Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

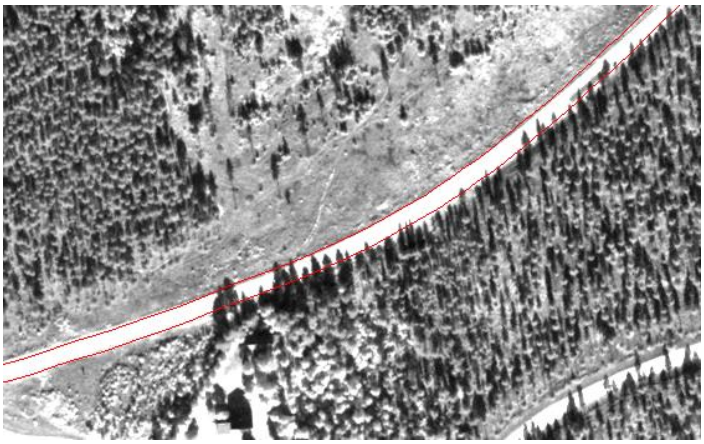
Note

SatTrafikk

Project Report 2010

Automatic road detection

Automatic cloud and cloud shadow detection



Note no SAMBA/57/10

Authors Arnt-Børre Salberg and Line Eikvil

Date Desember 2010

Contract JOP.12.10.2 (Norwegian Space Centre)

Norsk Regnesentral

Norsk Regnesentral (Norwegian Computing Center, NR) is a private, independent, non-profit foundation established in 1952. NR carries out contract research and development projects in the areas of information and communication technology and applied statistical modelling. The clients are a broad range of industrial, commercial and public service organizations in the national as well as the international market. Our scientific and technical capabilities are further developed in co-operation with The Research Council of Norway and key customers. The results of our projects may take the form of reports, software, prototypes, and short courses. A proof of the confidence and appreciation our clients have for us is given by the fact that most of our new contracts are signed with previous customers.

Norsk Romsenter

Norsk Romsenter (Norwegian Space Centre, NSC) is a government agency under the Ministry of Trade and Industry. NSC promotes the development, co-ordination and evaluation of national space activities as well as supports Norwegian interests in the European Space Agency (ESA). Earth observation involves all activities related to collection of information on the Earth's surface or atmosphere from instruments on board satellites. The Norwegian Space Centre's application programme supports users, research communities and businesses in testing the potential of Earth observation from satellites. Priority is given to the development of applications having public benefit.

Statens Vegvesen Vegdirektoratet

Statens Vegvesen Vegdirektoratet (The Norwegian Public Roads Administration (NPRA)) is responsible for the planning, construction and operation of the national and county road networks, vehicle inspection and requirements, driver training and licensing. It is also authorized to grant subsidies for ferry operations. The objective of the NPRA is to develop and maintain a safe, ecofriendly and efficient transport system. This is being done on a sound, professional basis by interacting with politicians, users and other interested parties.

Norsk Regnesentral
Norwegian Computing Center
Postboks 114, Blindern
NO-0314 Oslo, Norway

Besøksadresse
Office address
Gaustadalléen 23
NO-0373 Oslo, Norway

Telefon · telephone
(+47) 22 85 25 00
Telefaks · telefax
(+47) 22 69 76 60

Internett · internet
www.nr.no
E-post · e-mail
nr@nr.no

Title	**SatTrafikk Project Report 2010**
Authors	**Arnt-Børre Salberg and Line Eikvil**
Date	Desember
Year	2010
Publication number	SAMBA/57/10

Abstract

Traffic statistics is a key parameter for operation and development of road networks. Vehicle counts based on automated satellite image analysis can provide useful additional information to traditional ground based traffic surveillance. A significant advantage of satellite based technology is that it does not require installation and maintenance of equipment in the road or expensive manpower costs related to acquisition of traffic counts using mobile counting devices.

The SatTrafikk 2010 focused on deriving two key features needed in a traffic monitoring system. That is: (1) Automatic segmentation of the desired road, and (2) automatic detection of clouds and cloud shadows. The road segmentation is based on a snake model of the road, initialized by the road GIS vector data The GIS data is needed to find the correct road, and to find the location of the road in the image. Then, to simplify the snake based segmentation algorithm, the image is re-sampled along lines perpendicular to the road vectors. This transforms the road into a nearly straight line. The approach then segments the road by combining the multispectral and panchromatic bands of the re-sampled image.

The cloud detection is based on a pixel classification approach. However, the parameters describing the class distributions in the test image need to be adjusted due to the varying atmospheric, botanic, and phenological conditions between the training images and test image. By modelling the test image by means of a mixture of Gaussian distribution, the class specific test data distributions may be obtained directly from each component of mixture distribution, after the corresponding parameters have been updated (from the training data distributions) using a low-rank dataset shift modelling scheme.

Both the automatic road segmentation and cloud classification methods were successfully tested and evaluated on a large set of images. However, for urban environments, or environments with disturbing elements like roundabouts, intersections, etc., further development of the road segmentation algorithm is needed. More importantly, the vehicle detection algorithm needs to be evaluated using road and cloud masks constructed by the automatic routines for road segmentation and cloud detection.

Keywords	Remote sensing, pattern recognition, vehicle detection, road traffic statistics, very high resolution satellite images
Target group	Road traffic authorities
Availability	Open
Project number	220 465
Research field	Earth Observation
Number of pages	54

Contents

1 Executive summary SatTrafikk 2010

The main goal of the SatTraffic project is to develop methods for automatic estimation of the number of vehicles on a given road segment per hour (NVRH) from optical satellite images of Norwegian roads. This measure is directly applied by Statens Vegvesen Vegdirektoratet (SVV) to estimate the annual average daily traffic (AADT).

To achieve the main goal many components need to be derived (Figure 1), and in 2010 the focus was on deriving two critical components; (1) automatic road segmentation, and (2) automatic cloud and cloud shadow detection (shown with red border in Figure 1).

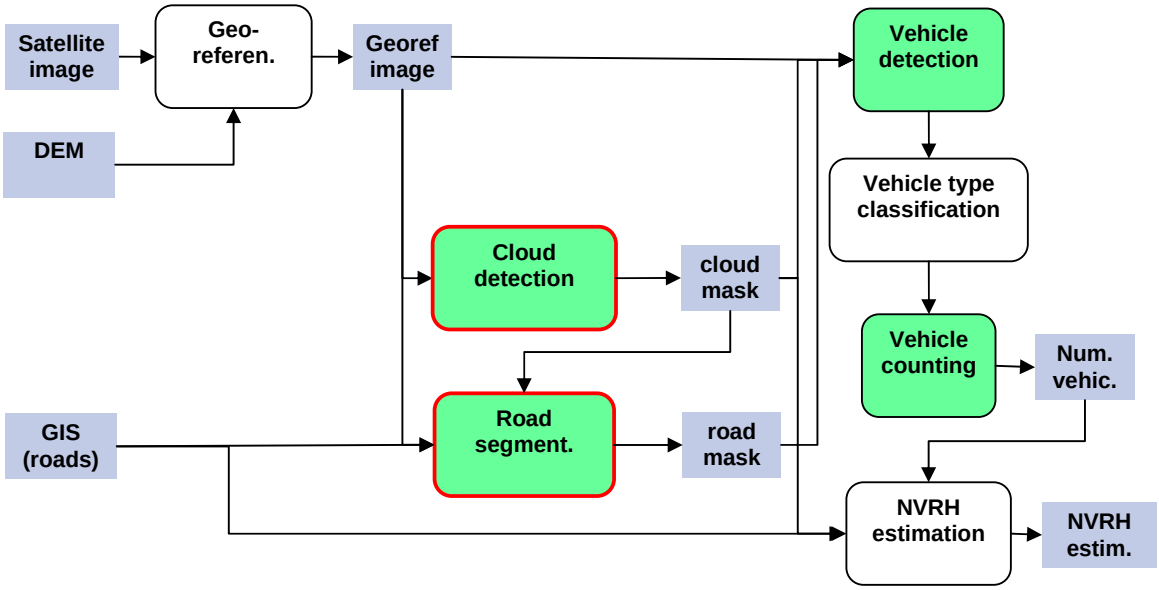


Figure 1: Block diagram of the SatTraffic system for automatic estimation of the number of vehicles on the road per hour. Green colour indicates existing modules and red lines indicate modules derived in 2010.

Automatic road detection: The detection of vehicles from satellite images requires that the road to be analysed has been identified and masked out in the image. In the first phases of this project the roads have been manually identified. However, to make the vehicle detection into a fully automatic process, the process of identifying the road has now also been automated.

Vector data (GIS) of the roads to be analysed are delivered from the Norwegian Public Road Administration. The vector data are necessary both to identify the right roads, as there may be several roads covered by an image, and to find the position of the road in the image. If the images and vector data were completely co-registered, the road could be delimited simply by selecting an area corresponding to the width of the road around the vector data. Unfortunately, this is not the case, and for some of our images the deviation between the two can be up to a hundred meters.

Although the vector data and image data are not accurately co-registered, the vector data still provides useful information about the approximate positions and directions of the road. We will therefore use this information to extract an area in the image around the road vector by re-

sampling the image along lines perpendicular to the road vectors. The result of this transformation is a long and narrow image along the road. The re-sampled multispectral and panchromatic images are then analysed in order to segment the road. Using a snake model of the road, the corresponding road in the images is segmented using the Viterbi algorithm to find the optimal path.

The road segmentation algorithm was successfully evaluated on several images. However, some challenges still remain when there are many disturbing elements like roundabouts, buildings, etc. near or on the road. The algorithm also needs to be evaluated together with the vehicle detection module.

Automatic cloud detection: Up till now, images containing clouds have not been used by the SatTrafikk project, but in order to increase the performance of the system, images that are partly covered by clouds need to be included. When using cloud contaminated images we need to construct a cloud and cloud shadow mask in order to (1) estimate observed road length such that the AADT estimate is not biased, (2) assist the vehicle detection algorithm by identifying road segments located in cloud shadow areas, and (3) assist the road segmentation algorithm by identifying road segments occluded by clouds.

