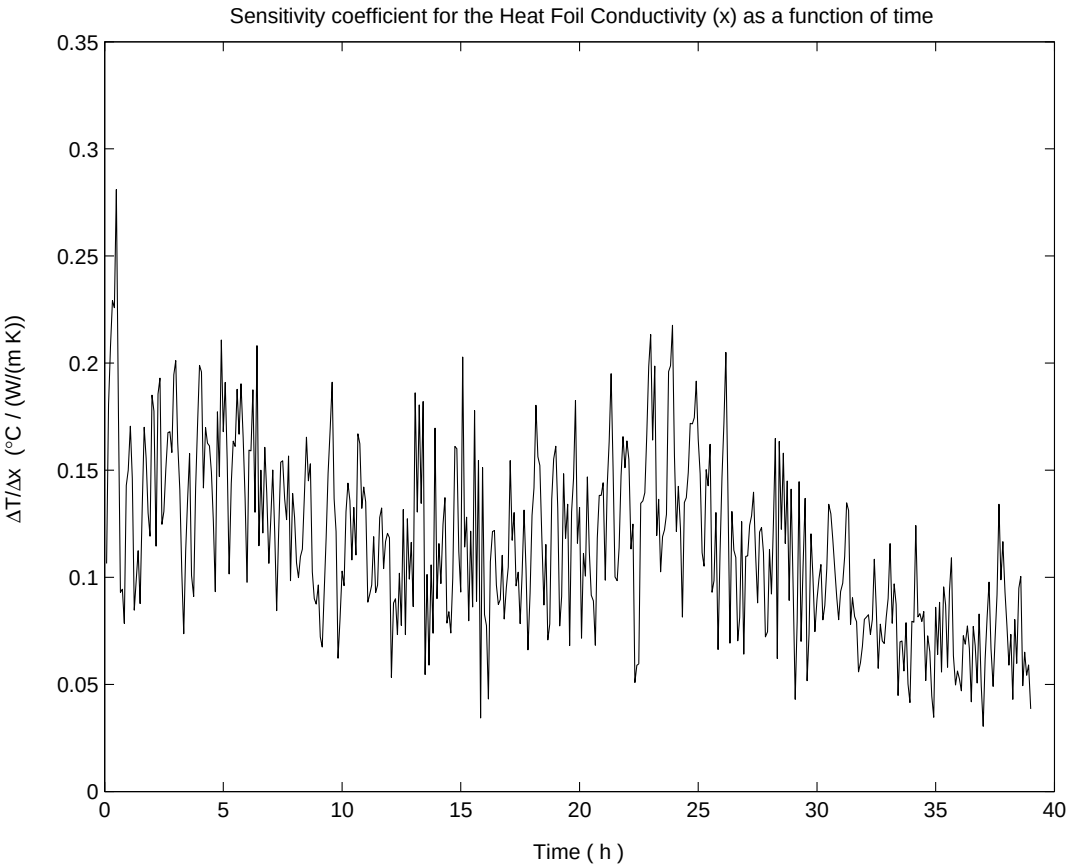

Sensitivity coefficient for the Convection Parameter C2 (x) as a function of time
ΔT/Δx (°C / (Ws/m³K))
Time (h)
1
0
−1
−2
−3
−4
−5
0
5
10
15
20
25
30
35
40

