



324, Ave St-Augustin, Breakeyville (Québec), Canada, G0S 1E1
telephone: (418) 832-1040, fax: (418) 832-9831, e-mail: info@aerex.ca

AEREX Report No: 2013-125216-004

***Simulation of subpixel atmospherically degraded target
detectability in cluttered scenes***

Submitted to

**CSA : Pierre Lahaie
DRDC, Valcartier Research Center
2459 de la Bravoure Road
Québec (Québec)
Canada, G3J 1X5
418-844-4000 x 4815**

in the framework of contract number

**PWGSC Contract Number: W7701-125216/001/QCL
"Atmospheric Optics"**

Authored by:

**Vincent Ross, M.Sc.
Guillaume Gagné, M.Sc.**

6 September 2013

UNCLASSIFIED

**© Sa majesté la reine du Canada, représentée par le ministre de la Défense nationale, 2013
© Her Majesty the Queen, as represented by the Minister of National Defence, 2013**

The scientific or technical validity of this Contract Report is entirely the responsibility of the Contractor and the contents do not necessarily have the approval or endorsement of the Department of National Defence of Canada.

Simulation of subpixel atmospherically degraded target detectability in cluttered scenes

AEREX Report Number: 2013-125216-004

Version: 1.0

Author:

(Original signed by)
Vincent Ross
Physicist

6 September 2013
(date)

Publication reviewed by:

(Original signed by)
Paul Lacasse, Eng., MBA
Senior Project Manager

6 September 2013
(date)

Publication approved by:

(Original signed by)
Daniel Pomerleau, Eng.
President, AEREX Inc.

6 September 2013
(date)

Abstract

This document describes a tool designed to assess the performance of detection algorithms when different atmospheric correction algorithms are employed to compensate for different, possibly detrimental, atmospheric and sensor hyperspectral image degradations. This is done by simulating atmospheric and sensor effects on background images, applying different correction algorithms to the generated images and running detection algorithms on the resulting images. Images with and without the presence of subpixel solid or gas targets are used to compute probability distributions of both detection and false alarms. These are then used to compute ROC curves that are useful for assessing each detection algorithm under the prevailing conditions.

This tool was developed under contract W7701-125216/001/QCL on behalf of DRDC, Valcartier Research Center between September 6th 2012 and September 6th 2013.

Résumé

Ce document décrit un outil conçu pour évaluer la performance d'algorithmes de détection lorsque différents algorithmes de correction atmosphériques sont utilisés pour contrer des conditions atmosphériques possiblement désavantageuses et des dégradations dues au senseur sur des images hyperspectrales. Cela s'accomplit par la simulation des effets de l'atmosphère et du senseur sur l'image de fond, par l'application de différents algorithmes de correction à l'image générée, et finalement par l'application de différents algorithmes de détection sur l'image résultante. Des images avec et sans la présence de la signature d'une cible sous-pixel ou d'un gaz sont utilisées pour calculer les distributions de probabilité de détection et de fausses alarmes. Celles-ci sont ensuite converties en courbes ROC qui sont utiles pour évaluer la performance de chaque algorithme de détection dans les conditions en place.

Cet outil a été développé dans le cadre du contrat W7701-125216/001/QCL pour le compte de RDDC, Centre de recherches de Valcartier entre le 6 septembre 2012 et le 6 septembre 2013.

This page intentionally left blank.

Executive summary

Simulation of sub pixel atmospherically degraded target detectability in cluttered scenes:

**Vincent Ross; Guillaume Gagné; AEREX Report Number: 2013-125216-004,
AEREX Avionics Inc.; September 2013**

Introduction or background: The atmosphere plays an important and often detrimental role in hyperspectral target detection. Downwelling irradiance and target emission alters the intrinsic emissivity spectrum of the target, while the atmosphere between the target and the sensor both absorbs and scatters radiation in and out of the beam further modifying the signature. Correction algorithms have been designed to attempt to remove atmospheric effects, and separate out the target's emissivity from its thermal emission, but none are perfect and can leave spectral artifacts. Sensor effects also degrade the hyperspectral images by adding noise and spectral distortions. The effect of these degradations on different target detection algorithms is not immediately obvious and requires further study.

This report describes the theoretical background and design of a tool that can be used to assess the performance of detection algorithms on atmospherically degraded and corrected LWIR hyperspectral sensor images. This is done by simulating atmospheric and sensor effects on background images, applying different correction algorithms to the generated images and running detection algorithms on the resulting images. Images with and without the presence of subpixel solid or gas targets are used to compute probability distributions of both detection and false alarms. These are then used to compute ROC curves that are useful for assessing each detection algorithm under the prevailing conditions.

This tool was developed under contract W7701-125216/001/QCL on behalf of DRDC, Valcartier Research Center between September 6th 2012 and September 6th 2013.

Results: Significant and representative cases are modeled and results are presented in an annex of the report. The effects of different atmospheric and sensor parameters, as well as correction algorithms on target detection, are shown.

Significance: The use of this tool could become an important part in the design of reconnaissance missions by adapting the algorithms to the sensor and atmosphere for optimal results. It can also become an effective tool to verify the effectiveness of newly designed algorithms and sensors.

Future plans: A number of recommendations are made to improve the tool for ease of use, speed, usability in more difficult atmospheric/geometric conditions and operational usability.

Sommaire

Simulation of sub pixel atmospherically degraded target detectability in cluttered scenes:

**Vincent Ross; Guillaume Gagné ; AEREX Report Number: 2013-125216-004,
AEREX Avionique inc.; septembre 2013.**

Introduction ou contexte : L'atmosphère joue un rôle important et souvent néfaste à la détection de cibles hyperspectrales. L'éclairement et l'émission thermique de la cible modifient le spectre d'émissivité intrinsèque de la cible, tandis que l'atmosphère entre la cible et le capteur absorbe et diffuse le rayonnement électromagnétique dans et hors du faisceau pour ainsi en modifier la signature. Des algorithmes de correction ont été conçus pour tenter d'éliminer les effets atmosphériques et de dissocier l'émissivité de la cible de son émission thermique, mais aucun algorithme n'est parfait, ceux-ci pouvant laisser des artefacts spectraux dans les signatures. Des effets capteurs peuvent également dégrader les images hyperspectrales en ajoutant du bruit et des distorsions spectrales. L'effet de ces dégradations sur différents algorithmes de détection de cibles n'est pas immédiatement évident et nécessite une étude plus approfondie.

Ce rapport décrit le contexte théorique et la conception d'un outil qui peut être utilisé pour évaluer la performance d'algorithmes de détection sur des images infrarouges de capteur hyperspectral dégradées et corrigées de certains effets de l'atmosphère. Cela s'accomplit par la simulation des effets atmosphériques et de capteur sur des images d'arrière-plan, par l'application de différents algorithmes de correction sur les images générées et par l'application d'algorithmes de détection sur les images qui en résultent. Des images avec et sans la présence de cibles sous-pixel solides ou gazeuses sont utilisées pour calculer des distributions de probabilité de détection et de fausses alarmes. Ceux-ci sont ensuite utilisés pour calculer des courbes ROC qui sont utiles pour l'évaluation de chaque algorithme de détection dans les conditions qui prévalent.

Cet outil a été développé dans le cadre du contrat W7701-125216/001/QCL pour le compte de RDDC, Centre de recherches de Valcartier du 6 septembre 2012 au 6 septembre 2013.

Résultats : Des cas significatifs et représentatifs sont modélisés et les résultats sont présentés en annexe au rapport. L'effet de différents paramètres atmosphériques et de capteur ainsi que l'effet des algorithmes de correction sur la détection de cibles sont démontrés.

Importance : L'utilisation de cet outil pourrait devenir un élément important dans la conception des missions de reconnaissance, en adaptant les algorithmes de détection à l'atmosphère et au senseur utilisé, dans le but d'obtenir des résultats optimaux. L'outil pourrait également devenir un outil efficace pour vérifier le rendement d'algorithmes et de capteurs nouvellement conçus.

Perspectives : Un certain nombre de recommandations sont formulées afin d'améliorer l'outil pour sa facilité d'utilisation, sa rapidité, la modélisation d'atmosphères et de géométries de mesures difficiles, et l'amélioration pour l'utilisation dans des conditions opérationnelles.

Table of contents

Abstract	i
Résumé	i
Executive summary	iii
Sommaire	iv
Table of contents	v
List of figures	viii
Acknowledgements	x
1 Introduction.....	1
2 Objectives	2
3 Methodology.....	3
4 Modeling an image degraded by the atmosphere	5
4.1 Uniform and broken cloud cover.....	5
5 Target modelling.....	8
5.1 Solid targets.....	8
5.1.1 Fractional pixel coverage.....	8
5.2 Gas plumes	8
6 Sensor modeling	10
6.1 Smile distortion	10
6.2 Spectral resampling	12
6.3 Sensor response	12
6.4 Noise.....	12
6.5 Radiometric calibration	13
7 Atmospheric compensation algorithms	14
7.1 MODTRAN.....	14
7.2 ELM.....	14
7.3 STAC.....	15
7.4 None	15
8 Temperature-emissivity separation.....	16
8.1 NEM	16
8.2 Defilte.....	16
9 Target Detection Algorithms	17
9.1 GLRT.....	17
9.2 Signed GLRT.....	17
9.3 ACE	17
9.4 Signed ACE.....	18
9.5 AMF	18

9.6	CEM	18
9.7	Hybrid.....	18
9.8	Gas detection	18
10	ROC curve analysis	20
10.1	Curve area index (CAI)	21
11	Future perspective and recommendations.....	22
11.1	Independent configuration of the MODTRAN compensation	22
11.2	Erroneous radiometric correction	22
11.3	Better smile implementation.....	22
11.4	More realistic broken cloud implementation.....	22
11.5	Sideways non-nadir sensor geometries.....	23
11.6	Reducing Beer's law violations.....	24
11.7	Parallelization of slow algorithms	24
11.8	Code cleanup and parameter verification	25
11.9	C++ (QT) user interface	25
11.10	Validation with field measurements	25
11.11	Validation with simulated images	25
11.12	Addition of other detection, atmospheric compensation and TES algorithms	25
11.13	Extend tool to the visible bands.....	26
11.14	Interface with meteorological predictions from major met centers.....	26
11.15	Simplified mode of use for operational contexts.....	26
11.16	More flexible file units	26
11.17	Enable scenario saving and loading.....	26
12	Conclusion.....	27
13	Additional information	27
Annex A	Results and sensitivity studies	28
A.1	Solid Target	29
A.1.1	Effect of altitude	30
A.1.2	Effect of nadir angle	33
A.1.3	Effect of cloud altitude/type (100% coverage)	34
A.1.4	Effect of cloud coverage	37
A.1.5	Effect of pixel fraction.....	40
A.1.6	Effect of target-air temperature difference	42
A.1.7	Effect of sunlight	44
A.1.8	Noise level	45
A.1.9	Smile effect.....	47
A.1.10	Effect of slit function shape.....	48
A.1.11	Effect of using a constant background temperature.....	52
A.1.12	Conclusion	53
A.2	Gas plumes	54
A.2.1	Effect of plume temperature	55

A.2.2 Smile	58
References	59
List of symbols/abbreviations/acronyms/initialisms	61

List of figures

Figure 1 Methodology outline	3
Figure 2: Schematic representation of the different MODTRAN calculations necessary to approximate simulations from above a cloud deck.	6
Figure 3 : Smile simulation effects from Aktaruzzaman [14].	11
Figure 4 : Smile curve for SEBASS sensor in band 102. [14]	11
Figure 5 : Smile curve for Hyperion in band 41. [14]	11
Figure 6: Representation of detection and false alarm probability calculations (figure from ref [13]).....	20
Figure 7: Example of ROC curves.	21
Figure 8: Ambiguous ROC curves example.....	21
Figure 9: Current non-nadir geometry.....	23
Figure 10: Example of possible non-nadir geometry not currently implemented	24
Figure 11: Average emissivity (left) and temperature (right) images used in the parameter sensitivity study.....	29
Figure 12: Emissivity spectrum of ammonium sulfate.....	30
Figure 13: ROC curve analysis for the altitude effect at 2 km	32
Figure 14: ROC curve analysis for the altitude effect at 20 km	33
Figure 15: ROC curve analysis for the nadir angle effect at 82 degrees	34
Figure 16: ROC curve analysis for cloud type and altitude for cumulus at 0.66 km.	35
Figure 17: ROC curve analysis for cloud type and altitude for altostratus at 3 km.....	36
Figure 18: ROC curve analysis for cloud type and altitude for standard cirrus.	37
Figure 19: ROC curve analysis for cloud coverage effect at 75%.	38
Figure 20: ROC curve analysis for cloud coverage effect at 50%.	39
Figure 21: ROC curve analysis for cloud coverage effect at 25%.	40
Figure 22: ROC curve analysis for pixel fraction at 0.25.....	41
Figure 23: ROC curve analysis for pixel fraction at 0.1.....	42
Figure 24: ROC curve analysis for target-air temperature difference at -10°C.	43
Figure 25: ROC curve analysis for target-air temperature difference at +10 °C.	44
Figure 26: ROC curve analysis for the effect of having no sunlight.	45
Figure 27: ROC curve analysis for the effect of sensor noise at 0.05 RMS.	46
Figure 28: ROC curve analysis for the effect of sensor noise at 0.1 RMS.....	47

Figure 29: ROC curve analysis for the effect neglecting the smile effect.	48
Figure 30: ROC curve analysis for the square slit function.	50
Figure 31: ROC curve analysis for the Gaussian slit function.	51
Figure 32: ROC curve analysis for the Lorentzian slit function.	52
Figure 33: ROC curve analysis for a constant temperature background.	53
Figure 34: Absorbitivity of NH_3 (ammonia) gas at 25 °C.	54
Figure 35: ROC curve analysis for a gas with temperature 5°C cooler than the air.	56
Figure 36: ROC curve analysis for a gas at air temperature.	57
Figure 37: ROC curve analysis for a gas with temperature 5°C warmer than the air.	57
Figure 38: ROC curve analysis for a gas without the smile effect.	58

Acknowledgements

The authors would like to thank Mr. Paul Lacasse for his thorough review and useful comments and insight regarding the production of this document.

1 Introduction

Atmospheric correction is usually considered an essential part of the process leading to the detection of a target of interest in a multispectral or hyperspectral airborne image. The result of such a correction is the emissivity of the surface, which is an intrinsic property of a material (independent of the environment) that can be compared to database entries or measurements. In the long wave infrared, atmospherically correcting an image is a two-step process. The first step consists in estimating or calculating the atmosphere's effect on the image and removing this influence. This is called atmospheric compensation. The second step is to separate the effect of the surfaces temperature and their emissivity on the measured radiance. These two steps are not necessarily independent, because removing the atmosphere's effect also requires knowledge of the emissivity. Once the emissivity is known and all effects of the temperature and atmosphere have been removed, target detection can be attempted.

In addition to atmospheric effects, other factors can impinge on the capability to detect a target in an image. Many of these factors are sensor based and include noise, spectral distortions (smile, frown) and spectral calibration errors.

Finally, the detection strategy will also have an obvious impact on the capacity to detect a target within an image. Many detection algorithms have been conceived to function optimally with different idealized background and target statistical compositions. It is likely that the treatment (correction and calibration) will have different impact on detection algorithms based on different hypotheses.

In this document, we describe a tool called CODE for COrrrection and Detection Evaluation (hereafter referred to as the tool) designed to evaluate the effect of different combinations of atmospheric conditions, atmospheric correction algorithms and sensor degradations on the outcome of some detection algorithms. Such a tool can be used to design reconnaissance missions more effectively by adapting the algorithms to the sensor and atmosphere for optimal results. It can also become an effective tool to verify the effectiveness of newly designed algorithms and sensors.

2 Objectives

The global objectives of this project, as stated in the statement of work attached to contract W7701-125216/001/QCL and to AEREX Proposal Number AAI-2012-125216 [1], are to conduct scientific research on the field of atmospheric optics relative to panchromatic, multispectral and hyperspectral airborne imaging applications. The objective of this particular work is to provide a tool to assess the impact of the atmosphere on the measurement of hyperspectral images, on the capacity of algorithms to correct these effects and on the capacity of algorithms to detect targets of interest in the corrected images.

3 Methodology

The methodology adopted for this project follows the steps outlined in Figure 1, and includes

- a. Bibliographic research;
- b. Familiarization with typical hyperspectral data, sensors and algorithms;
- c. Prototyping of a tool in Matlab;
- d. Validation of the design and features with the scientific authority.
- e. Incremental upgrades to the tool with feedback from the scientific authority.

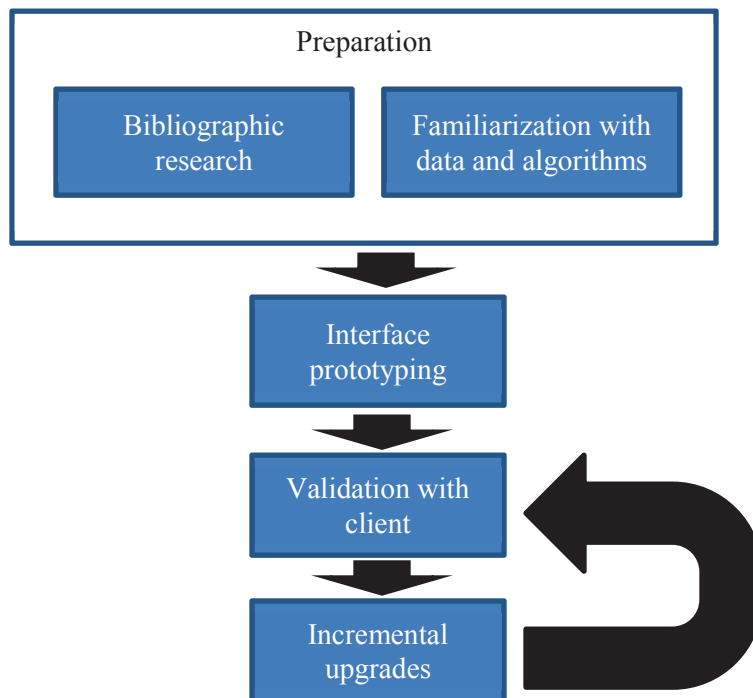


Figure 1 Methodology outline

In a preparation period, a thorough bibliographic research was done in order to understand the current state of the atmospheric correction field in the long wave infrared (LWIR), available algorithms as well as their benefits and drawbacks. Peripheral material was also gathered in connected fields such as radiative transfer. Concurrently, a typical corrected (emissivity) image to be used as a background image was acquired.