To construct a cloud mask we have applied a classification based approach, i.e. we aim for classifying the image into the following thematic classes: Clouds, cloud shadows, green vegetation, haze, and water. We have used a Gaussian distribution to model the respective classes, and the corresponding parameters were estimated from training data. The main challenge is that match between corresponding classes from the training data and the test data may be poor due to atmospheric, geographic, botanic, and phenologic variations in the image data. To solve this problem we model the test image using a Gaussian mixture distribution, and apply a low-rank model to estimate the parameters involved. The corresponding class-specific parameters are then directly given.

The cloud detection algorithm was evaluated visually on many cloud contaminated images. The overall performance of the algorithm is very good; however, for some cases it might be difficult to define the limit between clouds and haze.

Operational scenario (see NR-NOTE SAMBA/58/10): A work package with the aim of discussing the basic needs, use and implementation of the SatTraffic system (NRSAT) at SVV was conducted. A price/performance analysis were NRSAT was compared to existing mobile traffic counting was also performed. The results of this work package indicate that (1) NRSAT may be a black box solution connected to SVV's NorTraf system using a suitable interface. The interface between NRSAT and NorTraf will be implemented by SVV. (2) A fully automated NRSAT system implemented in Linux is preferred. (3) The price/performance analysis indicates that, given good prices for the satellite images, the cost of performing satellite based traffic statistics is about 50% of costs using existing technology. However, some reduces precision compared to existing technology is expected.

2 Introduction

Road networks are resources of major importance for the society. Operation and development of road networks is a central activity for several public institutions, such as the Norwegian Public Roads Administration (Statens Vegvesen Vegdirektoratet, SVV). Traffic statistics is a key parameter for this activity. The primary source of traffic statistics today is ground based counts generated using various types of equipment mounted in or close to the road. Such equipment, e.g., radar, induction loops or pressure sensors, counts the number of cars passing a given location on the road during a period of time. Important statistics describing traffic can be derived from these counts, most importantly the so-called AADT, i.e., the Annual Average Daily Traffic, which is the average number of vehicles passing a given location during one day, taken as an average over a year. In Norway AADT is estimated using ground based vehicle counts in combination with statistical tools developed at the Norwegian Computing Center (Norsk Regnesentral, NR).

For fairly large parts of the Norwegian road network AADT is still unknown. The reason is that installation and operation of measurement equipment for ground based counts are both difficult and expensive, hence there are relatively few counting locations as seen in a geographical scope. Especially, AADT is missing for most roads with low traffic density on national basis.

Over the last few years, very high resolution satellite sensors have opened up for alternative means of traffic monitoring. Vehicle counts based on automated satellite image analysis can provide useful additional information to traditional traffic surveillance. A significant advantage of satellite based technology is that it does not require installation or operation of equipment in the road, thus maintenance demands and operating costs are no longer an issue. Moreover, a satellite image can cover large geographical areas and in principle this allows for AADT estimation of all the roads in the region, as opposed to only a few roads, as one is restricted to using fixed ground based measurements.

In 2006-2007 NR conducted the European Space Agency project "Road Traffic Snapshot" (see http://dup.esrin.esa.it/projects/summaryp92.asp) in cooperation with SVV and Institute of Transport Economics (Transportøkonomisk Institutt). Fundamental algorithms for vehicle detection using satellite images were demonstrated in the project. In the following project "SatTrafikk", starting in the summer of 2007 and funded by SVV and the Norwegian Space Centre (Norsk Romsenter, NRS), the detection algorithm was further developed and optimized. In 2008 the methods were validated on a large data set containing a variety of road conditions from different parts of the country. While the original methods were developed using satellite images covering the urban area of Oslo, this larger data set contained images from Sennalandet, Bodø, Kristiansund, Østerdalen, Eiker and Sollihøgda (see Figure 2-1). These images revealed different types of challenges when it comes to automatic vehicle detection in different parts of the country, and especially there are differences between urban and rural areas.

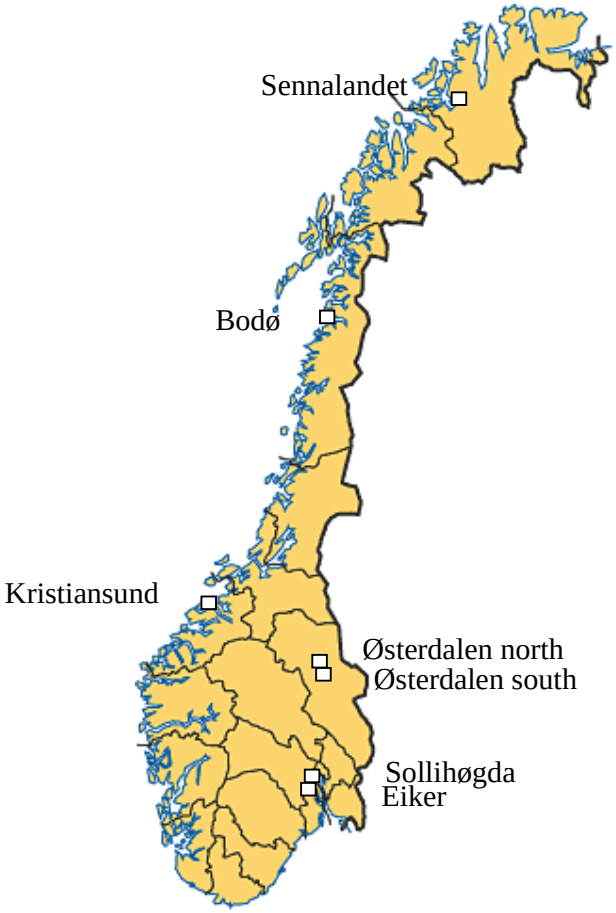


Figure 2-1. Locations of image acquisition for the 2008 QuickBird data set.

In 2008 we also conducted research on the expected quality of AADT estimates (derived from the statistical basis curve method), given the availability of one or a few satellite images per year. For roads with relatively large AADT as seen in a national context (i.e., AADT >20,000 vehicles) the results were promising (absolute error less than 20% given two satellite images a year), with the precondition that the vehicle detection algorithm is more or less completely accurate. For roads with smaller AADT (<20,000 vehicles) the corresponding average error was around 25%. With AADT less than 1,000 vehicles, even larger error can be expected, although, as traffic statistics hardly exist for such roads, there was no data evidence to verify this.

On the background of the mentioned experiences, the focus of the SatTrafikk project in 2009 was directed towards relatively low traffic density roads, mainly located in rural areas. Large parts of the previously suggested methods have been replaced or modified to meet the specific challenges related to vehicle detection under these conditions, and to optimize the detection strategy in general.

In order to apply the SatTrafikk vehicle detection software in operational scenarios, several key components are necessary and need to be developed. Two critical components are the automatic detection of the roads and automatic detection of clouds and cloud shadows. The former is necessary since we apply a road mask in the vehicle detection module to obtain counts from the desired road only, and traffic statistics are often based on the length of the road segment. The latter is necessary in order to obtain an estimate of the length of the road that can actually be observed, and thereby an unbiased measure of the AADT. This Report proposes methods that can perform the desired tasks of the two components.

The road segmentation module (Section 4) is based on GIS vector data delivered by SVV. Since the geographical precision of neither the satellite images nor the GIS data is sufficiently good, the GIS data is used to indicate the area where the road is situated. Then, using both the multispectral and panchromatic images the road is segmented using a snake based approach. The cloud and cloud shadow detection approach (Section 5) is designed such that it may be applied to any 5-bundle (4 multispectral channels and one panchromatic) Quickbird or WorldView-2 product. It is based on a set of training images acquired at different locations in Norway, and automatically adjusts to the conditions of the images in question.

3 New satellite images

For 2010 it was agreed that new images should be taken from the northern part of Norway. Two E6 road segments were selected, one north of Mosjøen in Nordland and one south of Norkjosbotn in Troms. The images were acquired on June, 10 (Mosjøen) and June, 11 (Nordkjosbotn) 2010 using the new satellite sensor WorldView-2. A bundle consisting of 4 multispectral and a panchromatic band was ordered. The images were ordered as "ortho-ready" products, meaning that geo-referencing to a digital elevation model (DEM) is possible if desired. The resolution of WorldView-2 is slightly better than Quickbird, with a panchromatic resolution of 0.5m and a 2.0m multispectral resolution (Quickbird has 0.6m and 2.4m, respectively).

Due to the shape of the selected road segment, several images were needed in order to cover the whole scene. The Mosjøen scene was divided into 8 images, while the Nordkjosbotn scene was divided into 7 images. The images covering the Mosjøen and Nordkjosbotn scene were acquired at 12:17 and 11:25, local time. An overview of some of the images is shown in Figure 6-24 - Figure 6-27.

4 Automatic road detection

The detection of vehicles from satellite images requires that the road to be analysed has been identified and masked out in the image. In the first phases of this project the roads have been manually identified and drawn in the images. However, to make the vehicle detection into a fully automatic process, the process of identifying the road must also be automated.

Vector data of the roads to be analysed are available, and delivered from the Norwegian Public Road Administration. The vector data are necessary both to identify the right roads, as there may be several roads covered by an image, and to find the position of the road in the image. If the images and vector data were completely co-registered, the road could be delimited simply

by selecting an area corresponding to the width of the road around the vector data. Unfortunately, this is not the case, and for some of our images the deviation between the two can be up to a hundred meters. Figure 4-1 shows an image with the road vector superimposed in purple, illustrating a case of quite large deviation.

Figure 4-1: Mismatch between road vector (purple) and actual road.

A first thought could then be to try and co-register the two prior to finding the roads. However, as the vector data contain only the roads, such a co-registration would require that the roads were already identified in the image. Hence, this approach will not work. Instead we have therefore found another way to exploit the information from the vector data and reduce the complexity of the problem of finding the road without performing co-registration. In addition, we afterwards suggest an approach for co-registration of identified road segments and road vectors that can work even for our case of quite sparse geographical information.

