Image Processing Techniques for Lateral Migration Radiography Land Mine Images

Technical Report
ARO Grant Number
DAAG-55-98-1-0400

By

Anthony Allard
Edward Dugan
Alan Jacobs

NUCLEAR AND RADIOLOGICAL ENGINEERING DEPARTMENT
UNIVERSITY OF FLORIDA
Gainesville, FL 32611
June, 2000

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June, 2000	August, 1999 - June, 2000

4. TITLE AND SUBTITLE
Image Processing Techniques for Lateral Migration Radiography Land Mine Images

5. FUNDING NUMBERS

6. AUTHOR(S)
Anthony Allard, Edward Dugan, and Alan Jacobs

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Nuclear & Radiological Engineering
P.O. Box 118300
University of Florida
Gainesville, FL 32611

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
research Triangle Park, NC 27709-2211

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ARO 38830.1-MA-LMD

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Lateral migration radiography (LMR) is a new imaging technique based on ideas of Compton backscatter imaging. In LMR, multiple image sets are generated. Uncollimated detectors provide images that are due primarily to single-collision photons which have been mainly scattered from the soil surface or near-surface. Collimated detectors provide images that are due primarily to multiple-collision photons that contain both surface and sub-surface information.

Simple noise removal and image enhancement techniques, including weighted filters, Weiner filters and histogram equalization improved the visibility of objects and of object details. Linear combinations of collimated and uncollimated images provided a fast, simple means of surfact-object removal and of yielding additional details for mine detection. Optimal filters that maximize the signal-to-noise ratio and include a prewhitening step were particularly useful for identifying deep-buried mines. Finally, simple neural networks gave good image enhancement results and are espically efficient for surface-laid object removal.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Landmine detection, Compton backscatter imaging, lateral migration radiography, image processing, neural networks	78
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Image Processing Techniques for Lateral Migration Radiography Land Mine Images

Technical Report
ARO Grant Number
DAAG-55-98-1-0400

By

Anthony Allard
Edward Dugan
Alan Jacobs

NUCLEAR AND RADIOLOGICAL ENGINEERING DEPARTMENT
UNIVERSITY OF FLORIDA
Gainesville, FL 32611
June, 2000

Acknowledgements

The author would like to thank Dr Dugan and Dr Jacobs for all their support and advice in this work. And the author would also like to thank Zhong Su and Chris J. Wells, two graduate students, for all their help.

The material in this report is based upon work supported by, or in part by, the U.S. Army Research Office under grant number DAAG-55-98-1-0400.

TABLE OF CONTENTS

	Pages
ACKNOWLEDGEMENTS	i
INTRODUCTION	1
SECTIONS	
1 INTRODUCTION TO X-RAY LATERAL MIGRATION RADIOGRAPHY	3
1- Principles	3
2- Interpretation of the images	5
3- Mathematical model of the system	6
2 IMAGE ENHANCEMENT	8
1- Noise removal	8
1-1- weighted average filters	9
1-2- Wiener filter	10
1-2-1- introduction to Wiener filter	10
1-2-2- mathematical model	10
1-2-3- results	11
2- Color manipulation	13
2-1- histogram equalization	13
2-2- other techniques	14
3- Conclusion	15
3 LINEAR COMBINATION TECHNIQUES	16
1- Image characteristics	16
2- Image subtraction	17
3- Removal of surface laid objects	18
3-1- model of the problem	18
3-2- simple subtraction	19
3-3- interpolation and subtraction	21
4- Conclusion	22

4 IMAGE FILTERING 23

1- Introduction 23
2- Signal source separation 23
2-1- Herault-Jutten signal separation method 24
2-2- results 25
3- Optimal filtering 28
3-1- optimal filtering theory 29
3-2- application to LMR images 31
3-3- prewhitening step 32
4- Conclusion 35

5 NEURAL NETWORK 36

1- Introduction 36
2- From biological neurons to artificial neurons 37
2-1- biological neurons 37
2-2- artificial neurons 38
3- Artificial neural networks 39
3-1- architecture of artificial neural networks 39
3-2- training an artificial neural network 42
3-2-1- supervised learning 42
3-2-2- backpropagation training for a multiplayer neural network 44
4- Application of neural networks to LMR images 46
4-1- build the input and the output 47
4-1-1- nature of the input and the output 47
4-1-2- rescale the input and the output vectors 48
4-2- set the parameters of the network 49
4-2-1- number of layers and neurons 49
4-2-2- the activation function 50
4-3- choice of the learning examples 51
5- Experimental results 52
5-1- size of the selection square 53
5-2- influence of the number of neurons 53
5-3- influence of the K coefficient 55
5-4- final results 55
6- Conclusion 61

6 AUTOMATIC IMAGE PROCESSING 63

1- Introduction 63
2- Object recognition 63
2-1- contour detection 63
2-2- labeling task 65
3- Conclusion 66

7 USER INTERFACE 67

1- Description of user interface 67
2- Command description 68
2-1- image commands 68
2-2- filter 68
2-3- linear combination 69
3- Architecture of the program 70

CONCLUSION 72

REFERENCES 73

Introduction

Lateral Migration Radiography (LMR) is a new imaging technique, developed at the University of Florida (UF) which based on ideas of Compton backscatter imaging systems [1]. The UF mine detection project developed an LMR system in a laboratory setting for the detection of land mines. This system consists of an x-ray generator and two pairs of detectors along with a data acquisition system. Monte Carlo simulations of the LMR system have shown that the uncollimated detectors mainly detect the once-scattered photons that have been reflected near the surface, while the collimated detectors mainly detect the multi-scattered photons that have undergone sub-subsurface collision. Thus, the uncollimated images primarily contain information about the surface or near surface features, while collimated images primarily contain information on the subsurface features.

When applied to a real or surrogate mine [2], this system has proven to be an efficient technique to get information not only on the position of the mine, but also on the main features of the mine. However, in some unfavorable situations, like a deep depth-of-burial (DOB) or in the presence of surface laid objects, the quality of the image of the buried mine can be greatly degraded. This report details the theoretical basis and the application to LMR images of several image processing techniques which use the characteristics of the LMR images to improve the quality of the image, to emphasize the photon energy deposition in the mine, or to remove the surface laid objects.

In Section 1 we look at the LMR principle and the characteristics of the measured images. In section 2 we study some general image enhancement techniques that can be applied to LMR images. In Sections 3, 4 and 5 we describe some linear combination techniques, an optimal filter and a neural network, respectively, used to emphasize the image of the mine or to remove the surface laid objects. In Section 6 we discuss the automatic treatment of the LMR images. And next, in Section 7, we present the user interface program.

Section 1

Introduction to X-ray

Lateral Migration Radiography

1- Principles

Lateral Migration Radiography is a technique that uses the lateral transport of multi-scattered photons in material to form images.

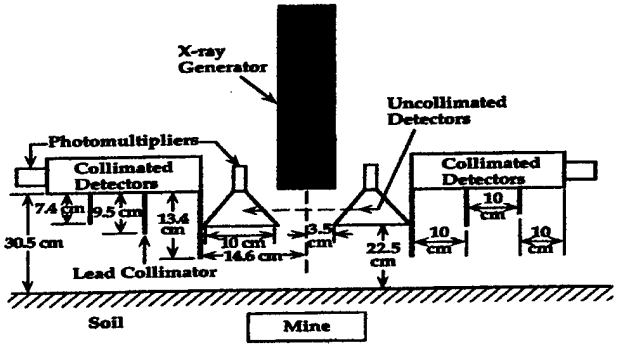


Figure 1: Schematic of a X-Ray Lateral Migration Radiography system.

This X-Ray Lateral Migration Radiography method is based on two different kinds of photon interactions: Compton scattering and the photoelectric effect. These two interactions are functions of electron density, effective atomic number of the migration

material, and photon energy. Thus, the energy of the backscattered photons which reach the detector depends on the nature of the materials the photons have been reflected from. The final image is made of m x n pixels, each pixel representing the energy deposition on the detectors when the beam is positioned at that pixel location.

The LMR system used for the landmine detection project utilizes one X-Ray source and two detector types to form images: collimated and uncollimated. The collimated detectors are used to detect photons that undergo lateral migration. The collimators prevent photons that have undergone only one collision from reaching the detector; thus, the collimated detector signals primarily contain information on the subsurface features. The uncollimated detectors effectively detect the photons that hit the surface and scatter directly back; thus the uncollimated detectors signals primarily contain information on the surface, or near surface, features.

Figure 2 is an example of an image of a surrogate mine with a flush depth of burial (DOB).

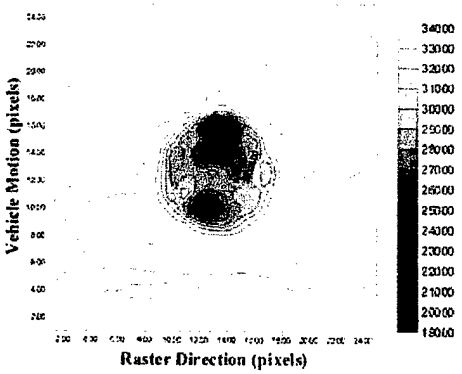


Figure 2: Front uncollimated detector image of a surrogate mine, flush DOB

2- Interpretation of the images

The energy deposition in the collimated detectors is mainly due to the energy deposition from multi-scattered photons.

The Compton scattering cross section is greater for a plastic mine than it is for the soil, so the photons migrating through the mine have a higher probability of being scattered out of the soil and being registered by one of the detectors. Therefore, when the x-ray beam is scanning toward the mine and just reaches the front edge of the mine, the forward direction migrating photons are transported more readily than in any other direction and are registered in the front collimated detector. Because of this, the highest intensity seen in the front collimated detector is seen earlier than the actual physical center of the mine and this appears as a backward shifting of the mine in the front collimated image. In the rear, the opposite appears. This shifting varies directly with the depth of burial of the land mine. Figure 3 shows the shift for a surrogate plastic mine with a flush depth of burial.

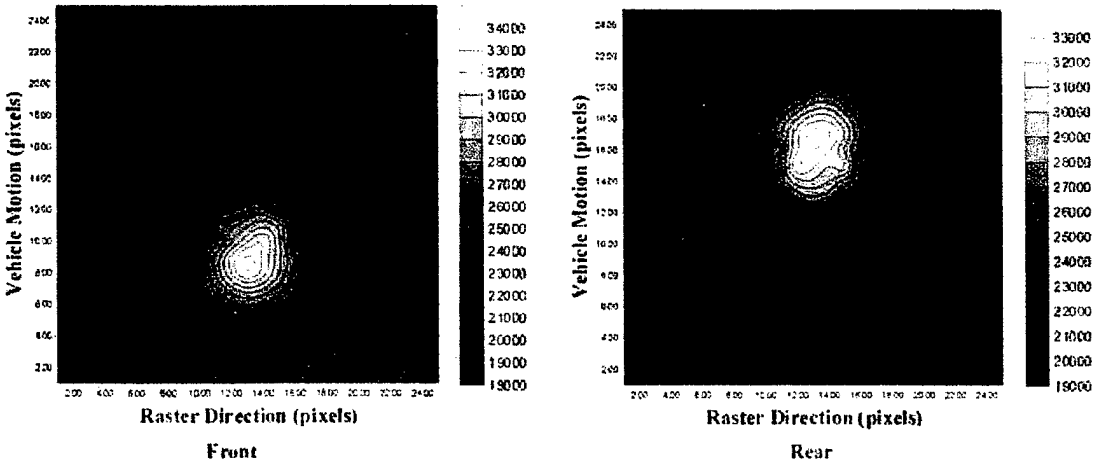


Figure 3: Front and rear collimated detector images of a surrogate mine with a flush DOB.

3- Mathematical model of the system

The subject of all the work has been to improve the LMR images, applying signal and image processing techniques. So we need to define a mathematical model of the system.

The images observed are a measurement of the energy deposition in the collimated and uncollimated detectors due to the reflection of the photons from the soil, the surface laid objects and the buried objects. Thus the total energy deposition on each detector can be modeled as:

$$\begin{cases} c(x,y) = b_c(x,y) + s_c(x,y) + l_c(x,y) \\ u(x,y) = b_u(x,y) + s_u(x,y) + l_u(x,y) \end{cases}$$

where : $c(x,y)$: is the energy deposition at pixel (x, y) in the collimated detector.

$u(x,y)$: is the energy deposition at pixel (x, y) in the uncollimated detector.

$b_c(x,y)$: energy deposition in the collimated detectors due to the reflection of photons from buried objects.

$b_u(x,y)$: energy deposition in the uncollimated detectors due to the reflection of photons from buried objects.

$s_c(x,y)$: energy deposition in the collimated detectors due to the reflection of photons from the soil.

$s_u(x,y)$: energy deposition in the uncollimated detectors due to the reflection of photons from the soil.

$l_c(x,y)$: energy deposition in the collimated detectors due to the reflection of photons from surface laid objects.

$I_u(x, y)$: energy deposition in the uncollimated detectors due to the reflection of photons from surface laid objects.

All these notations are used in this report.

So now that we have described the principles of the landmine LMR system, the characteristics of the images and a simple mathematical model of the images we can start studying some image processing techniques.

Section 2

Image Enhancement

The process of image acquisition frequently leads to image degradation. Due to instrumentation noise, non-uniform soil, surface clutter, etc., the LMR images can be de-gaded. The goal of image enhancement is to improve an image, where 'improve' is sometimes defined objectively (e.g., increase the signal-to-noise ratio), and sometimes subjectively (e.g., make certain features easier to see by modifying the colors or intensities).

1- Noise removal

As introduced in Section 1 and taking into account $n(x,y)$, the instrumentation noise, we can give a new model for the acquired images:

$$o(x,y) = b(x,y) + l(x,y) + s(x,y) + n(x,y)$$

where:

o(x ,y): total energy deposition in the detector

$b(x,y)$: energy deposition from the buried objects

$l(x,y)$: energy deposition from the surface laid objects

$s(x,y)$: energy deposition from the soil

In the case of a non-uniform soil, $s(x,y)$ can also be considered as a noise.

Sometimes the energy deposition due to the soil, considered as a noise, when added to the noise from the detector gives a signal that is higher than the energy deposition from the buried objects; so we can not see the buried object in the image. To improve these images a first idea would be to apply different noise removal techniques.

1-1- Weighted average filters

Weighted average filters are very simple, spatially, low pass filters that can be used to remove statistical noises and some small soil irregularities. These filters use a weighted mask filter to compute the new value of the pixel located at the center of this mask.

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Figure 1: Mask of the mean filter

But low pass filtering often leads to edge deterioration. So we introduce an adaptive low pass filter.

1-2- Wiener filter

1-2-1- introduction to Wiener filter

An adaptive low pass filter is a filter which adjusts its pass-band to each situation. When encountering a uniform region this filter applies a low pass filter with a small band-pass to smooth this region, but when encountering an object's contour this filter increases its band-pass to avoid deteriorating the edge.

1-2-2- mathematical model

We note $d(x,y) = b(x,y) + l(x,y)$ the signal we want to observe, and $nn(x,y) = n(x,y) + s(x,y)$ the noise added to the image. We assume that these two signals are centered (if not, we center them).

The noisy observation $o(x,y)$ is given by:

$$o(x,y) = d(x,y) + nn(x,y)$$

We wish to determine $d(x,y)$ from $o(x,y)$ using a linear estimator given by

$$\hat{d}(x,y) = o(x,y) * h(x,y)$$

where $*$ is the convolution product and $h(x,y)$ a linear filter.

The linear estimator we used is based on the Linear Minimum Square Error; the criterion used is the minimization of an error:

$$Error = E\{|e(x,y)|^2\}$$
$$e(x,y) = d(x,y) - \hat{d}(x,y)$$

This signal estimation problem can be solved using the orthogonality principle. This principle states that the Error function is minimized by requiring that $e(x,y)$ be uncorrelated with $o(x,y)$. This leads us to the following result:

$$H(\upsilon,\phi) = \frac{P_{do}(\upsilon,\phi)}{P_o(\upsilon,\phi)},$$

where P_{do} is the interaction power spectrum of d and o, and P_o the power spectrum of o. This filter $H(\upsilon,\phi)$ is called the noncausal Wiener Filter.

If we suppose that nn is a white noise , then we can show that

$$P_{do}(\upsilon,\phi) = P_d(\upsilon,\phi)$$
$$P_o(\upsilon,\phi) = P_d(\upsilon,\phi) + P_{nn}(\upsilon,\phi)$$

So the noncausal Wiener filter is given by:

$$\boxed{H(\upsilon,\phi) = \frac{P_d(\upsilon,\phi)}{P_d(\upsilon,\phi) + P_{nn}(\upsilon,\phi)}}$$

1-2-3- results

If we apply this filter to the whole image, then the resulting image is not well filtered. Indeed, when calculating the Wiener filter we made the assumption that the noise $nn(x,y)$ is a white noise. However, if we assume that the noise $nn(x,y)$ is larger than $d(x,y)$, then the autocorrelation function of $o(x,y)$ can be considered as the autocorrelation of the noise $nn(x,y)$, which should be a Dirac function for a white noise. However, this is not what we observe on Figure 2.