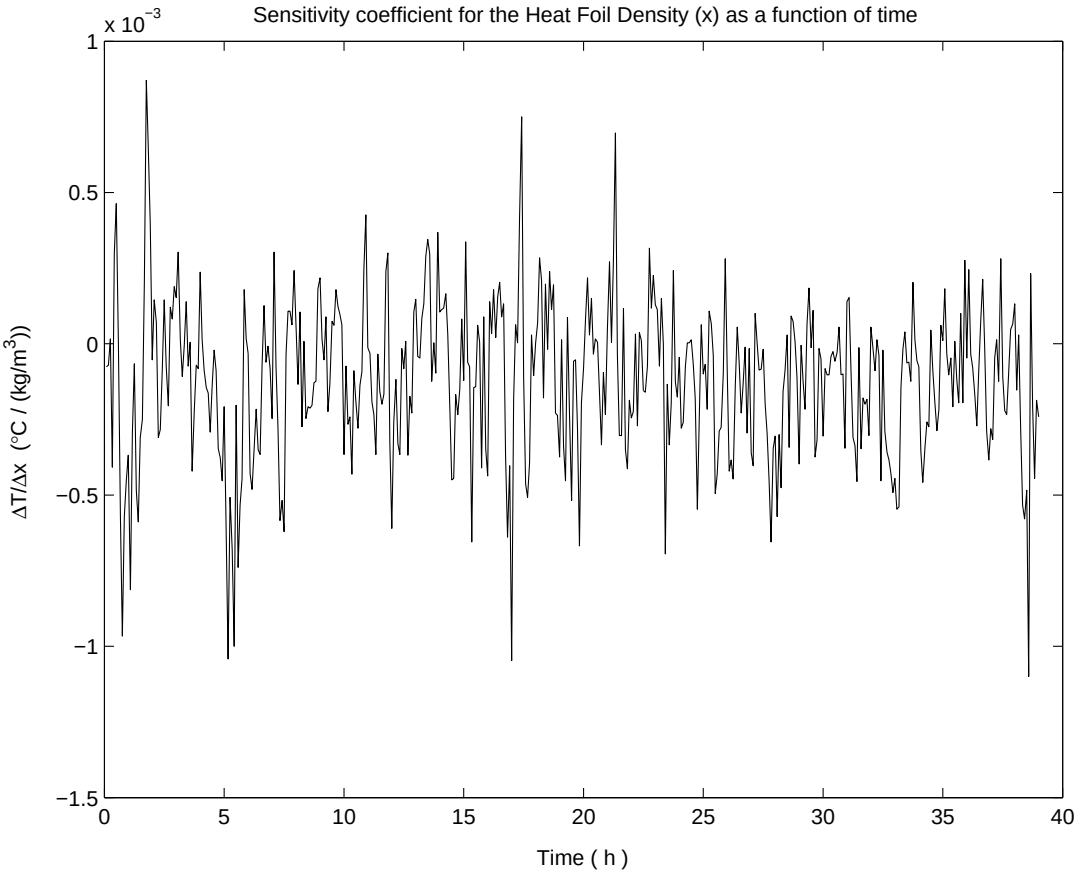

Sensitivity coefficient for the Divinycell Conductivity (x) as a function of time
ΔT/Δx (°C / (W/(m K))
Time (h)
30
20
10
0
−10
−20
−30
−40
−50
0
5
10
15
20
25
30
35
40

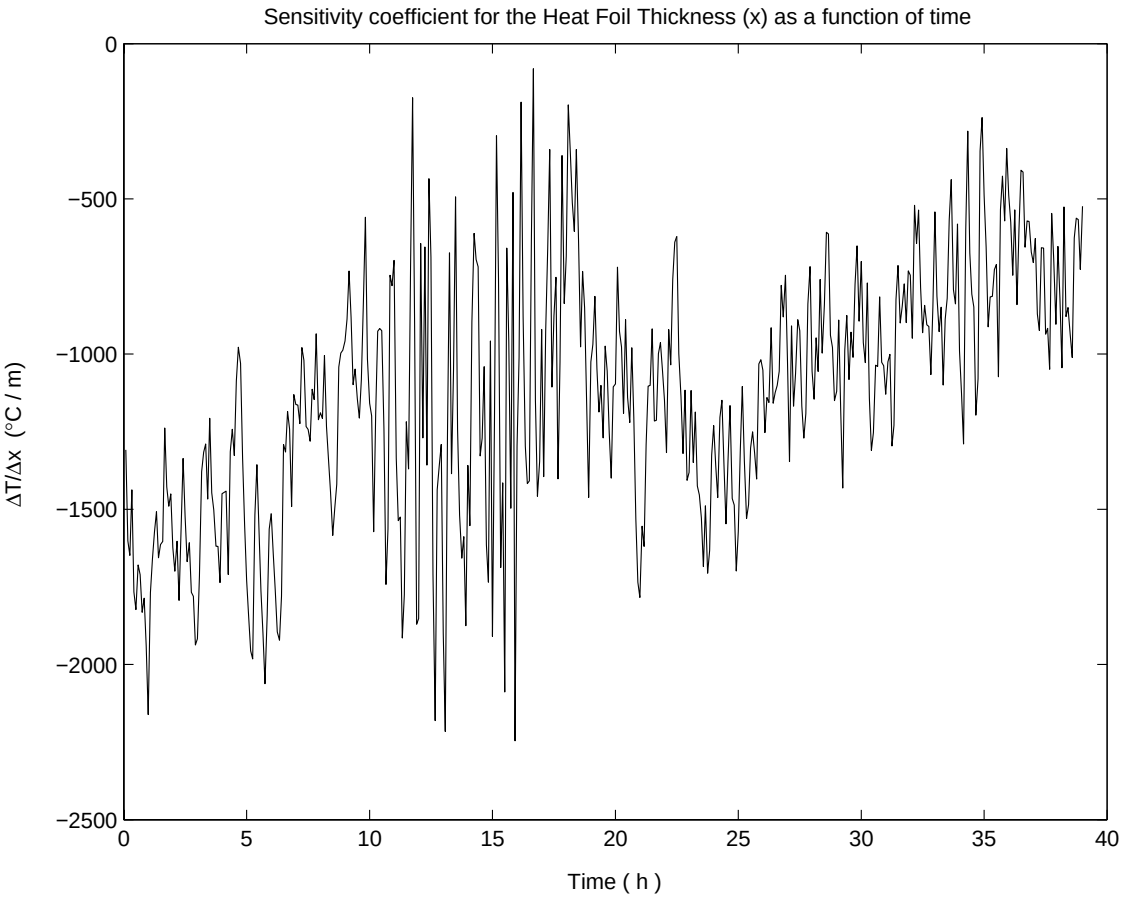

Sensitivity coefficient for the Divinycell Density (x) as a function of time
ΔT/Δx (°C / (kg/m3))
Time (h)
0.025
0.02
0.015
0.01
0.005
0
−0.005
−0.01
−0.015
−0.02
0
5
10
15
20
25
30
35
40