4.1 Road segmentation

4.1.1 Radial road representation

Although the vector data and image data are not accurately co-registered, the vector data still provides useful information about the approximate positions and directions of the road. We will therefore use this information to extract an area in the image around the road vector. The idea is that if this area is of a reasonable size relative to the expected magnitude of the geographical displacement, it will contain the road in the image. We have chosen to extract such an area by sampling the image along lines perpendicular to the road vectors, as shown in Figure 4-2, producing a radial road representation.

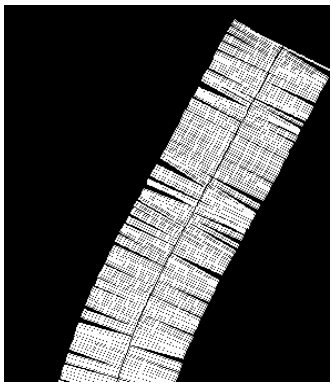


Figure 4-2: The image is sampled along lines perpendicular to the road vectors.

The result of this transformation is a long and narrow image along the road. In the case where the vector and image data are perfectly co-registered, the road will run as a straight line along the middle of this image. But still, when this is not the case, the position and movement of the road will be greatly reduced through this transformation. Hence, by doing this transformation we can limit both the search area and the search direction and thereby simplify the problem of finding the road in the image. In the transformation process we also store the original coordinates of each sampled point, to be able to easily retain the original representation.

Figure 4-3 shows a road segment before and after this transformation. The transformation has here been performed on the first (blue) band of the multispectral Quickbird image. The width of each sampling line has currently been set to 100 pixels, meaning that the width of the area that is covered will be around 240 meters. The width must be set large enough to cover for the expected deviation between image and vector data.

In this new representation we know that the road should run through all the lines of the transformed image. Hence, we should be able to find and trace the road by analysing this image line by line. To do this, we determine the external and the internal forces in a snake-like fashion. Internal forces are the ones that impose constraints on smoothness, while external forces arise from the data.

The external force is found from the transformed image data and designed to find the road which typically appears as a bright ribbon. Hence, for each line we compute the average pixel value over a window corresponding approximately to the width of the road.

$$E_{ext}(i) = \frac{1}{w} \sum_{x-w/2}^{x+w/2} f(i)$$

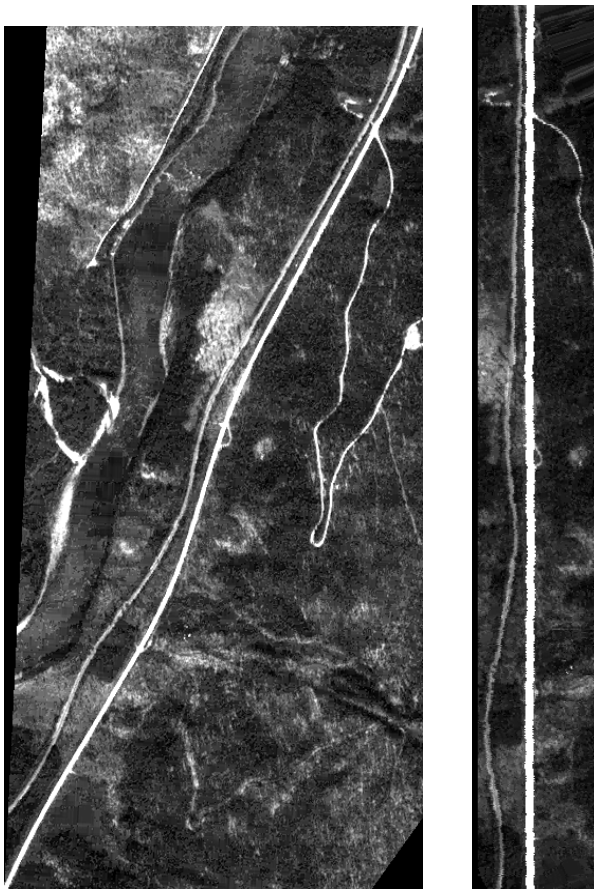


Figure 4-3: Image of road before (left) and after (right) transformation. The weaker line running alongside the road is the railway.

The internal force is based on local smoothness, where the assumption is that the position on two adjacent lines will not change dramatically and is realized through a cost term meant to discourage abrupt changes in direction:

$$E_{\text{int}}(i) = \text{cost}(i)$$

The cost term is determined from the difference in positions from the current to the previous position:

$$\text{cost}(i) = (x_i - x_{i-1})^2$$

In addition, we have included a penalty term that puts an extra penalty on discontinuity, i.e. when a point moves outside the window:

$$\text{penalty}(i) = E_{ext}(i) \cdot |x_i - x_{i-1}|, \qquad \text{if } |x_i - x_{i-1}| > \text{window}$$

Thus, this penalty term is made dependent both on the external and the internal forces.

Based on the terms defined above we then compute for each point of a line the sum of forces as:

$$E_{sum} = \alpha \cdot E_{ext} - \beta \cdot E_{int} - \gamma \cdot penalty$$

where $\alpha = 0.2$, $\beta = 1.0$, *while* $\gamma = 0.3$ if $|x_i - x_{i-1}| < window$, and $\gamma = 0.6$ otherwise.
For each line in the transformed image we select the position with the maximum value.

When using an approach like this we also need to find a starting position. We find this by first setting a pre-start point as the midpoint of the line. Then the first M (currently M=30) lines are analysed using the approach described above, and the median position is used as the initial position.

The result of this process in the transformed image is shown in Figure 4-4, left. Here the identified positions for the road are marked in black. To obtain a smoother line the final result may be filtered. This resulting line can then be transformed back to the original representation. To obtain a smooth and continuous road mask after this transformation, a morphological closing operation is performed to give a mask as shown in Figure 4-4, right.

A center line for this road in multispectral resolution is made from the midpoints of the identified segments in the transformed image. This sequence of x-coordinates is filtered by a median filter and transformed to the original image coordinates.

4.1.2 Extension to true snake

Using the approach above, we can only do an optimization of the line position line-by-line. A true snake on the other hand, uses dynamic programming (e.g. Viterbi) to find the set of points that optimize the sum of internal and external forces over *the whole set*. By using only an external and an internal term, excluding the combined penalty term, we can extend our quasi-snake to a full snake model and make use of the Viterbi algorithm. As the optimization process and the search are changed under this approach, the terms describing the external and internal forces were also somewhat reformulated, but following the same principle:

The *internal force* for the true snake is defined as

$$E_{int} = d_0 + d_1 + d_2$$

where d_0 is the distance from the previous position measured in the original coordinate system and defined as

$$d_0 = |y_i - y_{i-1}| + |x_i - x_{i-1}|,$$

d_1 is the difference in position within the radial scanline from the previous position with an additional penalty when the distance increases

$$d_1 = |pos_i - pos_{i-1}| + \alpha \cdot window.$$

Here *window* is the radii of the search window within the scanline, and α is defined as

$$\alpha = 0, \quad \text{if } |pos_i - pos_{i-1}| \leq 1$$
$$\alpha = 1, \quad \text{if } 1 < |pos_i - pos_{i-1}| \leq window$$
$$\alpha = 2, \quad \text{if } |pos_i - pos_{i-1}| > window$$

where d_2 is the distance from the prior position (currently defined as the initial position). The idea is that it should be more costly to move towards the edge of the transformed image, and we therefore define d_2 as

$$d_2 = |pos_i - pos_{prior}| + D_{penalty}$$

where the penalty term $D_{penalty}$ is used when $|pos_i - pos_{prior}|$ is larger than a distance D (currently $D=20$) and is defined as

$$D_{penalty} = \exp((pos_i - a_{prior}) - D)$$

The *external force* is derived from the image as the maximum over a window and normalized with respect to the range of the values within the image

$$E(i) = \begin{cases} -\log(E(i)) & if \quad E(i) > 0 \\ -E(i) & otherwise \end{cases}$$

where E(i) is defined as

$$E(i) = 255 \cdot \left(\frac{\sum_{x-w/2}^{x+w/2} f(i)}{w} - \min(scanline) \right) / range(scanline),$$

The value 255 transforms the grey levels into a range from 0 to 255.

The performance for the true snake on the image in Figure 4-4 is almost identical to that of the quasi-snake. However, for some of the more difficult cases, the results of this approach seem to be better. Figure 4-5 shows the result of this road segmentation approach on a WordView-2 scene.