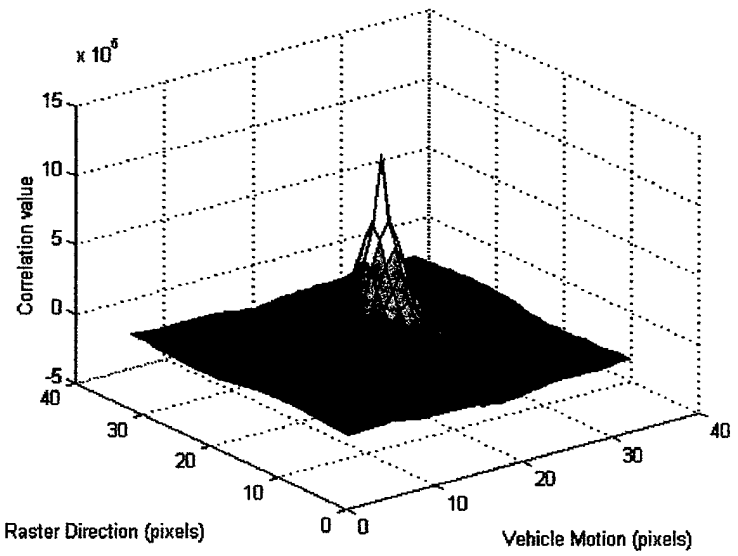


Figure 2: Autocorrelation function of $o(x,y)$

So to overcome this problem we apply this filter locally so that now we use an adaptive Wiener filter which locally estimates $P_o(x,y)$ and $P_d(x,y)$, where $nn(x,y)$ can be considered as a white noise. Then the new filtered signal is:

$$\hat{d}(x,y) = m_d(x,y) + \frac{{\sigma_d}^2(x,y)}{{\sigma_d}^2(x,y) + {\sigma_{nn}}^2(x,y)}(o(x,y) - m_d(x,y))$$

where $m_d(x,y)$ is the estimated mean value of d in the local neighborhood.

So this new filter is an adaptive filter which tailors itself to the local image variance. Where the variance of the image is large this filter performs little smoothing; where the variance is small, this filter performs large smoothing.

The results are shown on Figure 3b.

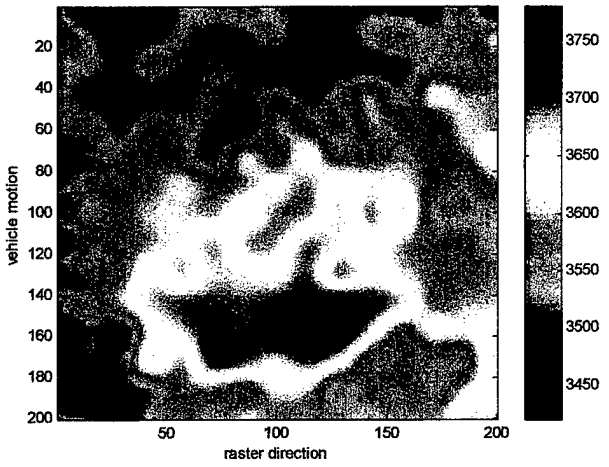


Figure 3a: Original image

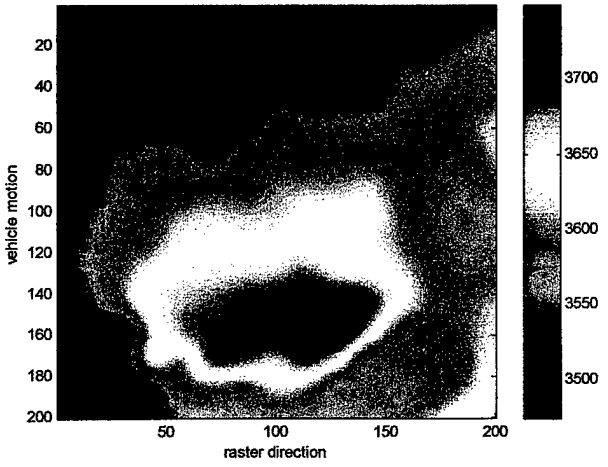


Figure 3b: Wiener filtered image

So this Wiener adaptive filtering method is a good way to reduce noise without altering the contour as would be done with a simple low-pass filter. However, this filter is not able to point out a buried mine when it is totally hidden under the energy deposition due to the soil.

2- Color manipulation

2-1- Histogram equalization

When plotting a 2D landmine image, Matlab software, like most of the software, uses colors to show the distribution of energy. In the case of low contrast images, for example when the mine is deep-buried, there is one predominant color and so we cannot see any details. A technique to increase the contrast of an image is to equalize the histogram of the image so that the number of pixels per energy container is about the same for each one.

Figures 4a and 4b represent the histogram and the image of a plastic mine centered with a DOB of 3 inches. Figure 5a and 5b represent the new histogram and the image after performing a *histogram equalization* .

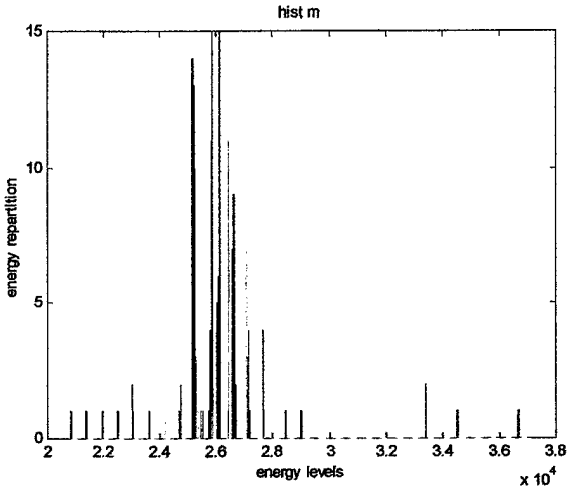


Figure 4a: Histogram of the original image

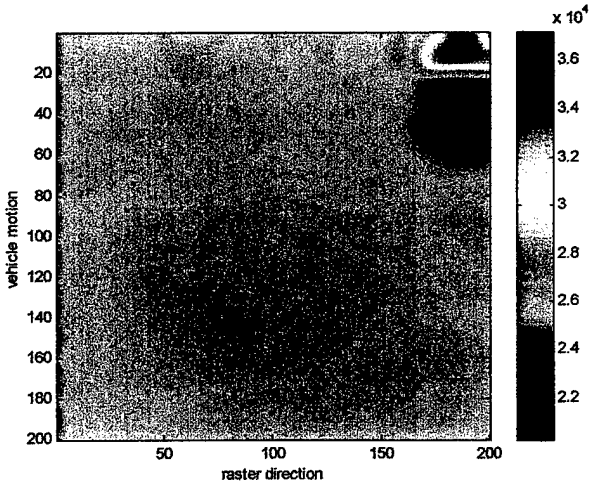


Figure 4b: Original image interpolated to 200*200 pixels

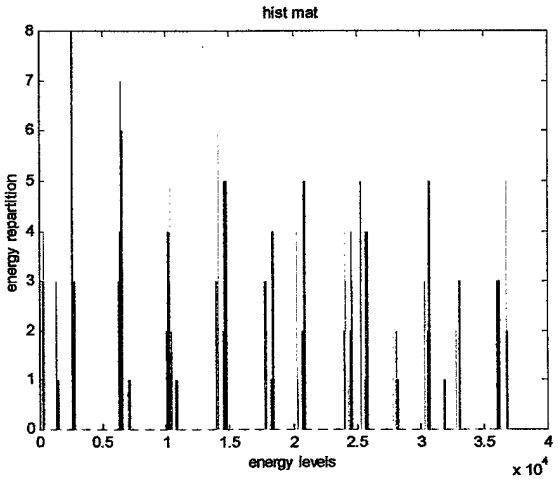


Figure 5a: Histogram after equalization equalization

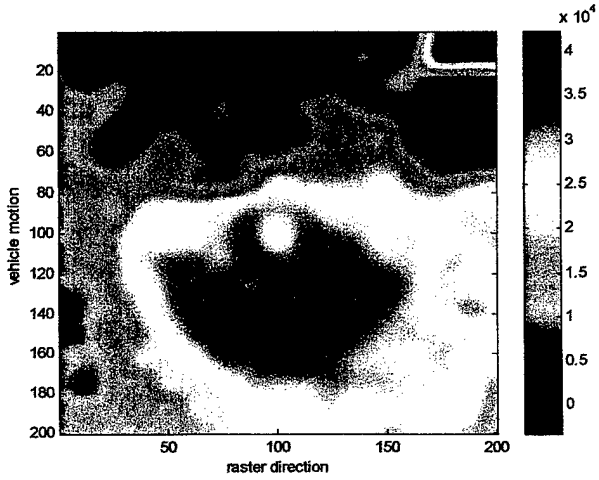


Figure 5b: Image after histogram

interpolated to 200*200 pixels

2-2- Other techniques

To increase the contrast of some images we can use other techniques such as the gamma correction which maps the value of the pixels to some other values using a specified function like a linear function.

These methods can be useful but are highly correlated to the nature of the image.

3- Conclusion

In this section we have presented some image enhancement techniques that improve the quality of the images. These techniques are very general and do not take into account the specificities of the LMR system. So in the next sections we introduce new signal and image processing techniques specific to the LMR system which will help the user to decide whether or not there is a buried mine in the soil.

Section 3

Linear Combination Techniques

In this section we study some linear combination techniques based on the characteristics of the LMR system that will help the user to conclude whether or not there is a mine present.

1- Image characteristics

As explained in Section 1, uncollimated images primarily contain information on the surface features and collimated images contain information on both surface and subsurface features. In addition, we can observe a shift in the apparent location of a buried mine in both front and rear collimated detectors.

So in this section we will use these characteristics to remove surface laid objects that may 'hide' buried objects, or point out these objects.

2- Image subtraction

Because of the shift in the position of the buried objects, and because surface laid objects remain at the same position, we could subtract collimated images to remove surface laid objects and point out buried objects.

To do so we have to normalize both collimated images. One possible normalization is to divide each image by its total energy.

The result of this technique is shown on Figure 1c for a surrogate plastic mine centered with a DOB of 1 inch and a stick of wood and a rock laid on the surface.

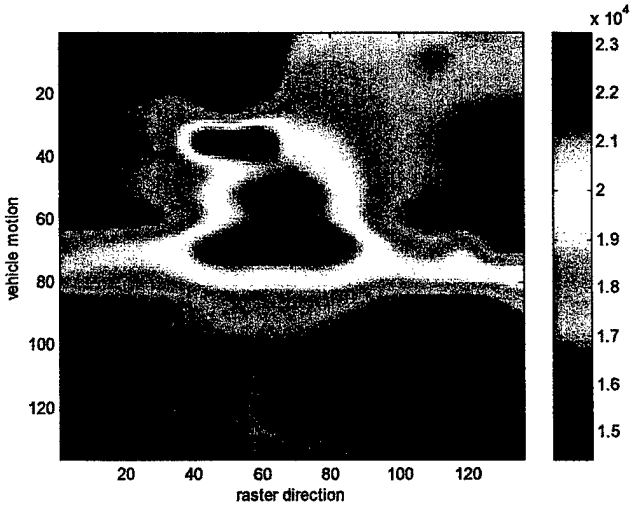


Figure 1a: Original rear collimated image

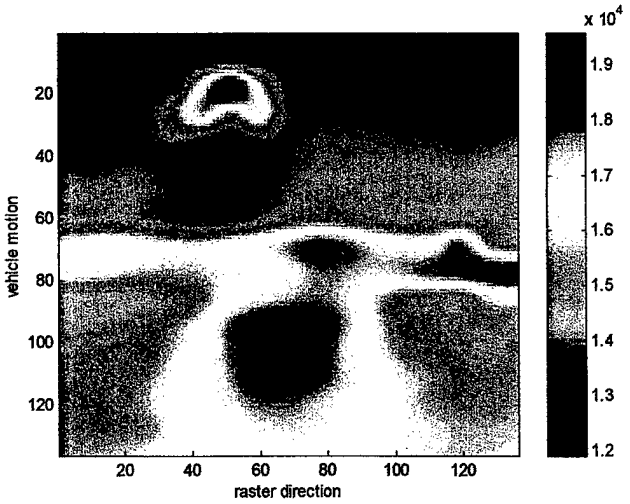


Figure 1b: Original front collimated image

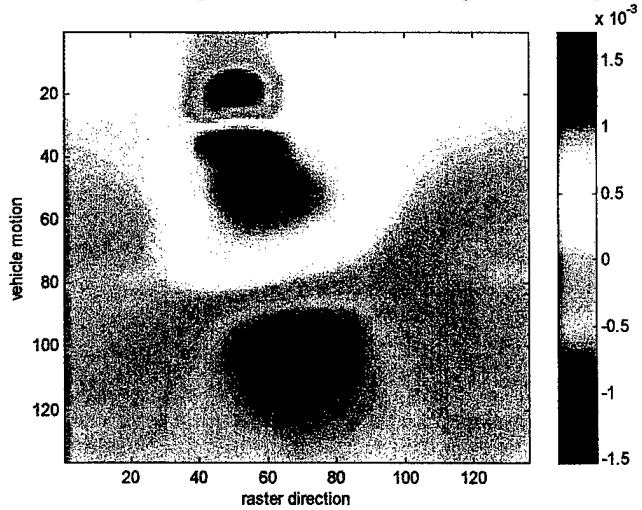


Figure 1c: Image obtained by subtraction of both collimated images

So on the one hand, the mine is more visible in the resulting image and appears at two locations, one showing a signal intensity increase and the other an intensity decrease. On the other hand, the stick of wood has been removed, but the rock still appears. This is due to the fact that there is a shift in the position of the rock too.

So this method is a very simple and efficient way to point out buried mines which then appear as two objects with the same shape but with opposite signal intensities, shifted in the vehicle motion direction. But sometimes it fails in removing surface laid objects.

3- Removal of surface laid objects

In some cases surface laid objects can 'hide' some buried mines in the collimated images. So we might need to remove them. To do so we use the fact that uncollimated detectors respond primarily to surface laid objects.

3-1- Model of the problem

A model of the energy deposition in both collimated and uncollimated detectors is (Section 1):

$$\begin{cases} c(x,y) = b_c(x,y) + s_c(x,y) + l_c(x,y) & (1) \\ u(x,y) = b_u(x,y) + s_u(x,y) + l_u(x,y) & (2) \end{cases}$$

If we assume that the soil is uniform without any slope, we can easily remove $s_c(x,y)$ and $s_u(x,y)$. To do so we consider that the energy deposition most represented in each detector is the energy deposition from the soil which we can extract using the histogram of the image.

3-2- Simple subtraction

In a first approach we can assume that the energy deposition in the uncollimated detectors is only due to the reflection of the photons from surface laid objects, which means that:

$$u(x,y) = l_u(x,y) = \alpha l_c(x,y).$$

So $$\hat{b}_c(x,y) = c(x,y) - l_c(x,y) = c(x,y) - \frac{1}{\alpha} l_u(x,y)$$

$$\boxed{\hat{b}_c(x,y) = c(x,y) - \frac{1}{\alpha} u(x,y)}$$

where α is a normalization factor.

Figure 2 summarizes the method:

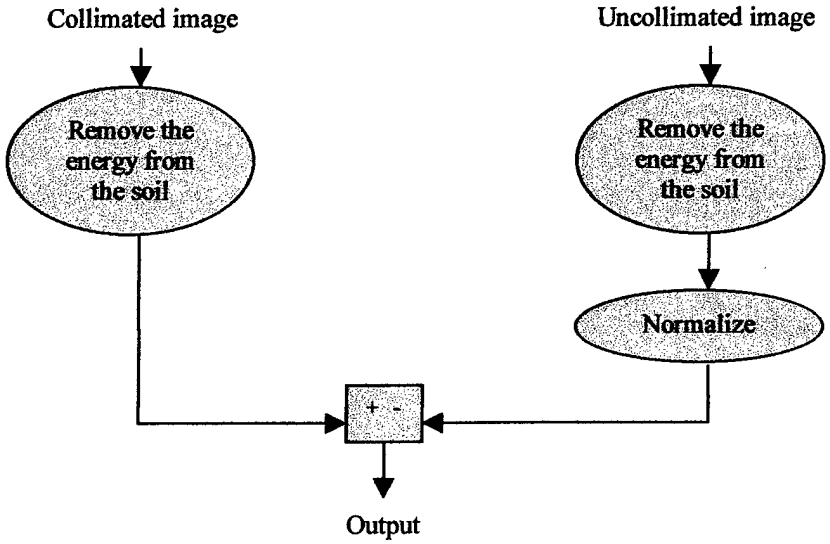


Figure 2: Block diagram of the surface object removal method

Figure 3a represents the original collimated image and Figure 3b the output image of this surface object removing technique.