Once possible approaches are assessed and balanced with the project requirements, a promising method can be proposed to assess target detection with varying atmospheric conditions, sensor effects and corrections. The method can be an existing and proven one with possible enhancements to minimize risk or an entirely new method for the possibility of even greater performance, but at the cost of greater risk.

Once the proposed method is approved, it is implemented behind a graphical user interface. The interface and algorithms are presented to the scientific authority who then provides feedback. The interface and algorithms are modified until the result is satisfactory.

4 Modeling an image degraded by the atmosphere

In the model developed under this project, we assume that we have access to images which have been atmospherically corrected perfectly. Although this assumption is not entirely reasonable, very good correction can be achieved when reference targets are located within the image with known emissivity and temperatures. In any case, the image will be considered good enough to reproduce a radiative background onto which the target of interest will be superimposed to assess its detectability.

For a given pixel within the image, if the emissivity ε and temperature T are known, the surface is considered to be flat and horizontal and its reflectance Lambertian, the radiance R measured from an airborne detector is

$$R = \left(\varepsilon B(T) + \frac{(1-\varepsilon)}{\pi} L_{\downarrow} \right) \tau + R_p \quad (1)$$

where B is the blackbody (Planck) function, $L_{\downarrow}$ is the downwelling irradiance, τ is the atmospheric transmittance and R_p is the atmospheric path radiance accumulated between the sensor and target by emissions and scattering in the air. All quantities in equation (1) have spectral dependence, but it is left implicit in this equation and throughout this document for conciseness. Quantities R_p , τ and $L_{\downarrow}$ are modelled using MODTRAN 5.3 [2] and resampled to the sensor resolution. Note that to minimize errors associated with applying Beer's law to band quantities, the product of the transmittance τ and downwelling irradiance $L_{\downarrow}$ is obtained before resampling is done.

4.1 Uniform and broken cloud cover

When a uniform cloud deck is present (100% cloud coverage) and the sensor is located lower than the bottom of the clouds, MODTRAN is simply configured to include clouds when calculating R_p and $L_{\downarrow}$. When the same situation arises with a broken cloud deck, two MODTRAN simulations are necessary to calculate R_{atm} and $L_{\downarrow}$. The first calculates clear sky R_p^{clear} and $L_{\downarrow}^{clear}$, and the other cloudy sky R_p^{cloud} and $L_{\downarrow}^{cloud}$. The transmittance τ is identical for both simulations. Results of both MODTRAN simulations are combined according to the cloud fraction (from 0 to 1) f_c with

$$\begin{aligned} L_{\downarrow} &= f_c L_{\downarrow}^{cloud} + (1 - f_c) L_{\downarrow}^{clear} \\ R_p &= f_c R_p^{cloud} + (1 - f_c) R_p^{clear} \end{aligned} \quad (2)$$

When the sensor is located in or above the cloud deck, the model assumes that the target and background are visible through an opening in the cloud deck, even when the cloud coverage is 100%. In this case, three different MODTRAN runs are necessary to approximate R_p and $L_{\downarrow}$ as shown in Figure 2:

- A) From the sensor to the bottom of the cloud deck with no clouds in the model

- B) From the bottom of the cloud to the ground with no clouds in the MODTRAN modelling
- C) From the bottom of the cloud to the ground with clouds enabled in the MODTRAN modelling

Transmittances from the A and B runs are also computed and stored.

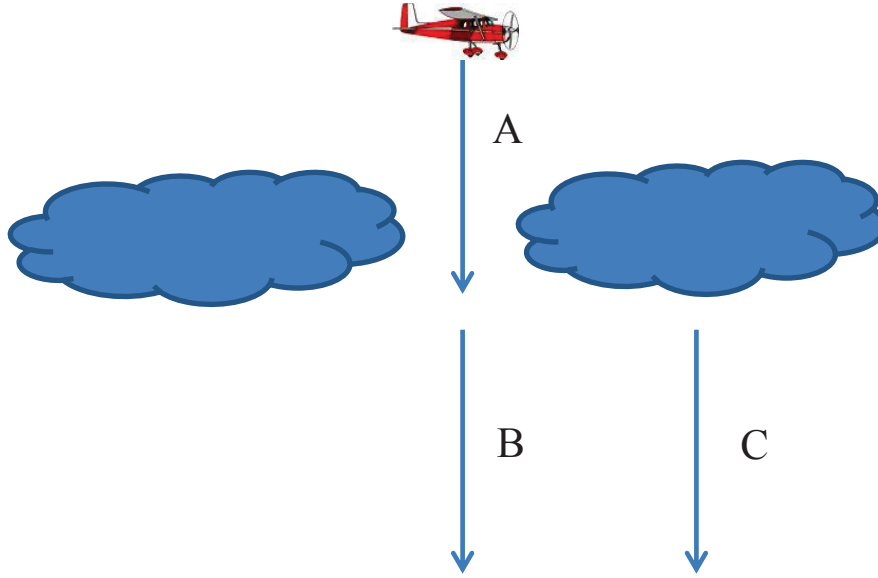


Figure 2: Schematic representation of the different MODTRAN calculations necessary to approximate simulations from above a cloud deck.

Using A, B and C superscripts with the appropriate quantities, the resulting downwelling flux, path radiance and transmittances are obtained from

$$\begin{aligned}
 L_{\downarrow} &= L_{\downarrow}^A + \tau^A \left[f_c L_{\downarrow}^C + (1 - f_c) L_{\downarrow}^B \right] \\
 R_p &= R_p^A + \tau^A \left[f_c R_p^C + (1 - f_c) R_p^A \right] \\
 \tau &= \tau^A \tau^B
 \end{aligned} \tag{3}$$

It is important to note that equation (3) makes several approximations. The first and probably the most important is the fact that the equations violate Beer's law. To minimize the error, all quantities are computed at full MODTRAN resolution (the actual resolution is configurable by the user) before being resampled at the image resolution. The other approximation that might affect flux and radiance calculation accuracy is the fact that path A calculations are done without a cloud deck. This means that the lower boundary condition in the multiple scattering calculations in MODTRAN will be incorrect.

Note that in the case of cirrus cloud models, the line of sight goes through the cloud layer, no matter the cloud coverage (equations (2) are used). This is because the cloud altitude and thickness are not known, since they are calculated internally in MODTRAN.

5 Target modelling

5.1 Solid targets

In the modeling tool developed here, targets are considered flat, Lambertian reflectors lying on the ground. Because of this, equations (1) to (3) are applicable directly to target apparent radiance calculations.

5.1.1 Fractional pixel coverage

Solid targets can be set to occupy only a fraction of a pixel. This means that the target pixel will be mixed with the background pixel, making it more difficult to detect. In this case, the target pixel spectrum depends on its location in the image, and is given by

$$R_{mixed} = f_p R_t + (1 - f_p) R_b \quad (4)$$

where f_p is the pixel fraction, R_t is pure target radiance and R_b is pure background radiance.

5.2 Gas plumes

Input files absorption coefficients are in units of $\text{ppm}^{-1}\text{m}^{-1}$ at 25 °C and 1013.1 mbar (hPa). The absorption coefficients in units of m^{-1} at plume temperature T_{plume} and pressure P_{plume} and at gas concentration C_g are given by

$$k_{abs} \left[\text{m}^{-1} \right] = \frac{298.15}{T_{plume}} \frac{P_{plume}}{1013.1} C_g \times k_{abs} \left[\text{ppm}^{-1}\text{m}^{-1} \right] \quad (5)$$

where the plume temperature is the temperature of the gas-air mixture. The plume pressure is the atmospheric pressure at the plume level (ground level in the present implementation).

Using the absorption coefficients in units of m^{-1} , the transmittance through a plume of thickness t is then (assuming that the coefficients are in e base)

$$\tau_{plume} = \exp(-k_{abs} t) \quad (6)$$

while the radiance emitted by the plume is

$$R_{plume} = (1 - \tau_{plume}) B(T_{plume}). \quad (7)$$

The background radiance measured through an absorbing and emitting plume of gas can be calculated with equation (1), substituting R_p , τ and $L_{\downarrow}$ with

$$\begin{aligned}
L'_{\downarrow} &= L_{\downarrow} \tau_{plume} \\
\tau' &= \tau \tau_{plume} \\
R'_p &= R_p + \tau R_{plume}
\end{aligned}
\tag{8}$$

Note that equation (8) for R'_p is only valid if the thickness of the plume is much smaller than the altitude of the sensor because we use τ from the ground instead of from the top of the plume. This approximation is usually the case for airborne remote sensing.

6 Sensor modeling

6.1 Smile distortion

The smile or frown effects in hyperspectral images corresponds to an optical defect that is caused by a difference in the position of the band center for a given band depending on its position on the detector array. It is in fact a spectral distortion as demonstrated in Figure 3 taken from [14]. Wavelengths as well as the spectral resolution are not identical in pixels in the center or on the edges of the image [15][16][17]. This effect, if not corrected adequately will affect the spectrum extracted from the hyperspectral images. The phenomenon is typically observed in images taken from “push-broom” type sensors [14] as is the case for Hyperion and SEBASS. These sensors possess a spectrometer whose detector consists of a single pixel line.

The tool being developed, the smile (frown) curves are provided by the user in file format or described by the coefficients of a fifth order polynomial. The use of a polynomial is a good approximation as demonstrated by the modeled smile curves for the Hyperion and SEBASS sensors illustrated in Figure 4 and Figure 5. These curves are for any given spectral band and are extracted from [14].

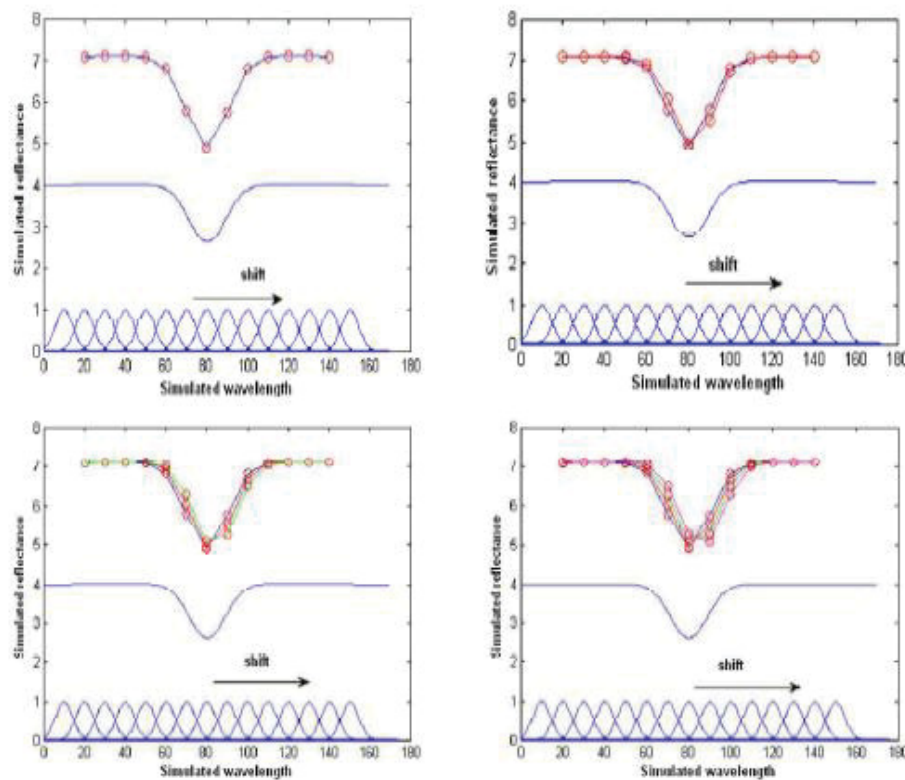


Figure 2-5 Simulation of smile effect; resampled spectrum without any shift in response function (top left), resampled spectra with 1 nm shift (top right), with 1+ 2 nm shift (bottom left) and with 1+2+3 nm shift (bottom right).

Figure 3 : Smile simulation effects from Aktaruzzaman [14].

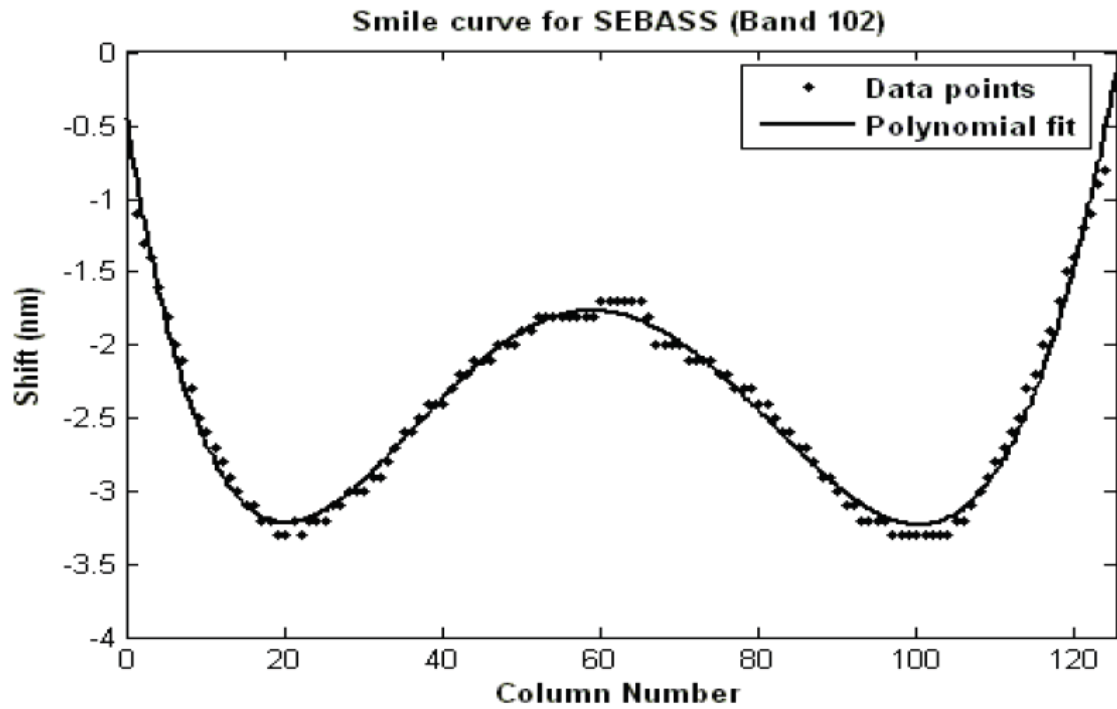


Figure 4 : Smile curve for SEBASS sensor in band 102. [14]

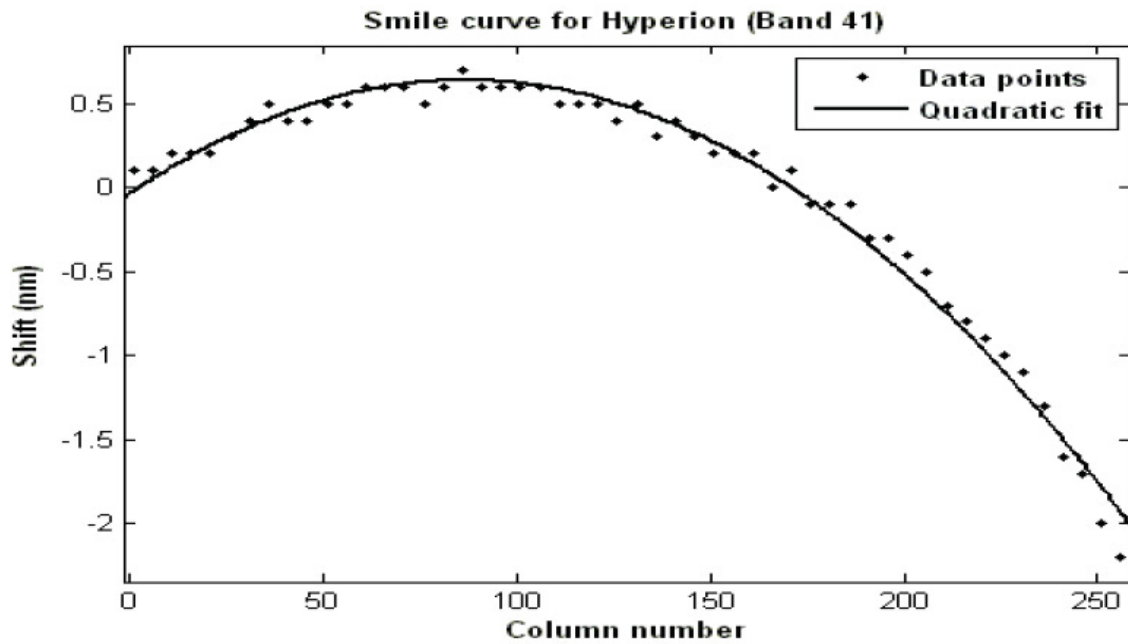


Figure 5 : Smile curve for Hyperion in band 41. [14]

6.2 Spectral resampling

In order to resample the higher resolution target signature and MODTRAN simulations to the sensor resolution, an appropriate slit (window) function can be selected approximating the sensor slit shape. Four different functions are available, represented in Table 1.