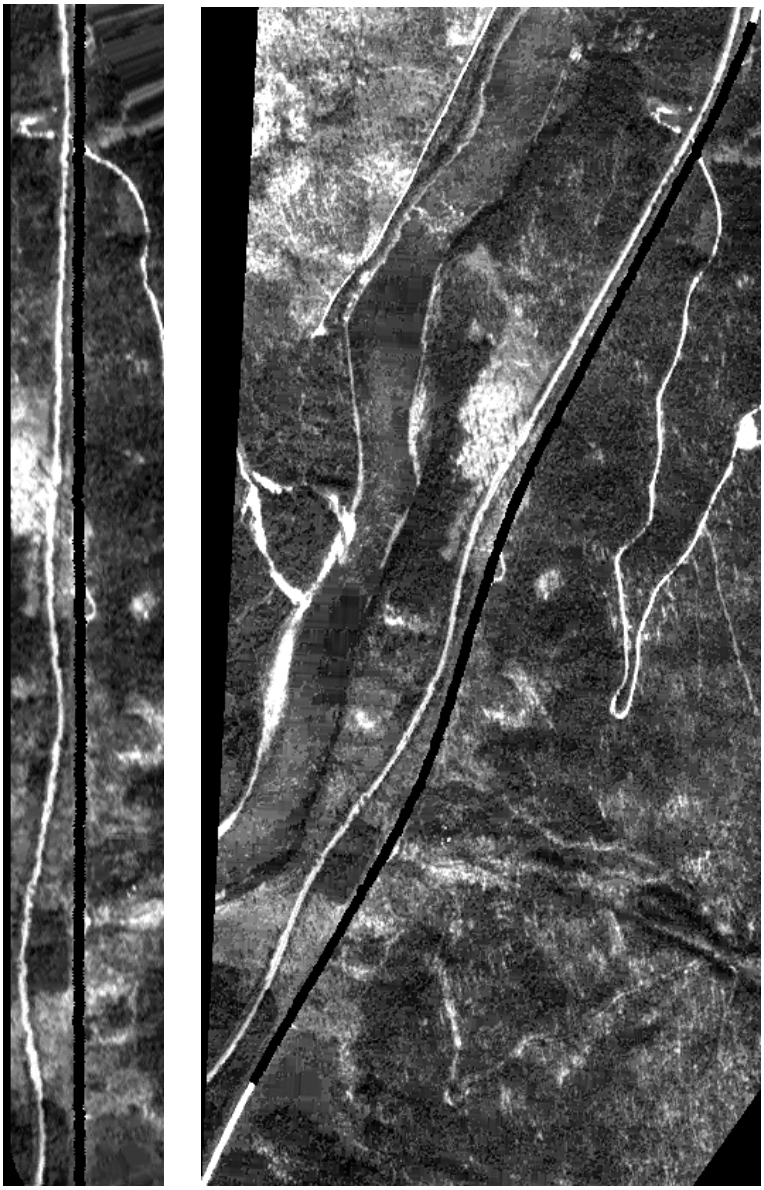


Figure 4-4: Left: Traced road in transformed image, marked as black. Right: Traced road transformed back to original image coordinates.

4.1.3 Viterbi-algorithm

The snake is the set of nodes (image points) $V = v_1, v_2, \ldots$ that minimizes the snake energy, $E_{snake}(V)$. The problem is to find the set of points V that minimizes:

$$\sum_{v_i \in V} \left| E_{ext}(v_i) + E_{\text{int}}(v_{i-1}, v_i) \right|$$

where E_{ext} is the external energy from the image data, and E_{int} determines the elasticitiy of the curve. To do this we use dynamic programming using the Viterbi algorithm. Since the internal energy is a local property of neighbouring vertices, we can decompose it into a sum of local terms:

$$V = E_1(v_1, v_2) + E_2(v_2, v_3) + \ldots + E_{n-1}(v_{n-1}, v_n)$$

where:

$$E_{i-1}(v_{i-1}, v_i) = E_{ext}(v_i) + E_{\text{int}}(v_{i-1}, v_i)$$

In particular this means that each vertex position v_i, influences the total energy only through the terms E_{i-1} and E_i. The Viterbi algorithm capitalizes on this property.

To find the best-fit contour, the algorithm starts from the first line of the transformed image and traces the road through the scanlines until the last line. For every pixel along a scanline, we can find the optimal energy for a contour ending at that point. We find the best trace by finding the minimum energy point for the last scanline and then back-trace from this to find the whole trace that gives this optimal minimum energy.

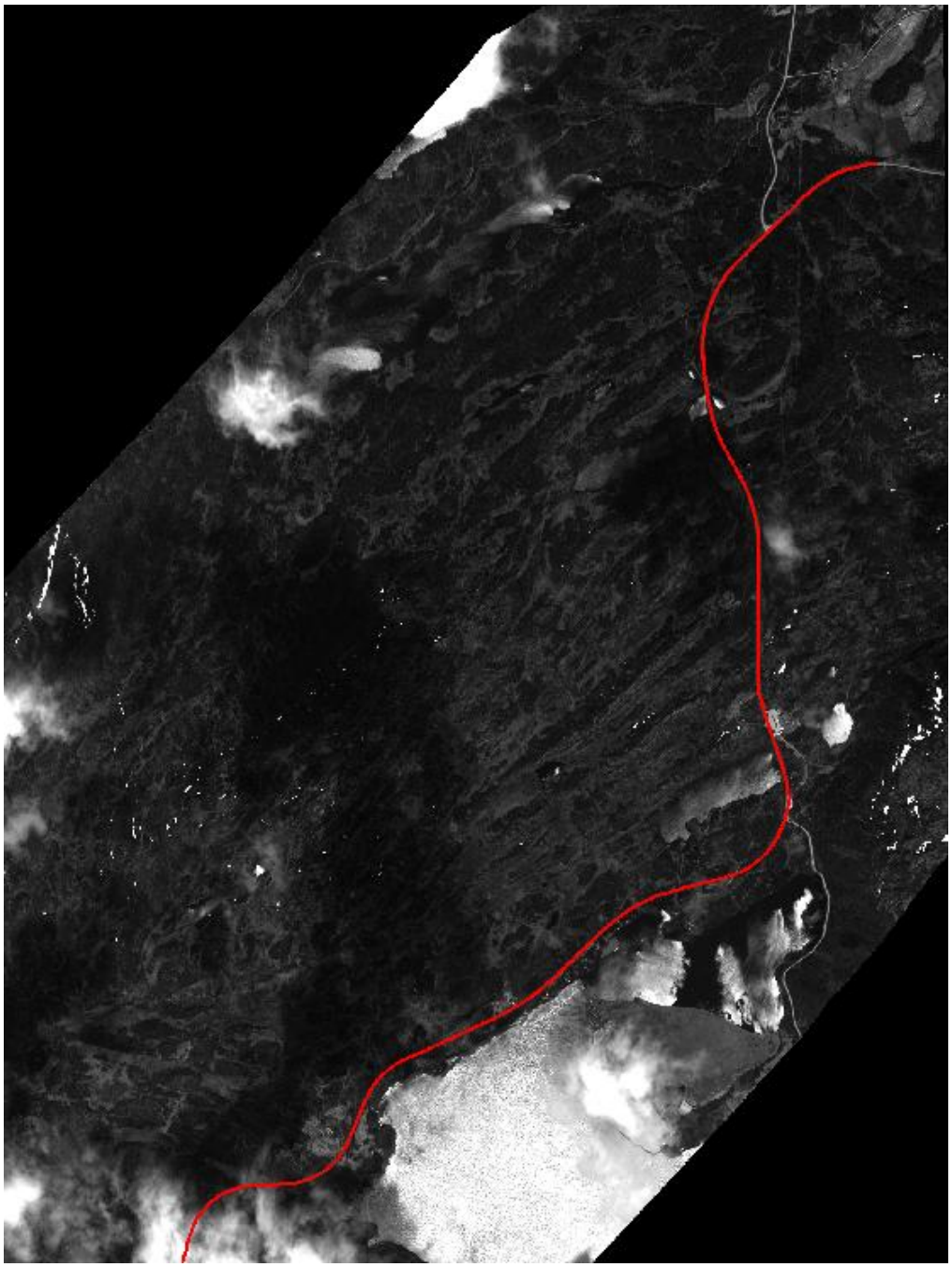


Figure 4-5: Result of snake-based road segmentation on a WorldView-2 image covering part of the Nordkjosbotn scene.

4.1.4 Transferring result to panchromatic resolution

The segmentation approach has been applied at the multispectral resolution level. In theory the approach could be adapted to panchromatic images, but initial experiments showed that the multispectral data was best for this, probably due to the fact that the road is more homogeneous and well defined in these images. However, the subsequent analysis concerns vehicles, and for

this we need to work with the panchromatic data. Hence, the multispectral mask needs to be transferred to this higher resolution.

A very simple way to achieve this is to simply resample the multispectral mask to achieve an image of panchromatic resolution. We tested this approach initially, but this leads to a quite coarse road mask which may not be sufficiently accurate for the vehicle detection. To achieve a smoother mask, the mapping to panchromatic resolution was instead performed already on the transformed image. The filtered positions of the multispectral center line were re-sampled to panchromatic resolution, using interpolation to estimate the intermediate points and the resulting panchromatic positions were filtered again. Then the coordinate mapping was also re-sampled and interpolated, and based on this both the centerline and road mask were transformed back to image coordinates in panchromatic resolution. This approach leads to a much smoother centerline and mask (see Figure 4-6). However, a potential weakness of this mask is that it is not based on the actual pixel information of the panchromatic image.

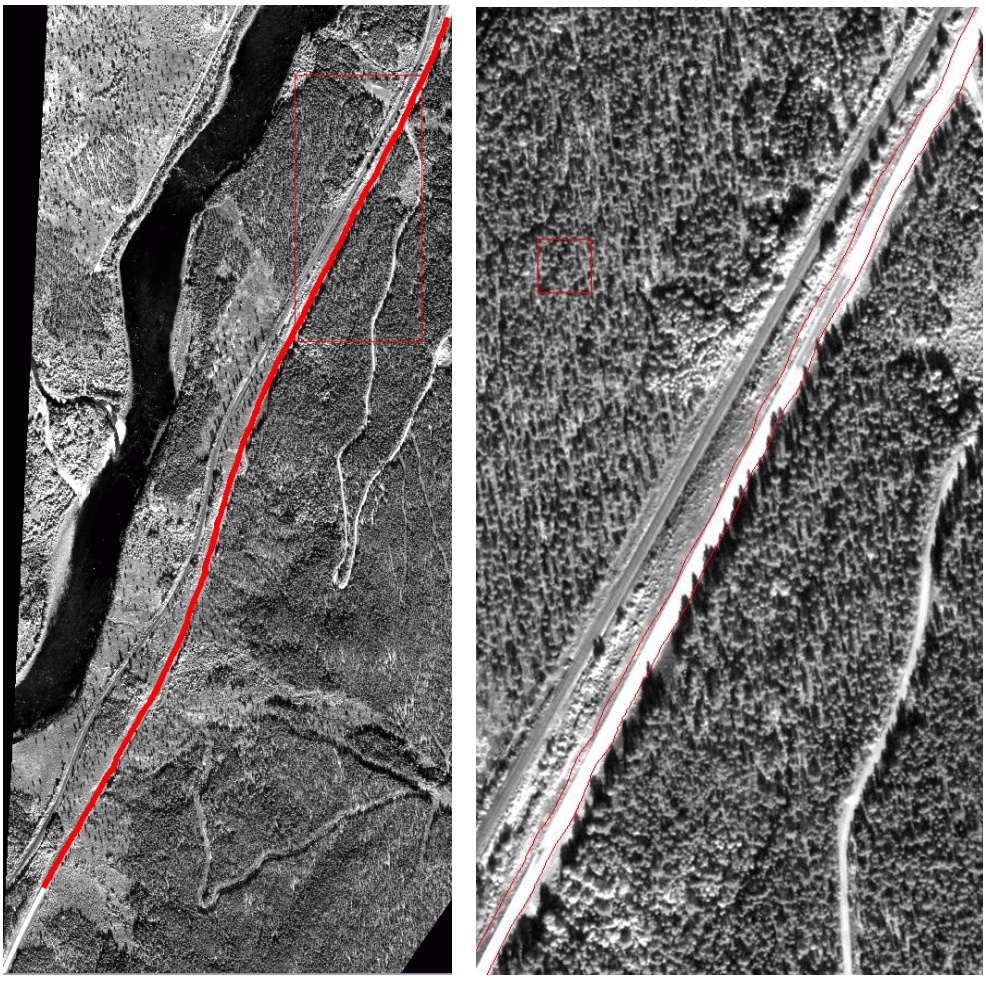


Figure 4-6: Road mask derived only from the multispectral information.