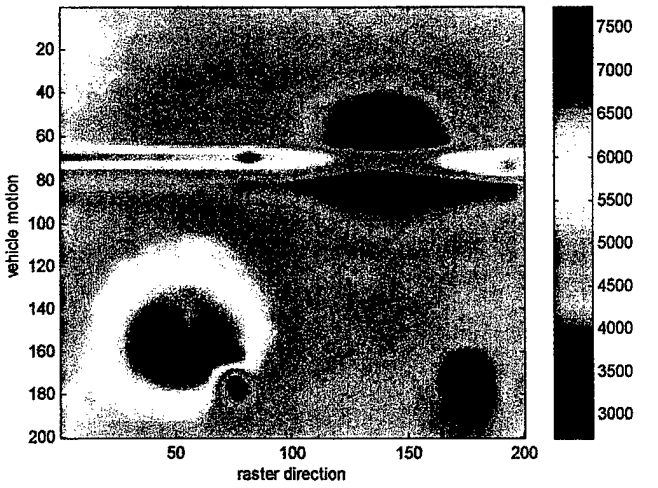


Figure 3a: Original collimated image. 2 buried mines centered at (150,70) and (50,160) and 3 surface laid objects

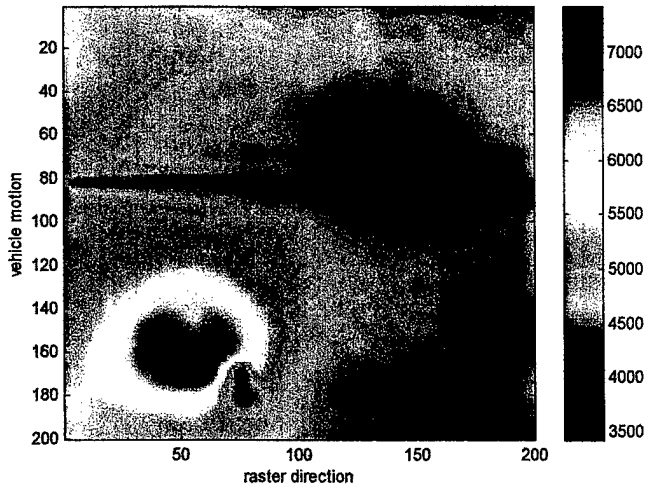


Figure 3b: Image after removing surface laid objects

The resulting image is not very good because when removing the stick of wood from the collimated image we also remove a part of the energy deposition from the mine. Indeed, the uncollimated detector also responds to the buried objects. So *u(x,y)* is not equal to *l(x,y)* but $u(x,y) = l_u(x,y) + b_u^{\varepsilon}(x,y)$ (3), where ε means that the energy deposition due to the reflection of photons from buried objects exists but its contribution to the total energy deposition is low. So to improve the result of the technique, taking into account this new element, we can develop a new technique.

3-3- Interpolation and subtraction

We can estimate $b_u^{\varepsilon}(x,y)$ using the frontier pixels of the object on the uncollimated image and an interpolation technique based on Laplace's equation.

Then, using equation (3), $l_u(x,y)$ can be estimated by :

$$\hat{l}_u(x,y) = u(x,y) - \hat{b}_u^{\varepsilon}(x,y)$$

And so, using equation (1), $\hat{b}_c(x,y)$ is :

$$\hat{b}_c(x,y) = c(x,y) - \hat{l}_c(x,y) = c(x,y) - \frac{1}{\alpha}.\hat{l}_u(x,y)$$

$$\boxed{\hat{b}_c(x,y) = c(x,y) - \frac{1}{\alpha}.\left[u(x,y) - \hat{b}_u^{\varepsilon}(x,y)\right]}$$

Figure 4 shows the output of this new method when using the same collimated and uncollimated images as for Figure 3.

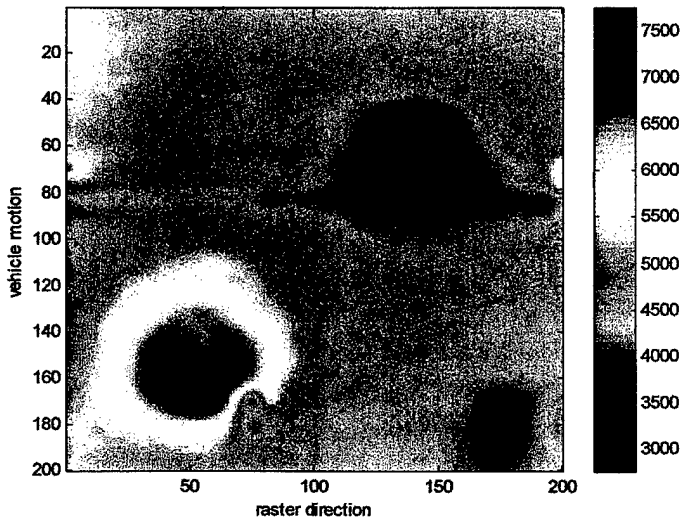


Figure 4: Image after removing surface laid objects (using interpolation and subtraction)

So this method is more efficient in removing surface laid objects than the first one which didn't use the interpolation step. However, this second method requires a selection of the surface laid objects to remove. So far this selection has been done manually. We will discuss automatic selection in Section 6.

4- Conclusion

In this section we have presented some linear combination techniques based on the properties of the collimated and uncollimated images. Using these techniques we were able to remove surface laid objects and to point out buried mines. These methods appear to give an indication to the user that allows him to decide whether or not there is a mine buried in the soil. In the next section we introduce some filtering techniques.

Section 4

Image Filtering

1- Introduction

In this section we study two filtering techniques designed to point out the buried objects in the collimated images.

In Section 1 the physical system is modeled by:

$$\begin{cases} c(x,y) = b_c(x,y) + s_c(x,y) + l_c(x,y) \\ u(x,y) = b_u(x,y) + s_u(x,y) + l_u(x,y) \end{cases}$$

The main purpose of this study is to apply some signal and image processing techniques that would help the user to make a decision regarding whether or not there a buried mine.

So in an ideal situation we would like to extract $b_c(x,y)$ from the collimated images.

2- Signal source separation

Over the last decades, Signal Source Separation has been an important field of research in Signal Processing. This technique aims at extracting source signals from mixture signals.

In our case we would like to extract $b_c(x,y)$ from collimated and uncollimated images.

Most of the Signal Source Separation techniques are based on the statistical independence of the source signals and aim at computing statistically independent outputs by linear combination of the inputs (the mixture signals). Figure 1 illustrates the principle:

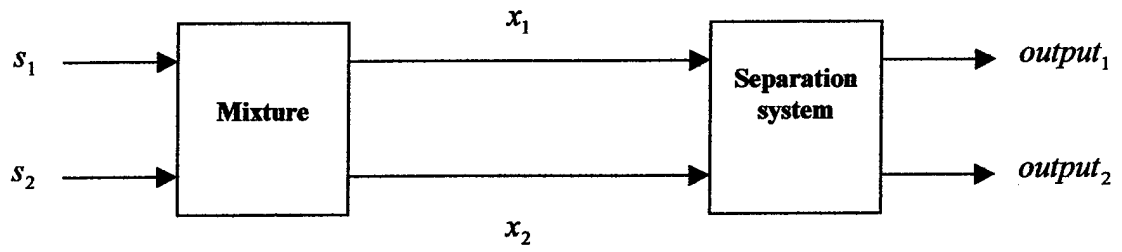


Figure 1: Signal Source Separation principle

s_1, s_2: statistically independent source signals.

x_1, x_2: mixture signals.

$output_1, output_2$: statistically independent signals.

2-1- Herault-Jutten signal separation method

To test signal separation methods, when applied to the LMR imaging system, we used the Herault-Jutten method [3]. This method is based on a neural network which adapts its internal coefficients to achieve the statistical independence of the outputs. Figure 2 explains the principle of this method:

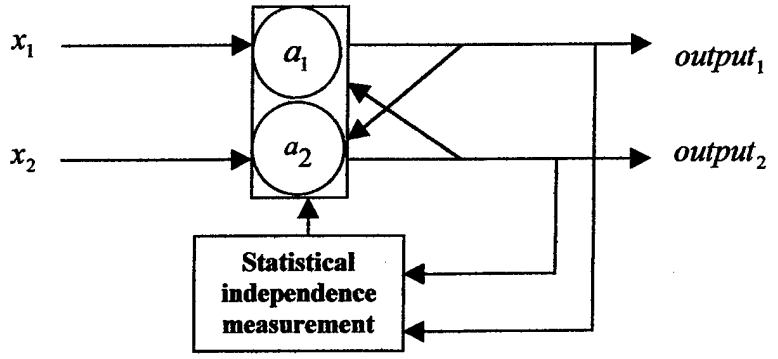


Figure 2: Herault-Jutten Source Separation method

$$\begin{cases} output_1 = x_1 - a_1.output_2 \\ output_2 = x_2 - a_2.output_1 \end{cases}$$

By an iterative algorithm a_1 and a_2 are adapted to achieve the statistical independence of the outputs.

2-2- Results

To test this method of signal separation, we first used a mixture of two spatially independent signals with a correlation coefficient of –0.29. This correlation coefficient is defined for two functions *a* and *b* by:

$$\rho = \frac{\langle a,b \rangle}{\sigma_a \sigma_b} = \frac{\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}(a(i,j)-\bar{a})(b(i,j)-\bar{b})}{\text{var}\,iance(a).\text{var}\,iance(b)}$$

where $\bar{a}$ and $\bar{b}$ are the mean values of a and b.

$|\rho| \in [0;1]$. As $|\rho|$ increases, the spatial correlation between the two images increases.

For these two test images the Herault-Jutten method is a success because the outputs correspond to the two original images, as shown in Figures 3.

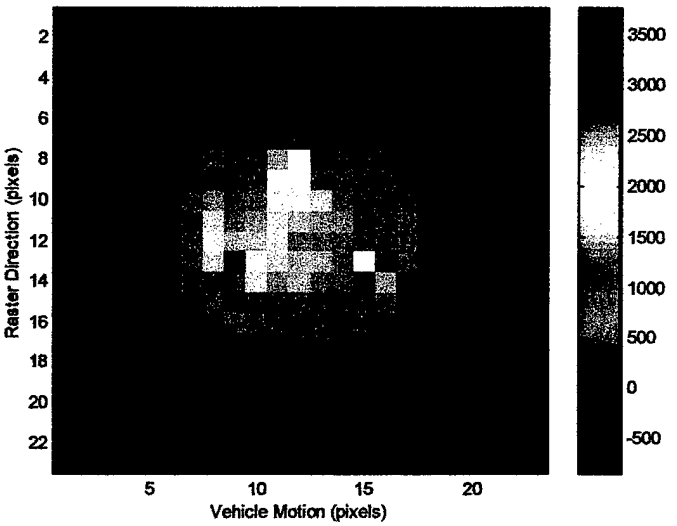


Figure 3a: Source image 1

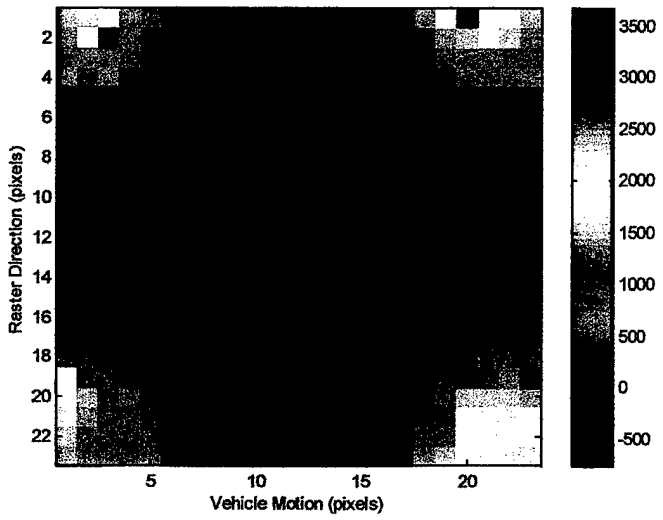


Figure 3b: Source image 2

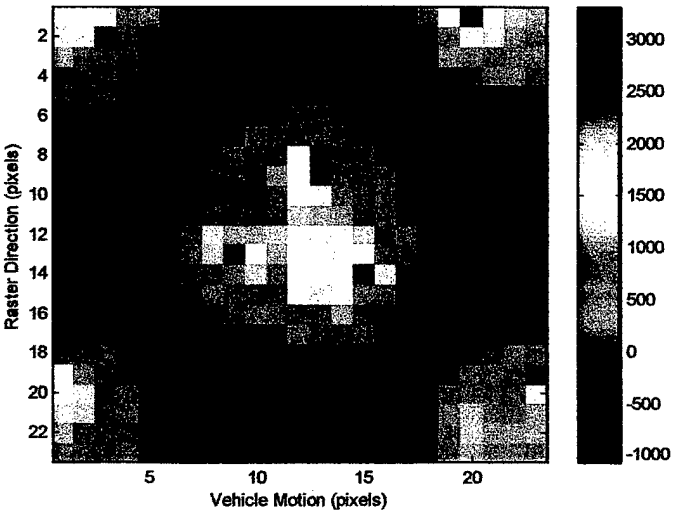


Figure 3c: Mixture 1

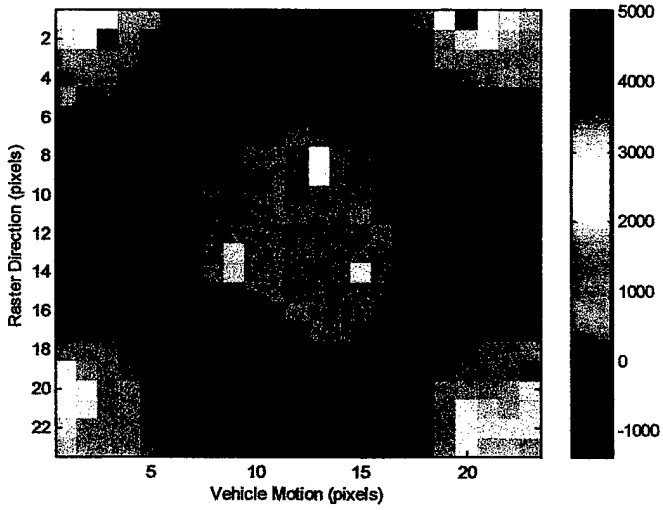


Figure 3d: Mixture 2

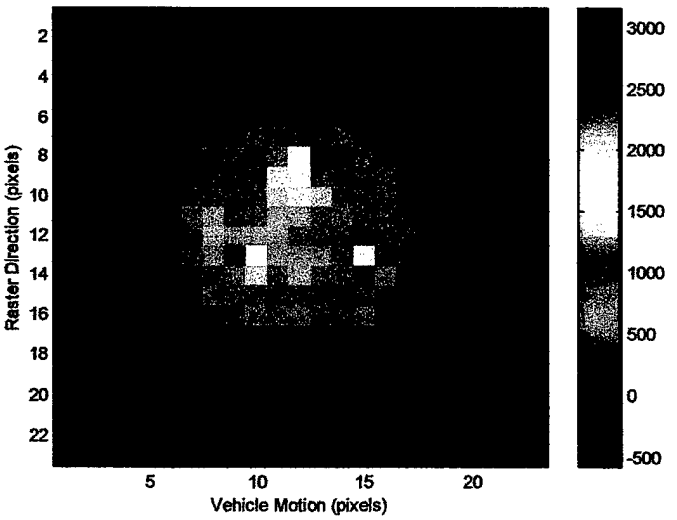


Figure 3e: Signal estimated 1

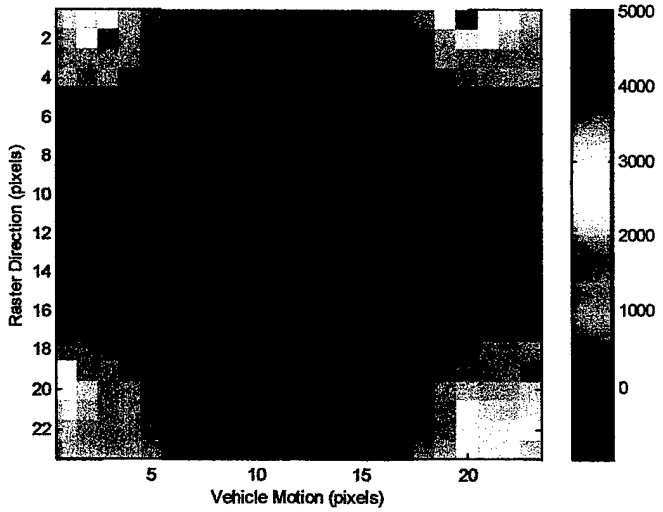


Figure 3f: Signal estimated 2

Next, we apply this signal source separation method to different real LMR images.

Each image was taken in standard conditions with a buried mine and a rock laid on the surface, and a flat and uniform soil.

We didn't succeed in obtaining two output images, each representing the mine and the rock, as shown in Figure 4.