Table 1: Resampling windows

	Triangular	Square	Gaussian	Lorentz
Parameter	$w = FWHM$	$w = FWHM$	$\sigma = \frac{FWHM}{2\sqrt{2\ln(2)}}$	$\gamma = \frac{FWHM}{2}$
Wavelength offset vector	$[-w \ 0 \ w]$	$[-w/2 \ -w/2 \ w/2 \ w/2]$	$[-3\sigma \ -2\sigma \ -\sigma \ 0 \ \sigma \ 2\sigma \ 3\sigma]$	$[-4\gamma \ -2\gamma \ -\gamma \ 0 \ \gamma \ 2\gamma \ 4\gamma]$
Weight vector	$[0, 1, 0]$	$[0, 1, 1, 0]$	$[0.011109, 0.135335, 0.606531, 1.000000, 0.606531, 0.135335, 0.011109]$	$[0.058824, 0.200000, 0.500000, 1.000000, 0.500000, 0.200000, 0.058824]$

6.3 Sensor response

Before noise (section 6.4) is added to the background and target radiances, the response function f_s of the sensor is applied. This function represents the spectral sensitivity of the instrument. By this definition, the data output by the sensor is the normalized true radiance, convolved by the normalized slit function and multiplied by the sensor response. So, if R is the resampled radiance of target or background, the sensor signal S is

$$S = Rf_s \quad (9)$$

6.4 Noise

Noise is modeled as being normally distributed with a signal standard deviation (RMS) σ_s . The noisy raw detector signal S_n is then

$$S_n = Sr\sigma_s \quad (10)$$

where r is a random number function producing normally distributed numbers with unit standard deviation.

6.5 Radiometric calibration

Radiometric calibration is simply the process of removing the sensor response from the raw sensor signal to convert it back to radiance units. This is done after noise is added

$$R_n = \frac{S_n}{f_s} \quad (11)$$

where R_n is the final simulated radiance of either the target or the background.

7 Atmospheric compensation algorithms

Atmospheric compensation consists in obtaining the atmospheric degradation to the image in order to remove it from the measurements. This degradation originates from atmospheric absorption and scattering spectrally attenuating the surface radiance (transmittance), from atmospheric radiance emitted and scattered into the sensor's line of sight, and from downwelling irradiance that reflects off the surface modifying its spectra.

7.1 MODTRAN

One straightforward way to obtain the atmospheric radiative quantities, which intervene in degrading the image, is to model them with MODTRAN59[2]. In the context of the tool being developed here, there is an important caveat to be aware of since MODTRAN is also used to degrade the images to be corrected. In reality, MODTRAN would never be able to model the atmosphere perfectly. The correction should therefore be considered to be the *best possible* attainable with this method. The correction is not perfect however, since in spectral regions of low transmittance, the loss of signal will still result in increased noise. Furthermore, spectral distortions found in the image (section 6.1) are not reproduced in the MODTRAN quantities.

A partial solution to the problem for obtaining a more realistic correction is proposed in section 11.1.

7.2 ELM

The empirical line method (ELM) technique is commonly used to apply corrections to visible\ short-wave hyperspectral airborne or satellite hyperspectral image measurements. In the long-wave, ELM must be modified to take into account the emissivity (or reflectivity) of the observed surface. DiStasio [3] presents the Emissivity Empirical Line Method (EELM) that consists of taking into account the emissivity in the calculation. Similar calculations are presented in this document, but are based on the long-wave radiance equation (1).

In the ELM technique for reflective bands (no emissions), two known surfaces at the ground are necessary. In the LWIR three surfaces characteristics must be known. Measurements of the three known surfaces are represented by:

$$\begin{aligned} R_1 &= (\varepsilon_1 B(T_1) + (1 - \varepsilon_1) L_{\downarrow} / \pi) \tau + R_p \\ R_2 &= (\varepsilon_2 B(T_2) + (1 - \varepsilon_2) L_{\downarrow} / \pi) \tau + R_p \\ R_3 &= (\varepsilon_3 B(T_3) + (1 - \varepsilon_3) L_{\downarrow} / \pi) \tau + R_p \end{aligned} \tag{12}$$

Atmospheric components are considered identical in all three equations. This method can also correct the gain and offset due to the sensor. The gain is included in the transmittance factor and the offset in the path radiance. Unknown elements in equation (2) are determined using the matrix equation:

$$\begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix} = \begin{bmatrix} \varepsilon_1 B(T_1) & (1 - \varepsilon_1) & 1 \\ \varepsilon_2 B(T_2) & (1 - \varepsilon_2) & 1 \\ \varepsilon_3 B(T_3) & (1 - \varepsilon_3) & 1 \end{bmatrix} \begin{bmatrix} \tau \\ \tau L_{\downarrow}/\pi \\ R_p \end{bmatrix} \quad (13)$$

With the value of τ , R_p and $L_{\downarrow}$, atmospheric and sensor correction can be applied to the radiance in the image.

7.3 STAC

STAC is an algorithm developed at DRDC Valcartier by Pierre Lahaie [4]. The algorithm consists in finding a high emissivity pixel (also referred to as the cold pixel) in the image as well as warm pixel with low spectral variance. These two pixels, along with simple assumptions, let STAC compute the atmospheric transmittance and path radiance. Although STAC can estimate the atmosphere's mean temperature, no downwelling irradiance is provided.

7.4 None

The tool also offers the choice of applying no atmospheric compensation before temperature-emissivity separation and target detection. This equates to setting $t = 1$, $R_p = 0$ and $L_{\downarrow} = 0$ on all spectral bands.

8 Temperature-emissivity separation

Temperature-emissivity separation (TES) algorithms, as their name suggests, separate the temperature (through the Plank function) from the surface emissivity. The result is the surface emissivity, which can then be used for detection purposes.

8.1 NEM

The simplest and fastest method is the normalized emissivity method first described in [5]. The method consists in assuming that the pixel has constant emissivity across the entire spectral band (usually close to 1), calculating the equivalent temperature by inverting the Blackbody function, and choosing the maximum spectral temperature as the surface's actual temperature. Once the temperature is known, the emissivity is obtained by isolating it in equation (1), that is:

$$\varepsilon = \frac{R - R_p - L_{\downarrow}\tau/\pi}{B(T) - L_{\downarrow}\tau/\pi} \quad (14)$$

where τ , R_p and $L_{\downarrow}$ are provided by the selected atmospheric compensation method.

8.2 Defilte

The TES technique known as Defilte [6] is based on iterations on temperature principle, where a total square error criterion is used to estimate the temperature. Once the temperature is estimated, the emissivity is obtained with (14). The algorithm involves spectra filtering and function minimizations, so it is much slower than using NEM, but has the potential to be more accurate.

9 Target Detection Algorithms

9.1 GLRT

The Generalized likelihood ratio test (GLRT), also known as Kelly's detector [7] is optimal for Gaussian probability distributions of the target and background. The detector assumes that the background has the same covariance structure and same variance whether or not a target is present. The detector is given by

$$GLRT = \frac{\left[(s-u)^T \Gamma^{-1} (x-u) \right]^2}{\left[(s-u)^T \Gamma^{-1} (s-u) \right] \left[1 + \frac{1}{N} (x-u)^T \Gamma^{-1} (x-u) \right]} \quad (15)$$

where s is the reference spectrum (target), x is the spectrum of the observed pixel, u is the background average, Γ is the covariance matrix of the background and N is the number of background samples.

9.2 Signed GLRT

The signed GLRT is basically the same as the GLRT, but with the sign of the square in the numerator, or,

$$sGLRT = \frac{\left[(s-u)^T \Gamma^{-1} (x-u) \right] \left[(s-u)^T \Gamma^{-1} (x-u) \right]}{\left[(s-u)^T \Gamma^{-1} (s-u) \right] \left[1 + \frac{1}{N} (x-u)^T \Gamma^{-1} (x-u) \right]} \quad (16)$$

This algorithm generally performs better for solid opaque targets since anticorrelated targets are least likely to produce detection false alarms. The normal GLRT on the other hand should perform better for gases, as they can emit (correlated detection) or absorb (anticorrelated detection).

9.3 ACE

The adaptive cosine estimator (ACE) [8] is similar to the GLRT, assuming that the background has the same covariance structure whether or not a target is present, but allows for a different variance. The detector is given by

$$ACE = \frac{\left[(s-u)^T \Gamma^{-1} (x-u) \right]^2}{\left[(s-u)^T \Gamma^{-1} (s-u) \right] \left[(x-u)^T \Gamma^{-1} (x-u) \right]} \quad (17)$$

9.4 Signed ACE

The signed ACE is basically the same as the ACE detector, but with the sign of the square in the numerator, or,

$$sACE = \frac{\left[\left[(s-u)^T \Gamma^{-1} (x-u) \right] \right] \left[(s-u)^T \Gamma^{-1} (x-u) \right]}{\left[(s-u)^T \Gamma^{-1} (s-u) \right] \left[(x-u)^T \Gamma^{-1} (x-u) \right]} \quad (18)$$

This algorithm generally performs better for solid opaque targets since anticorrelated targets are least likely to produce detection false alarms. The normal ACE on the other hand should perform better for gases, as they can emit (correlated detection) or absorb (anticorrelated detection).

9.5 AMF

The Adaptive Matched Filter (AMF) [9] detector is optimal when target and background follow multivariate normal distributions with same covariance matrix. Even though this situation is unlikely to occur, the AMF can still offer interesting performances. The detector is given by

$$AMF = \frac{\left[(s-u)^T \Gamma^{-1} (x-u) \right]}{\left[(s-u)^T \Gamma^{-1} (s-u) \right]} \quad (19)$$

9.6 CEM

The Constrained Energy Minimization (CEM) [10] detector minimizes the energy of the matched filter output in order to suppress the noise and unknown signal sources. It is very similar to the AMF filter but uses a correlation matrix instead of a covariance matrix

$$CEM = \frac{\left[(s-u)^T R^{-1} (x-u) \right]}{\left[(s-u)^T R^{-1} (s-u) \right]} \quad (20)$$

where R is the correlation matrix of the background.

9.7 Hybrid

The Hybrid detector was developed at DRDC Valcartier. It Combines Kelly's GLRT and CEM to attempt a best of both worlds approach. The more complex algorithm can be somewhat slower than other detectors.

9.8 Gas detection

As can be seen in this section's previous subsections, all algorithms require a reference emissivity spectrum. For solid targets, this is trivially the emissivity spectrum. For gases, it is not as obvious.

From equation (7), we can see that the emissivity of the gas plume is $1 - \tau_{plume}$. This means that the emissivity depends on the transmittance through the plume, which in turns means that it depends on the plume concentration and thickness (though equations (5) and (6)). But in a real detection scenario, these quantities are *a priori* unknown. In the current implementation of the tool, we assume that these quantities are known exactly, so results should be considered a best case scenario.

10 ROC curve analysis

In order to determine the effectiveness of detection algorithms, the tool uses graphical aids called ROC (Receiver Operating Characteristic) curves. These are calculated by applying the detector to the background image (no target present), and computing the probability distribution of the detector output f_0 . The detector is then applied to a second image where the target signature is mixed in with the background according to pixel fraction or gas concentration, yielding the target probability distribution f_1 . For each value of detection threshold η , the probabilities of detection P_D and of false alarms P_{FA} are calculated with

$$\begin{aligned} P_{FA} &= \int_{\eta}^{\infty} f_0(y) dy \\ P_D &= \int_{\eta}^{\infty} f_1(y) dy \end{aligned} \quad (21)$$

where y represents possible detector output values. This is represented by Figure 6 extracted from ref [13].

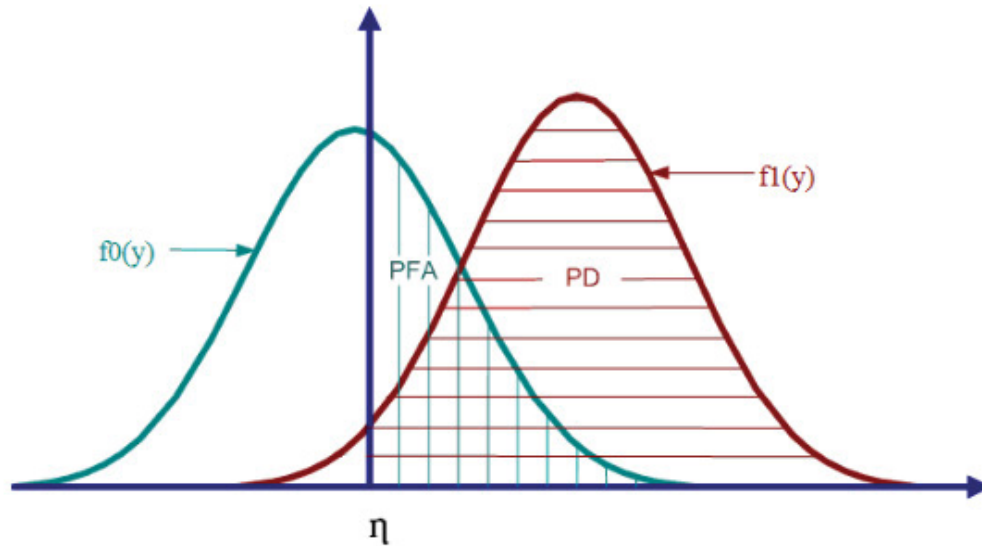


Figure 6: Representation of detection and false alarm probability calculations (figure from ref [13])

Plotting P_D against P_{FA} on a logarithmic scale produces the so-called ROC curves. Typical examples are shown in Figure 7. Since a better algorithm produces a lower rate of false alarms with higher detection probabilities, a ROC curve dropping down further to the left of the graph is desirable. In the example of Figure 7, the red ROC curve represents a better performing algorithm than the blue curve.

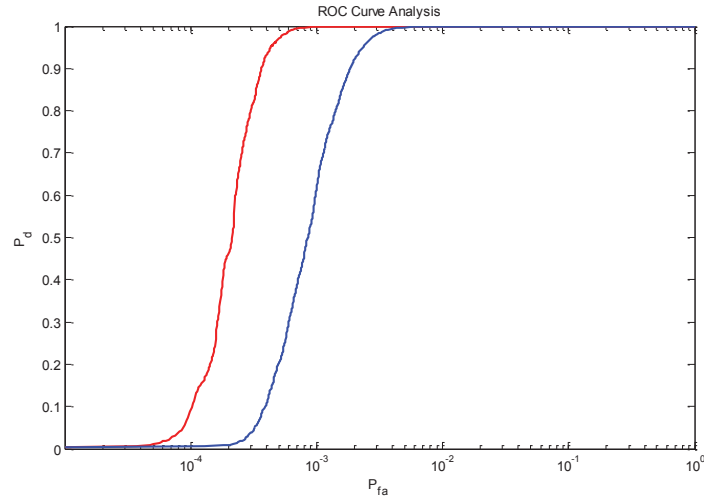


Figure 7: Example of ROC curves.

10.1 Curve area index (CAI)

Rating the performance of different detection algorithms is not always as obvious as in the example of Figure 7. Take the example of Figure 8 where the blue curve takes longer to drop, but starts sooner than the red curve. In these cases, another metric, the curve area index can help. The CAI is simply the integrated area under the ROC curve. In the example of Figure 8, the blue curve has a CAI of 0.99664, while the red curve has a CAI of 0.99972 making it the stronger algorithm under the CAI metric.

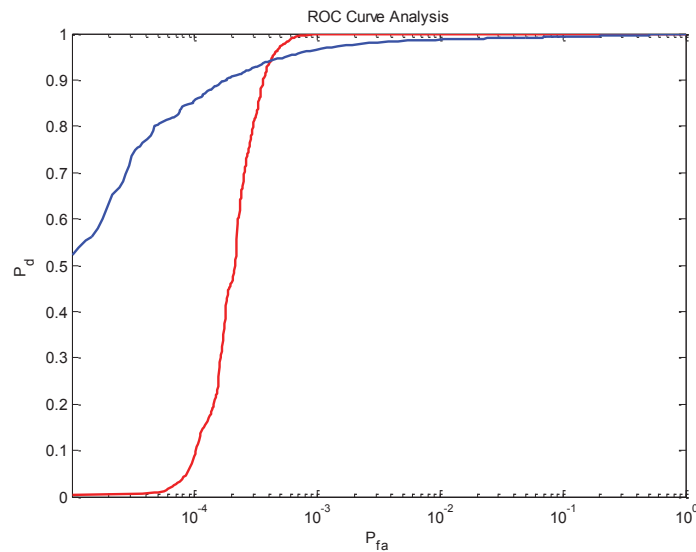


Figure 8: Ambiguous ROC curves example

11 Future perspective and recommendations

11.1 Independent configuration of the MODTRAN compensation

As mentioned in section 7.1, image degradation and atmospheric compensation use the same MODTRAN parameters, and thus the exact same results. This is a problem as the result of the compensation will always be unrealistically good. One way to attempt to simulate a more approximate compensation would be to have the possibility to configure MODTRAN independently for the degradation and correction. That way, the effect of errors in atmospheric parameterization or the use of more approximate algorithms (multiple scattering) could be explored.

11.2 Erroneous radiometric correction

Just as with the MODTRAN compensation, the radiometric correction that converts the simulated radiance into raw sensor measured quantities is used exactly to convert back to radiance. This might cause overestimation of detector capabilities. The possibility to configure the radiometric calibration independently for the degradation and correction would permit the simulation of calibration errors.

11.3 Better smile implementation

Currently, the smile effect only depends on the column position of the pixel in the image. All wavelengths are shifted in an identical fashion. In reality, the amplitude of the smile shift also depends on the wavelength itself. Such a distortion could easily be implemented, but would make the sensor characterization and user interface more complicated.

For some atmospheric compensation algorithms, notably MODTRAN, the smile effect is considered unknown, meaning that the distortion is not applied to the atmospheric quantities before correction. When the smile distortion is known, it is possible to consider the smile effect when doing the atmospheric compensation, yielding better results. An option to do this could be included in the interface.

11.4 More realistic broken cloud implementation

In the current implementation, broken clouds are treated as a weighed mean of cloudy and clear conditions (section 5.1.1). Work being done concurrently to the present contract task, ref [11], demonstrates that this is in fact a good approximation in the LWIR. The problem is that this weighed average is applied to the entire image background uniformly. In reality, the effective cloud coverage over the span of large images might change, adding background variations. Since most detection algorithms are concerned with approximating the background true statistical assumptions, they could be affected by such variability.