To exploit also the information from the panchromatic image in the process, an additional step has therefore been included. In this step the results from the previous steps are used in combination with the panchromatic image data. First, the centerline in multispectral resolution is just crudely re-sampled to panchromatic resolution and used as a basis for region growing along the road in the panchromatic image. The multispectral line is chosen as the basis rather than the panchromatic line simply because it covers more pixels in the panchromatic image and provides more seed pixels for the region growing. The region growing is performed by the REGION_GROW function in IDL on pixels that fall within the range of the mean of the region's pixel values plus or minus a given multiplier times the sample standard deviation as follows: Mean +/- StdDevMultiplier * StdDev. The multiplier for the standard deviation has been set to two.

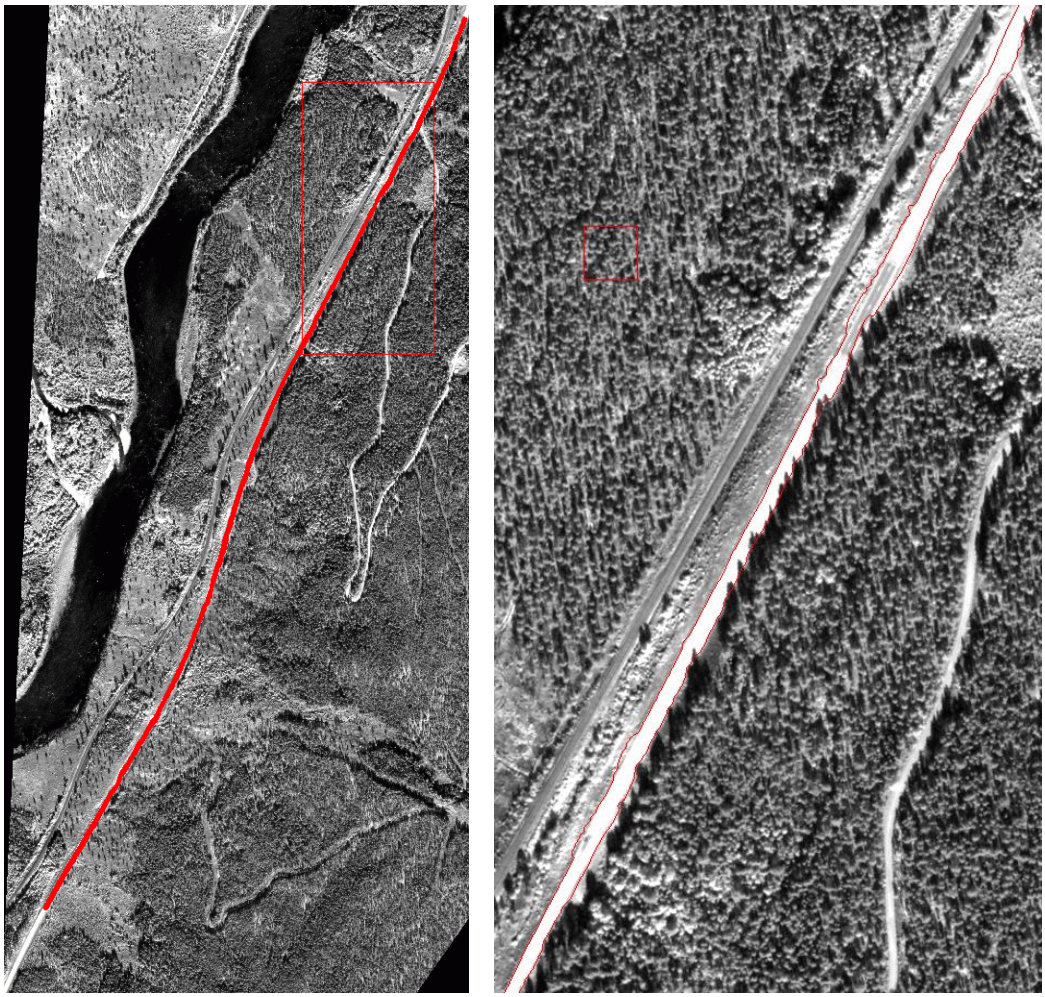


Figure 4-7: Road mask derived from tracing in multispectral data and region growing in panchromatic data.

The result from this growing will be a quite rough road mask, which at points may have grown into crossing road segments. To limit the extension of the grown mask, a distance map is produced from the center line. For this process a detailed and narrow line is the best option. Hence, we use the centerline estimated for the panchromatic resolution. The distance map gives

the distance to the center line in all points that are closer to the line than to the edge of the image. This map is used to exclude all points from the grown mask that are further from the road center than a certain distance and to include all points that are very close to the center line. Finally, the resulting mask is morphologically filtered to fill small holes and remove noise. The resulting grown road mask is shown in Figure 4-7.

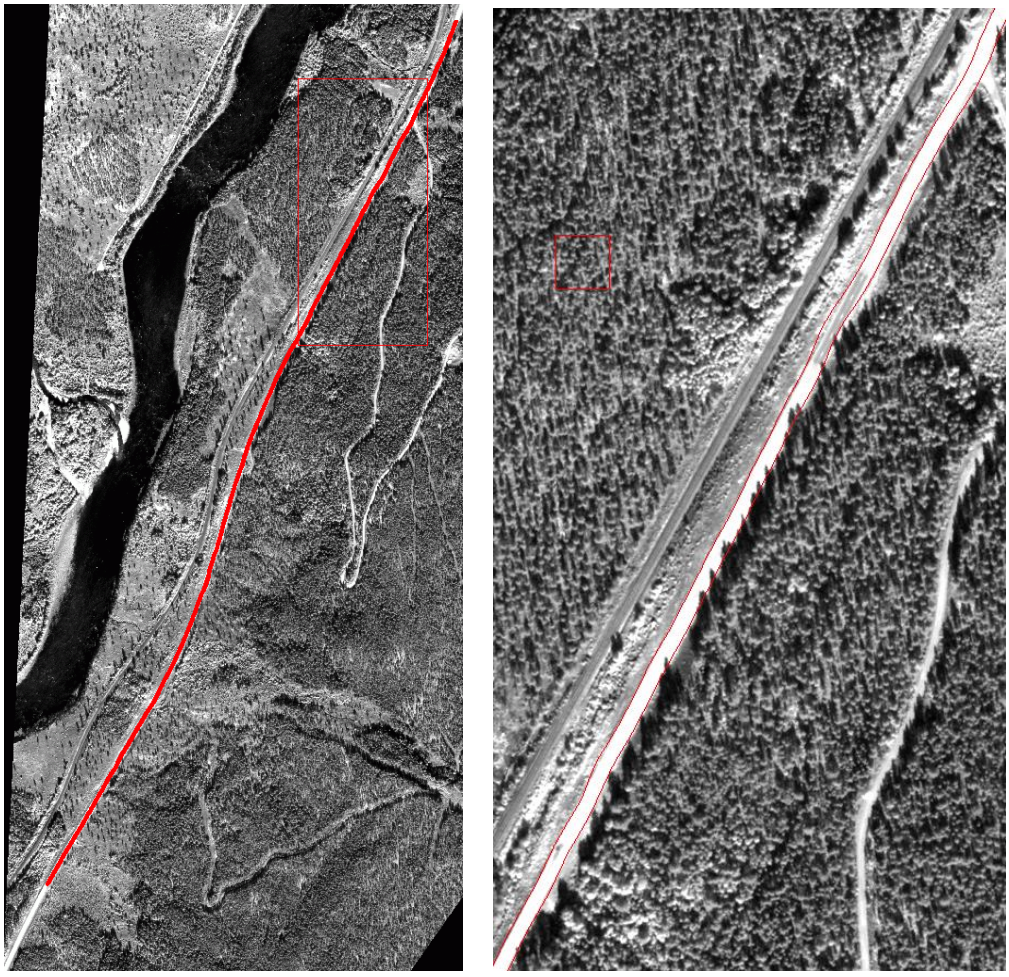


Figure 4-8: Road mask derived from tracing in multispectral image and distance mapping relative to centerline from region grown mask in panchromatic data.

The advantage of this grown mask is that it is estimated more directly from the panchromatic image data. However, this is also a disadvantage as it results in a road mask that is not smooth along the edges due to variations in grey levels along the road and the roadside. As a compromise between this mask and the one based solely on the filtered and interpolated multispectral data, we have therefore also produced a third mask. This mask incorporates information from the panchromatic data by using the grown mask to obtain a new center line. This center line is produced by thinning the grown road mask. From this center line a new distance map is produced, and then this distance map is smoothed through the use of median and mean filters. This has a similar effect to filtering of the actual center line. Finally, this distance map can be thresholded according to the road width. The distance based road mask

resulting from this process is shown in Figure 4-8. Any one of these three alternative road masks may be chosen as basis for the vehicle detection. Testing in combination with the vehicle detection should decide which is the best solution.