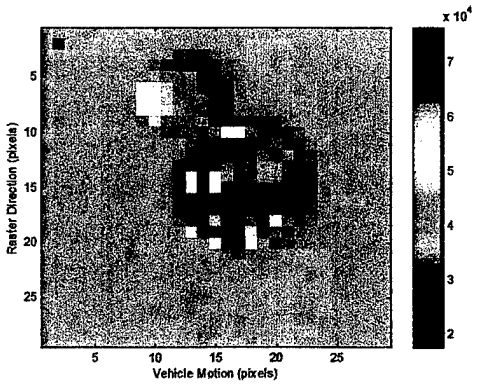


Figure 4a: Measured image 1

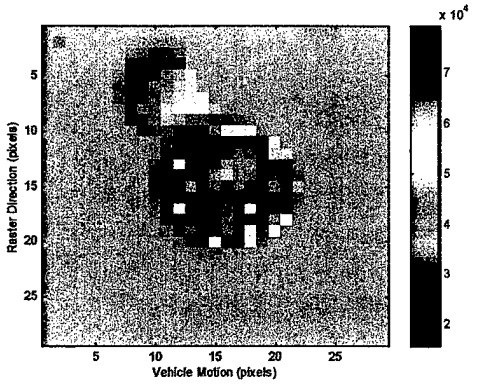


Figure 4b: Measured image 2

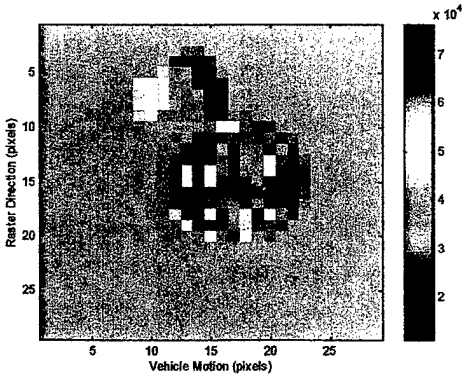


Figure 4c: Output image 1

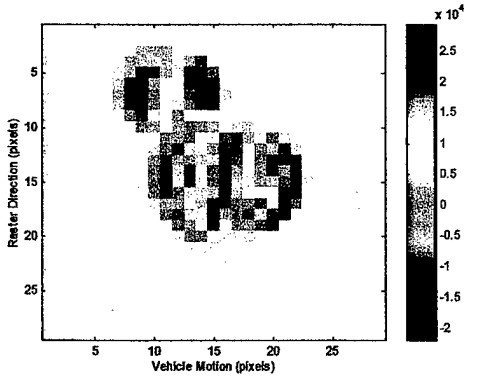


Figure 4d: Output image 2

Figures 4a, b, c, d: : Input and Output images of the Herault-Jutten method.
These images are 29x29 pixels images with 2 mines centered in pixels (12;7) and (17;15).

However, the Herault-Jutten algorithm converged, which means that the final outputs were uncorrelated images. So we can assume that the source images, which are the

images of the mine and the rock, are statistically dependent images (because they can be considered as a linear combination of two statistically independent images (the outputs)), which means that the fundamental assumption of the signal source separation techniques is not satisfied. So we cannot apply signal separation source techniques to estimate the image of each mine.

The idea of using signal separation source techniques was maybe optimistic because we tried to retrieve the images of each single object (buried mines or surface laid objects), without giving the algorithm any information about the signal we were looking for.

3- Optimal Filtering

All we are interested in is the mine; everything else can be considered as noisy information and treated as noise. So far we have not used the information we have about the mine signal we are looking for. Thus, a new idea is to use a method that could distinguish the mine from the noise, using some information about the mine.

3-1- Optimal Filtering Theory

First, we assume that we know exactly the signal of the mine we are looking for, $b(x,y)$, and we note that $n(x,y)$ is the noise in the image. Thus, the collimated image is

$$c(x,y) = b(x,y) + n(x,y)$$

If (x_0, y_0) is the center of the buried mine, then what we want to compute is the linear filter that maximizes the following signal-to-noise ratio that will point out the mine.

$$\chi^2(x_0,y_0) = \frac{Power_{b_1}(x_0,y_0)}{Power_{n_1}(x_0,y_0)} \text{ at the location } (x_0,y_0).$$

where, $\begin{cases} b_1(x,y) = g * b(x,y) \\ n_1(x,y) = g * n(x,y) \end{cases}$

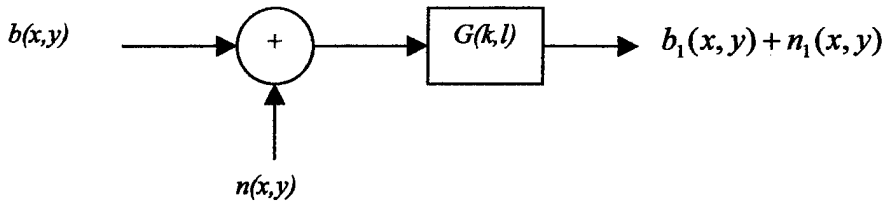


Figure 5: Filtering sytem

3-1-1- $Power_{b_1}(x_0,y_0)$

$b_1(x,y)$ is a deterministic signal, so $Power_{b_1}(x_0,y_0) = b_1^{\,2}(x_0,y_0)$. Using the inverse Fourier transform we can write:

$$b_1(x_0,y_0) = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} B_1(k,l)\exp(2i\pi(\frac{kx_0}{M} + \frac{ly_0}{N})),$$

where $B_1(k,l)$ is the Fourier transform of $b_1(x,y)$.

$$B_1(k,l) = G(k,l)B(k,l)$$

$$b_1(x_0,y_0) = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} G(k,l)B(k,l)\exp(2i\pi(\frac{kx_0}{M}+\frac{ly_0}{N}))$$

where $B(k,l)$ is Fourier Transform of $b(x,y)$.

3-1-2- $Power_{n_1}(x_0,y_0)$

We assume that $n(x,y)$ is a statistical signal, stationary to the second order so:

$$Power_{n_1}(x_0,y_0) = E\{n^2{}_1(x_0,y_0)\} = \Gamma_{n_1n_1}(0) = TF^{-1}\{\Omega_{n_1n_1}(k,l)\}(0,0)$$
$$= \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}\Omega_{n_1n_1}(k,l).\exp(2i\pi(\frac{k.0}{M}+\frac{l.0}{N}))$$
$$= \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}|G(k,l)|^2\Omega_{nn}(k,l)$$

where $\Omega_{nn}(k,l) = N(k,l).\overline{N}(k,l)$ and $\Omega_{n_1n_1}(k,l) = N_1(k,l).\overline{N}_1(k,l)$. $N(k,l)$ and $N_1(k,l)$ are the Fourier Transforms of $n(x,y)$ and $n_1(x,y)$.

3-1-3- Optimal Filter

Finally,

$$\chi^2(x_0,y_0) = \frac{\left|\frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} G(k,l)B(k,l)\exp(2i\pi(\frac{kx_0}{M}+\frac{ly_0}{N}))\right|^2}{\frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}|G(k,l)|^2\Omega_{nn}(k,l)} \qquad (1)$$

If we assume that $n(x,y)$ is a white noise, then $\Omega_{nn}(k,l) = K,\ \forall(k,l)$. Then, using the Cauchy-Schwartz inequality, we can show that the linear filter $g(x,y)$ that maximizes $\chi^2(x_0,y_0)$ is:

$$\chi^2_{b,b_1}(x_0,y_0)\ \text{MAX} \Leftrightarrow \begin{cases} G(k,l) = \alpha.\overline{B}(k,l)\exp(2i\pi(\frac{kx_0}{M}+\frac{ly_0}{N})) \\ g(x,y) = \alpha.\dfrac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} B(k,l)\exp(2i\pi(\frac{k(x_0-x)}{M}+\frac{l(y_0-y)}{N})) \end{cases}$$

$\boxed{g(x,y) = \alpha.b(x_0-x, y_0-y)}$, where K is a constant.

3-2- Application to LMR images

In point 3-1 we supposed that we exactly knew the energy repartition $b(x,y)$ of the buried object and that we knew its location (x_0,y_0) too, which is obviously not true. However, we can approximate $b(x,y)$ by a zero signal, except in a region where the value of the pixel is a constant. This region is selected by the user using the mouse. We applied this this filter to a collimated image as shown in Figure 6.

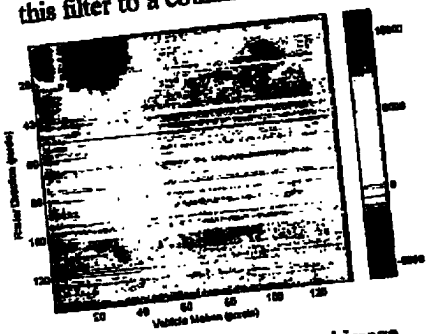


Figure 6a: Original collimated image. Mine with a DOB of 3" centered at (90,80), and a rock at (20,15).

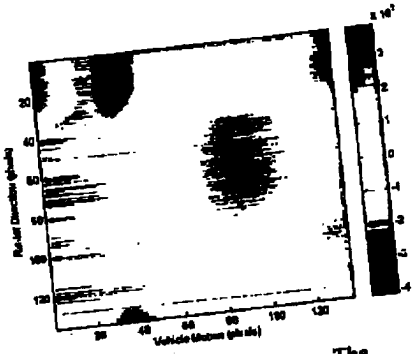


Figure 6b: Filtered image. The cross marks the pixel where χ^2 has been maximized.

So this filter has pointed out the mine which was hardly visible in the original collimated image. However, we have to be careful when looking at the filtered image because the filter $g(x,y)$ has been computed to maximize the signal-to-noise ratio at the (x_0,y_0) location, and so we can not conclude whether or not there is a mine at any other location, even if a peak in the energy deposition appears at another location.

3-3- Prewhitening step

When computing the optimal filter we made the assumption that the noise signal $n(x, y)$ is a white noise. In the previous situation the energy deposition from the soil can be considered as a white noise. Indeed, if we consider that the energy deposition from the mine is very low compared to the energy deposition from the soil, then the autocorrelation function of the noise is very similar to the autocorrelation function of the collimated image. Figure 7 displays this autocorrelation function which is very close to the autocorreleation function of a white noise, a Dirac function.

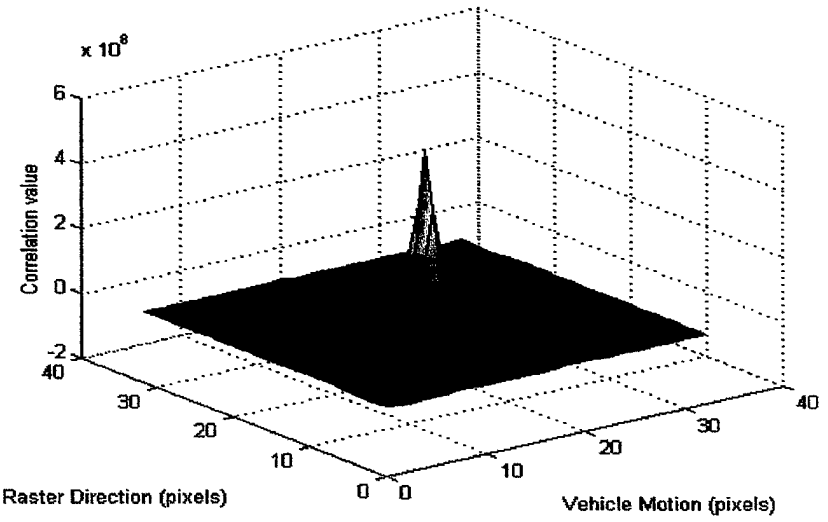


Figure 7: Autocorrelation function of the collimated image

However, this condition is not always true, and so the filter we use may sometimes not be the optimal filter. So in this case we must compute the optimal filter without making this assumption. To avoid resolving the difficult equation (1), a solution to is to bypass this difficulty by using a prewhitening step.

The prewhitening step consists of a filter, $h(x,y)$, that 'whitens' the noise $n(x,y)$, before filtering the collimated signal $c(x,y)$.

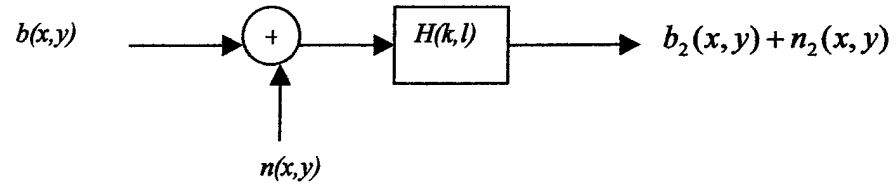


Figure 8: Prewhitening system

Several filters have been developed to whiten a signal. The one we used in this study is the one of L.P. Yaroslavsky [4].

If we let $\alpha(k,l)$ be the Fourier Transform of the collimated image where all the pixels in part occupied by the buried objects have been put to zero, then the whitening filter frequency response is:

$$H(k,l)=\frac{1}{|\alpha(k,l)|}$$

Then what we maximize is:

$$\chi^2(x_0,y_0)=\frac{Power_{b_2}(x_0,y_0)}{Power_{n_2}(x_0,y_0)},$$ where the noise $n_2(x,y)$ is a white noise.

Using the previous method we can easily compute the optimal filter that separates $b_2(x,y)$ from the noise. However, we are not interested in $b_2(x,y)$, which is a filtered image of $b(x,y)$ by $h(x,y)$, but in $b(x,y)$. So we can add a last step which is the inverse filtering $h^{-1}(x,y)$. Thus, the global filtering system is:

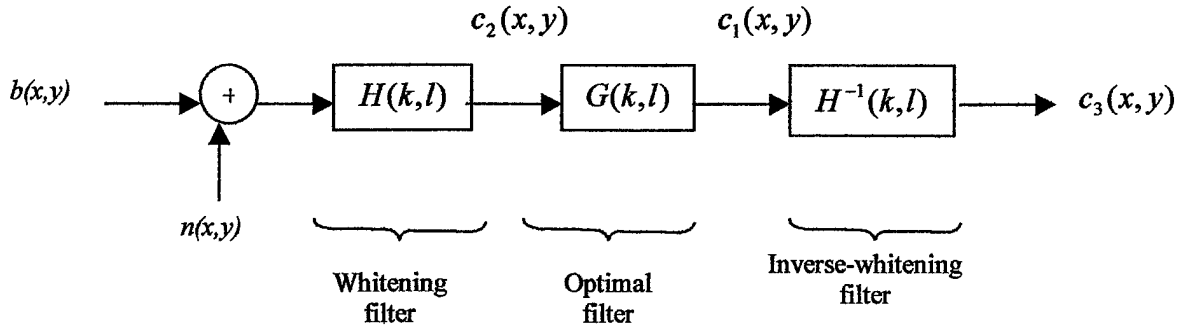


Figure 9: Global optimal filtering system with prewhitening and inverse whitening step.

In this case $g(x,y)$ is the filter that optimizes $\chi^2{}_{b_2 b_2}$, but we are not sure that it is the filter that maximizes the signal-to-noise ratio after the last inverse-filtering step.

This method gives better results than the previous one without the prewhitening step when the noise is not white. This is shown in figures 10b and 10c.

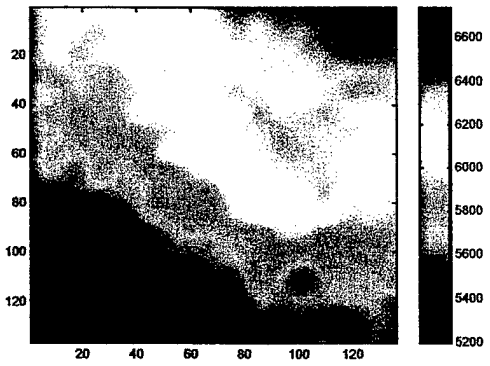


Figure 10a: Collimated image. Plastic mine centered at pixel (95,70) with a DOB of 3 inch.

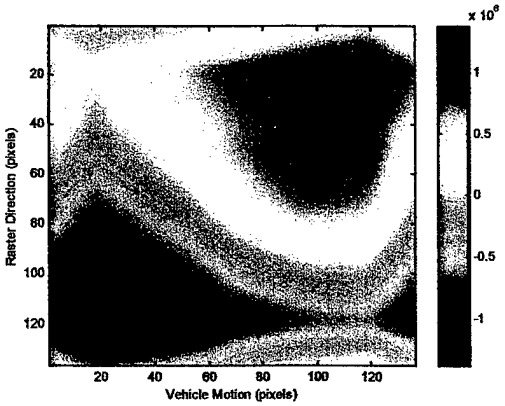


Figure 10b: Filtered image using the method without prewhitening step. (the cross marks the pixel where χ^2 has been maximized)

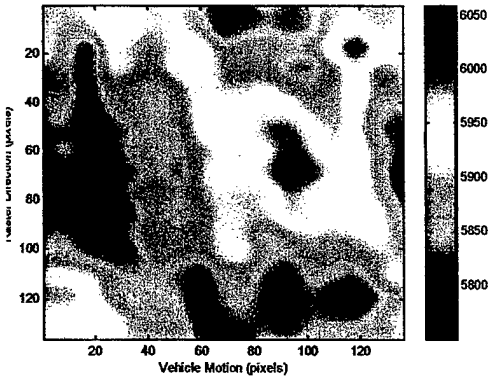


Figure 10c: Filtered image using the method with prewhitening step. (the cross marks the pixel where χ^2 has been maximized)

These two different methods of optimal filtering are able to point out some buried objects that were not visible in the original collimated image. The choice of the method (prewhitening or not prewhitening) mainly depends on the nature of the noise. However, the method without the prewhitening step appears to be very sensitive to the nature of the noise.