The concurrent work reported in ref [11] produced a model that stems from Monte-Carlo principles to estimate the variability of the downwelling flux. The span of the variability (plus or minus the standard deviation) could be applied to two copies of the same background image. These two images side-by-side would then constitute a new background encompassing the downwelling flux variability due to broken cloud coverage. A similar two-pane image would be produced for the target-background mixture image. Feeding these new background and target images into the detectors would produce more realistic ROC curves in these circumstances.

11.5 Sideways non-nadir sensor geometries

The current tool has the option to set a non-nadir viewing geometry for the sensor. This geometry is then applied to each pixel uniformly. This simulates the situation where a push-broom type sensor is held at an angle in front of the airplane (Figure 9).

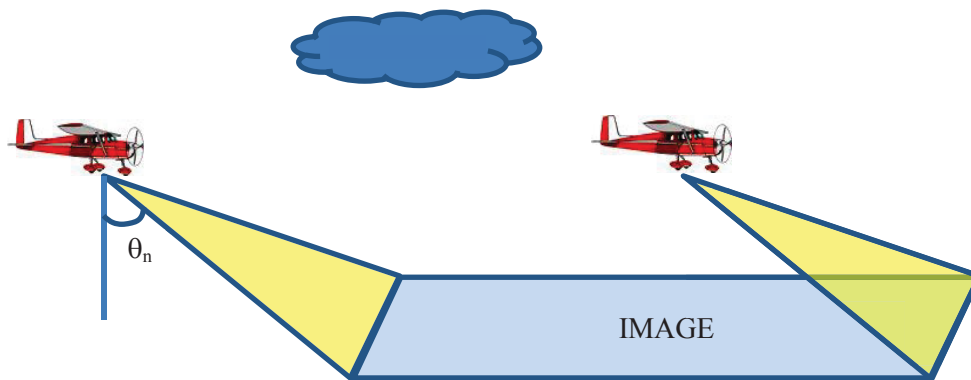


Figure 9: Current non-nadir geometry

In reality, this geometry is not representative of many true non-nadir measurements. Another possible non-nadir measurement geometry is shown in Figure 10 where the push-broom sensor is directed sideways from the flight line. This measurement geometry is problematic, since transmittance and path radiance might differ significantly from the close to the far side of the image, because of path length differences. This would in turn affect the background estimation of the detectors and possibly alter the ROC curves significantly.

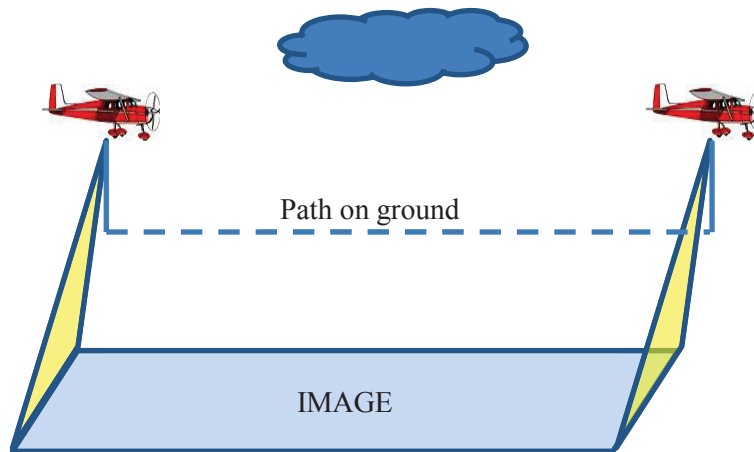


Figure 10: Example of possible non-nadir geometry not currently implemented

The problem of implementing this viewing geometry resides in calculating different transmittance for each image column with reasonable computational times. On the other hand, interpolating between two MODTRAN runs at the extremities involves violating Beer's law and would be another source of error. One solution is to interpolate in the new correlated-k MODTRAN outputs, which should dramatically reduce the effect of Beer's law violations (see section 11.6).

11.6 Reducing Beer's law violations

In many sections of this document, algorithms are described that make assumptions that violate Beer's law for multiplying segment transmittances. This leads to some loss in accuracy. One solution to this would be to use the newly added correlated-k output in MODTRAN. Applying Beer's law to multiply transmittances in the correlated-k space leads to much better results. Interpolation of transmittances in this way is also more accurate, since the correlated-k transmittances can be considered exponential by segment, an assumption that is erroneous for band transmittances.

11.7 Parallelization of slow algorithms

Some algorithms are somewhat slow to the point of being difficult to use in realistic operational contexts, where fast results are often a prerequisite to fast decision making. In the current implementation, this is the case of the Defilte correction (section 8.2) and to a lesser extent of the Hybrid (section 9.7) detection algorithm. As many algorithms that deal with hyperspectral imagery, these algorithms might benefit from parallel computing implementation to benefit from multicore architectures, of even of the hyper-parallel GPU architectures.

Implementation could be attempted in Matlab using the parallelization tool box, or third party Matlab plug-in such as Jacket for GPU implementation. Another, more flexible and portable solution would be to implement these algorithms directly in the C++ programming language using multithreading libraries.

11.8 Code cleanup and parameter verification

The current form of the tool can be described as a prototype. The tool would benefit greatly from code cleanup and better documentation for easier maintenance.

User entered parameters are for the most part not verified for correctness, and even often for existence. In order to protect from unexpected crashed or unintended behaviour, parameter verification should also be better implemented.

11.9 C++ (QT) user interface

Although great effort was deployed to make the graphical user interface as convivial as possible, there are some intrinsic limits to using a tool like Matlab to create such interfaces. A more streamlined interface created in C++ with a tool like QT (free library from the Nokia Corporation) would make for a more refined tool and enjoyable user experience.

11.10 Validation with field measurements

As with any scientific tool, validation is important if it is to be trusted as the basis for potentially important decision making. For a tool such as this, validation is not trivial though. One cannot go about hiding thousands of small unresolved targets in a scene and verify the false alarm vs. detection rate in real airborne images.

One possibility for validation, which would include all the effects of the sensor and the atmosphere, can be devised. If a single target, large enough to be fully resolved by the sensor, can be placed within the scene, both the background and target signatures degraded by both atmosphere and sensor could be measured. The target could then be removed from the image (by replacing the target area with pixels from the nearby background for example). The target signature could then be mixed within each background pixel according to the desired pixel coverage, and ROC curves obtained from purely measured data. These could then be compared with purely simulated ROC curves using the exact same corrected background and laboratory target measurements as input.

11.11 Validation with simulated images

It is expected that for operational uses around the world, well corrected and representative background images will not always be available. It is therefore of interest to verify if synthetic emissivity images calculated pixel temperature can be used instead.

11.12 Addition of other detection, atmospheric compensation and TES algorithms

Although the current tool implements a relatively large number of sample detection, compensation and TES algorithms, many more exist and could be of interest to the DRDC. With the guidance of the DRDC and intended users, other algorithms of interest could be added to the current roster to make the tool even more useful.

11.13 Extend tool to the visible bands

Currently, only the LWIR is modeled by the tool, but the detection and correction principles often apply to other bands such as the visible and short/midwave infrared. It might be of interest to expand and modify the tool to also function in other bands.

11.14 Interface with meteorological predictions from major met centers.

For operational uses, the objective of the tool will often be to predict the performance of different combinations of correction and detection algorithms to make appropriate decisions for planning missions. This implies knowledge of the atmospheric conditions in the near future. This process could be streamlined by connecting the tool directly to available prediction data issued by large meteorological agencies such as Environment Canada, and possibly others like the UK Met Office or Météo France.

11.15 Simplified mode of use for operational contexts

For many operational users who do not necessarily possess knowledge of such things as MODTRAN parameters, using the tool could seem at first like a daunting endeavour. A simplified version or mode of the tool could hide such parameters and use defaults or guess the parameters from other or simplified input.

11.16 More flexible file units

In the current version, the user is responsible for providing all file entries (except the emissivity image) with wavelengths in units of microns and fixed units for the relative quantities. This burden might be eliminated by providing the user the option to choose file units from a list.

11.17 Enable scenario saving and loading

Saving and loading of the parameters for a scenario is not currently possible. This feature would be a welcomed addition for future versions.

12 Conclusion

This report described the CORrection and Detection Evaluation (CODE) tool, which makes it possible for a user to evaluate different combinations of atmospheric conditions, atmospheric compensation tools, and detection algorithms to determine the possibility of detecting subpixel targets or gas plumes on a cluttered background. This is done by simulating the radiative signature of both target and background under the influence of atmospheric and sensor degradation of the underlying emissivity spectra. ROC curves are then used to assess the performance of chosen detectors when combined with different correction algorithms under the prevailing, possibly detrimental atmospheric conditions. This tool will be useful when planning missions where target detection plays an important role.

13 Additional information

Documentation (user guide) for the resulting CODE tool can be found in [18] and work relating to this contract on modelling the effect of broken clouds on downwelling irradiance can be found in [11].

Annex A Results and sensitivity studies

Even though a tool has been developed to be able to study specific cases quickly in an operational context, it can be very instructive to study the effect of certain parameters on the results for a typical case. Such a study provides clues to which parameters have the most effect, so that mission planning can ensure they are measured. It can also be of use to decision makers in order to develop a feel of how parameters will eventually affect the results in the instances where actual circumstances differ slightly from the predicted mission parameters used in the pre-mission simulations.

The main caveat is that it is impossible to model all combination of parameters because of their great number. Doing this would not be useful either, since no person would be able to assimilate such a large quantity of information. Furthermore, detection algorithms are highly sensitive to the image statistics. Since only a single background image and one solid and gas target are used in this study, it is not useful to discuss the relative efficiency of the algorithms. Because of this, only the overall effect of the parameters in a more general sense will be discussed. The determination of the best algorithms to use in a particular case should always be done with the tool provided.

The atmosphere is modeled using the Midlatitude Summer profile from MODTRAN, with the ground temperature shifted by -0.4K to match the average ground temperature of 293.8. Rural aerosols with default 23 km visibility are included in the calculations. All runs of the model use 2-stream multiple scattering scaled with 8-stream DISORT. Note that DISORT scaling has been extended to the infrared in the house version of MODTRAN 5.3 used here. Unless otherwise stated, all simulations include no clouds and set the sun position at the zenith.

The background image was taken with a SEBASS hyperspectral camera from 1 km altitude above CFB Valcartier in June 2004 during the “Black Fly” measurement campaign. The image was atmospherically compensated using an ELM technique using known ground targets. Temperature emissivity separation was then done with the Defilte algorithm. The danger in using a real image in such a study is that no correction is perfect. The residual spectral artifacts and noise could skew the results in favor of a given detection or correction algorithm. Since the objective here is not to compare algorithms, but to gather a feeling for the effect of various parameters, the background image will likely be more than adequate.

To match the background image, the target and atmospheric spectral quantities are resampled using the SEBASS sensor FWHM, with (unless otherwise stated) the triangular slit function and smile shift of Figure 4. The sensitivity curve is approximated as being flat and equal to 1.0 on the entire spectral region of interest.

All results are statistics extracted from 16 runs, with the random seed for noise calculation reset each time. This eliminates some of the variability that can occur because of the limited number of pixels in the background image.

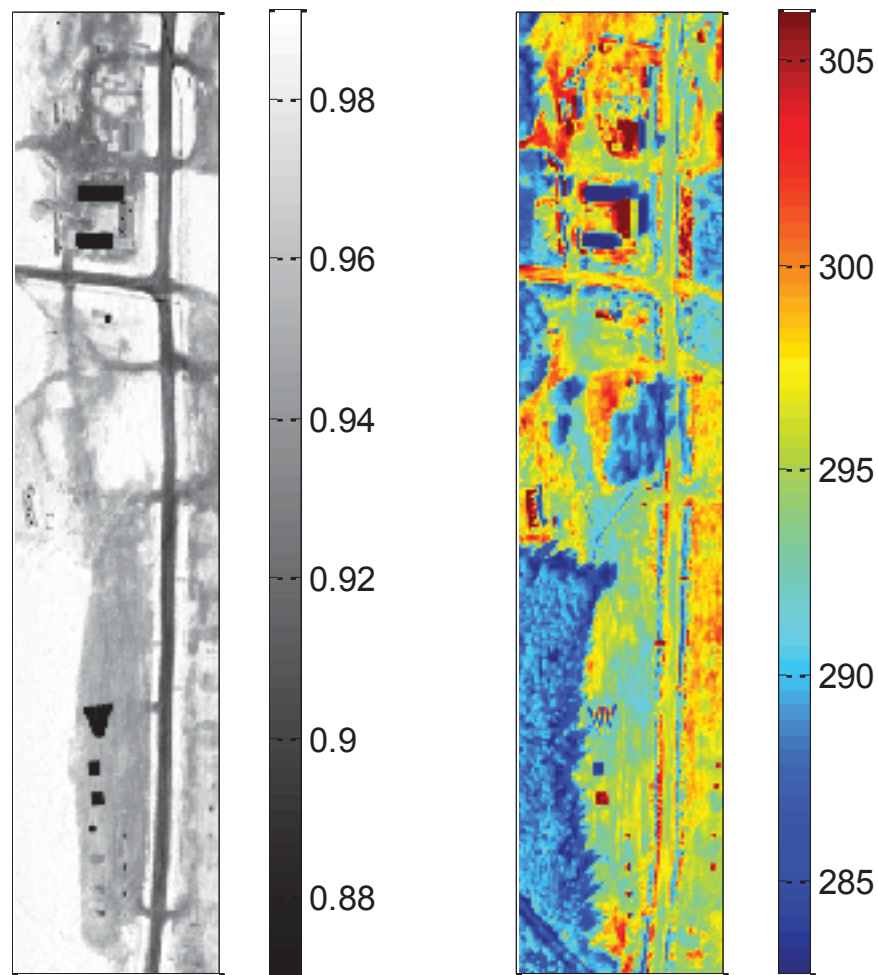


Figure 11: Average emissivity (left) and temperature (right) images used in the parameter sensitivity study.

A.1 Solid Target

The first set of results are obtained by modeling a solid target (emissive and reflective, but not transmissive) in the synthetic scene. The material composing the target is ammonium sulfate ($(\text{NH}_4)_2\text{SO}_4$). This material was chosen because it has a prominent emissivity feature at 9 microns (see Figure 12) which makes it a potential candidate for infrared detection.

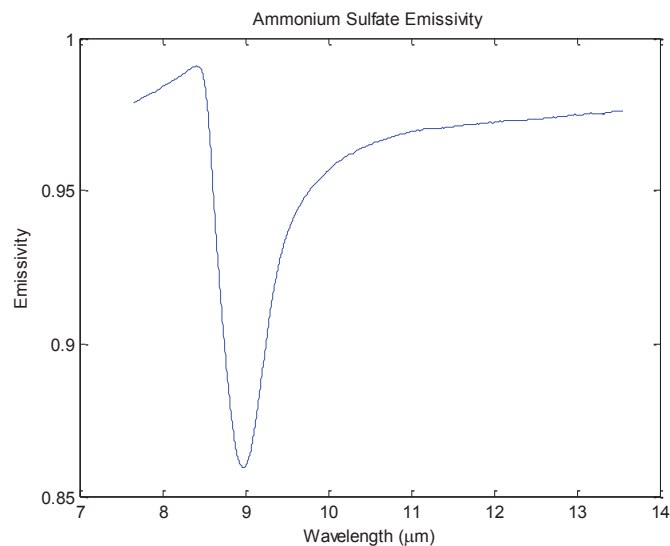


Figure 12: Emissivity spectrum of ammonium sulfate.

All solid target simulations are made using the MODTRAN 1 cm^{-1} band model since the target has no sharp spectral features, and the SEBASS sensor resolution does not exceed 2 cm^{-1} over the band of interest.

To reduce clutter in the resulting graphics, only the Signed GLRT, Signed Ace, CEM, AMF detection algorithms are used. Signed versions of the GLRT and ACE algorithms are used since they are less likely to produce false alarms for pixels that are by chance anti-correlated to the target, and therefore superior for solid targets. The Hybrid algorithm is not used simply because it requires significantly more computation time. Atmospheric compensation algorithms used are MODTRAN and None, noting that the MODTRAN correction is the exact inverse of the degradation, and should be considered the best case scenario. The choice of not correcting the image for atmospheric effects also provides an interesting baseline for the comparisons. Finally, because of speed considerations, all temperature-emissivity separation is done with the NEM algorithm with baseline emissivity set at 1.0 between 9 and 11 microns.

A.1.1 Effect of altitude

Target Temperature: 293.8 K
Pixel Fraction: 0.5
Nadir angle: 0 deg
Noise RMS: 0.01

The altitude of the sensor mostly affects the quantity of air between the sensor and the target and background. ROC curves for a sensor at 2 km are shown in Figure 13 while the CAI results are shown in Table 2. At this low altitude, all detection-correction algorithm combinations produce very good results. For comparison, ROC curves for a sensor located at 20 km altitude are shown

in Figure 14 with numerical CAI results listed in Table 3. As expected, the overall detection capabilities are diminished because of the increased effect of the atmosphere.

Of interest is the fact that at 2 km altitude, applying no atmospheric compensation is better than using the ideal MODTRAN correction. It is possible that in this case, the detection algorithms are good enough to estimate the atmospheric effects as part of their background statistics. On the other hand, the MODTRAN correction probably introduces spectral artifacts by amplifying noise in higher absorbing spectral regions, which cannot be incorporated into the background characterization as noise varies on a pixel basis. This trend is completely reversed at 20 km where the MODTRAN correction improves detection for all algorithms. It can also be noted that the MODTRAN corrected results are very similar in both the 2 and 20 km cases, but that the uncorrected versions drop significantly at 20 km. This is most obvious with the Signed ACE result that goes from best to worse. This is probably because the capacity of the detection algorithms to estimate the atmospheric effects into their background statistics becomes less efficient at 20 km.

Finally, it should be noted that in reality, changing the altitude would also probably affect the pixel fraction of the target in the pixel. This effect is not reproduced here in order to isolate the effect of the altitude only. The effect of changing the pixel fraction is studied in section A.1.5.

Table 2: CAI results for the altitude effect at 2 km in descending order.