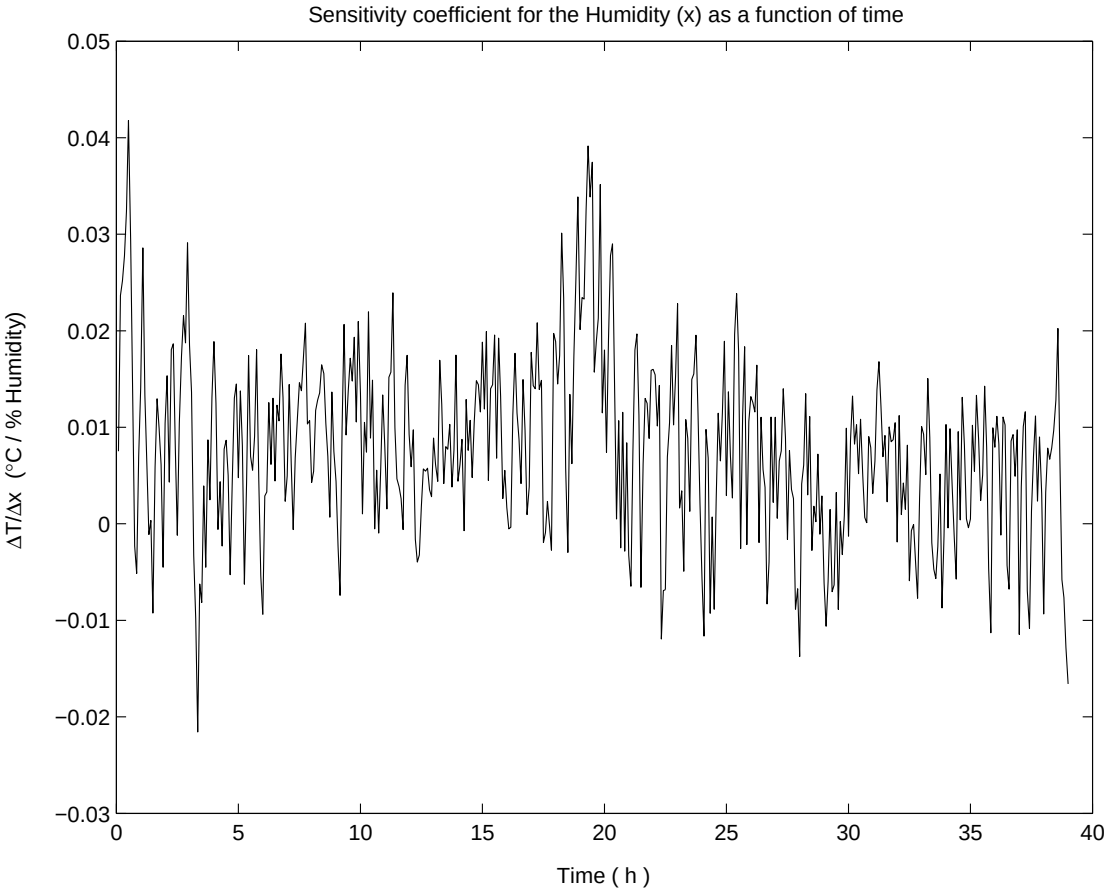

Sensitivity coefficient for the Divinycell Specific Heat (x) as a function of time
x 10^−4
ΔT/Δx (°C / (J/(kg K))
Time (h)