4- Conclusion

In this section we first introduced some signal separation techniques which failed in producing images containing only buried objects because the assumption of the statistical independence of the source signals is not true. Next we studied an optimal filter which produced some interesting results because it is able to point out buried objects even when they are deep-buried.

The last technique of optimal filtering required the user to select a region where a mine seemed to be buried. In the next section we introduce a new technique of image processing, neural networks, that could help us to move toward an automatic filtering process of the images.

Section 5

Neural Network

1-Introduction

The human brain is a very complex system capable of thinking, remembering and problem solving using neurons. Some studies have shown that the brain stores information as patterns. Some of these patterns are very complicated and give us the ability to recognize objects from many different angles. This process of storing information as patterns, utilizing those patterns, and then solving problems, encompasses a new field in computing that we use in this section to improve LMR images with the goal of deciding whether or not there is a buried mine.

2- From biological neurons to artificial neurons

2-1- Biological neurons

The individual neurons are complicated. They have a myriad of parts, sub-system, and control mechanisms, and there are over one hundred different classes. We will focus only on the simplest neuron.

This neuron is a simple processing element that receives and combines signals from other neurons through input paths called dendrites. If the combined input signal is strong enough, the neuron 'fires', producing an output signal along the axon that connects to the dendrites of many others neurons.

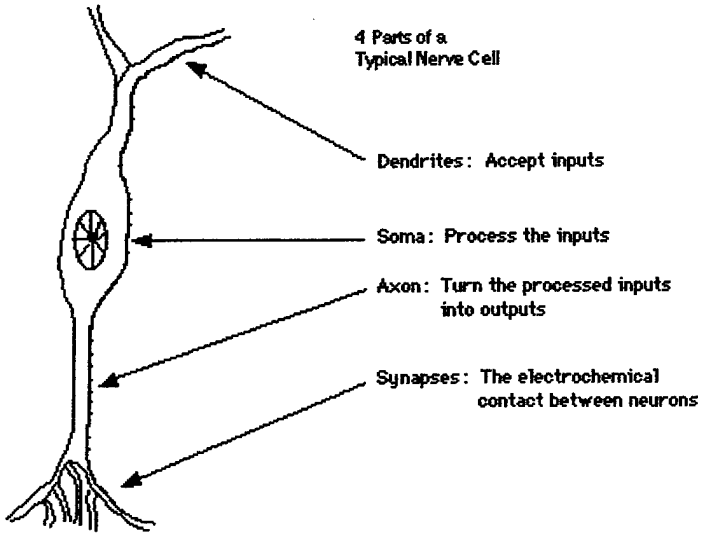


Figure 1: A simple neuron

2-2- Artificial neurons

An artificial neuron tries to imitate the previous biological neuron. Figure 2 shows the schematic representation of an artificial neuron.

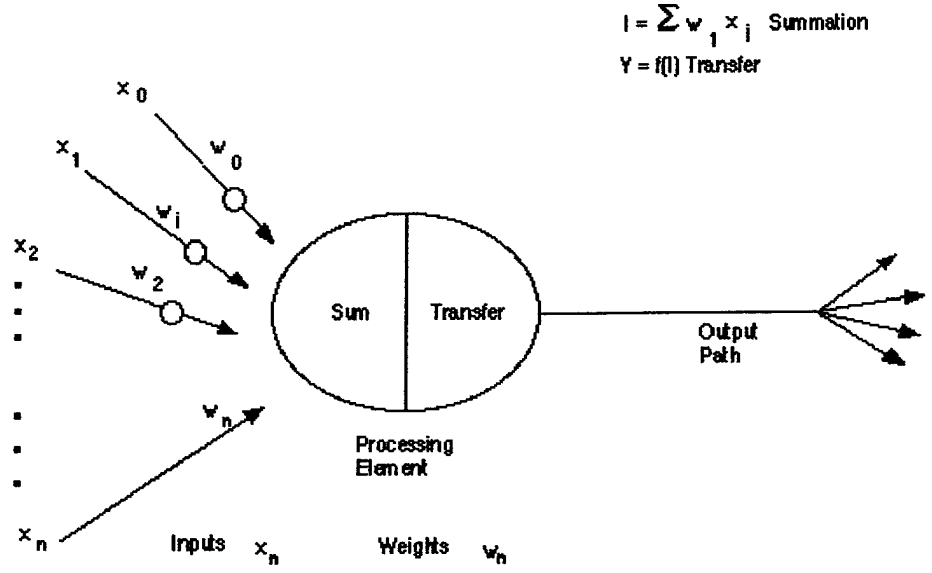


Figure 2: A basic artificial neuron.

The input signals are represented by $x_0, x_1, x_2, ..., x_n$; these signals are continuous variables. Each of the inputs is modified by a weight, $w_0, w_1, w_2, ..., w_n$, whose function is analogous to that of the synaptic junction in a biological neuron. This artificial processing element consists of two parts: The first simply sums the weighted inputs resulting in a quantity *I*. The second part is a nonlinear filter, called the activation function, through which the signal flows. Figure 3 shows a typical activation function.

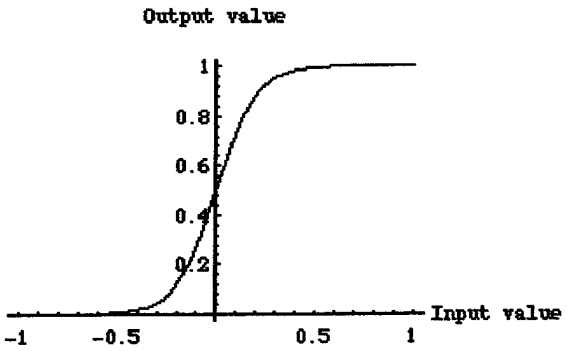


Figure 3: A typical activation function.

$\Phi(I) = \frac{1}{1+e^{-\alpha.I}}$, where α is a coefficient that adjusts the abruptness of this function as it changes between the two asymptotic values.

3- Artificial neural networks

3-1- Architecture of artificial neural networks

An artificial neural network can be defined as:

A data processing system consisting of a large number of simple, highly interconnected neurons in an architecture inspired by the structure of the cerebral cortex of the brain.

These processing elements (neurons) are usually organized into a sequence of layers with full connections between the layers. This arrangement is shown in Figure 4, where the input layer is a buffer that presents data to the network. The top layer is the output layer

which presents the output response to a given input. The other layers are called hidden layers because they usually have no connection to the outside world.

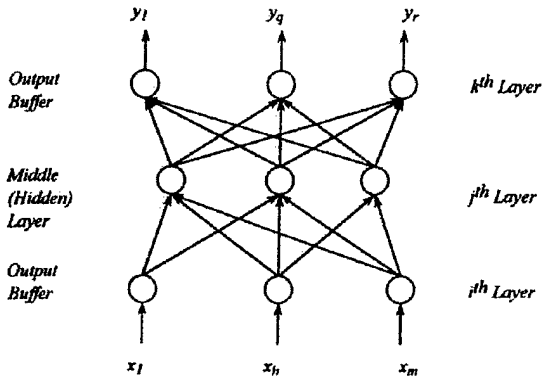


Figure 4: Example of a neural network architecture.

Two general kinds of neural networks are in use: the heteroassociative neural network in which the output vector is different than the input vector, and the autoassociative neural network in which the output is identical to the input.

A typical neural network is 'fully connected', which means that there is a connection between each of the neurons in any given layer with each of the neurons in the next layer as shown in Figure 5. However, this is not the only kind of neural network; others have have been developed, like the feedback neural network where the output of one layer routes back to a previous layer. This one is shown in Figure 6. In our study we will use a heteroassociative, feedforward neural network.

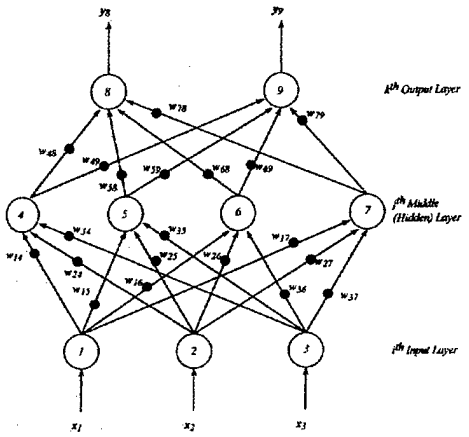


Figure 5: Simple feedforward neural network

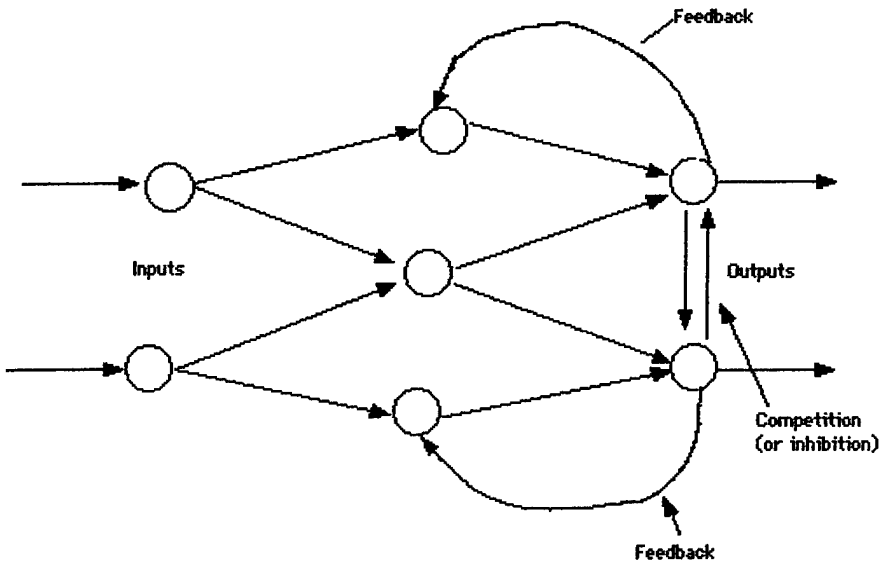


Figure 6: Simple network with feedback and competition.

3-2- Training an artificial neural network

Artificial neural networks perform two major functions: learning and recall. Learning is the process of adapting the connection weights in an artificial neural network to produce the desired output vector in response to a stimulus vector presented to the input buffer. Recall is the process of accepting an input stimulus and producing an output response in accordance with the network weight structure.

The learning rules of neural computation indicate how connection weights are adjusted in response to a learning example. In *supervised learning*, the artificial neural network is trained to give the desired response to a specific input stimulus. In *unsupervised learning* there is no specific response sought, but rather the response is based on the network's ability to organize itself. Only the input stimuli are applied to the input buffers of the networks. The network then organizes itself internally so that each neuron responds strongly to a different set of input stimuli. These sets of input stimuli represent clusters in the input space.

In our application of neural networks to LMR images we will use the supervised learning technique.

3-2-1- Supervised learning

The goal of supervised learning is to train the neural network to give a desired response to a specific input stimulus. We will call X the input learning vector, Z the output learning vector and Y the computed output of the neural network. Figure 7 shows the principle of the supervised learning technique.

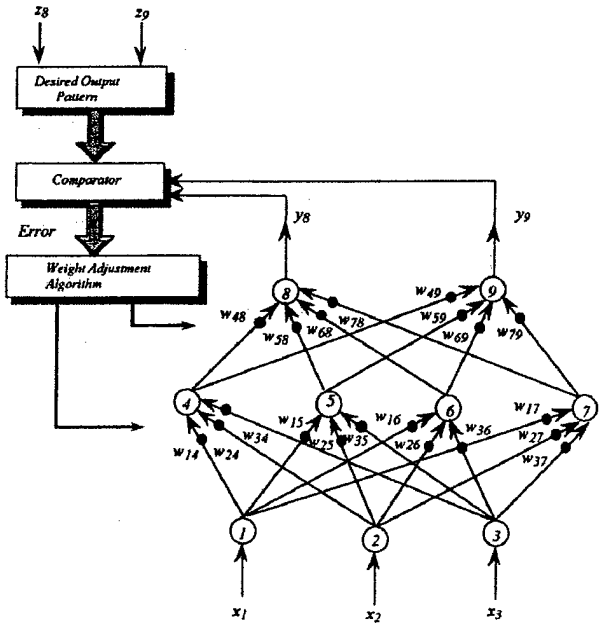


Figure 7: A neural network with supervised learning.

To start the learning process, we first randomly initialized all the coefficients w_{ij} of each layer and computed the first output of the network *Y*. The inputs to the comparator are the desired output vector *Z* and the actual output pattern *Y*. The error coming from the comparator is then utilized in the weights-adjusting algorithm to determine the amount of the adjustment to be made in the weights in each layers. This is an error-backpropagation process. This process occurs over and over until the error reaches a desired value.

3-2-2 Backpropagation training for a multilayer neural network

The error used in the backpropagation learning process is the square error defined by:

$$Error = \varepsilon^2 = [Z - Y]^2$$

The method used to decrease this error, modifying the weights of the neural network, is the *delta rule*. This rule indicates that the change in a weight is proportional to the rate of the square error, ε^2 with respect to that weight:

$$\Delta w_{pq,r} = -\eta \frac{\partial \varepsilon^2}{\partial w_{pq,r}}$$

In our study we used some neural networks with 2 and 3 layers. To understand how this delta rule is applied we will study a 2 layer neural network (the 3 layer neural network is very similar). Figure 8 represents a 2 layer neural network.

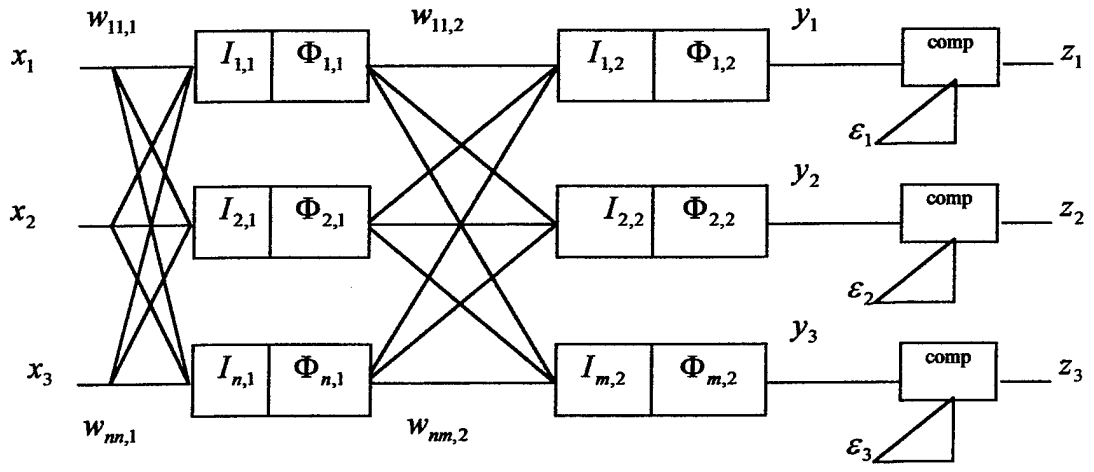


Figure 8: Sketch of multilayer network showing the symbols and indices used in deriving the backpropagation training algorithm.

$I_{i,j}$: result of the summation of the weighted inputs for the neuron i of the layer j.

$\Phi_{i,j} = f(I_{i,j})$, where f is the activation function.

$w_{ki,j}$: weight of the output of neuron k, layer j-1, sent to neuron i, layer j.

3-2-2-1- Calculation of weights for the output-layer neurons

The error of the neuron q of the output layer is:

$$\varepsilon^2 = \varepsilon^2{}_q = \left[z_q - \Phi_{q,2}\right]^2$$

The delta rule indicates that the change in a weight is proportional to the rate of the square of the error with respect to that weight, that is:

$$\Delta w_{pq,2} = -\eta_{p,q} \frac{\partial \varepsilon^2{}_q}{\partial w_{pq,2}}$$

where $\eta_{p,q}$ is a constant of proportionality called the learning rate. To evaluate this partial derivative, we use the chain rule of differentiation:

$$\frac{\partial \varepsilon^2{}_q}{\partial w_{pq,2}} = \frac{\partial \varepsilon^2{}_q}{\partial \Phi_{q,2}} \frac{\partial \Phi_{q,2}}{\partial I_{q,2}} \frac{\partial I_{q,2}}{\partial w_{pq,2}}$$

Then we can show that:

$$\Delta w_{pq,2} = -\eta_{p,q} \frac{\partial \varepsilon^2{}_q}{\partial w_{pq,2}} = -\eta_{p,q}.2\alpha\left[z_q - \Phi_{q,2}\right]\Phi_{q,2}\left[1 - \Phi_{q,2}\right]\Phi_{p,1}$$

$$w_{pq,2}(N+1) = w_{pq,2}(N) + \Delta w_{pq,2}$$

where N is the number of the iteration involved, and α the coefficient of the activation function. An identical process is performed for each weight of the output layer to give the adjusted values of the weights.