4.2 Preparation for integration with vehicle detection

In addition to the original image data the vehicle detection needs the road mask, the centre line of the road and a vegetation mask:

- Road mask: An ENVI classification file with two classes; background and road.
- Center line: A 1-pixel wide centre line marked with pixel value 3.
- Vegetation mask: A two-level mask, where areas identified as vegetation are marked as zero. (The process for this is described below).

All of these can now be produced automatically based on the images and the vector data. This process has been integrated and tested with the first step of the vehicle detection, where potential vehicles are detected as blobs. Currently it is only verified that the road segmentation produces the right type of data, while the detection and classification performance using automatically segmented roads compared to that of using manually drawn road masks has not yet been evaluated.

4.2.1 Vegetation mask

The vehicle detection also uses a vegetation mask, which is computed from the multispectral image. This is done in two steps. First a normalized vegetation index is computed using the following formulae:

$$NVI = 1.0 - (2.0 \cdot red + \varepsilon)/(\varepsilon + NIR + red), \quad \text{where } \varepsilon = 0.0001$$

Then the resulting image is thresholded using Otsu's method for threshold selection. This is the same approach as used before, but it is now performed automatically rather than interactively and has been included in the automatic processing chain preparing input for the vehicle detection.

4.3 Vector-and-image co-registration

When the road has been identified in the image, we have information from both the vector data and the image data that are corresponding, i.e. the road data, and which makes it possible to do a co-registration of the two data sources. Although this co-registration is not needed for the road segmentation (it's actually the other way round), it can be useful for other purposes and as we will see, it can also be used to do a quality control of the segmented road.

We have chosen to use an image-based registration approach. The road vector is therefore first converted to a raster image, and the objective is then to co-register this raster road with the road mask achieved from segmentation of the satellite image. As the actual road masks contain very little information to help guide the co-registration, we make distance maps based on the masks. These distance maps will for all the pixels surrounding the road mask indicate the distance to the road, and to provide even stronger guiding information to the co-registration, logarithmic

distance to the road mask is used. In Figure 4-9 such a distance map is shown along with a plot of a profile crossing the road mask showing the logarithmic structure.

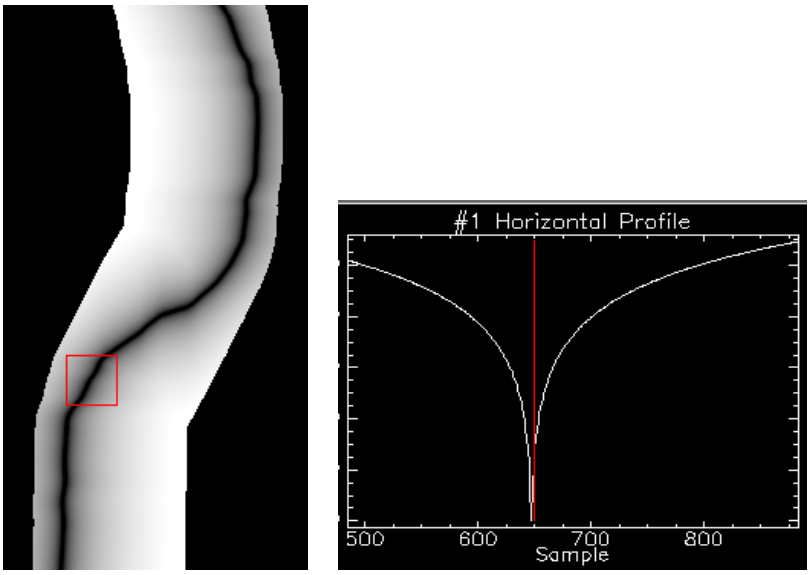


Figure 4-9: Logarithmic distance map for a road segment.

The co-registration is performed on two distance maps, the one resulting from the image data and the one from the vector data, using the MIR co-registration system. This is a system developed at NR that does adaptive co-registration in the way that it automatically selects the regions in the image suited for co-registration and can also for each region select the registration algorithm expected to be most appropriate. The selection of the regions and the algorithms is based on the image contents. This approach is well suited for the registration of the distance maps as these will contain large homogeneous areas which are automatically ignored by the registration approach. The registration will then only focus on the areas containing useful information, and this makes it fast.

Figure 4-10 shows the result of using this co-registration approach. The image on the left shows a mosaic of the vector and image roads before registration, while the image on the right shows the same thing after registration. The mosaic contains image regions selected in a check board pattern from the two images, and when the two are perfectly co-registered the road should appear as a continuous line through this check board. The image on the right in Figure 4-10 shows the mosaic after registration, where the road appears as a more continuous line. Discrepancies can however appear along directions where the co-registration is not able to get enough information.

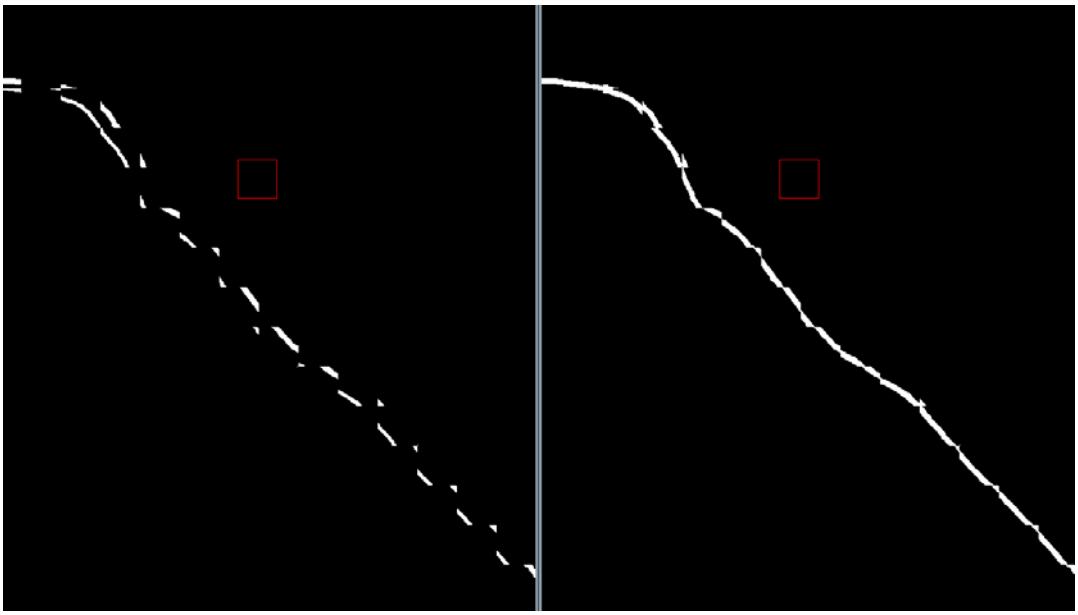


Figure 4-10: Mosaic of vector and segmented road before (left) and after (right) registration.

One interesting feature of the result of the co-registration is that it can be used to investigate the quality of the segmented road. This can be done by first performing a co-registration where the distance map based on the vector data is used as master and the distance map resulting from the segmented road is used as slave. Then the co-registered image can be resampled to get a new radial representation of the road. In the transformed image the road should now appear as a straight line (this can be checked automatically), and if it does we know that the segmentation is correct (if this is not the case, the road segmentation or the co-registration may have failed at the corresponding points).

5 Automatic cloud and cloud shadow detection

Up till now, images containing clouds have not been used by the SatTrafikk project, but in order to increase the performance of the system, images that are partly covered by clouds need to be included. When using cloud contaminated images we need to construct a cloud and cloud shadow mask in order to:

- Estimate observed road length such that the AADT estimate is not biased.
- To assist the vehicle detection algorithm by identifying road segments located in cloud shadow areas.
- To assist the road segmentation algorithm by identifying road segments occluded by clouds.

The goal is to develop a method for precise and automatic detection of clouds and cloud shadows in very high resolution images. The main challenge is that training data may poorly

match the test data due to atmospheric geographic, botanic, and phenologic variations of the image data.

Calibration of satellite data to ground reflectance is not an easy problem (see e. g., Liang , Fang and Chen 2001, and the references therein), and many algorithms have been proposed for atmospheric calibration of an image set. However, a further complicating factor is that we have images of different scenes, and that geographic, botanic and phenologic variation, in addition to atmospheric variation results in a much harder calibration problem. Many cloud detection algorithms are developed for sensors with a thermal channel (Irish, et al. 2006), which excludes very high resolution (VHR) satellites such as Quickbird, WordView-II, GeoEye and Ikonos. However, some have addressed cloud and cloud shadow detection in VHR images (Lu 2007).

5.1 Classification based approach

To detect clouds and cloud shadows in the images we apply a classification based approach. The goal is to classify the image into the following thematic classes:

- Clouds
- Cloud shadows
- Green vegetation
- Haze
- Water
- Other (concrete, asphalt, soil, etc)

Since there is no need to resolve the clouds with 0.5-0.6m resolution, we perform the cloud classification on an image down-sampled by a factor of 8 (e.g. 19.2m for QuickBird and 16m for WorldView-2). Training data for each class is constructed by visual inspection of a set of training images with reduced resolution and by labelling a selection of regions that contain the respective classes. The labelling of the training images is guided by the images of nominal resolution (2.4m and 2.0m). For cloud classification we use the following two spectral features:

- band 2 (green)
- band 3 (red)

For cloud shadow classification, we will in addition to these use the following features:

- band 2 / band 4 (green/near infrared)
- (band 4 – band 3)/(band 4 + band 3) (NDVI)

We model the data representing each class using a multivariate Gaussian distribution, i.e. for class *i* the PDF of the data is given as

$$f(\mathbf{x} \mid \omega_i) = f(\mathbf{x} \mid \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_i|} \exp\left(\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right)$$

where the mean vector and covariance matrix are estimated from the training data.