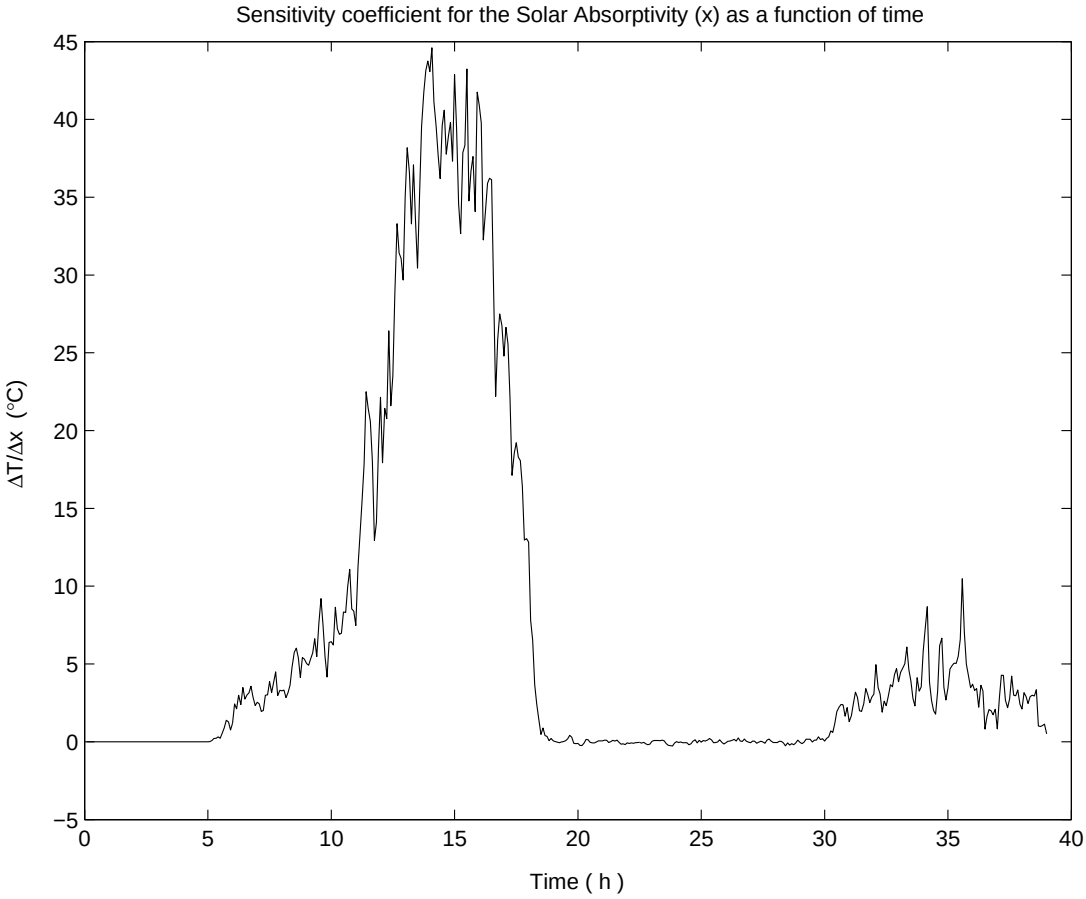

Sensitivity coefficient for the Divinycell Thickness (x) as a function of time
ΔT/Δx (°C / m)
Time (h)

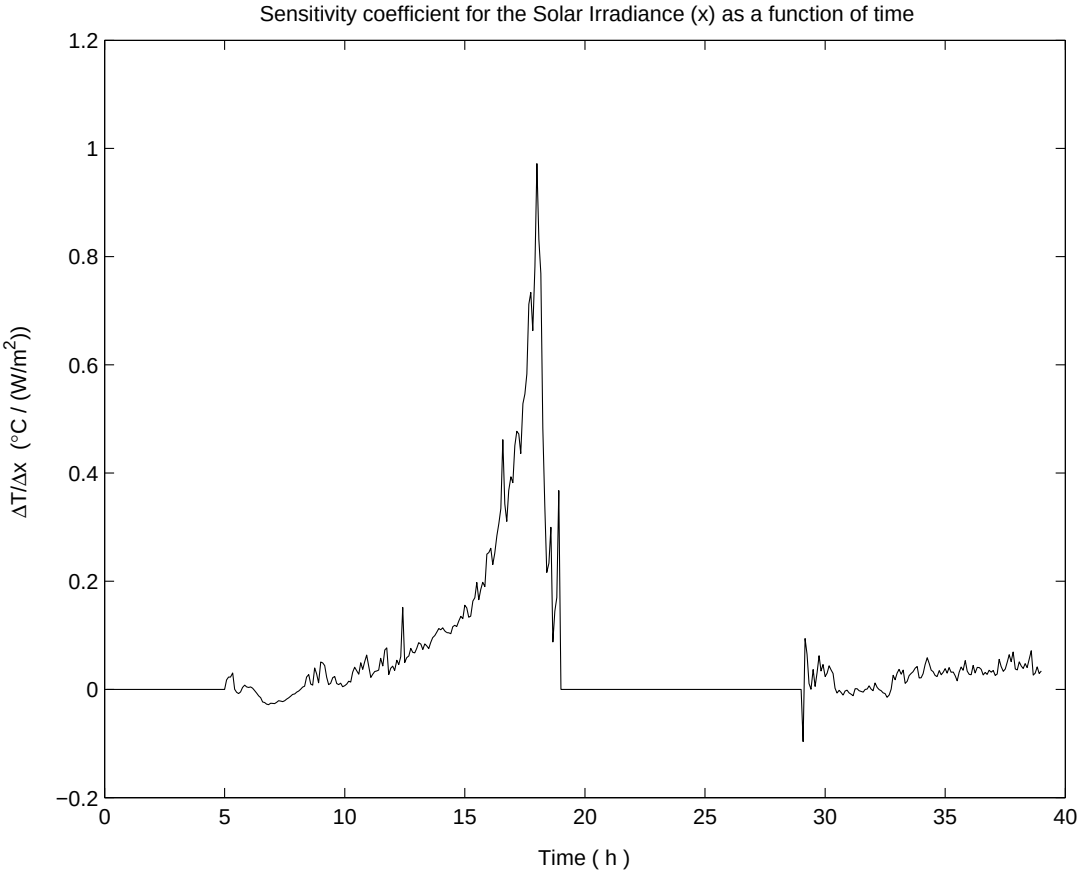

Sensitivity coefficient for the Heat Foil Conductivity (x) as a function of time
ΔT/Δx (°C / (W/(m K))
Time (h)
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0
5
10
15
20
25
30
35
40