3-2-2-2- Calculation of weights for the hidden layer neurons

Since the hidden neurons have no target vectors the problem is a bit different. We must use the errors of all the output neurons:

$$\varepsilon^2 = \sum_{q=1}^{n} \varepsilon_q{}^2$$

So using the delta rule:

$$\Delta w_{hp,1} = -\eta_{h,p} \frac{\partial \varepsilon^2}{\partial w_{hp,1}} = -\eta_{h,p} \sum_{q=1}^{n} \frac{\partial \varepsilon_q{}^2}{\partial w_{hp,1}}$$

And then using once more the chain rule of differentiation we can show:

$$\Delta w_{hp,1} = \eta_{h,p} \sum_{q=1}^{n} 2\alpha(z_q - \Phi_{q,2})\left[\Phi_{q,2}(1-\Phi_{q,21})\right] w_{pq,2}\, \alpha\left[\Phi_{p,1}(1-\Phi_{p,1})\right] x_h$$

$$w_{hp,1}(N+1) = w_{hp,1}(N) + \Delta w_{hp,1}$$

4- Application of neural networks to LMR images

As a human, looking at both collimated and uncollimated images, we are able to identify surface-laid objects. So we would like to teach the neural network to recognize surface-laid objects and remove them from collimated images. Thus, during the learning step we will present to the network some original input images containing buried mines and surface-laid objects, and the corresponding output collimated images from which the surface-laid objects have been removed.

4-1- Build the input and the output

4-1-1- Nature of the input and the output

We want the neural network to learn how to remove surface-laid objects using collimated and uncollimated image properties (shift in the position of the buried objects in the collimated images and information about the surface-laid objects in the uncollimated images). Information in both uncollimated images are very similar. Because the larger the input vector is, the more complicated the neural network is, we will only use one of the uncollimated images in the input vector. So the input vector will contain information from both collimated images and from one of the uncollimated images.

The corresponding output vector is an image of the exact same mine as for the input image, at the same location and the same DOB, but without any surface-laid object, or without any slope.

During the learning step we will present to the system some images containing some surface-laid objects and some buried mines at certain positions. However, we want the network to be able to give good results wherever the objects are. So we will not present to the system all of the images, but some region of the images, so that the input contains information about the local variation of the energy but no information about the position of the object. We used some 3*3 or 5*5 pixel square regions. So the input vector contains the values of the pixels of 3*3 pixel square regions, centered at the same position on both collimated images and on the uncollimated image.

To use the information from the neighbor pixels, we associated to the input vector a single output pixel which represents the value of the pixel of the output image at the same location as the center of the squares. Figure 9 represents the input and the output vectors of the neural network.

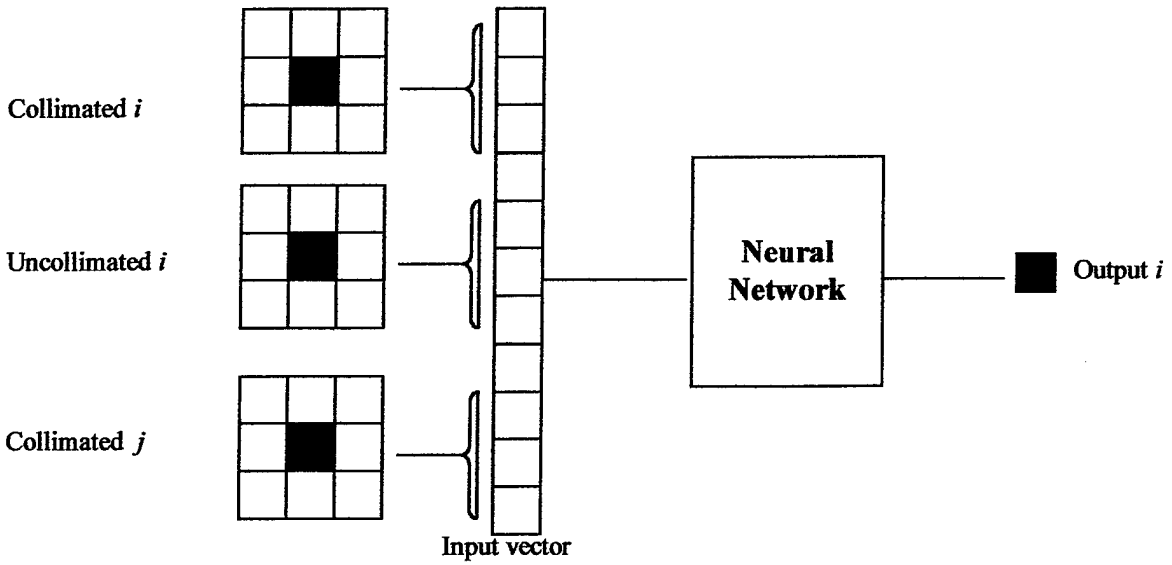


Figure 9: Representation of the input and the output of the neural network (*i* and *j* mean the front and the rear).

4-1-2- Rescale the input and the output vectors

The energy deposition in the detectors depends on different factors (nature of the soil, kind of mine,...etc.), so to design a network able to achieve good results in every situation, we have to rescale the input and the output images so that each image contains values included in the same range so that the neural network could be used in as many cases as possible.

We used the following normalization method:

$$I_{norm}(x,y) = \frac{(I(x,y) - E_{soil}) / E_{soil}}{Max_{x,y}(I_1(x,y))}, \text{ where } I_1(x,y) = \frac{(I(x,y) - E_{soil})}{E_{soil}}$$

where $I(x,y)$ is the original image and E_{soil} ,the energy deposition from the soil, is supposed to be the energy deposition most represented in the image which we extract by looking at the histogram of the image.

This normalization method computes the relative enegy value of each pixel compared to the energy of the soil, and then divides by the maximum relative value for all of the pixels.

4-2- Set the parameters of the network

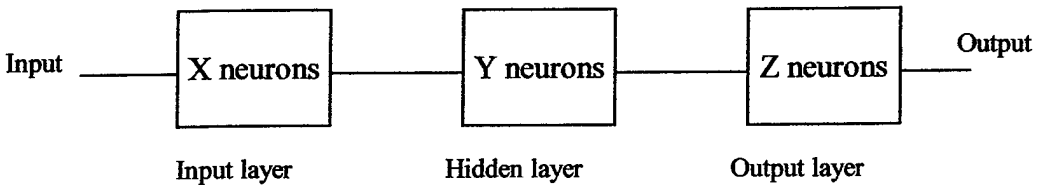
When the architecture of the neural network has been chosen we have to choose the number of layers, the number of neurons in each layer, and the activation function.

4-2-1- number of layers and neurons

In this study we arbitrarily decided to use a 3 layer neural network. So now we have to choose the number of neurons in each layer.

As the number of neurons increases, the performance of the system increases; but the number of neurons is limited by the number of learning examples. Indeed, it has been proven that to design a system that is able to generalize what it has learned from examples, the number of neurons must be about 10 times lower that the number of learning vectors.

The system can been represented the following way:



If the size of the input vector is: num_row x num_col, then the condition to be satisfied for the number of neurons is:

$$num\_col \geq 10.(X.Y.Z)$$

4-2-2- The activation function

To choose the activation function we have to take into account the characteristics of the normalized images which contain pixels into the interval [-1;1] where 0 corresponds to the energy deposition due to the soil. So to keep values included into the interval [-1;1] and centered around 0 we use the activation function:

$$\Phi(x) = \frac{2}{\pi}\arctan(\alpha.x)$$

where α is the coefficient of the activation function which can be set to different values.

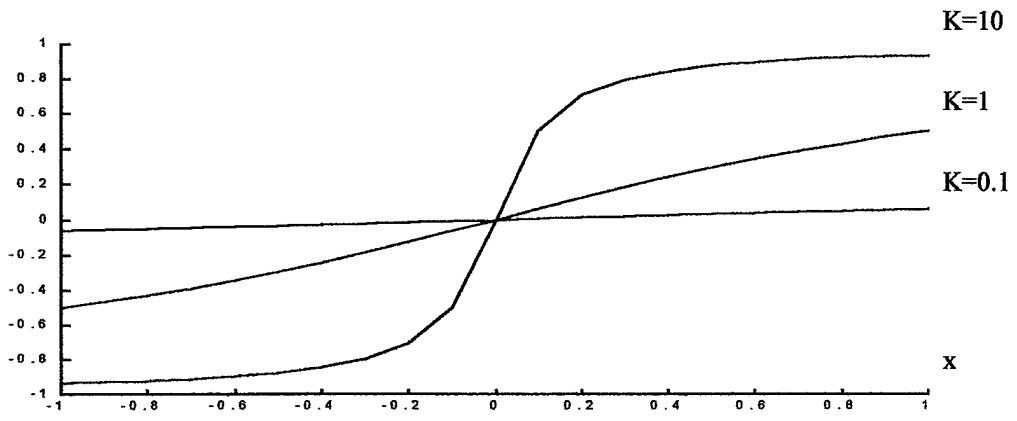


Figure 11: Activation function for different values of α.

So increasing α allows the neural network to be very selective, while decreasing α causes the activation function to become less sensitive to the variation of x.

4-3- Choice of the learning examples

The choice of the learning input and output vectors is very important because the performances of the system will mostly depend on it. We used three sets of images:

Image set 1: surrogate plastic mine with a DOB of 1". Surface laid objects: triangular wood piece, soda can, bolt, nut.

Image set 2: surrogate steel mine with a DOB of 1". Surface laid objects: 2 pieces of wood and a rock.

Image set 3: surrogate plastic mine with a DOB of 1". Surface laid objects: 2 pieces of wood and 2 rocks.

These 3 image sets provide 2147 learning input vectors. Figures 9a, b, c, d represent Image set 1.

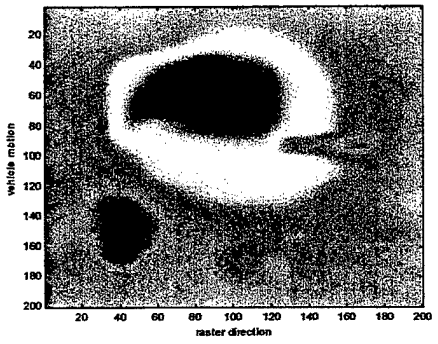


Figure 9a: Input rear collimated image

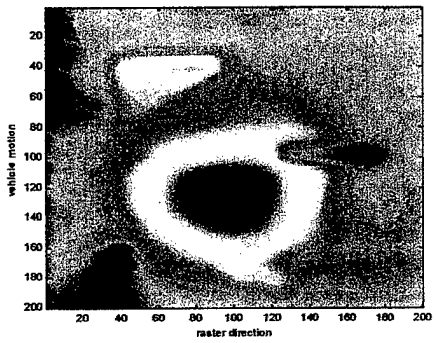


Figure 9b: Input front collimated image

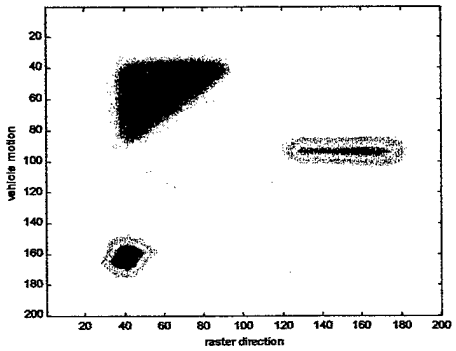


Figure 9c: Input rear uncollimated image

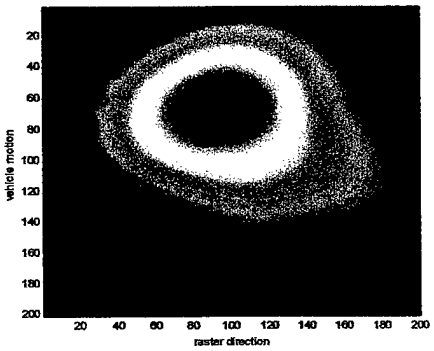


Figure 9d: Output rear collimated image

Figure 9a, b, c, d: Example of a typical learning set of images

5- Experimental results

To test this method we have trained the network under different conditions, using different sizes of the input vectors [(3*3) pixels or (5*5) pixels from each of the three input images], different number of neurons, different values for the α coefficient of the activation function, and different learning images from the three previous sets.

Because a backward shifting appears in the front collimated image and a forward shifting appears in the rear collimated image, we trained two different networks to teach them to

recognize the backward shifting for one and the forward shifting for the other. These networks are then used to respectively filter the front and the rear collimated images. The following results were obtained with the network trained to filter the rear collimated image.

5-1- Size of the selection square

Using a 5*5 square pixel to take into account the neighborhood of a pixel appeared to give better results, which tends to prove that using information from neighbor pixels is an interesting idea. However, the drawback of such a method is that the network output associated with an n*n input image is a (n-2)*(n-2) image.

5-2- Influence of the number of neurons

When using a large number of neurons (more than the number of learning vectors), the system computes very good output images when applied to one of the learned images, but fails to provide good results for non-learned images, as shown in Figures 10 and 11. In this example we used a neural network containing 75 neurons in the first layer, 50 in the second layer, and 1 in the output layer (we call it 75*50*1 network). The total number of neurons is 9425 which is larger than the number of learning vectors (2147).

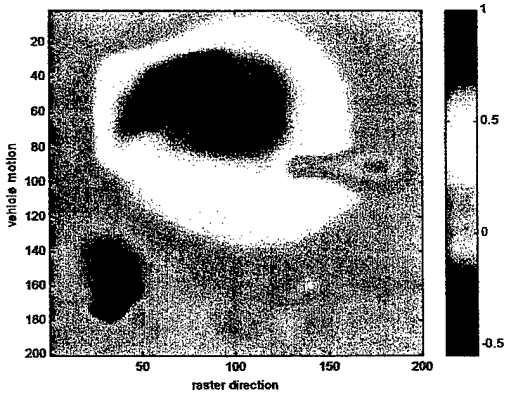


Figure 10a: Input rear collimated image

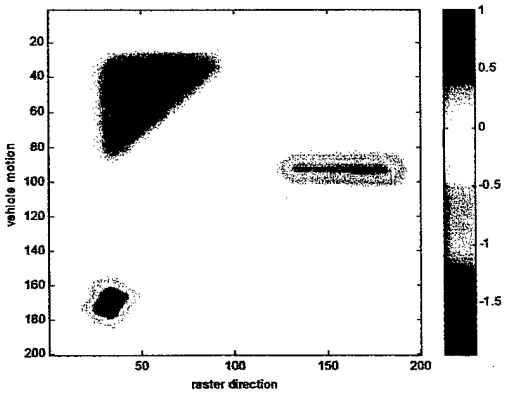


Figure 10b: Input rear uncollimated image

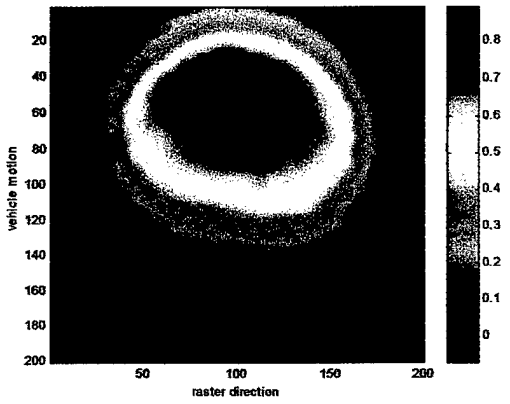


Figure 10c: Output rear collimated image

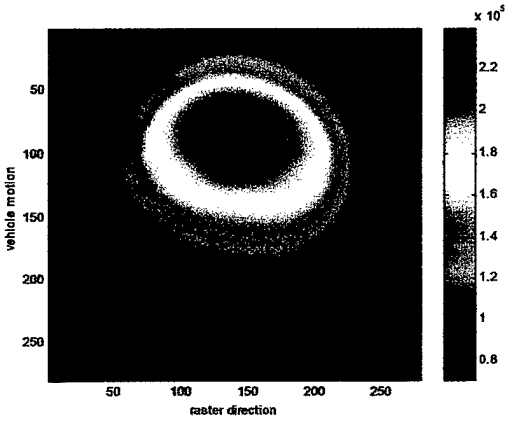


Figure 10d: Theoretical output rear collimated image used during the learning step

Figures 10: Inputs and output of the neural network, when using a set of images presented to the system during the learning step.