The classification process is a 2-stage procedure, where we in the first stage classify the clouds, and in the second stage classify the cloud shadows. This will be described in detail in Section 5.3.

5.2 Low-rank dataset shift modelling

One of the major challenges of extracting the training data from a set of images and classifying another image is that the data distribution for a given class varies between the training images, and varies between the training images and the test image (see Figure 5-1). In terms of the probability density functions (PDFs) that models the data for each of the classes in each of the images, we may express this as

$$f_{train}(\mathbf{x} \mid \omega_i) \neq f_{test}(\mathbf{x} \mid \omega_i)$$

and

$$f_{train\,k}(\mathbf{x} \mid \omega_i) \neq f_{train_n}(\mathbf{x} \mid \omega_i)$$

However, the training data domain and test data domain are generally neither identical nor uncorrelated. In these situations it may be possible to utilize the existence of an intrinsic relationship between the two data domains, and adapt the training data distributions to the distributions describing the classes in the test domain (Quiñonero-Candela, et al. 2009)

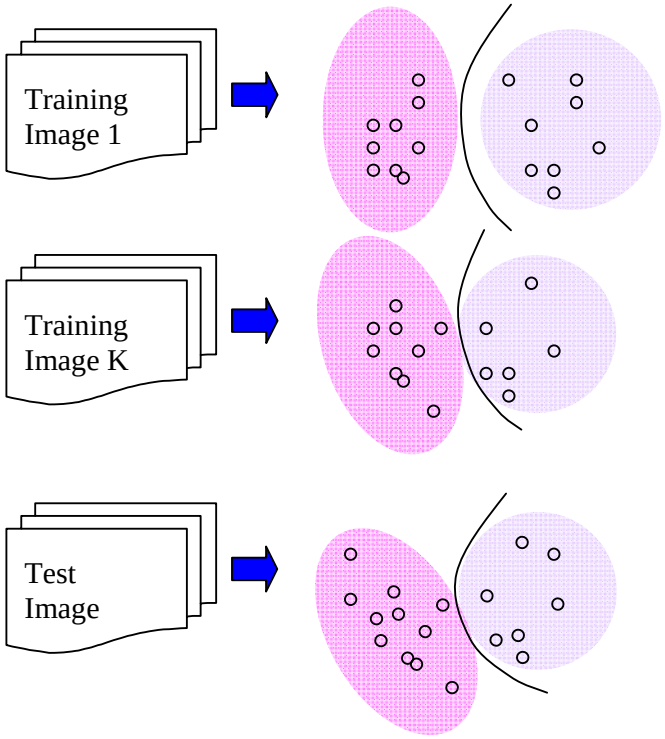


Figure 5-1: Illustration of the dataset shift between training images and test image. For each case we see that both the shape of the distribution and, hence, the decision line varies between the images.

5.2.1 An existing method for retraining the classifiers

Bruzzone and Prieto (2001) proposed a method for retraining a classifier when test data differs (slightly) from the acquired training data. The approach in (Bruzzone og Prieto 2001) models the test image using a mixture of Gaussians and estimates all parameters involved using the expectation-maximization (EM) algorithm, with the parameters estimated from the training data as initial values, i.e. it solves the following optimization problem

$$\begin{Bmatrix} \boldsymbol{\mu}_1,...\boldsymbol{\mu}_C \\ \boldsymbol{\Sigma}_1,...,\boldsymbol{\Sigma}_C \\ P_1,...,P_C \end{Bmatrix} = \arg \max_{\substack{\boldsymbol{\mu}_1,...\boldsymbol{\mu}_C \\ \boldsymbol{\Sigma}_1,...,\boldsymbol{\Sigma}_C \\ P_1,...,P_C}} \log \sum_{\mathrm{n}=1}^{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{C}} \mathrm{P_i} f_N\left(\mathbf{x}_\mathrm{n} \mid \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\right) + \lambda\left(\sum_{i=1}^{C} P_i - 1\right)$$

Even though the approach seems appealing it suffers from a weakness in the sense that many parameters need to be estimated. When applying this method to our data we obtain a very good statistical fit of the likelihood to the test image, but the mixture components have no longer a physical meaning in the sense that, e.g. the mixture component of a given land cover type no longer models that land cover type, but something else.

The approach we propose as an alternative is based on the method suggested in (Bruzzone og Prieto 2001), but apply a low rank modelling of the parameters in order to reduce the number of degrees of freedom and flexibility of the model. By doing so, we force the class structure of the training data to be maintained in the test image. We will also extend the method by incorporating several training images, each with different class dependent data distributions.

5.3 Data modelling

We model each class of a given training image as a Gaussian distribution with mean vector and covariance matrix of class *k* and class *i* given by

$$\boldsymbol{\mu}_{i,k} \text{ og } \boldsymbol{\Sigma}_{i,k}$$

The PDF of representing the *i*th class in the *k*th training image is denoted by $f_N(\boldsymbol{x} \mid \boldsymbol{\mu}_{i,k}, \boldsymbol{\Sigma}_{i,k})$.

5.3.1 Cloud detection

We model each class in the test image as a Gaussian distribution, with mean vector and covariance matrix defined as

$$\boldsymbol{\mu}_i = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\mu}_{i,k} + \nu_i \boldsymbol{\delta}_i \alpha$$

$$\boldsymbol{\Sigma}_i = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\Sigma}_{i,k}$$

where $\boldsymbol{\delta}_i$ is the eigenvector corresponding to the largest eigenvalue ν_i of the matrix

$$\mathbf{M}_i = \frac{1}{K} \sum_{k=1}^{K} \left(\boldsymbol{\mu}_{i,k} - \boldsymbol{\mu}_i\right)\left(\boldsymbol{\mu}_{i,k} - \boldsymbol{\mu}_i\right)^T$$

Hence, the covariance matrix is simply the average covariance matrix of all training images, and the mean vector is the average mean vector of all training images plus a component of rank 1

along the direction $\boldsymbol{\delta}_i$. The principal eigenvector is oriented along the direction where the mean vectors vary most across the images (Figure 5-2).

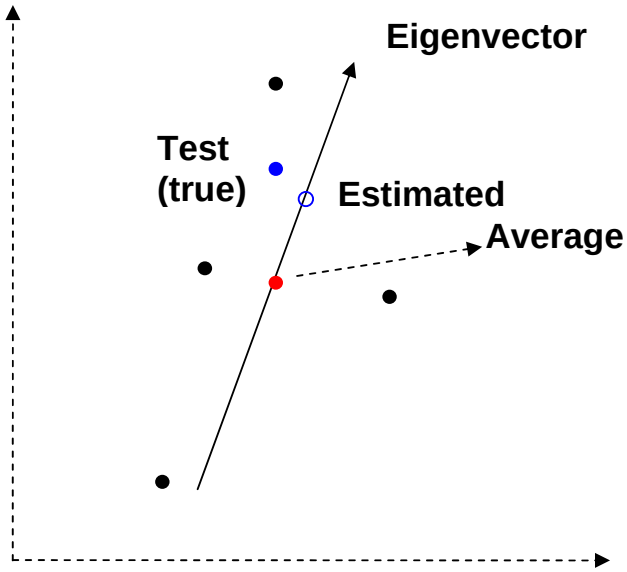


Figure 5-2: Illustration of data modeling of a given class. The black dots represents the mean vector of a given class of each training image, the red dot is the average of the mean vectors of all images (fixed class), the blue dot represent the true class mean vector of the test image, the blue circle represent the estimated mean vector of the test image. The arrow indicates the subspace spanned by the principal eigenvector of $\mathbf{M}_i$.

The parameter α is unknown and needs to be estimated from the test data. This is done under the assumption that the mean vector of the test image is located in the range of the eigenvector (when centred at the average mean vector).

To estimate α we select 10000 random samples from the test image (Figure 5-3), and model the data using a as mixture of Gaussian distribution, i.e. the PDF is given as

$$f(\mathbf{x}) = \sum_{i=1}^{C} \mathrm{P}_i f_N(\mathbf{x} \mid \boldsymbol{\mu}_i + \boldsymbol{\delta}_i \alpha, \boldsymbol{\Sigma}_i)$$

where P_i denotes the class prior probabilities.

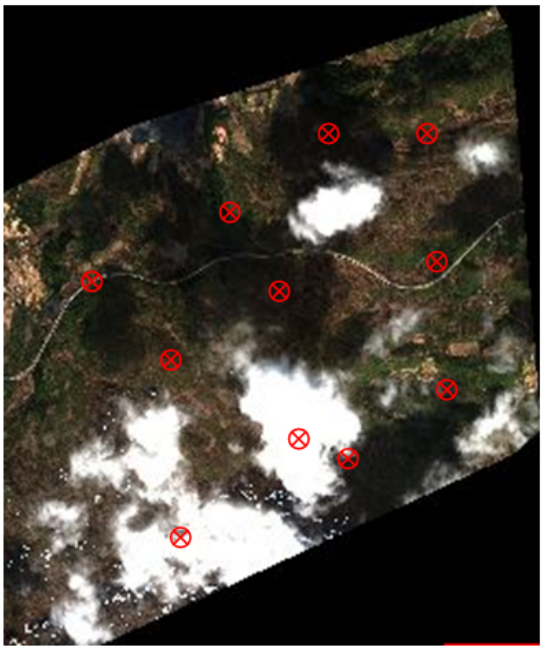


Figure 5-3: Random sampling of the test image for unsupervised modeling.