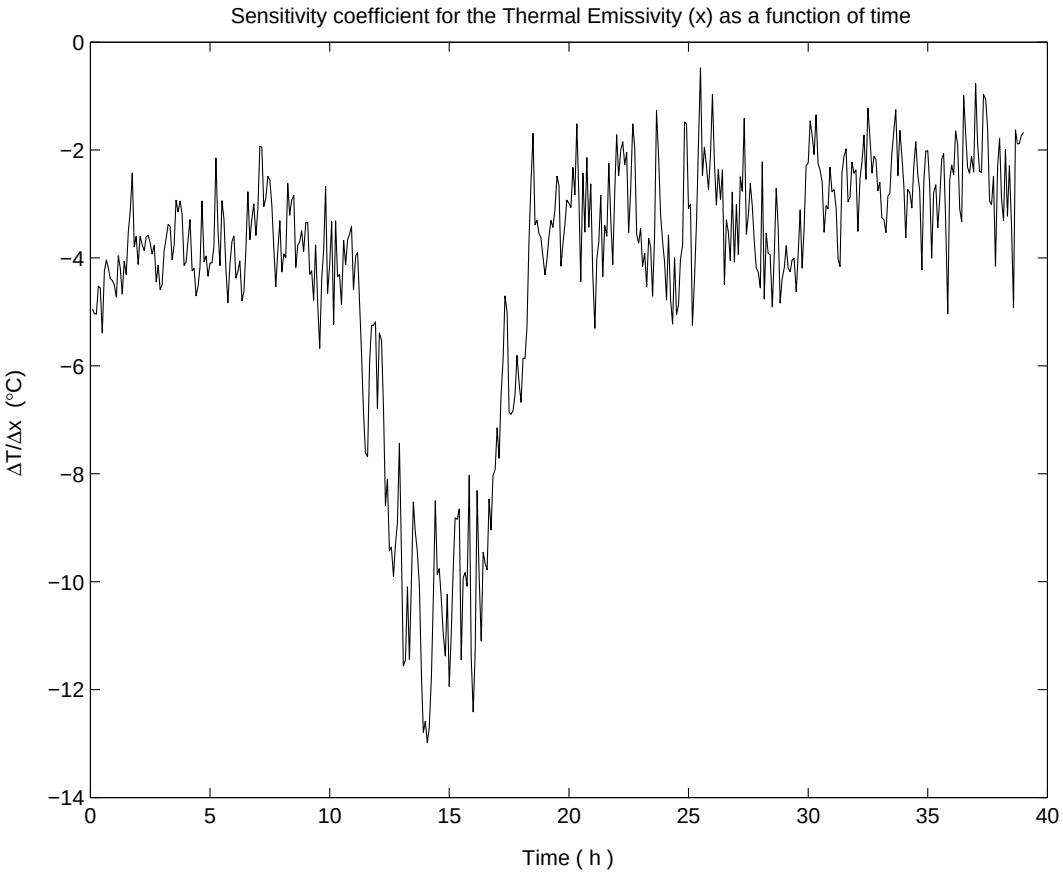

Sensitivity coefficient for the Heat Foil Density (x) as a function of time
x 10^-3
ΔT/Δx (°C / (kg/m3))
Time (h)
−1.5
−1
−0.5
0
0.5
1
0
5
10
15
20
25
30
35
40

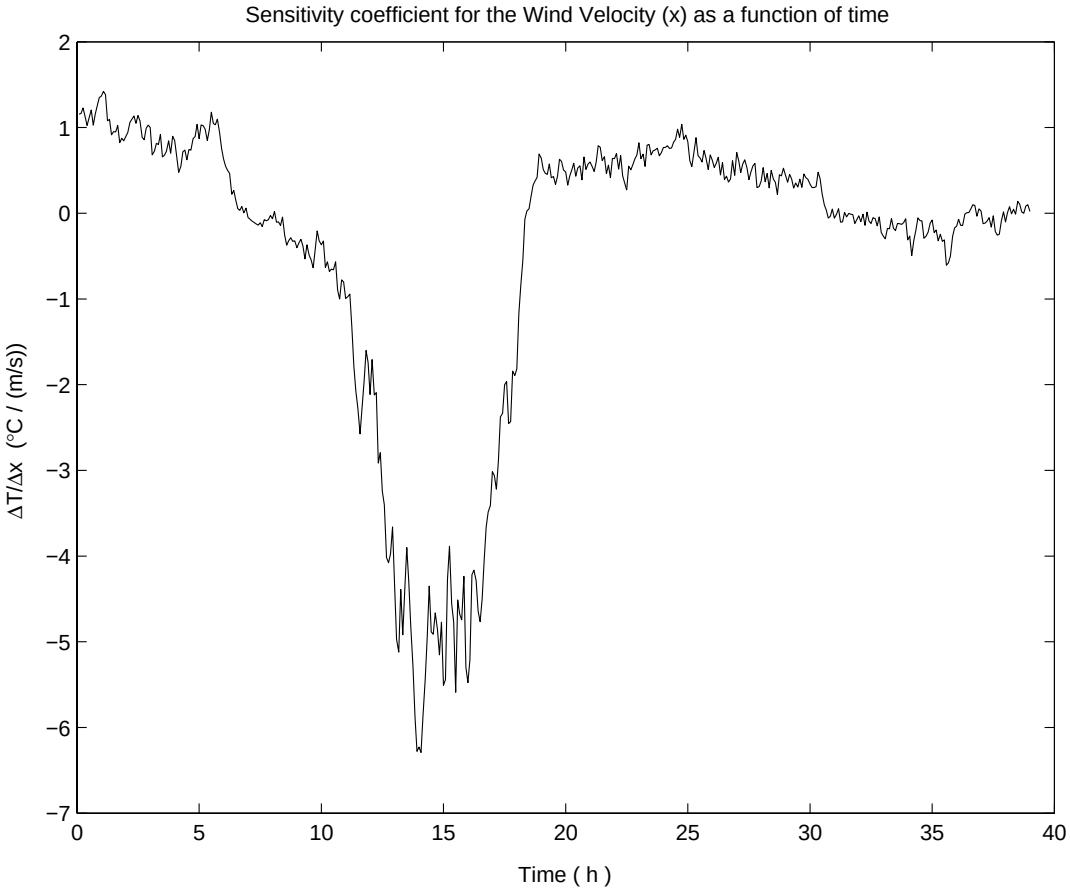

Sensitivity coefficient for the Heat Foil Thickness (x) as a function of time
ΔT/Δx (°C / m)
Time (h)
0
−500
−1000
−1500
−2000
−2500
0
5
10
15
20
25
30
35
40