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999996587249
CEM	None	0.999982804627
AMF	None	0.999979098948
Signed GLRT	None	0.999978688361
AMF	MODTRAN	0.999850016550
Signed GLRT	MODTRAN	0.999848486484
CEM	MODTRAN	0.999838919160
Signed ACE	MODTRAN	0.999418625778

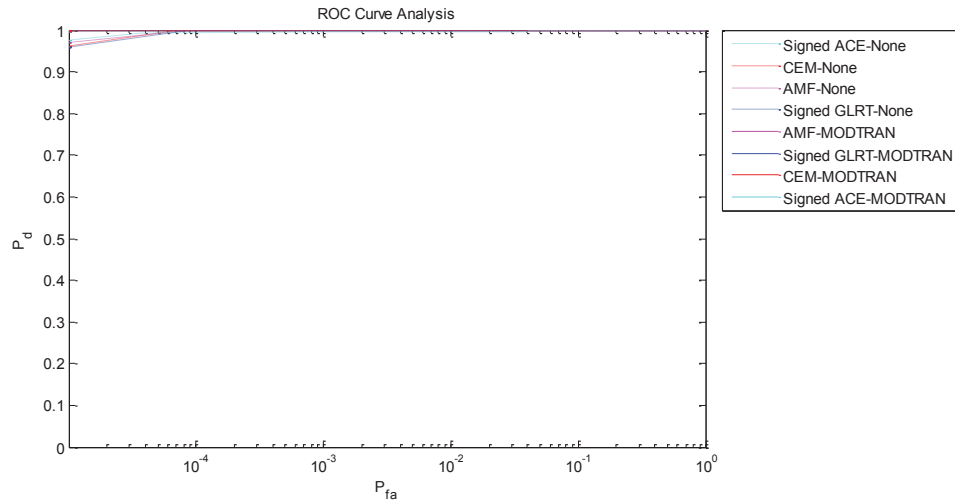


Figure 13: ROC curve analysis for the altitude effect at 2 km

Table 3: CAI results for the altitude effect at 20 km in descending order.

Detection Algorithm	Correction Algorithm	CAI
Signed GLRT	MODTRAN	0.999881629868
AMF	MODTRAN	0.999875300867
CEM	MODTRAN	0.999874610549
Signed ACE	MODTRAN	0.999459475204
CEM	None	0.998235770083
AMF	None	0.990404018134
Signed GLRT	None	0.990348899053
Signed ACE	None	0.937564277310

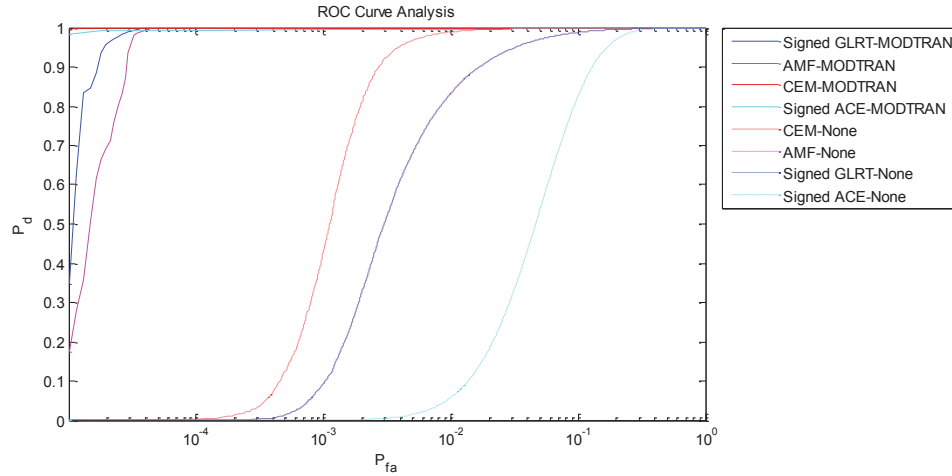


Figure 14: ROC curve analysis for the altitude effect at 20 km

A.1.2 Effect of nadir angle

Target Temperature: 293.8 K
Pixel Fraction: 0.5
Sensor Altitude: 2 km
Noise RMS: 0.01

In the current implementation of the tool, changing the nadir angle increases the quantity of air along the optical path from the sensor to the target and background. While the results at an angle of 0 degree are shown in Figure 13 and Table 2 of section A.1.1, the ROC curves and CAI results for an angle of 82 degree is shown in Figure 15 and Table 4 respectively. As expected, even though the rank of algorithms is not identical, the discussion in section A.1.1 is applicable here also.

Table 4: CAI results for the nadir angle effect at 82 degrees in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	MODTRAN	0.999915556898
CEM	MODTRAN	0.999817709814
Signed GLRT	MODTRAN	0.999810994348
AMF	MODTRAN	0.999802899577
CEM	None	0.999620000949
Signed ACE	None	0.998692368518
Signed GLRT	None	0.998063460415
AMF	None	0.998060220210

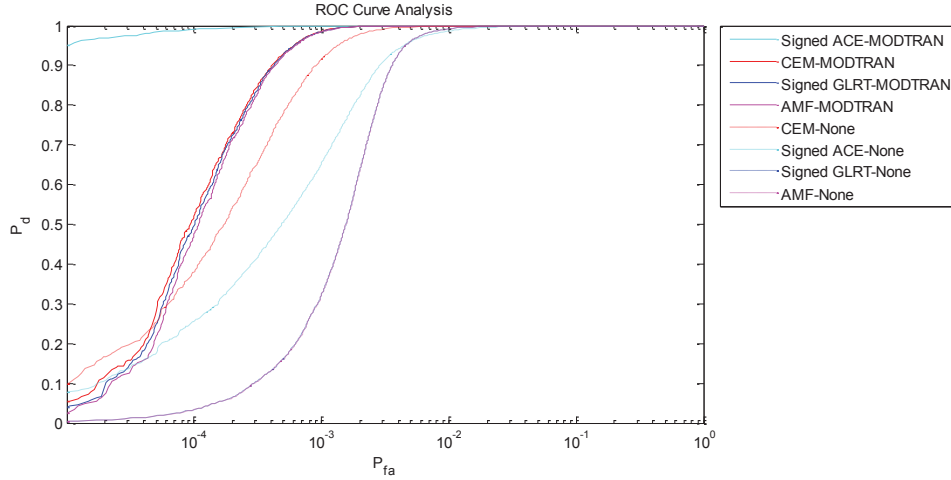


Figure 15: ROC curve analysis for the nadir angle effect at 82 degrees

A.1.3 Effect of cloud altitude/type (100% coverage)

Target Temperature: 293.8 K
Pixel Fraction: 0.5
Sensor Altitude: 1 km
Nadir angle: 0 deg
Noise RMS: 0.01

Clouds, since they are relatively opaque, emit strongly according to their temperature. Furthermore, the lower they are in the atmosphere, the warmer they will be. When clouds get low enough, they start emitting with a blackbody function very similar to the target and background, which are usually close to air temperature. When this happens, we get $L_d/\pi \approx B(T)$ in equation (1) so that it becomes

$$\begin{aligned}
 R &\cong (\varepsilon B(T) + (1 - \varepsilon) B(T)) \tau + R_p \\
 &= [(\varepsilon + 1 - \varepsilon) B(T)] \tau + R_p \\
 &= B(T) \tau + R_p
 \end{aligned} \tag{22}$$

Meaning that as the cloud temperature approaches the surface temperature, the surface gradually loses its only distinguishable intrinsic characteristic: its emissivity. Trying to correct this surface in order to retrieve the emissivity will likely only exacerbate the noise, which has then become the only spectrally distinct feature of the pixel.

This behaviour is evident when looking at the ROC curves when a low (0.66 km) cumulus cloud layer is present (Figure 1) as well as the corresponding CAI results (Table 5). MODTRAN compensated results are much worse in this case, probably because the spectral artifacts from the atmospheric compensation are even further amplified.

With a 3 km high altostratus (Figure 17 and Table 6) the detection capacity increases, as expected, and with a high cirrus cloud layer (Figure 18 Table 7), it becomes almost as good as when no clouds are present (Figure 13).

Table 5: CAI results for cloud type and altitude for cumulus at 0.66 km in descending order

Detection Algorithm	Correction Algorithm	CAI
CEM	None	0.969028471636
Signed ACE	None	0.959222219614
Signed GLRT	None	0.957521106643
AMF	None	0.957508914783
AMF	MODTRAN	0.565748521888
Signed GLRT	MODTRAN	0.540591421160
Signed ACE	MODTRAN	0.528173741006
CEM	MODTRAN	0.393031624300

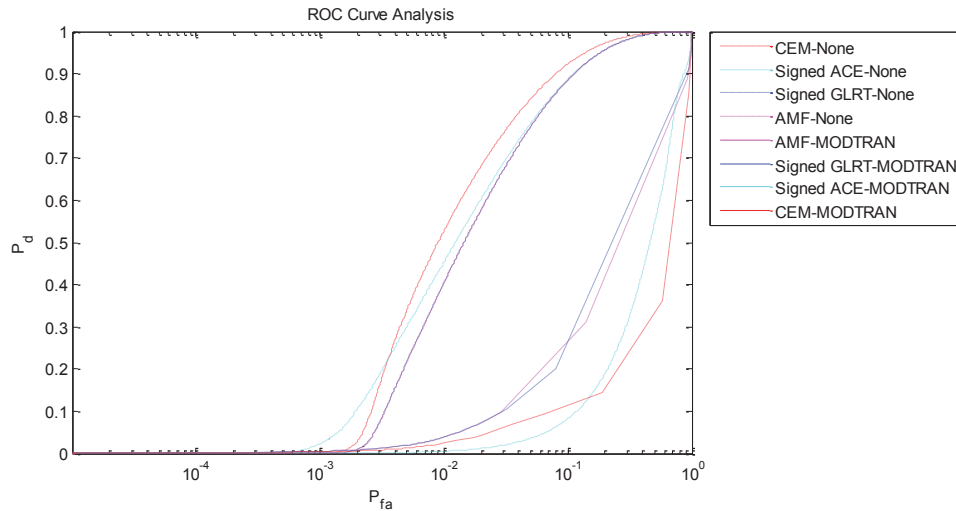


Figure 16: ROC curve analysis for cloud type and altitude for cumulus at 0.66 km.

Table 6: CAI results for cloud type and altitude for altostratus at 3 km in descending order

Detection Algorithm	Correction Algorithm	CAI
CEM	None	0.999919951273
Signed ACE	None	0.999664771707
Signed GLRT	None	0.999651821131
AMF	None	0.999650190239
AMF	MODTRAN	0.778015958759
Signed GLRT	MODTRAN	0.775346440357
Signed ACE	MODTRAN	0.760729361756
CEM	MODTRAN	0.697733469074

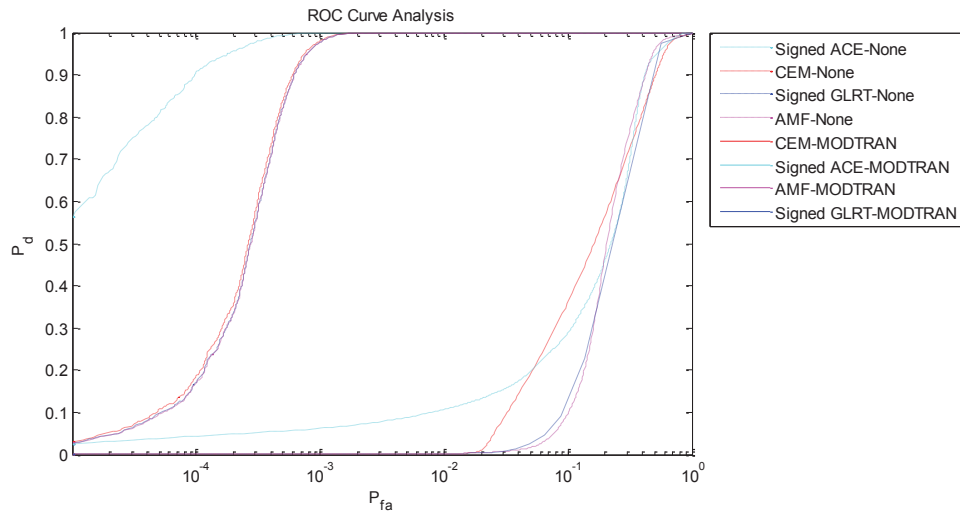


Figure 17: ROC curve analysis for cloud type and altitude for altostratus at 3 km.

Table 7: CAI results for cloud type and altitude for standard cirrus in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999996877171
CEM	None	0.999979510348
AMF	None	0.999979298917
Signed GLRT	None	0.999979094509
AMF	MODTRAN	0.999955001966
Signed GLRT	MODTRAN	0.999953840782
CEM	MODTRAN	0.999953163902
Signed ACE	MODTRAN	0.999739304664

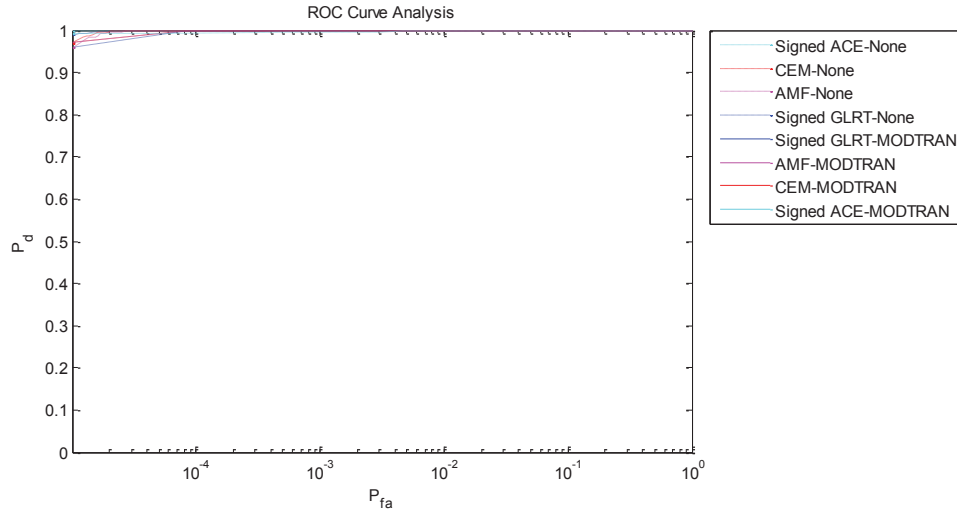


Figure 18: ROC curve analysis for cloud type and altitude for standard cirrus.

A.1.4 Effect of cloud coverage

Target Temperature: 293.8 K
Pixel Fraction: 0.5
Sensor Altitude: 1 km
Nadir angle: 0 deg
Noise RMS: 0.01

In the current implementation, cloud coverage affects downward flux by averaging 100% and 0% coverage calculations according to the coverage fraction. It is therefore expected that the effect of cloud coverage will gradually bring the ROC curves from their aspect at 0% (Figure 13 and Table 2 of section A.1.1) to that of 100% coverage (Figure 16, Table 5 for cumulus clouds for example). This is in fact what can be observed by looking at the results for a cumulus layer at 0.66 km altitude and with 75% coverage (Figure 19, Table 8), 50% (Figure 20, Table 9) and 25% (Figure 21, Table 10). As with the previous section, MODTRAN compensated images produce markedly worse results when clouds are present, even at a low 25% coverage.

Table 8: CAI results for cloud coverage effect at 75% in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999847259516
CEM	None	0.999725099204
Signed GLRT	None	0.999434033223
AMF	None	0.999430540610
Signed ACE	MODTRAN	0.872002902467
AMF	MODTRAN	0.843798880892
Signed GLRT	MODTRAN	0.835381758514
CEM	MODTRAN	0.798300703102

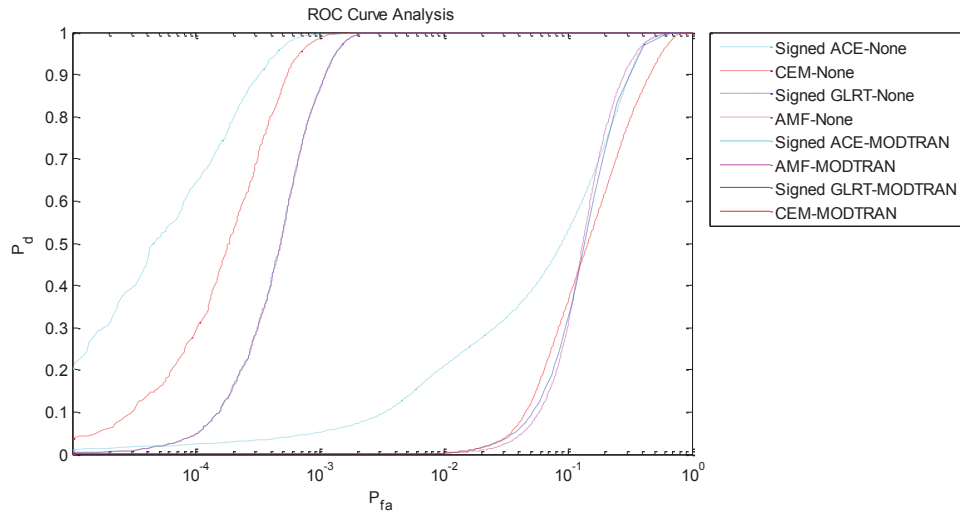


Figure 19: ROC curve analysis for cloud coverage effect at 75%.

Table 9: CAI results for cloud coverage effect at 50% in descending order.