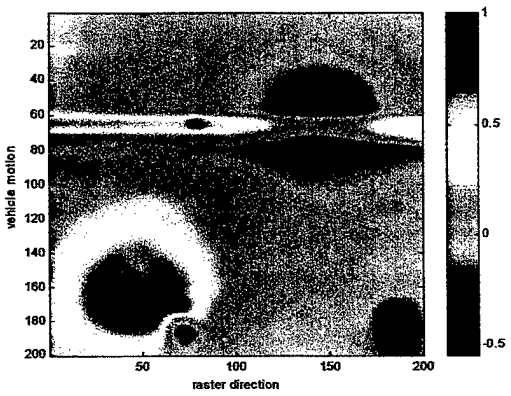


Figure 11a: Input rear collimated image

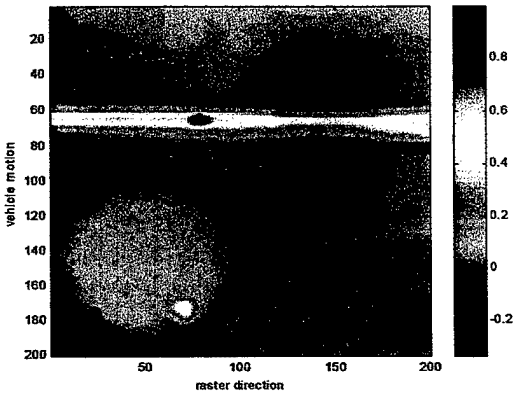


Figure 11b: Input rear uncollimated image

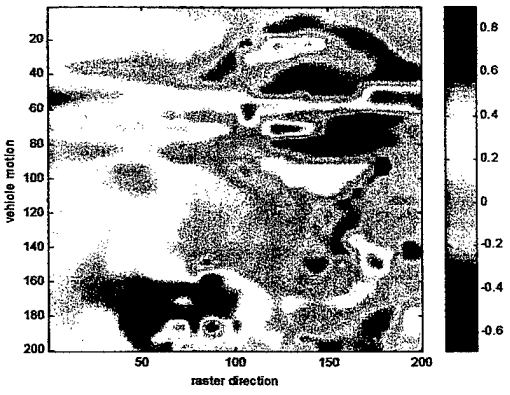


Figure 11c: Output rear collimated image

Figures 11: Inputs and output images of the neural network, when applied to a non-learned set of images

To get a network to be able to compute good results in every situations we need to verify the condition $num\_col \geq 10.(X.Y.Z)$. Thus, we used a 5*3*1 network (five neurons in the input layer, 3 in the hidden layer and one in the output layer) and a 3*2*1 network. These networks do not provide as good results as the previous one when the input is a learned-image, however, they are better when applied to non-learned vectors.

We need the total number of neurons to be less than the number of learning vectors, but we also need enough neurons to model the system. This is the reason why the 5*3*1 network gives better results than the 3*2*1 network.

5-3- Influence of the α coefficient

This coefficient sets the way the output of a neuron reacts to an input signal. The bigger this coefficient is, the more difference a neuron makes between two values close to 0 and the less difference it makes between two larger values. The lower this coefficient is, the less difference a neuron makes between two values close to 0. Experimentally it appears that 0.5 is a good value for this coefficient.

5-4- Final results

Finally the neural network used was a three layer, 5*3*1 network, with an α coefficient of the activation function equal to 0.5, a 3*3 pixel selection square, and using image sets 1, 2 and 3, or the image sets 1 and 3 as learning images.

Figures 12,13, 14, 15 and 16 show the outputs of the network when different images are applied.

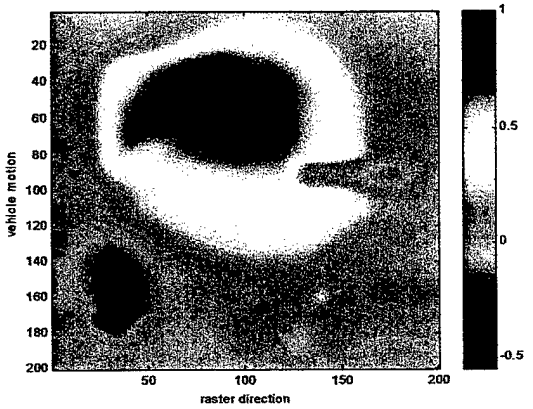


Figure 12a: Input rear collimated image

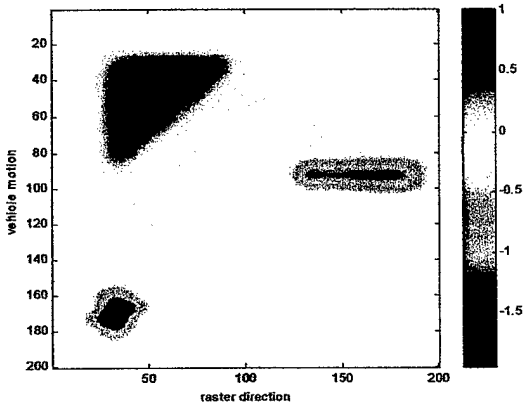


Figure 12b: Input rear uncollimated image

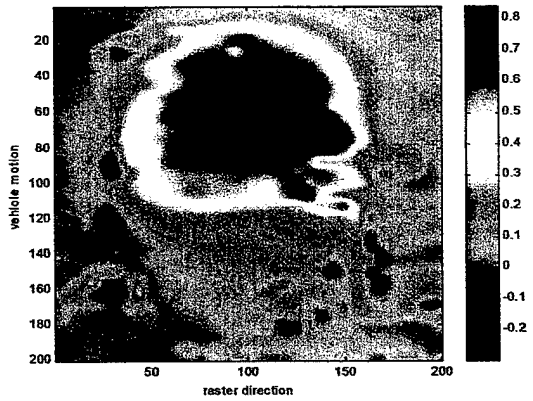


Figure 12c: Output rear collimated image

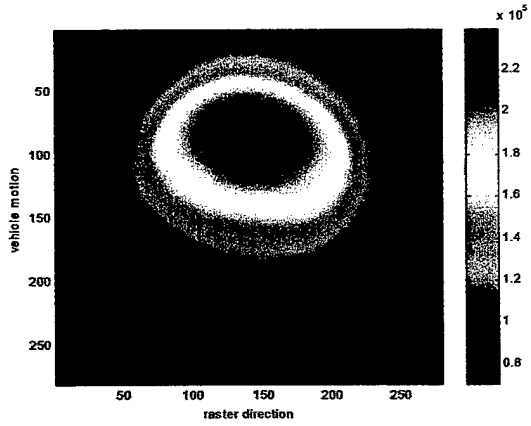


Figure 12d: Theoretical output rear collimated image used during the training step

Figures 12: Inputs and output of the neural network, when using a set of images presented to the system during the learning step.
The learning images are made of the image sets 1,2 and 3.

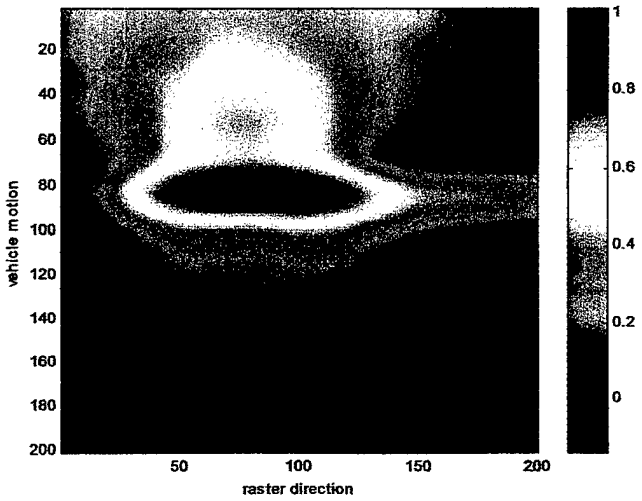


Figure 13a: Input rear collimated image

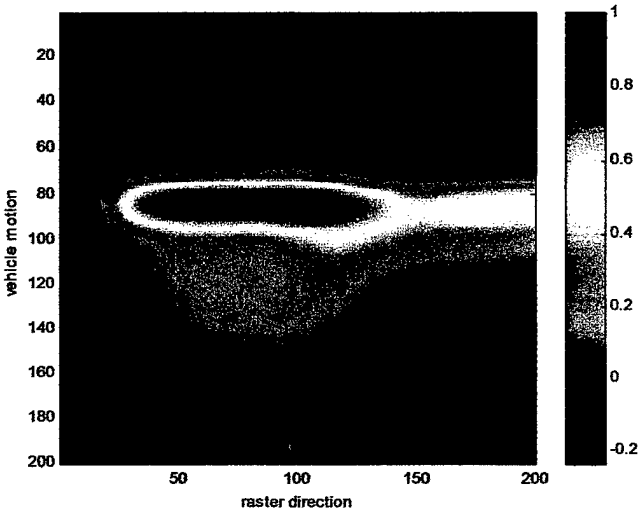


Figure 13b: Input rear uncollimated image

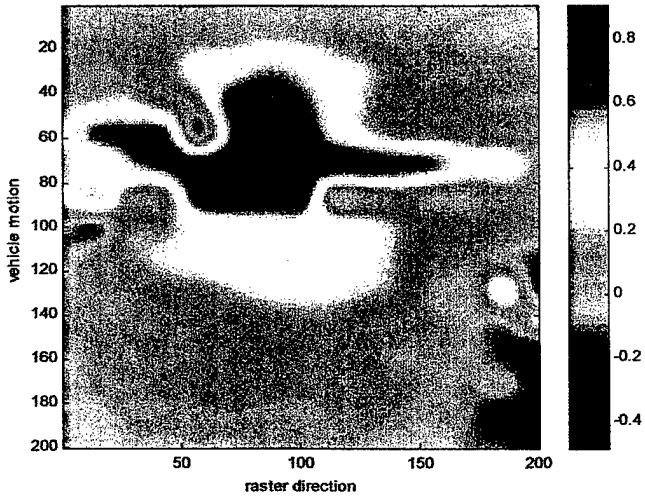


Figure 13c: Output rear collimated image

Figures 13: Inputs and output images of the neural network,
when applied to a non-learned set of images.
The learning images are made of the image sets 1, 2 and 3.

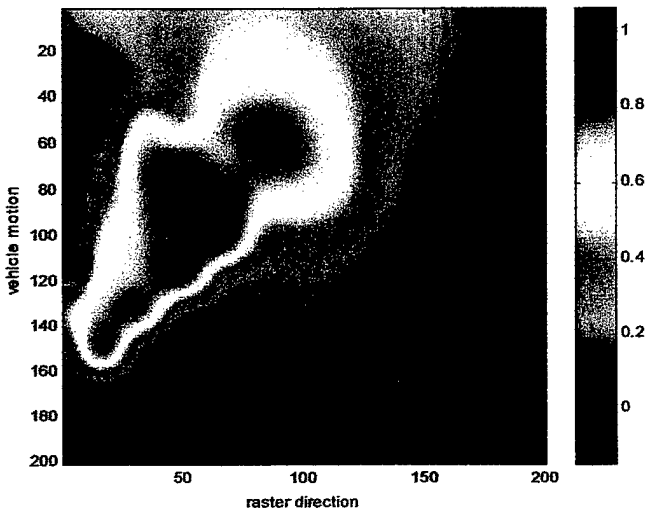


Figure 14a: Input rear collimated image

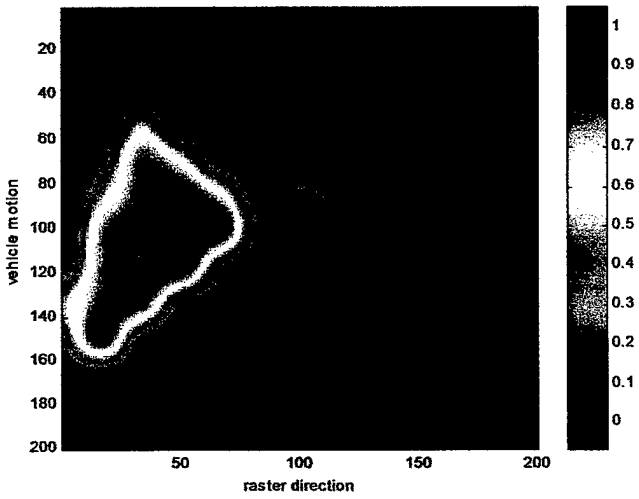


Figure 14b: Input rear uncollimated image

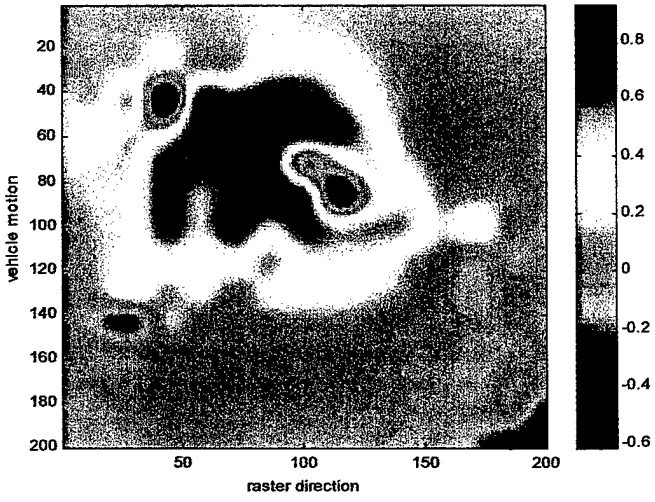


Figure 14c: Output rear collimated image

Figures 14: Inputs and output images of the neural network,
when applied to a non-learned set of images.
The learning set is made of the image sets 1, 2 and 3.

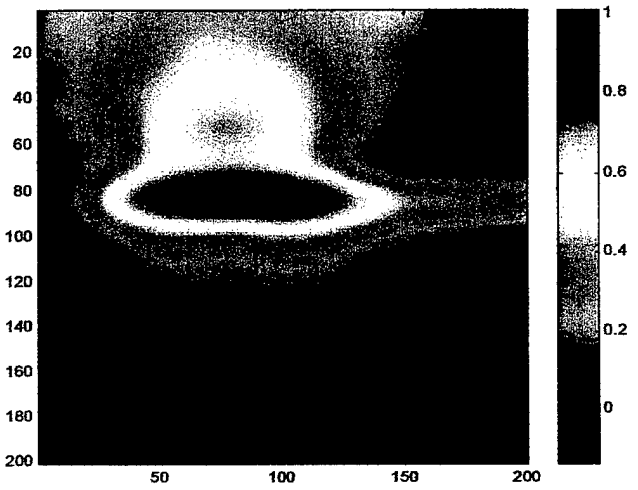


Figure 15a:Input rear collimated image

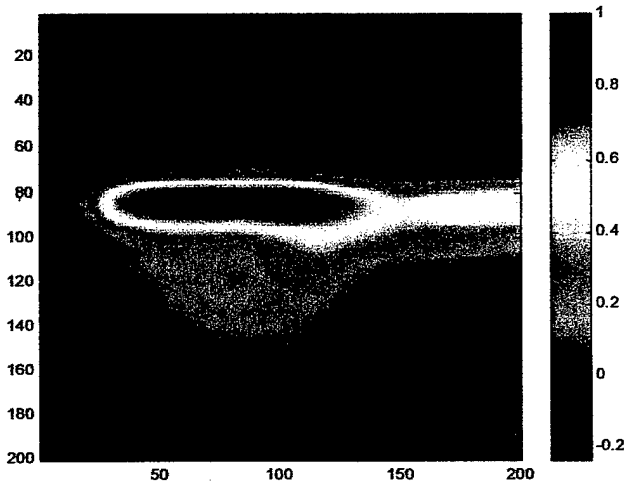


Figure 15b: Input rear uncollimated image

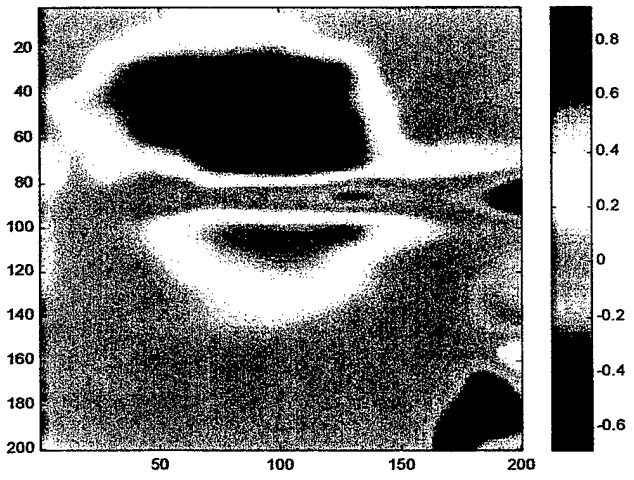


Figure 15c: Output rear uncollimated image

Figures 15: Inputs and output images of the neural network,
when applied to a non-learned set of images.
The learning set is made of the image sets 1 and 3.