Sensitivity coefficient for the Humidity (x) as a function of time
ΔT/Δx (°C / % Humidity)
Time (h)
0.05
0.04
0.03
0.02
0.01
0
−0.01
−0.02
−0.03
0
5
10
15
20
25
30
35
40

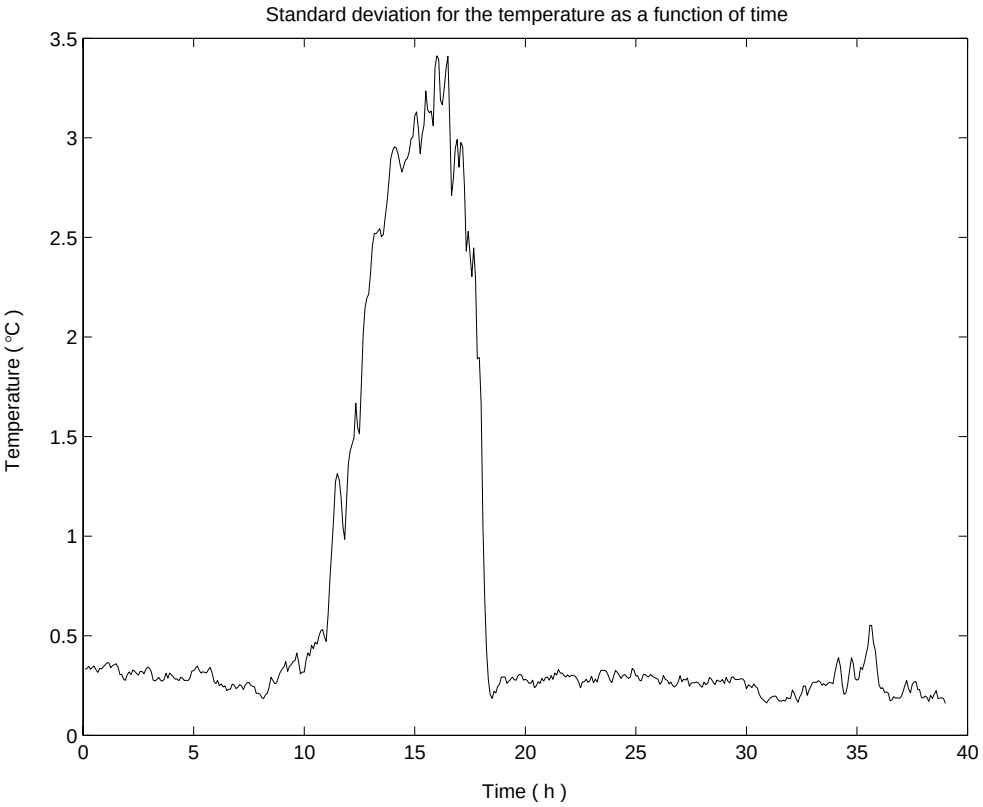

Sensitivity coefficient for the Solar Absorptivity (x) as a function of time
ΔT/Δx (°C)
Time (h)