Detection Algorithm	Correction Algorithm	CAI
CEM	None	0.999997757939
Signed GLRT	None	0.999996040739
AMF	None	0.999996009542
Signed ACE	None	0.999987147989
Signed ACE	MODTRAN	0.999345303625
CEM	MODTRAN	0.999061919339
Signed GLRT	MODTRAN	0.999048124146
AMF	MODTRAN	0.998999947917

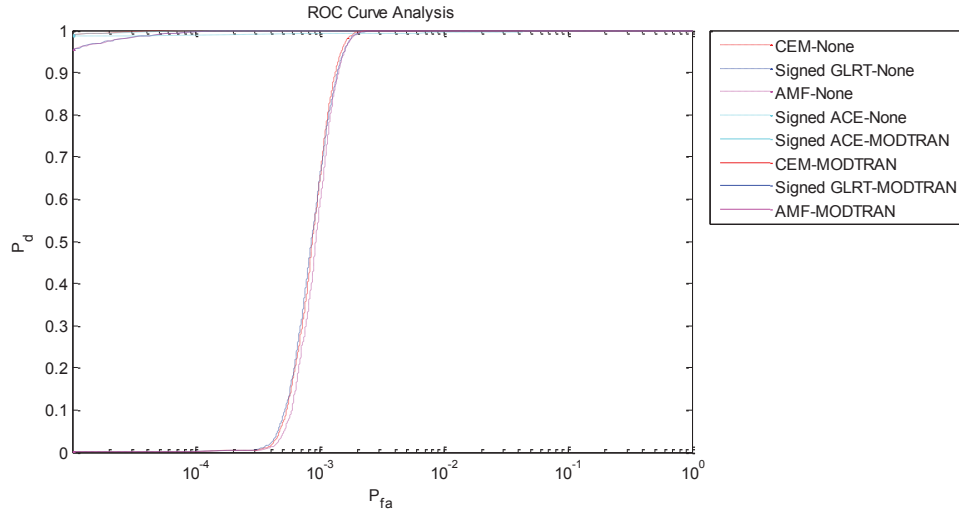


Figure 20: ROC curve analysis for cloud coverage effect at 50%.

Table 10: CAI results for cloud coverage effect at 25% in descending order

Detection Algorithm	Correction Algorithm	CAI
AMF	None	0.999999480783
Signed GLRT	None	0.999999480126
Signed ACE	None	0.999995279378
CEM	None	0.999972907441
Signed GLRT	MODTRAN	0.999801527040
CEM	MODTRAN	0.999789095252
AMF	MODTRAN	0.999788222859
Signed ACE	MODTRAN	0.999410886482

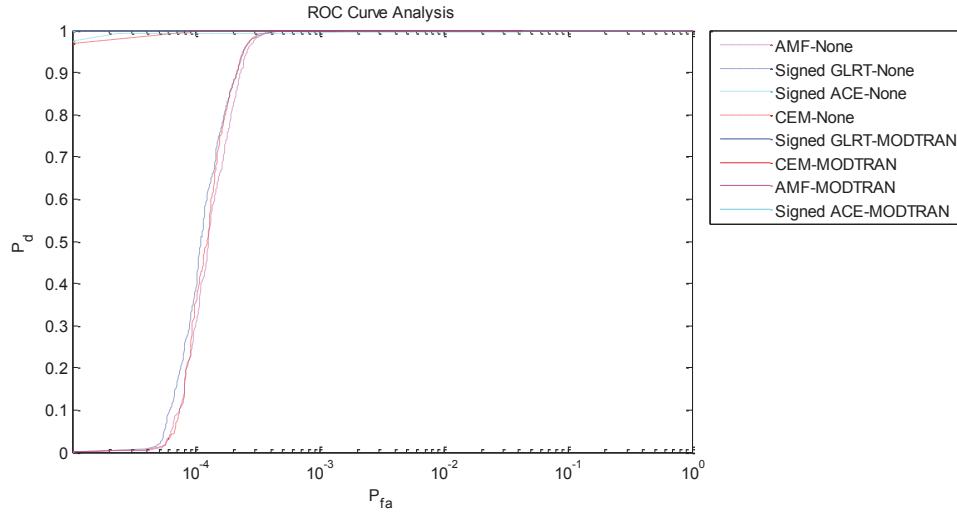


Figure 21: ROC curve analysis for cloud coverage effect at 25%.

A.1.5 Effect of pixel fraction

Target Temperature: 293.8 K
 Sensor Altitude: 2 km
 Nadir angle: 0 deg
 Noise RMS: 0.01

The pixel fraction is the amount of a pixel covered by the target. As the pixel fraction decreases, the target signal also decreases and can become more difficult to detect. The noise level will obviously have an effect on how quickly the target becomes undetectable (i.e. when the target signal within the pixel becomes comparable to the noise level in that pixel). Here, noise is set at 0.01. Compared to a pixel fraction of 0.5 (Figure 13 and Table 2 of section A.1.1), it is clear that detection becomes less efficient at 0.25 (Figure 22, Table 11) and even less so at 0.1 (Figure 23, Table 12).

Interestingly, the MODTRAN compensation becomes more of an asset as the pixel fraction diminishes. This effect is similar to that observed as the sensor altitude increases (with constant pixel fraction). Since in reality, pixel fraction is likely to decrease as altitude increases, this would compound the effect and make the need for proper atmospheric compensation even more important.

Table 11: CAI results for pixel fraction at 0.25 in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999633812000
CEM	None	0.999338413731
Signed GLRT	None	0.998814312542
AMF	None	0.998807863323
CEM	MODTRAN	0.998416420194
Signed ACE	MODTRAN	0.998391178170
Signed GLRT	MODTRAN	0.998322433658
AMF	MODTRAN	0.998214133034

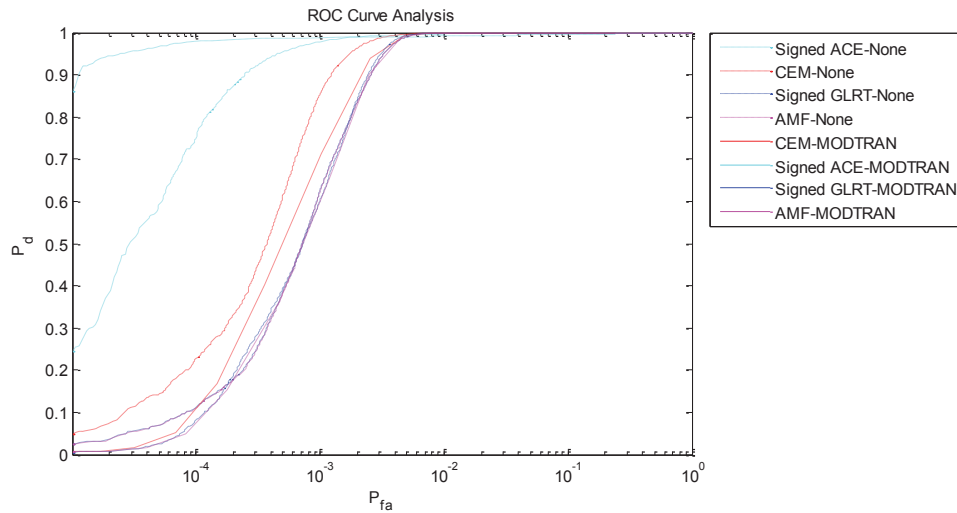


Figure 22: ROC curve analysis for pixel fraction at 0.25.

Table 12: CAI results for pixel fraction at 0.1 in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	MODTRAN	0.977294390657
CEM	MODTRAN	0.972062539371
Signed GLRT	MODTRAN	0.971007990390
AMF	MODTRAN	0.970903630939
CEM	None	0.960574009846
Signed ACE	None	0.957844956650
Signed GLRT	None	0.952255663208
AMF	None	0.952240092696

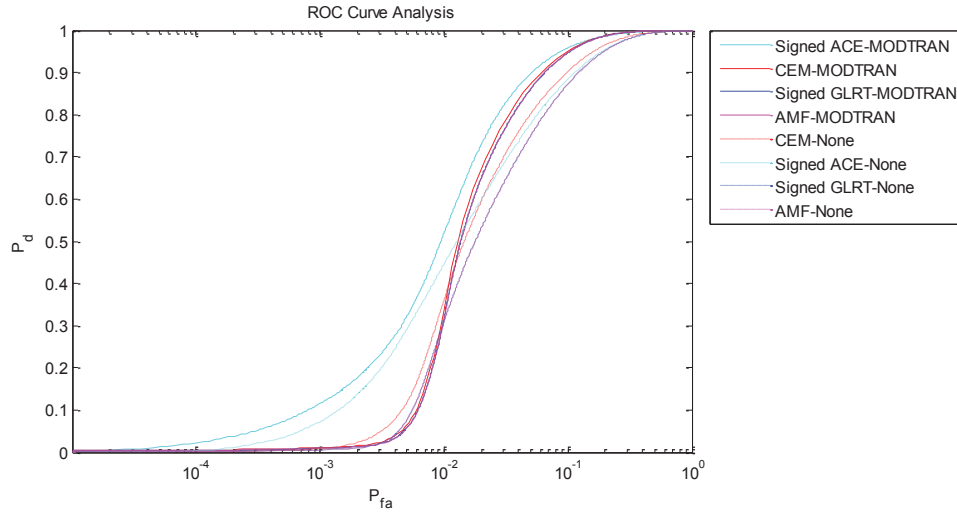


Figure 23: ROC curve analysis for pixel fraction at 0.1.

A.1.6 Effect of target-air temperature difference

Pixel Fraction: 0.25
 Sensor Altitude: 2 km
 Nadir angle: 0 deg
 Noise RMS: 0.01

For solid targets a lower surface temperature implies a weaker emission from the surface. Because of this, it can be expected that the behaviour of detection algorithm will follow the same pattern as when lowering pixel fraction. When compared to results where the surface is at the same temperature as the air and as the average background (Figure 22, Table 11), a cooler surface is more difficult to detect (Figure 24, Table 13) while a warmer surface is easier to detect (Figure 25, Table 14). As with pixel fraction, Table 13 shows that the importance of applying the MODTRAN atmospheric compensation prior to detection also becomes greater when the target temperature becomes cooler than the surrounding background and air.

Table 13: CAI results for target-air temperature difference at -10°C in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	MODTRAN	0.997518808939
Signed ACE	None	0.997417255202
CEM	MODTRAN	0.996443119448
CEM	None	0.996268951011
Signed GLRT	MODTRAN	0.996082262979
AMF	MODTRAN	0.996050722157
Signed GLRT	None	0.994936963092
AMF	None	0.994923899051

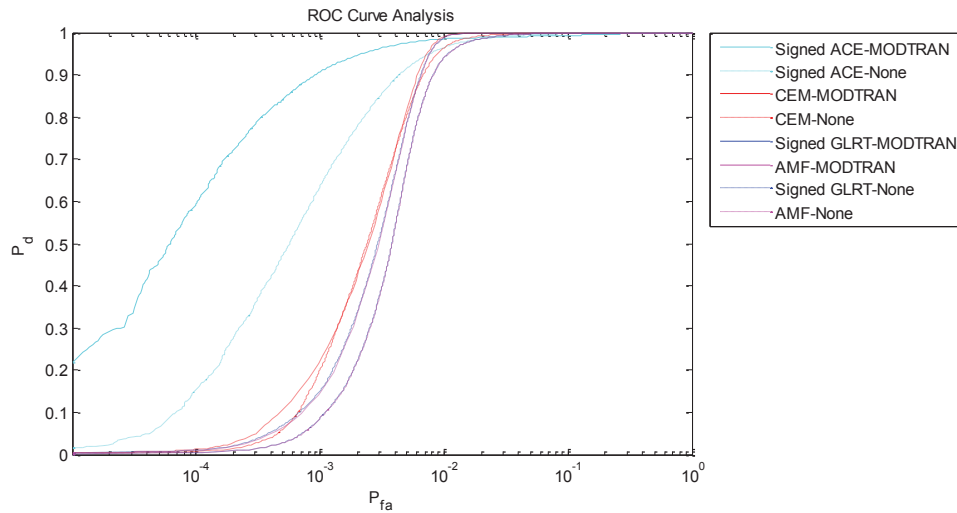


Figure 24: ROC curve analysis for target-air temperature difference at -10°C .

Table 14: CAI results for target-air temperature difference at $+10^{\circ}\text{C}$ in descending order

Detection Algorithm	Correction Algorithm	CAI
CEM	None	0.999924985594
Signed ACE	None	0.999906179067
Signed GLRT	None	0.999877838601
AMF	None	0.999876742175
CEM	MODTRAN	0.999525605117
Signed GLRT	MODTRAN	0.999491942062
AMF	MODTRAN	0.999476903102
Signed ACE	MODTRAN	0.999200998227

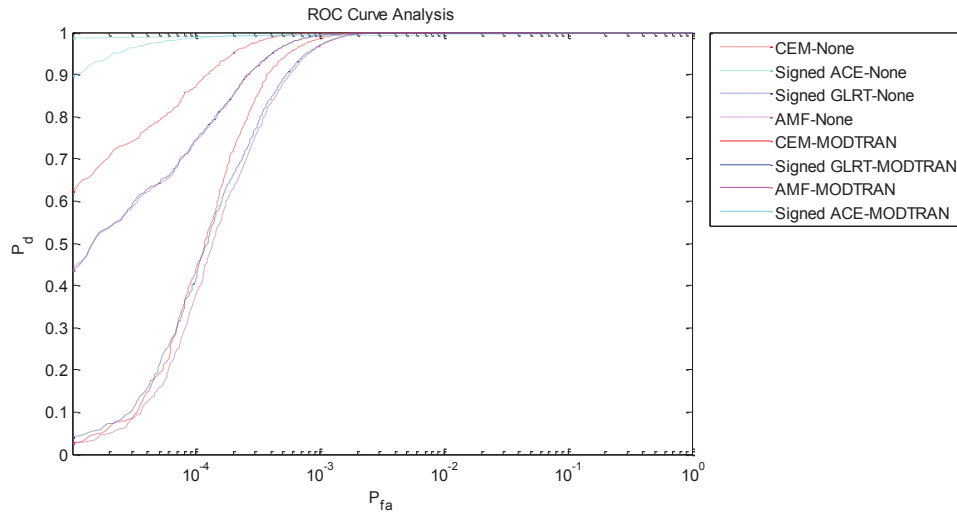


Figure 25: ROC curve analysis for target-air temperature difference at +10 °C.

A.1.7 Effect of sunlight

Target Temperature: 293.8 K
Pixel Fraction: 0.25
Sensor Altitude: 2 km
Nadir angle: 0 deg
Noise RMS: 0.01

In the thermal infrared bands, the sun adds very little signal to the overall scene. It is thus expected that the presence of the sun, moon, or none of these will have very little impact on the detection results. This is obvious when comparing results for a sunlit scene (Figure 22, Table 11 in section A.1.5) with those for a moonless night (Figure 26, Table 15). Both simulations produce practically the same results within expected variations of the CAI from one run to the other.

Note that here, the only difference between the simulations occurs in the downwelling flux calculation. In reality, the presence or not of the sun would also affect the temperature of the surface as well as warming rates compared to the surrounding air and background. In that regard, the diurnal cycle will probably have an impact closer to what is discussed in section A.1.6.

Table 15: CAI results for the effect of having no sunlight in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999665739059
CEM	None	0.999334193598
Signed GLRT	None	0.999021539752
AMF	None	0.999015944430
Signed ACE	MODTRAN	0.998850200886
CEM	MODTRAN	0.998843114787
Signed GLRT	MODTRAN	0.998624717248
AMF	MODTRAN	0.998599365161

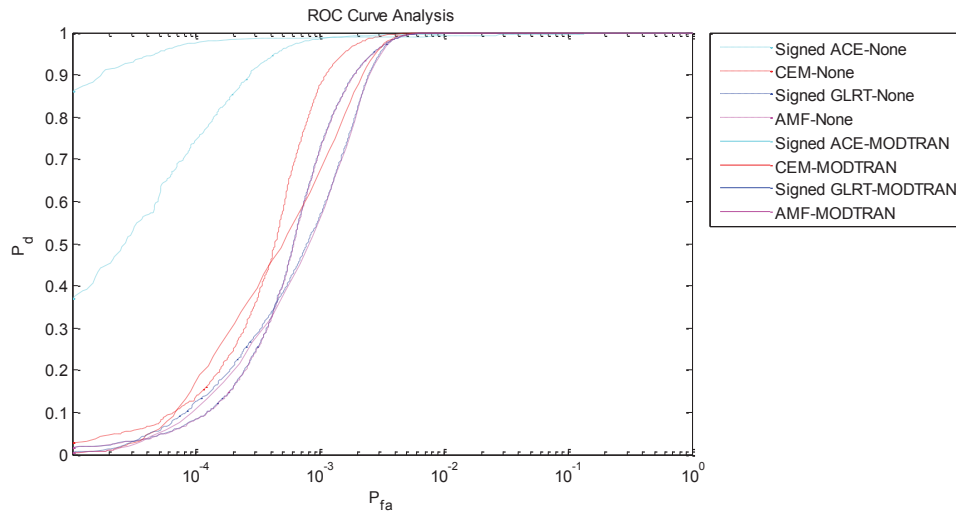


Figure 26: ROC curve analysis for the effect of having no sunlight.

A.1.8 Noise level

Target Temperature: 293.8 K
Pixel Fraction: 0.25
Sensor Altitude: 2 km
Nadir angle: 0 deg

A low target signal to noise ratio is usually considered the principal cause of detection failure. In many cases studied up to this point (high sensor altitude, low pixel fraction, cold target), it is the target signal that is decreased making detection more difficult. Another way signal to noise ratio can be increased is by increasing the noise itself. Here, sensor noise RMS is measured in radiance units of $W/m^2/sr/\mu m$ since we use a flat unit sensor sensitivity curve. Starting with the baseline of 0.01 RMS (Figure 22 and Table 11 in section A.1.5), it is obvious that, as expected, detection becomes more difficult at 0.05 RMS (Figure 27, Table 16) and even more so at 0.1 RMS (Figure

28, Table 17). As with the other cases studied previously, decreasing the signal to noise ratio gradually increases the need for prior atmospheric compensation.