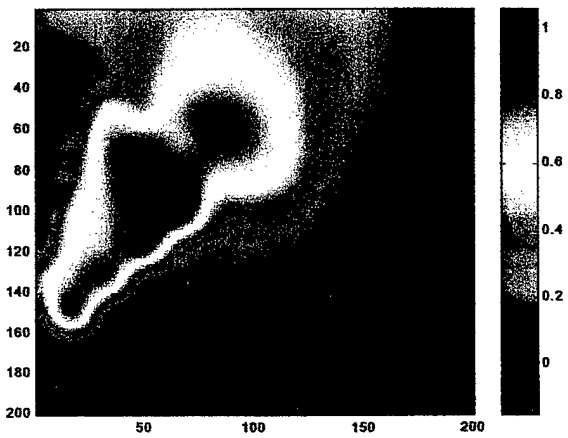


Figure 16a: Input rear collimated image

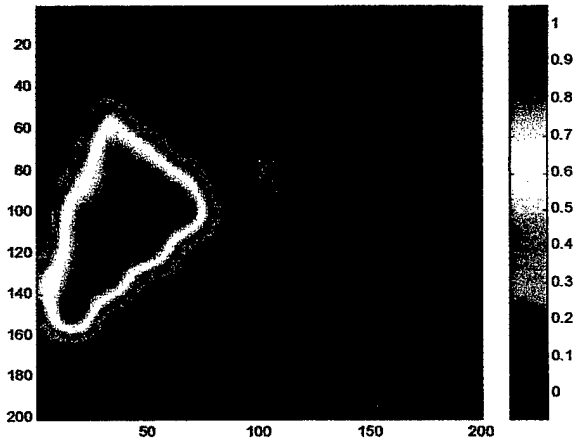


Figure 16b: Input rear uncollimated image

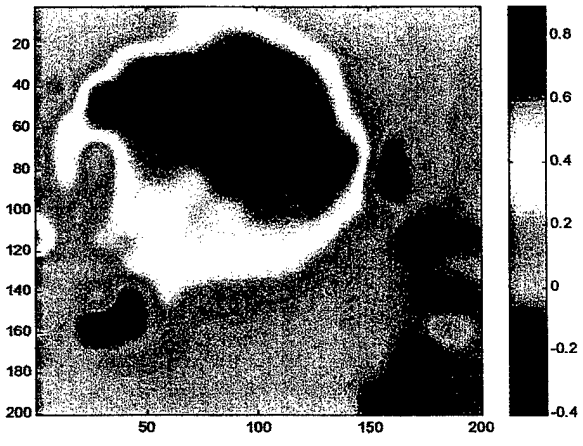


Figure 16c: Output rear collimated image

Figures 16: Inputs and output images of the neural network,
when applied to a non-learned set of images.
The learning set is made of the image sets 1 and 3.

So the network is able to remove the surface-laid objects when applied to non-learned images. However, the results seem to depend highly on the learning step. Indeed, we trained two different networks with the exact same internal characteristic but with two different kinds of learning images, one containing only the images of some plastic mines, another containing the images of plastic and metal mines. The network trained with the first set gives better results when applied to a plastic mine. This can be explained by the fact that the first network focuses on energy peaks, while the second network computes its internal weights to model both a peak and a depression of energies. So the first one is better when applied to plastic mines.

6- Conclusion

In this section we introduced the neural networks, their biological basis and their ability to learn. We designed a simple feedforward network in an attempt to remove surface-laid objects, using information from collimated and uncollimated images.

The results of this very first attempt, even if they are not perfect, are interesting because this system succeeded in removing surface-laid objects from unknown collimated images; we employed a very simple feedforward neural network and we used only three sets of images during the learning step.

For these reasons we continue to investigate using neural networks as a surface-laid object removal technique, and improve the system.

Three important elements to work on to improve the quality of the results are:

(1) the architecture of the network (a neural network with feedback would allow the system to compute the output using the neighbor pixels it has already calculated)

(2) the number of learning sets of images (the more information we give to the network during the learning step, the better the system is).

(3) The learning images. Computing different networks for specific situations (plastic mine, metal mine,...) could highly improve the quality of the results.

Section 6

Automatic Image Processing

The final goal of the signal and image processing techniques applied to the LMR, would be an automatic processing of the images. The introduction of the neural networks has been a first step in that direction. In this section we study this aspect more carefully.

1- Introduction

The neural network technique does not need human intervention during the process; the images are automatically processed. However, some other methods that have been previously developed need the user to select an object. So to achieve an automatic treatment in these cases, we need to design a technique that is able to recognize these objects; this is called object recognition.

2- Object recognition

2-1- Contour detection

Contour detection is the first task in the object recognition process. Numerous contour detection methods have been developed so far. Most of these methods are based on the

computation of the gradient of the image and on the use of a threshold. They are very efficient when applied to images with very sharp edges. However, in the case of LMR images this condition does not hold, because energy transition from the soil to a buried object is soft. Thus, these methods are not efficient when applied to LMR images, as shown in Figure 1b.

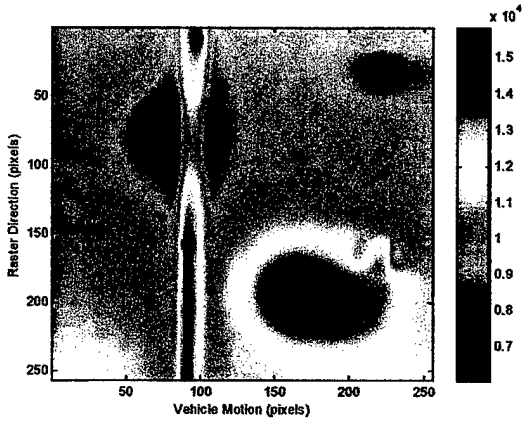


Figure 1a: Collimated image

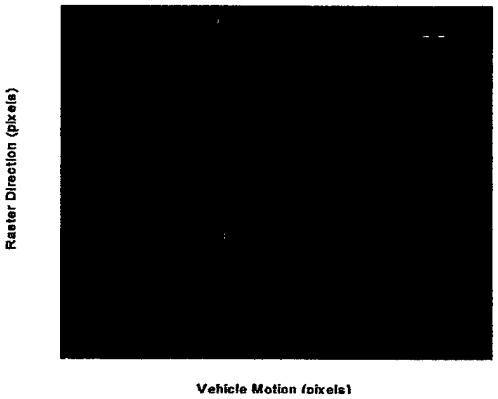


Figure 1b: Result of the Sobel contour detection method

Another method of contour detection would be the use of the energy level. Indeed, the contour of a mine is defined by a constant energy level. However, in this case, as shown in Figure 2, we have to choose the energy contour levels corresponding to the edge of a mine.

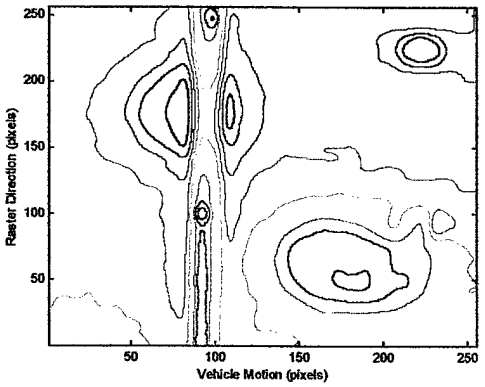


Figure 2: Energy level detection method (applied to Figure 1a image)

So these two contour detection methods are not very efficient when applied to LMR images.

A solution for this problem may be the use of the neural network. It might be possible to teach the network to recognize contours by providing it with the gradient function at each pixel location, and the value of the neighbor pixels.

2-2- Labeling task

Once the contours have been detected, the next step is to label each object and to then put into relation the objects in each image. This labeling task can be easily done by using connected neighborhood relations. At the end of this step we have an image where each pixel contains the number of the object it is part of (see Figure 3).

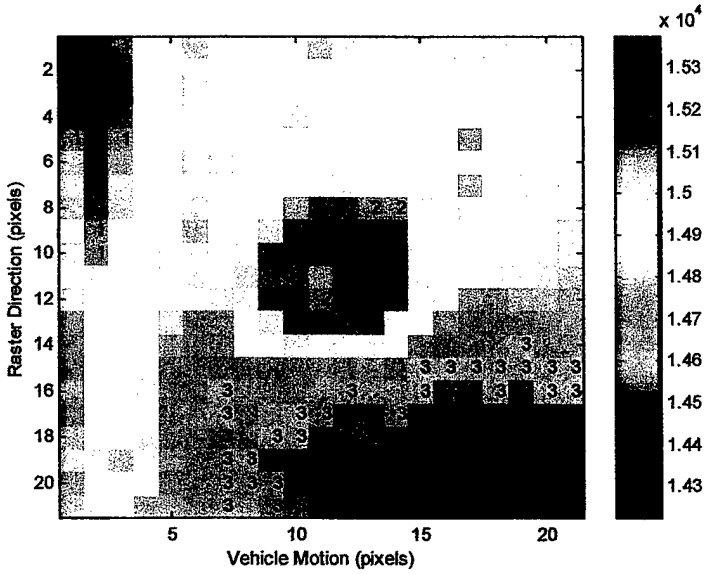


Figure 3: Resulting image of the labeling task.

3- Conclusion

The major difficulty in the process of object recognition is the contour detection task. It may be possible to use a neural network to do this.

An automatic object recognition method would allow us to develop new image processing techniques based on shape comparison of objects, symmetry between objects, object shifting, geometry, and so on. All these methods might give precious information that would help the user to decide whether or not there is a buried mine.

Section 7

User Interface

1-Description of user interface

Several signal and image processing techniques have been developed in this study that the user may want to use on site to improve the images. So we designed a user interface that allows the user to easily apply these filters to LMR images. To run this interface the user needs to go to the directory :\programs\image_filter, and run the program *main*.

This interface, programmed in the Matlab language, is made of a control window which contains three major control commands, as shown in Figure 1:

Images command: open, save, print and clear the desired image.

Filter command: perform the selected filtering on the selected image

Linear combination command: perform linear combination manipulation on the selected images.

Using this interface the user can easily apply successive filters to one image.

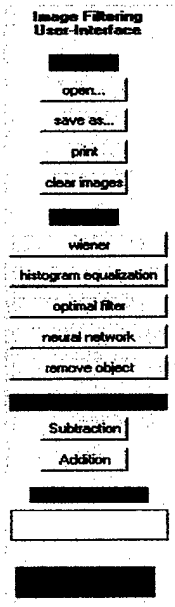


Figure 1: Command window of the user interface

2- Command description

In this subsection we describe each command. But each filter is based on the same scheme: select the filter, select the image to filter and follow the instructions displayed in the dialog box.

2-1- Image command

open... : opens a desired image of format *matfile* or *textfile*.

save as... : asks the user to select an image and then save it.

print : asks the user to select an image and then print it.

clear images : clear all the images.

2-2- Filter

wiener : asks the user to select an image and then apply the Wiener filter. This filter may be used as a first step to remove noise from the image before applying any other filter.

histogram equalization : asks the user to select an image and then equalize its histogram. This filter may be used to point out some objects which do not appear in the original image because their energy deposition is very low compared to the energy deposition of the soil.

optimal filter : asks the user to choose an image and to select a region with the mouse and then apply the optimal filter to that image. This filter maximizes the signal-to-noise ratio in the selected region, pointing out the buried object that may be in that zone.

neural network: asks first which collimated image the user wants to filter (front or rear), then ask the user to select 3 images (2 collimated and 1 uncollimated), and then apply the neural network. This filter may be used to remove surface laid objects, but also to point out some buried objects which weakly appear in the original collimated image.

remove object : asks the user to select a pair of collimated and uncollimated images, to choose the number of objects to remove from the selected collimated image, and then to select with the mouse the object. Finally, the program performs a subtraction-interpolation method to remove the objects. This method may be used to remove some surface-laid objects that may hide some buried objects.

2-3- Linear combination

subtraction : asks the user to select the subtraction factor and then the two images to subtract: normalization factor (noted '*norm*') performs:

$$output = image_1 - \frac{energy\ image_1}{energy\ image_2}.image_2$$

Selected value (noted '*value*') ask the user to enter the subtraction factor α and then performs:

$$output = image_1 - \alpha.image_2$$

This method can be used to remove surface-laid objects (select a collimated image, an uncollimated image and the normalization factor), or to point out a buried object which then appears as two different objects of opposite intensity and shifted in the vehicle motion direction (select the two collimated images and the normalization factor).

addition : ask the user to enter the value of the addition factor α, the two images to add, and then performs:

$$output = image_1 + \alpha.image_2 .$$

This method may be used to remove the energy deposition from the soil in some specific situations.

3- Architecture of the program

In this study of LMR signal and image processing techniques we used Matlab to test each method. We used this same software to design the user interface.

Each program contains at its very beginning a brief description of what it has been designed for. Figure 2 presents the architecture of the program.

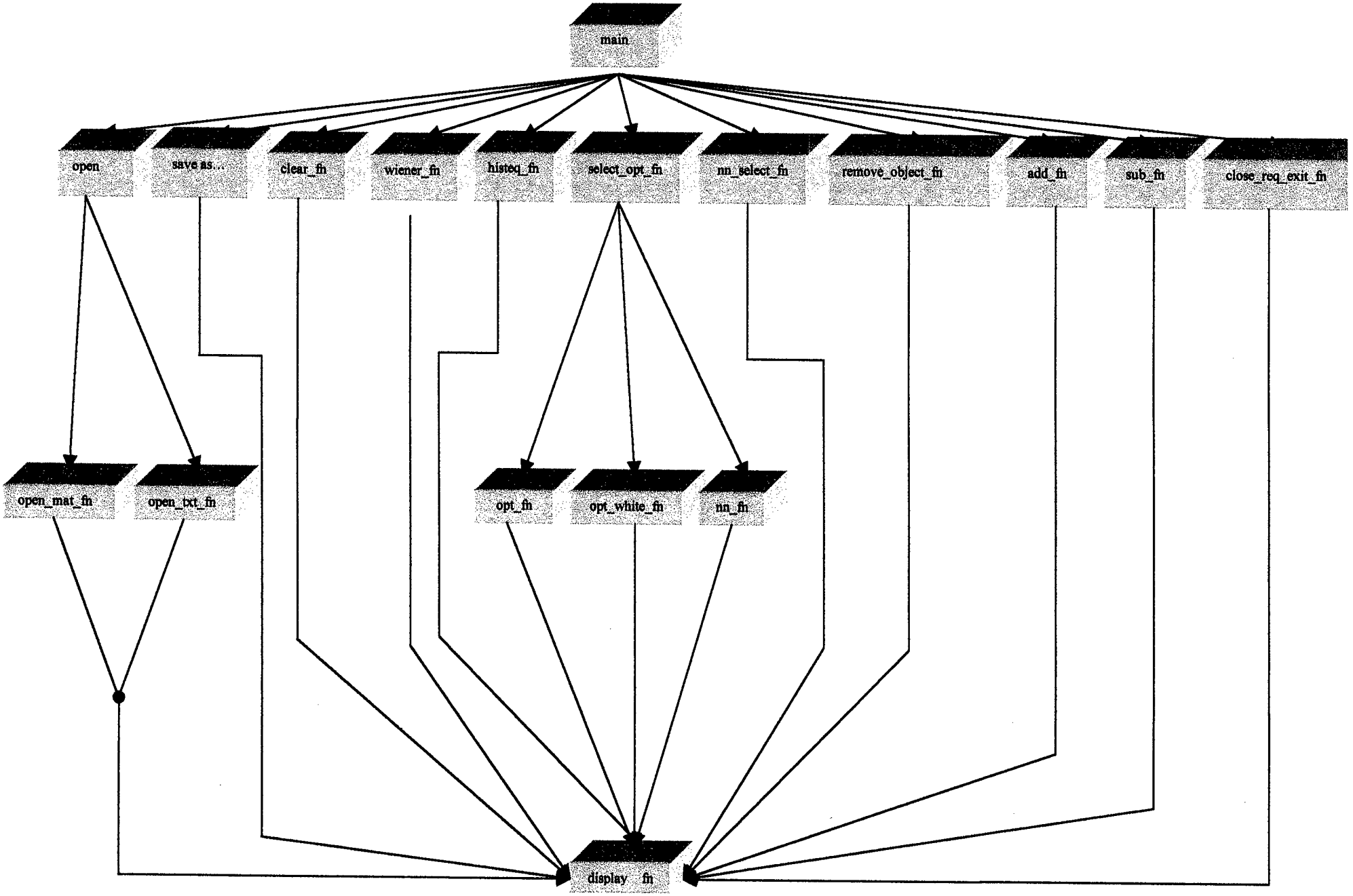


Figure 2: Architecture of the user interface program

Conclusion

In this report we studied different kinds of image filtering techniques applied to LMR images. We first introduced some image enhancement techniques that improve the visibility of the image and sometimes point out some details. Next, we looked at some linear combinations of the images which gave us information regarding the presence of a buried mine. Then, we introduced the optimal filter which is an efficient way to pick out mine from the soil, and at the end we studied the use of the neural network as a surface-laid object removal technique. The neural network method appears to be a very efficient method because even a very simple network, trained with very few images, gave good results. We will continue studying these networks, working on new architectures and acquiring new learning images, because the more images we use during the learning step the better the network is.

These image processing techniques sometimes give very useful information regarding the presence of a buried mine. This is why we will continue to investigate other techniques, especially object recognition techniques, that would lead to an automatic treatment of the images.

We should also look to develop neural networks that could be used to detect an object's contour, to compute the DOB of a mine, and/or to perform distortion removal (image enhancement) of distorted images.

References

1. J. Campbell, A. Jacobs, 'Detection of Buried Land Mines by Compton Backscatter Imaging', Nuclear Science and Engineering **110**, pp. 417-424, 1992.

2. C.J. Wells, Z. Su, A. Jacobs, E. Dugan, A. Allard 'Suitability of Simulated Land Mines for Detection Measurements Using X-ray Lateral Migration Radiography', SPIE Proceedings on Detection And Remediation Technologies for Mines and Minelike Targets V, Orlando, FL, April, 2000.

3. Herault, Jutten ,'Une solution neuromimetique au probleme de separation de sources', Traitement du signal, vol.5,No.1, 1988.

4. L.P. Yaroslavsky, *Digital Picture Processing, an introduction*, Springer-Verlag, pp. 246-256, 1985.