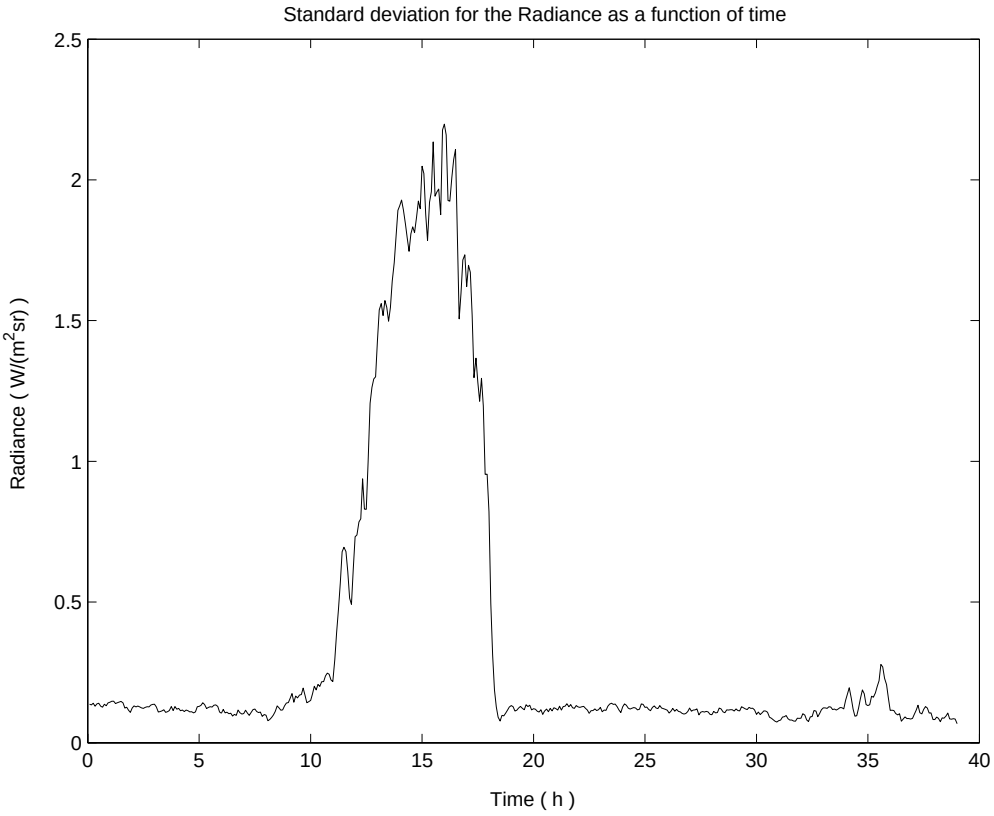

Sensitivity coefficient for the Solar Irradiance (x) as a function of time
ΔT/Δx (°C / (W/m2))
Time (h)


Sensitivity coefficient for the Thermal Emissivity (x) as a function of time
ΔT/Δx (°C)
Time (h)
0
−2
−4
−6
−8
−10
−12
−14
0
5
10
15
20
25
30
35
40


Sensitivity coefficient for the Wind Velocity (x) as a function of time
ΔT/Δx (°C / (m/s))
Time (h)
2
1
0
−1
−2
−3
−4
−5
−6
−7
0
5
10
15
20
25
30
35
40

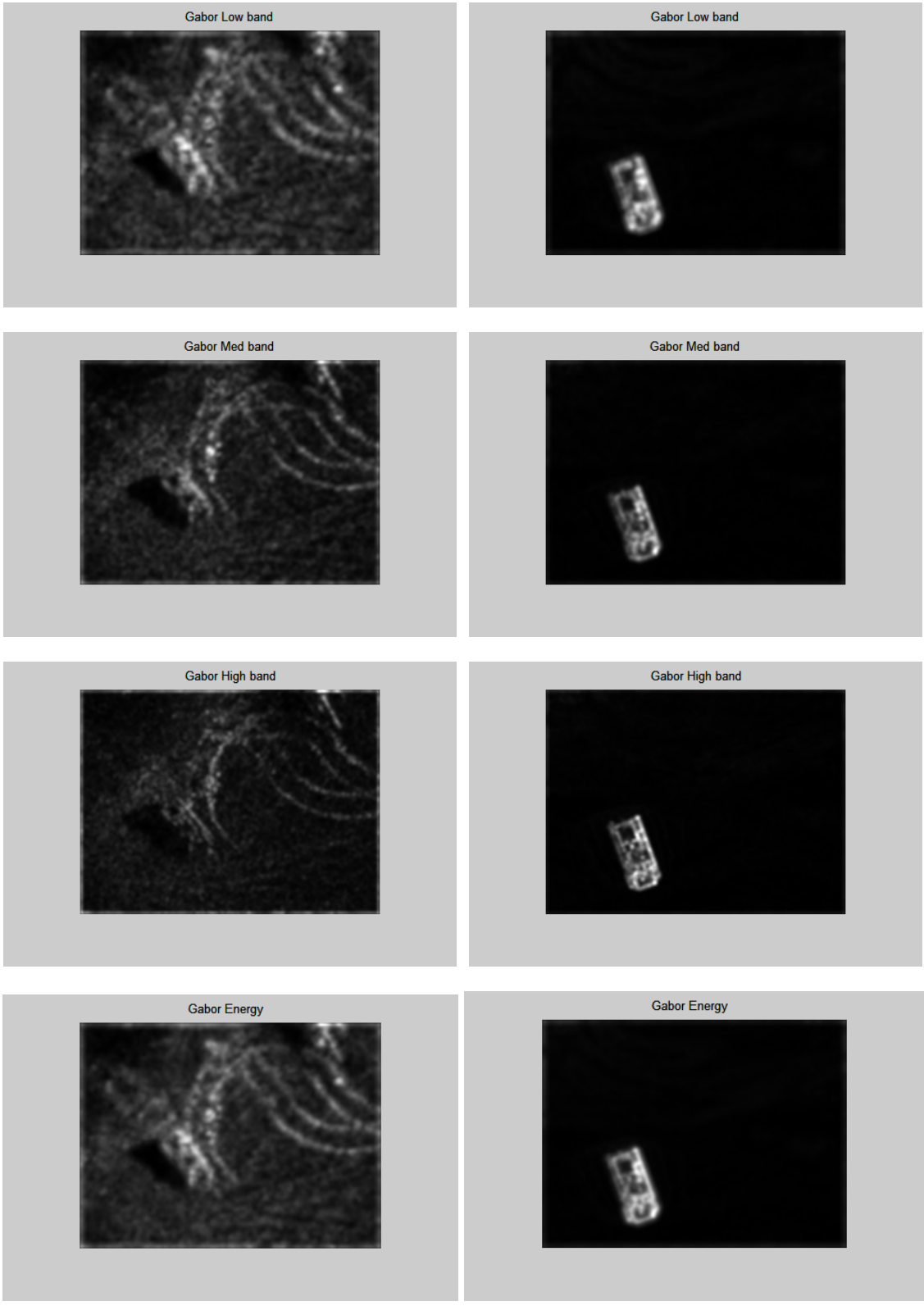
Appendix B Calculated standard deviations in temperature and radiance versus time