Table 16: CAI results for the effect of sensor noise at 0.05 RMS in descending order

Detection Algorithm	Correction Algorithm	CAI
CEM	None	0.999573217000
Signed ACE	MODTRAN	0.999051766554
Signed ACE	None	0.999035448625
CEM	MODTRAN	0.999031928738
Signed GLRT	MODTRAN	0.999019370182
AMF	MODTRAN	0.998999872528
Signed GLRT	None	0.998375658472
AMF	None	0.998371379891

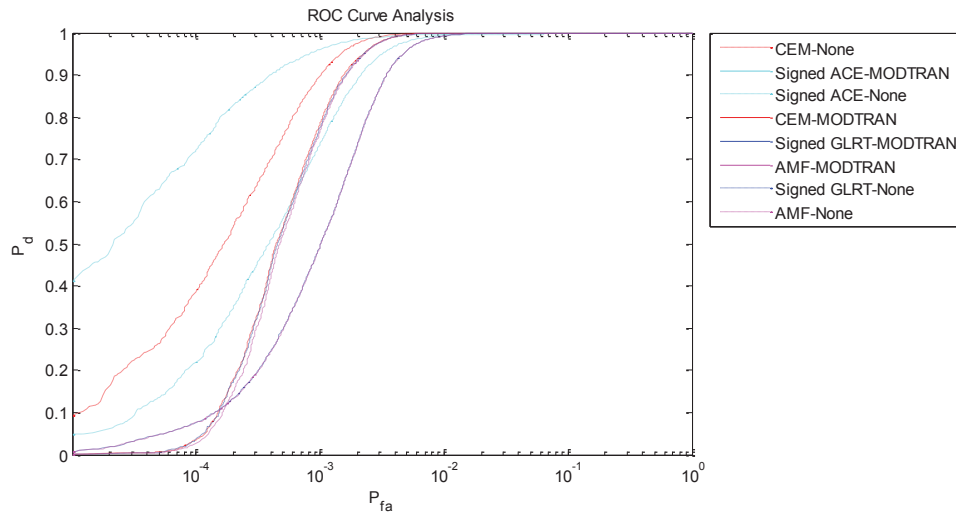


Figure 27: ROC curve analysis for the effect of sensor noise at 0.05 RMS.

Table 17: CAI results for the effect of sensor noise at 0.1 RMS in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	MODTRAN	0.989524516531
Signed GLRT	MODTRAN	0.989511544598
AMF	MODTRAN	0.989477117569
CEM	MODTRAN	0.989389338193
CEM	None	0.987002110424
Signed ACE	None	0.980894479952
Signed GLRT	None	0.979327972062
AMF	None	0.979323126044

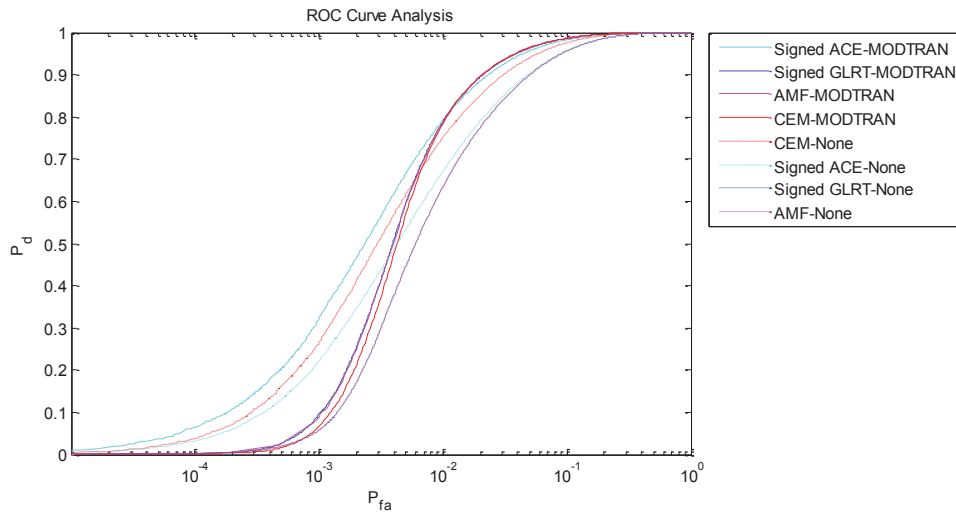


Figure 28: ROC curve analysis for the effect of sensor noise at 0.1 RMS.

A.1.9 Smile effect

Target Temperature: 293.8 K
Pixel Fraction: 0.25
Sensor Altitude: 2 km
Nadir angle: 0 deg
Noise RMS: 0.01

The smile effect is a spectral distortion that shifts spectral features into one direction or another along the spectral axis of the detector. If the shift is large enough, important spectral features can be sent into a different spectral bin making detection based on these features more difficult. The spectral signature of most solids varies relatively smoothly along the spectral axis, with very few sharp features. Since the smile effect is usually of much lower amplitude than the average spectral feature, it is not expected that smile will have a very large impact on detection. Another way the smile effect can affect the detection is by shifting the atmospheric spectral features, making atmospheric compensation more difficult.

When compared to detection in the presence of the SEBASS smile effect (Figure 22 and Table 11 in section A.1.5), Figure 29 and Table 18 show that the smile distortion has very little effect on the detection capabilities.

Table 18: CAI results for the effect neglecting the smile effect in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999658649483
CEM	None	0.999316804903
Signed GLRT	None	0.999005215629
AMF	None	0.998999429322
CEM	MODTRAN	0.998600047393
Signed ACE	MODTRAN	0.998499746590
Signed GLRT	MODTRAN	0.998369072209
AMF	MODTRAN	0.998343768589

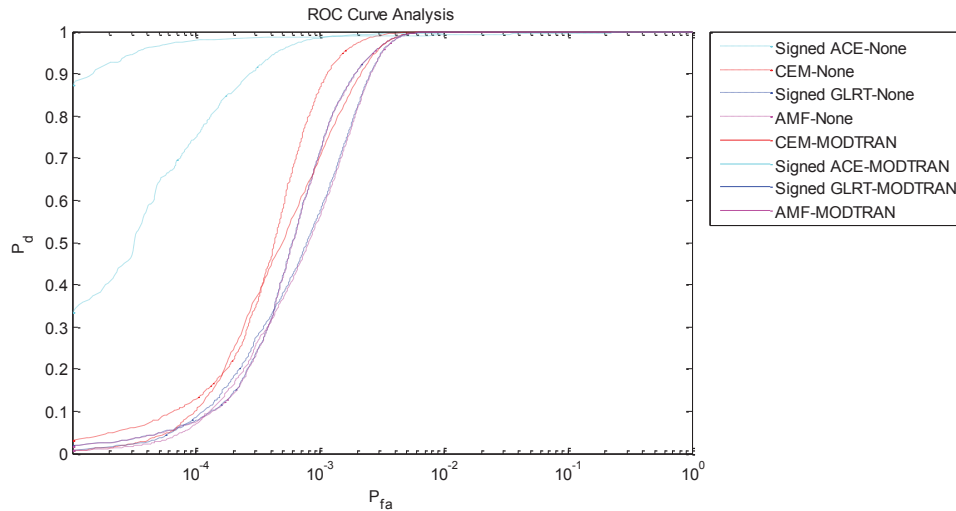


Figure 29: ROC curve analysis for the effect neglecting the smile effect.

A.1.10 Effect of slit function shape

Target Temperature: 293.8 K
Pixel Fraction: 0.25
Sensor Altitude: 2 km
Nadir angle: 0 deg
Noise RMS: 0.01

The atmospheric propagation and targets are not usually modeled at the sensor resolution. Although solid targets do not usually have sharp spectral features making the modelling resolution less of an issue, the atmosphere does. In fact, to properly model the atmospheric propagation effects as measured by the sensor, the atmosphere model should have at least twice the resolution of the sensor so that it can then be resampled properly down to the sensor resolution, while taking into account the Nyquist resampling theorem. In order to model the sensor measurements as closely as possible, the resampling window should match the sensor slit function shape. The tool only incorporates basic slit function shapes and cannot model the exact sensor slit function. It is therefore important to verify the sensitivity of the results to slit function shape.

The simulation was already done for the triangular slit shape (Figure 22 and Table 11 in section A.1.5), and is done for the square slit (Figure 30, Table 19), the Gaussian slit (Figure 31, Table 20) and the Lorentzian slit (Figure 32, Table 21). Results are practically identical for all slit shapes allowing for small rank changes, where CAI results are extremely close. These rank changes are mostly due to random variability in the Monte-Carlo simulation rather than true differences. The only results that differ more than randomness would allow are those for the square slit function, probably due to the fact that the slit function is more compact than the others (the slit width never goes beyond the FWHM). Even in that case, the first two CAI ranks are identical to the other simulations, so in practice algorithm selection would not have been affected.

Note that these simulations have been made with the MODTRAN 1.0 cm^{-1} resolution. Since resolution might be important in a slit shape analysis, these tests were also made with the 0.1 cm^{-1} resolution (not shown here), and were found to be very similar with identical conclusions.

Table 19: CAI results for the square slit function in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999580524298
CEM	None	0.999179499527
CEM	MODTRAN	0.999035733885
Signed GLRT	MODTRAN	0.998890427152
AMF	MODTRAN	0.998874433191
Signed ACE	MODTRAN	0.998793544764
Signed GLRT	None	0.998708668778
AMF	None	0.998701616827

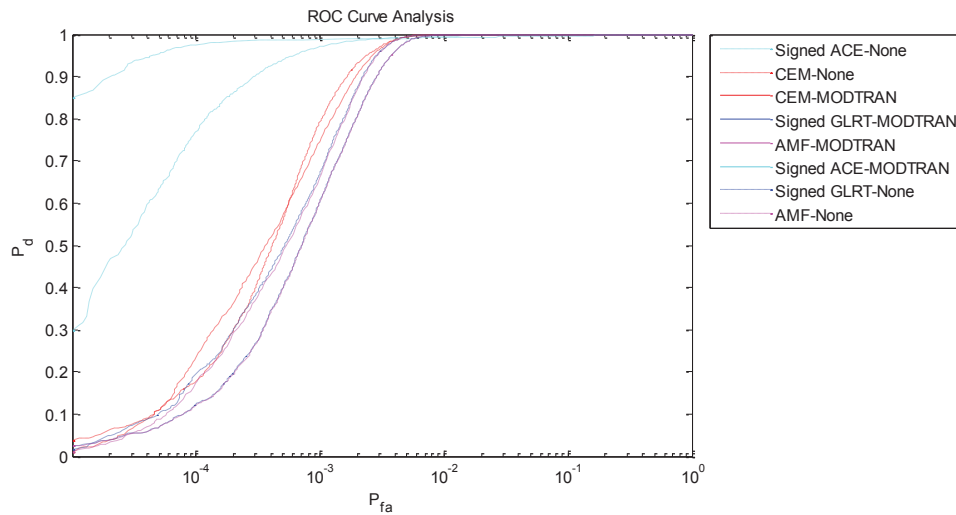


Figure 30: ROC curve analysis for the square slit function.

Table 20: CAI results for the Gaussian slit function in descending order.

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999660743574
CEM	None	0.999391742872
Signed GLRT	None	0.998912085725
AMF	None	0.998906256915
Signed ACE	MODTRAN	0.998418461335
CEM	MODTRAN	0.998254595181
Signed GLRT	MODTRAN	0.998057604020
AMF	MODTRAN	0.998030521744

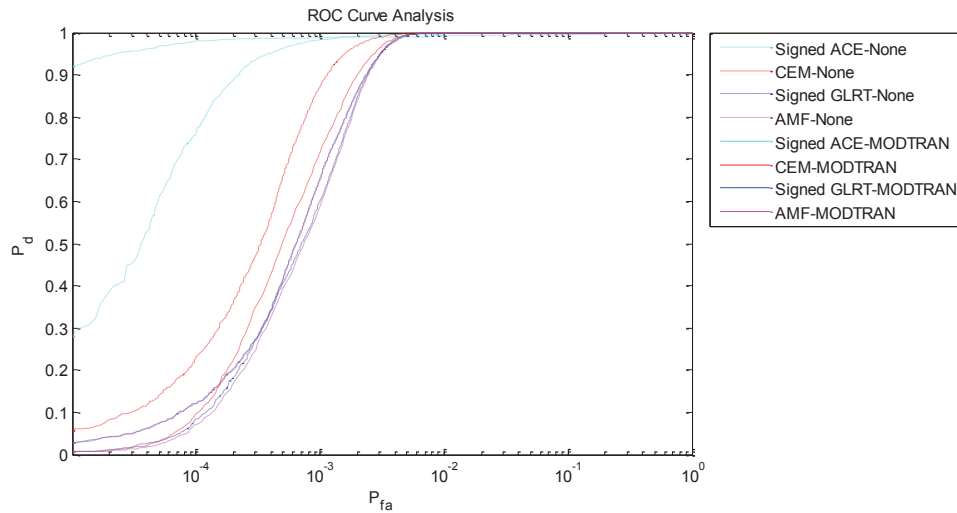


Figure 31: ROC curve analysis for the Gaussian slit function.

Table 21: CAI results for the Lorentzian slit function in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	None	0.999676130291
CEM	None	0.999376670187
Signed GLRT	None	0.998892075996
AMF	None	0.998886141162
Signed ACE	MODTRAN	0.998809549728
CEM	MODTRAN	0.998686944297
Signed GLRT	MODTRAN	0.998415382937
AMF	MODTRAN	0.998390296343

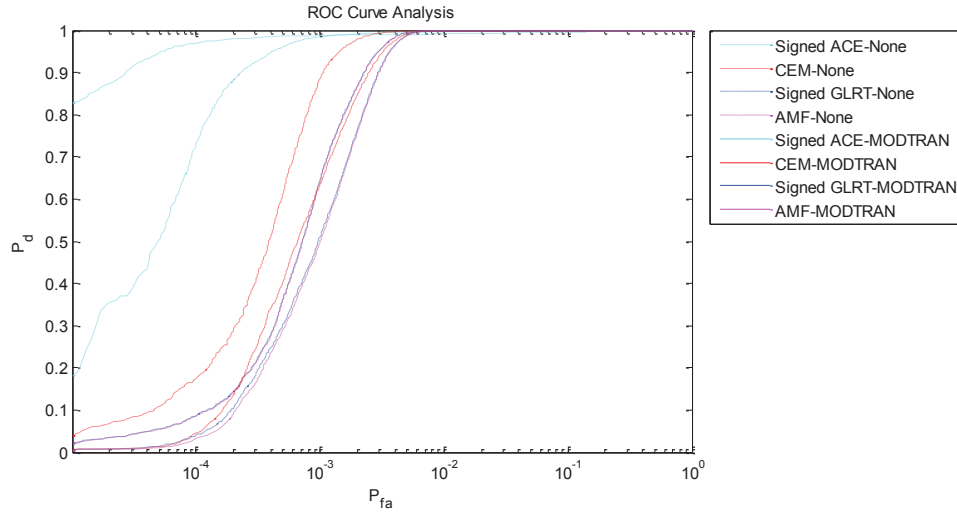


Figure 32: ROC curve analysis for the Lorentzian slit function.

A.1.11 Effect of using a constant background temperature

Target Temperature: 293.8 K
Pixel Fraction: 0.25
Sensor Altitude: 2 km
Nadir angle: 0 deg
Noise RMS: 0.01

Ideally, a temperature image should always accompany the emissivity image for proper simulation. This image might not always be available. Here we test the effect of using a uniform temperature equal to the air temperature (also the target and average background temperature). The results are shown in Figure 33 and Table 22. When compared to the simulation using the full temperature image (Figure 22 and Table 11 in section A.1.5), it becomes apparent that the MODTRAN compensated results are almost identical, while the uncompensated results fall back significantly. As mentioned in section A.1.1, the uncompensated results can sometimes be better since atmospheric effects can be estimated from the background statistics used by the detection algorithms. In this case, it is obvious that the uniformity of the background makes this statistical estimation much more difficult.

Table 22: CAI results for a constant temperature background in descending order

Detection Algorithm	Correction Algorithm	CAI
Signed ACE	MODTRAN	0.999711430960
CEM	MODTRAN	0.999399912436
Signed GLRT	MODTRAN	0.999293156015
AMF	MODTRAN	0.999275310392
CEM	None	0.997600736319
Signed GLRT	None	0.995452408514
AMF	None	0.995444172054
Signed ACE	None	0.995338274662

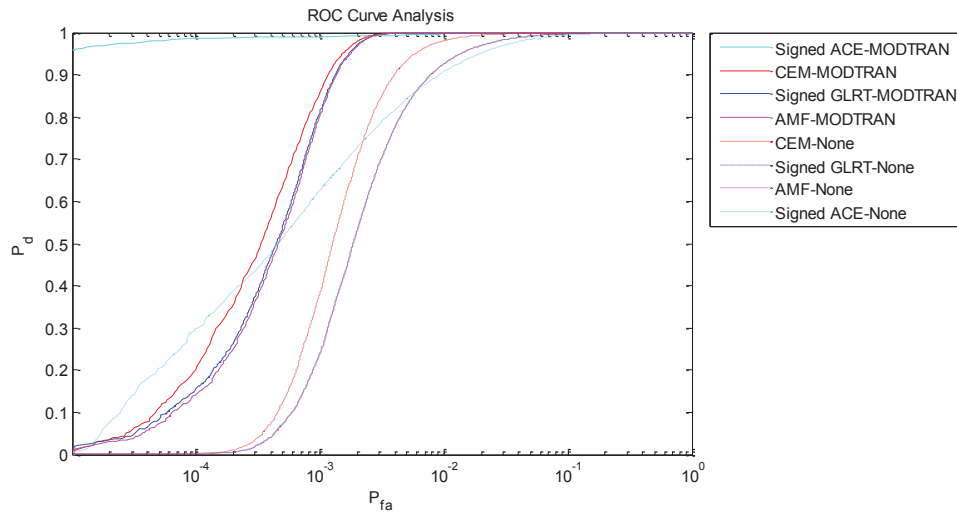


Figure 33: ROC curve analysis for a constant temperature background.

A.1.12 Conclusion

From the analysis on solid targets shown above, it is evident that very few general trends can be extracted. In fact, when varying most parameters of impact the rank of detection-compensation algorithm combinations are seldom repeated. This means that choosing the most appropriate algorithm combination should always be done on a case by case basis, with very few rules of thumb to rely upon. This demonstrates quite nicely the usefulness of a tool such as the one developed here.

One interesting trend was noticed though: lower target signal to noise ratios either from higher sensor altitude, lower pixel fraction, colder target or simply increased noise increases the need for accurate atmospheric compensation. On the other hand, in high target signal to noise ratio cases, the atmospheric compensation, even when considered very good (in this case, the best possible as it is an almost exact inversion of the degradation) can actually hinder detection.

A.2 Gas plumes

The second set of results is obtained using gas plumes as targets instead of solids. In the case of gas plumes, targets are non-reflective, but are emissive and transmissive. Because of this, depending on the temperature of the gas plume, the effect on the background can either be addition of emission or absorption features. Also, gases have the tendency to have sharper spectral features when compared to solids.

Overall, many of the behaviours noted in section A.1 will likely still be valid. This section will concentrate mostly on the behaviours that might be affected by the differences between solid and gaseous targets noted above.

The gas selected for the analysis is ammonia (NH_3) because it features a series of interesting spectral lines in the band of interest (Figure 34). The plume is modeled as a 1 m thick uniform slab at ground level with a 200 ppm concentration. In all simulations, fixed parameters are identical to the solid gas simulations, except for the MODTRAN resolution, which is set at 0.1 cm^{-1} in order to model the propagation of the sharp spectral features of the gas through the atmosphere. Other parameters are as follow:

Sensor Altitude: 2 km
Nadir angle: 0 deg
Noise RMS: 0.01

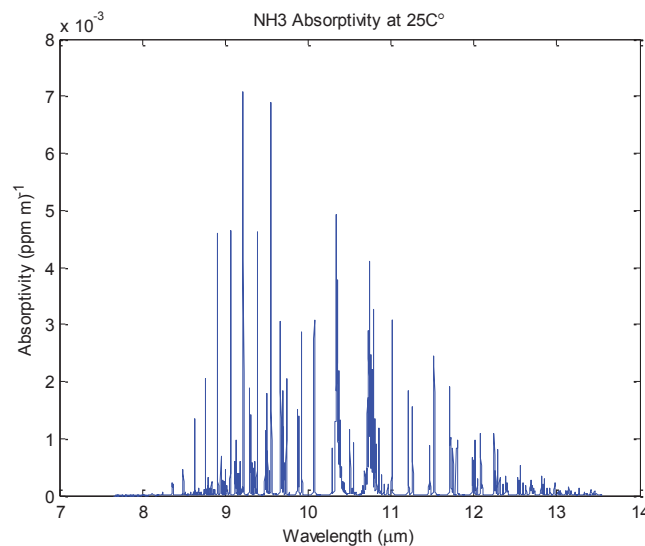


Figure 34: Absorbtivity of NH_3 (ammonia) gas at 25 °C.

The correction algorithms used are the same as for the solid target, while the detection algorithms are the signed and unsigned versions of the GLRT and ACE detectors. This should let us verify the hypothesis that the unsigned versions are superior since they can detect bot the emitting and absorbing cases.

A.2.1 Effect of plume temperature

In principle, a gas plume that is colder than the apparent temperature of the background pixel will tend to show absorption features, while a gas that is warmer will tend to show emission features. Both of these characteristics can lead to detection of the gas plume. It is also expected that a gas plume at the exact same temperature than its surrounding will be more difficult to detect. To verify this, a NH_3 plume 5°C colder than the average background (average background is 293.8 K) is used as a target for the results in Figure 35 and Table 23. Compared to this, results are also shown for a gas plume at the same temperature (Figure 36, Table 24) and 5°C warmer (Figure 37, Table 25) than the air and average background.

As expected, the warmer and cooler plumes are more easily detected than the plume at air temperature. The plume at air temperature is still detected because few background pixels are actually exactly at the same temperature as the plume.

These results also confirm the fact that the unsigned versions of the ACE and GLRT are usually better at detecting gases, especially when the temperature of the plume is cooler than the background pixel. The worst result occurs for the case where the plume is 5 degrees cooler. This is because the plume is cooler than most background pixels. Since the signed versions of the detection algorithms can only detect emission and not absorption, the plume is detected against only the coolest pixels in the scene. On the other hand, as the gas becomes warmer, the signed algorithms become more efficient, but never as efficient as the unsigned versions.

Table 23: CAI results for a gas with temperature 5°C cooler than the air in descending order

Detection Algorithm	Correction Algorithm	CAI
GLRT	None	0.946017863134
GLRT	MODTRAN	0.944834667672
ACE	None	0.941383261272
ACE	MODTRAN	0.932848919906
Signed ACE	MODTRAN	0.189707831669
Signed GLRT	MODTRAN	0.188388271474
Signed ACE	None	0.174266249947
Signed GLRT	None	0.173396582504

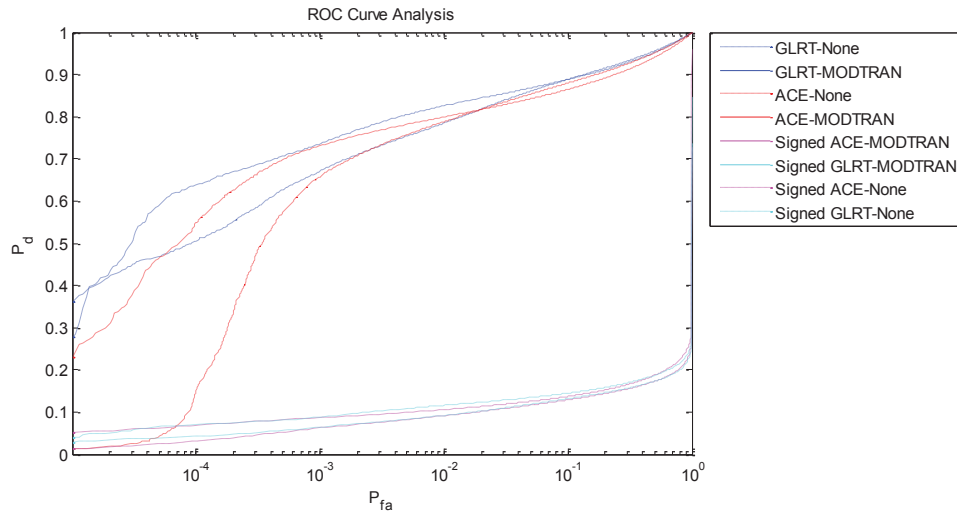


Figure 35: ROC curve analysis for a gas with temperature 5°C cooler than the air.

Table 24: CAI results for a gas at air temperature in descending order

Detection Algorithm	Correction Algorithm	CAI
ACE	None	0.915617566862
GLRT	None	0.913361020993
ACE	MODTRAN	0.911496196272
GLRT	MODTRAN	0.909152061046
Signed ACE	MODTRAN	0.528488227599
Signed GLRT	MODTRAN	0.528324793780
Signed GLRT	None	0.508416488480
Signed ACE	None	0.508395975063

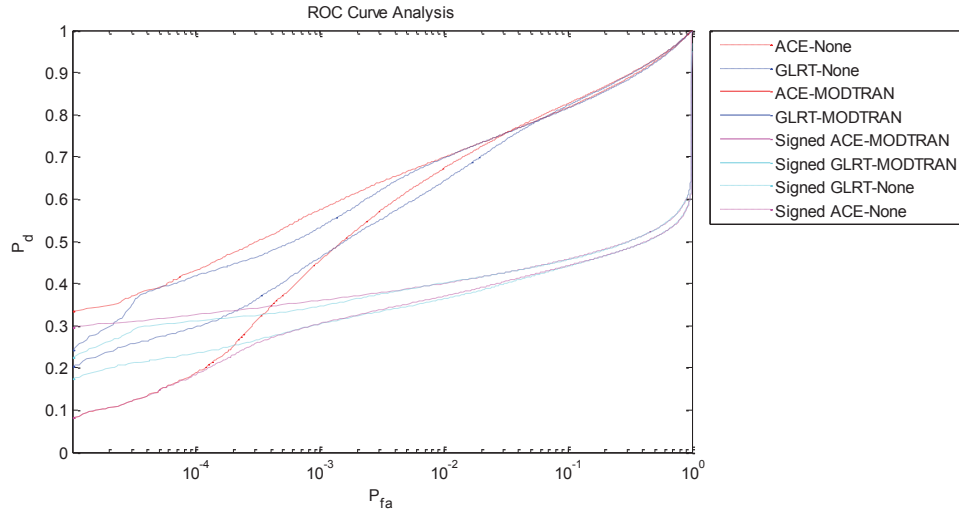


Figure 36: ROC curve analysis for a gas at air temperature.

Table 25: CAI results for a gas with temperature 5°C warmer than the air in descending order

Detection Algorithm	Correction Algorithm	CAI
ACE	MODTRAN	0.959741776568
ACE	None	0.956521829793
GLRT	MODTRAN	0.955198314380
GLRT	None	0.954944032848
Signed ACE	MODTRAN	0.916846078739
Signed ACE	None	0.915798749136
Signed GLRT	MODTRAN	0.91555320125
Signed GLRT	None	0.914696197673

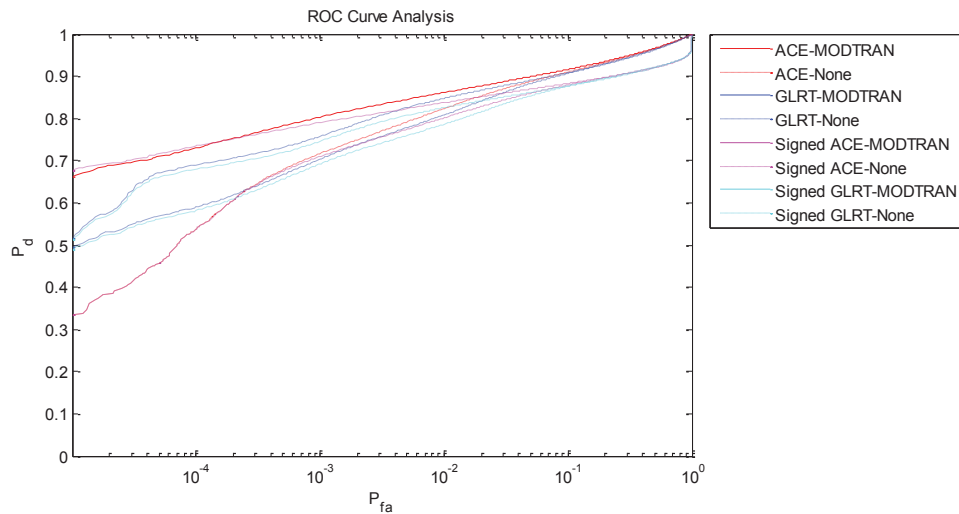


Figure 37: ROC curve analysis for a gas with temperature 5°C warmer than the air.

A.2.2 Smile

For solid targets, the smile distortion had very little impact on the capacity of detection mainly because of the lack of sharp spectral features. Since gases often have much sharper structures, it would be legitimate to wonder if the smile distortion has a larger impact when detecting plumes. Figure 38 and Table 26 show results for the same simulation as Figure 36 and Table 24 (plume at same temperature as average background and air). Even for a gas plume with very sharp lines such as NH_3 the difference is very minor. This is likely due to the fact that the smile effect distortion has an amplitude of less than 10% of the actual instrument resolution. As the smile distortion approaches the sensor resolution, a higher impact could be expected.

Table 26: CAI results for a gas without the smile effect in descending order

Detection Algorithm	Correction Algorithm	CAI
ACE	None	0.917778977779
GLRT	None	0.915446499974
ACE	MODTRAN	0.914938910564
GLRT	MODTRAN	0.912224472029
Signed ACE	MODTRAN	0.529866713310
Signed GLRT	MODTRAN	0.529155364275
Signed GLRT	None	0.508400513483
Signed ACE	None	0.508308426274

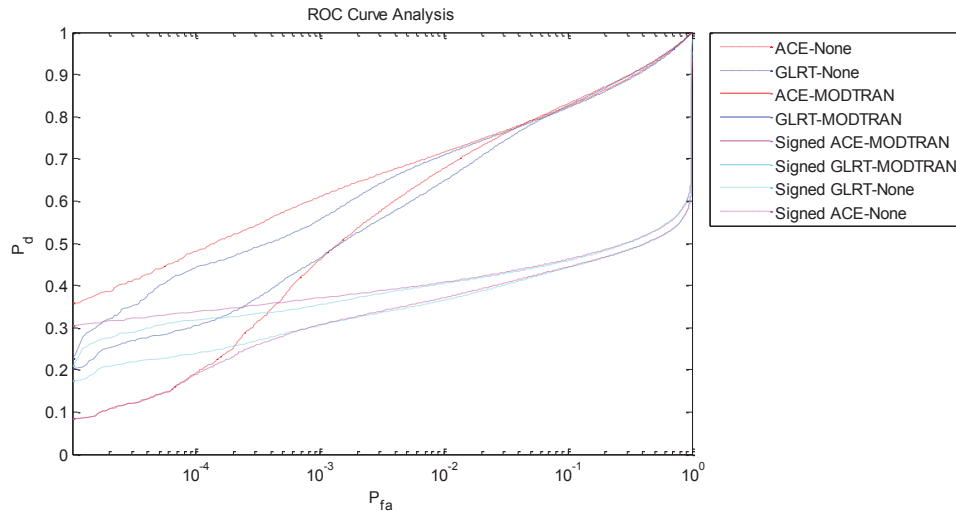


Figure 38: ROC curve analysis for a gas without the smile effect.

References

- [1] AEREX avionique inc., “Travaux en optique atmosphérique.”, Proposition AAI-2012-125216, 2012.
- [2] Berk, A., G.P. Anderson, P.K. Acharya, S.M. Adler-Golden, L.S. Bernstein, J.H. Chetwynd, T.W. Cooley, J.A. Gardner, M.L. Hoke, R.G. Kennett, J. Lee, P.E. Lewis, R.B. Lockwood, R.W. McMullen, L. Muratov and E.P. Shettle, “MODTRAN™ 5: 2006 Update”, *Proceedings of the 28th Review of Atmospheric Transmission Models Meeting*, Lexington MA., June 2006.
- [3] DiStasio, R. J. and Resmini, R. G., *Atmospheric Compensation of Thermal Infrared Hyperspectral Imagery with the Emissive Empirical Line Method and the In-Scene Atmospheric Compensation Algorithms: A Comparison*, 2010, The MITRE Corporation, Technical Paper.
- [4] Lahaie, P., *STAC : A simple technique for atmospheric correction*, internal DRDC document.
- [5] Gillespie, A. R. (1985), “Lithologic mapping of silicate rocks using TIMS.” In *The TIMS Data User’s Workshop*, JPL Publication 86-38, Jet Propulsion Laboratory, Pasadena, CA, pp. 2944.
- [6] Lahaie, P., “Performance estimation tools for: decoupling by filtering of temperature and emissivity (Defilte), an algorithm for thermal hyperspectral image processing”, *Int. J. Hi. Spe. Ele. Syst.*, **18**, 675 (2008).
- [7] Kelly, E. J. (1986). An adaptive detection algorithm. *Aerospace and Electronic Systems, IEEE Transactions on*, (2), 115-127.
- [8] Kraut, S., Scharf, L. L., & McWhorter, L. T. (2001). “Adaptive subspace detectors.” *Signal Processing, IEEE Transactions on*, 49(1), 1-16.
- [9] Robey, F. C., Fuhrmann, D. R., Kelly, E. J., & Nitzberg, R. (1992). A CFAR “adaptive matched filter detector”. *Aerospace and Electronic Systems, IEEE Transactions on*, 28(1), 208-216.
- [10] Harsanyi, J. C., Farrand, W. H., & Chang, C. I. (1994, April). “Detection of subpixel signatures in hyperspectral image sequences”. In *Proceedings of the American Society for Photogrammetry and Remote Sensing* (pp. 236-247).
- [11] Gagné, G and Ross, V., “Irradiance Estimation in Various Sky Conditions Hemispherical Cloud Model”, AEREX Report No. 2013-125216-003, September 2013.
- [12] Roy, V. "Hybrid algorithm for hyperspectral target detection." *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 2010.
- [13] Sun, Xia, Na Li, and Hui-jie Zhao. "Performance evaluation for hyperspectral target detection algorithms." *Seventh International Symposium on Instrumentation and Control Technology*. International Society for Optics and Photonics, 2008.

- [14] Aktaruzzaman, Md. (2008). Simulation and Correction of Spectral Smile Effect and its Influence on Hyperspectral Mapping. Thesis, International Institute for Geo-informaiton Science and Earth Observation, Netherlands.
- [15] Gao, B. C. et al. (2009), Atmospheric correction algorithms for hyperspectral remote sensing data of land and ocean, Elsevier Inc., Remote Sensing of Environment 113 (2009) S17–S24.
- [16] Davis, C. O., Bowles, J., Leathers, R. A., Korwan, D., Downes, T. V., Snyder, W. A., Rhea, W. J., Chen, W., Fisher, J., Bissett, W. P., & Reisse, R. A. (2002). Ocean PHILLS hyperspectral imager: design, characterization, and calibration. Optics Express, 10, 210–221.
- [17] Mourioulis, P., Green, R. O., & Chrien, T. G. (2000). Design of pushbroom imaging spectrometers for optimum recovery of spectroscopic and spatial information. Applied Optics, 39, 2210–2220.
- [18] Ross, V and Gagné, G, “CODE User Guide”, AEREX Report No. 2013-125216-005, September 2013.

List of symbols/abbreviations/acronyms/initialisms

ACE	Adaptive Cosine Estimator
AMF	Adaptive Matched Filter
BRDF	Bidirectional Reflectance Distribution Function
CAI	Curve Area Index
CEM	Constrained Energy Minimization
CODE	COrrrection and Detection Evaluation
GPU	Graphical Processing Unit
DRDC	Defence Research & Development Canada
ELM	Empirical Line Method
FWHM	Full Width Half Max
GLRT	Generalized Likelihood Ratio Test
GPU	Graphical Processing Unit
LWIR	Long Wave InfraRed
NEM	Normalized Emissivity Method
ROC	Receiver Operating Characteristic
TES	Temperature Emissivity Separation

This page intentionally left blank.