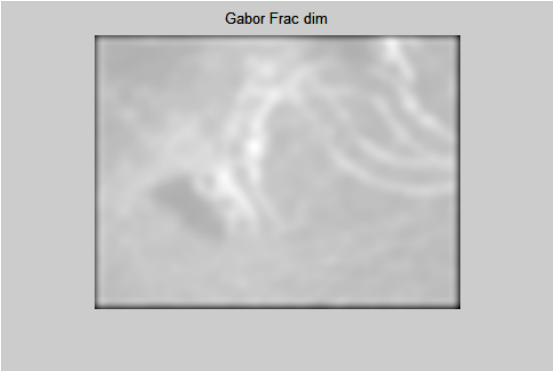




Appendix C Images used for feature generation

Ten pairs of Gabor feature images. Figures in the left column are feature images corresponding to the registered image and figures in the right column correspond to the modeled image.

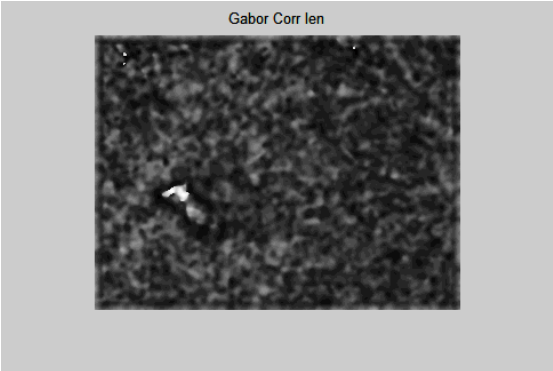


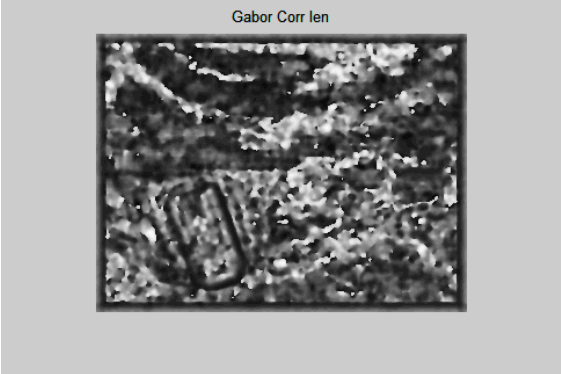

Gabor Frac dim

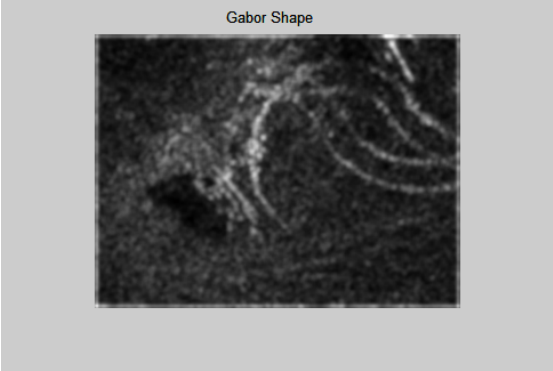

Gabor Frac dim


Gabor Frac err

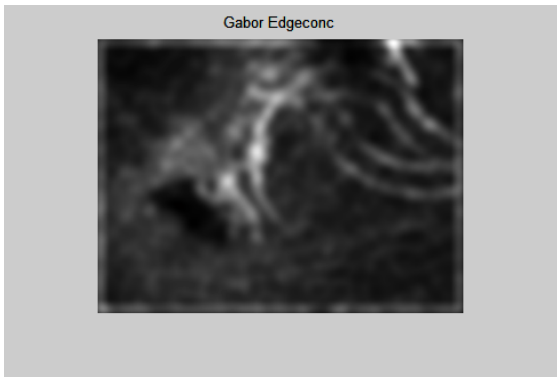

Gabor Frac err


Gabor Corr len

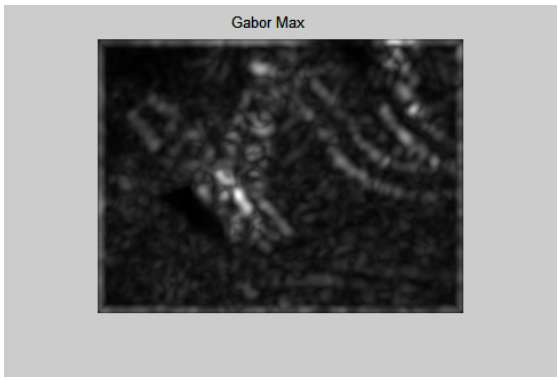

Gabor Corr len


Gabor Shape


Gabor Shape


Gabor Edgeconc


Gabor Edgeconc


Gabor Max


Gabor Max