The coefficient α and the unknown class prior probabilities $P_1,\ldots,P_C$ of the test image are estimated by maximizing the likelihood of the random samples $\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_N$ obtained from the test image

$$\{\hat{\alpha}, P_1,\ldots, P_C\} = \arg \max_{\alpha,P_1,\ldots,P_C} \log \sum_{\mathrm{n}=1}^{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{C}} \mathrm{P_i} f\left(\mathbf{x}_{\mathrm{n}} \mid \boldsymbol{\mu}_i + \boldsymbol{\delta}_i \alpha, \boldsymbol{\Sigma}_i\right) + \lambda\left(\sum_{i=1}^{C} P_i - 1\right)$$

The parameters are solved iteratively using the EM-algorithm (Fukunaga 1990):

$$P_i = \frac{1}{N} \sum_{n=1}^{N} q_{i,n}$$

$$\alpha = \left(\sum_{n=1}^{N} \sum_{i=1}^{C} q_{i,n} \boldsymbol{\delta}_i^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}_i \right)^{-1} \sum_{n=1}^{N} \sum_{i=1}^{C} q_{i,n} \boldsymbol{\delta}_i^T \boldsymbol{\Sigma}_i^{-1} \left(\mathbf{x}_n - \boldsymbol{\mu}_i\right)$$

$$\text{where } q_{i,n} = \frac{P_i f_i\left(\mathbf{x}_n\right)}{\sum_{i=1}^{C} P_i f_i\left(\mathbf{x}_n\right)} \text{ and } f_i\left(\mathbf{x}_n\right) = f\left(\mathbf{x}_{\mathrm{n}} \mid \boldsymbol{\mu}_i + \boldsymbol{\delta}_i \alpha, \boldsymbol{\Sigma}_i\right)$$

5.3.2 Cloud shadow detection

To detect cloud shadow areas in the image we first detect the cloud areas using the algorithm suggested in Section 5.3.1. The detected cloud pixels are then masked out in the test image. To detect cloud shadow areas we apply (almost) the same procedure again, and acquire a set of random samples from the test image (but now we have excluded the cloud pixels). For the cloud shadow detection task, we model the class dependent mean vector and covariance matrix as

$$\boldsymbol{\mu}_i = \left[\boldsymbol{\mu}_{i,1},\ldots,\boldsymbol{\mu}_{i,K}\right]\boldsymbol{\alpha} = \mathbf{M}_i\boldsymbol{\alpha}, \quad \boldsymbol{\alpha} \geq \mathbf{0}$$

$$\boldsymbol{\Sigma}_i = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\Sigma}_{i,k}$$

The parameter vector $\boldsymbol{\alpha}$ is unknown, but constrained to contain only positive elements, and will be estimated from the random samples obtained from the test image. The test image is again modelled as a mixture of Gaussians, but with the mean vector of class *i* equal to $\mathbf{M}_i\boldsymbol{\alpha}$

$$f(\mathbf{x}) = \sum_{i=1}^{C} \mathrm{P}_i f_N(\mathbf{x} \mid \mathbf{M}_i\boldsymbol{\alpha}, \boldsymbol{\Sigma}_i)$$

The unknown parameters are solved by optimizing the likelihood

$$\{\hat{\boldsymbol{\alpha}}, P_1, ..., P_C\} = \arg \max_{\boldsymbol{\alpha}, P_1, ..., P_C} \log \sum_{n=1}^{N} \sum_{i=1}^{C} \mathrm{P}_i f(\mathbf{x}_n \mid \mathbf{M}_i\boldsymbol{\alpha}, \boldsymbol{\Sigma}_i) + \lambda\left(\sum_{i=1}^{C} P_i - 1\right)$$

$$\text{subject to } \boldsymbol{\alpha} > 0.$$

In general, to solve optimization problems with equality and non-equality constraints is challenging (Chong og Zak 2001). We propose a solution that combines the EM-algorithm with the non-negative least squares (Lawson og Hanson 1974) in an alternating fashion. The parameters are solved iteratively as:

$$P_i = \frac{1}{N}\sum_{n=1}^{N} q_{i,n}$$

$$\hat{\boldsymbol{\alpha}} \text{ is solved by minimizing } \left\|\mathbf{b} - \mathbf{E}\boldsymbol{\alpha}\right\|^2, \boldsymbol{\alpha} \geq \mathbf{0} \quad \text{(non - negative least squares)}$$

$$\text{where } \mathbf{E} = \left(\sum_{n=1}^{N}\sum_{i=1}^{C} q_{i,n}\mathbf{M}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{M}_i\right), \quad \mathbf{b} = \sum_{n=1}^{N}\sum_{i=1}^{C} q_{i,n}\mathbf{M}_i^T\boldsymbol{\Sigma}_i^{-1}\mathbf{x}_n,$$

$$q_{i,n} = \frac{P_i f_i(\mathbf{x}_n)}{\sum_{i=1}^{C} P_i f_i(\mathbf{x}_n)} \text{ and } f_i(\mathbf{x}_n) = f(\mathbf{x}_n \mid \mathbf{M}_i\hat{\boldsymbol{\alpha}}, \boldsymbol{\Sigma}_i)$$

The proposed approach for retraining the training data to the test data distribution is based on a constrained low-rank modelling of the unknown parameters. For cloud detection we simply adjust the mean vector along the principal direction of the training data mean vector distribution. For the cloud shadow detection, the test mean vector is modelled as a linear combination of the training data mean vectors. Hence, the number of unknown parameters is *C·K*, which is much less than the approach proposed in (Bruzzone og Prieto 2001) that consists of

$$C \cdot K \cdot (1 + N_f + N_f^2)$$

parameters, where N_f is the number of features. The result of this low-rank estimation of the class dependent mean vectors is that the relative spectral differences between the classes are to a larger degree maintained after retraining than what is the case for the approach suggested in (Bruzzone og Prieto 2001). This is beneficial for our case since the physical meaning of the

classes is maintained after retraining, even if the spectral variation between the images is quite large.

5.3.3 Cloud shape matching

In order to reduce the number of falsely detected cloud shadows, we apply context information about the azimuth and elevation angle of the satellite and the sun.

The shadow of each cloud is located in the opposite direction relative to the sun apparent azimuth. Since the view angle of the satellite may be larger than zero, the projected position of the cloud in the image is different than its actual geographical position. However, even if the cloud height is unknown, it is possible to detail the shadow direction relative to the projected cloud position (Le Hégarat-Mascle og André 2009).

Let ϕ_{sat} denote the satellite azimuth angle (between north and the camera looking direction), and let ϕ_{sun} denote the sun azimuth angle. Further, let φ_{sat} and φ_{sun} denote the satellite and sun elevation angles, respectively. The direction from the projected cloud position and the cloud shadow is (Figure 5-4)

$$\theta = \tan^{-1}\left(\frac{-\tan(\varphi_{sun})\sin(\phi_{sun}) + \tan(\varphi_{sat})\sin(\phi_{sat})}{\tan(\varphi_{sun})\cos(\phi_{sun}) - \tan(\varphi_{sat})\cos(\phi_{sat})}\right)$$

For each cloud, the corresponding shadow is searched for by translating a binary mask B of the cloud shape in the direction θ. In practice this means that for each geometric translation step i in the θ direction, $B(\theta,i)$ is being the translated mask, and we calculated the mean image intensity $m(\theta,i)$ over $B(\theta,i)$. Note that when the intersection of the cloud and $B(\theta,i)$ is not empty, the pixels belonging to the intersection is not considered for the estimation of $m(\theta,i)$. The shadow location is given by the argument corresponding to the minimum of $m(\theta,i)$

$$i_{\min}(\theta) = \arg\min_{i} m(\theta, i)$$

The final shadow mask is the detected shadow area obtained by the cloud shadow classification, and corresponds to the location of i_{min}.

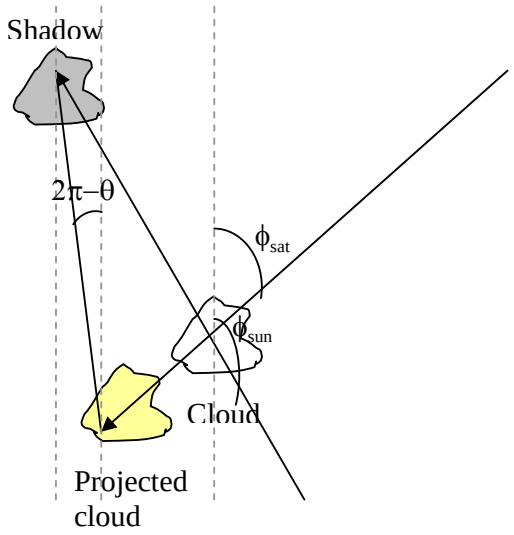


Figure 5-4: Sun, satellite and cloud geometry.

5.3.4 Contextual classification

Due to the size of clouds, a single pixel will rarely (or never) contain a cloud when its neighbours are non-cloud pixels, and vice versa. Thus to exploit this contextual information we apply a Markov random field (Besag 1986) (Solberg 2004) to model the context between neighbouring pixels in the classification. We solve for the optimal class configuration using iterated conditional modes (ICM) (Besag 1986).

5.4 Summary cloud and cloud shadow detection

The method for cloud and cloud shadow detection may be summarized as follows. First, the clouds are detected by a classification procedure (Section 5.3.1), then ignoring the cloud positions the cloud shadows are detected (Section 5.3.2), and finally the false detected clouds are removed by the cloud shape matching algorithm (Section 5.3.3).

6 Results and discussion

6.1 Automatic road detection results

In the following we show a selection of images and subimages to illustrate results obtained with the road segmentation for different areas and images. The image set consists of the following:

- Quickbird, E16, Sollihøgda
- Quickbird, Østerdalen south, RV3
- Quickbird, Østerdalen north, RV3.
- (Quickbird, Eiker, EV134, RV35.)
- World View 2, Nordkjosbotn.

6.1.1 E16, Sollihøgda

The Quickbird image covering Sollihøgda was processed for automatic extraction of the road E16 (Figure 6-1 - Figure 6-3). We see that a smoother road mask may be obtained by applying the panchromatic image in addition to the multispectral image (Figure 6-2). The algorithm worked well in general (e.g. Figure 6-3), but problems probably due to changes in elevation occurred (Figure 6-4, left). This is expected to work better if the images are geo-referenced using a DEM. The algorithm was also capable of handling tunnels (Figure 6-4, right), but GIS info is needed in order to label the tunnel segments prior to the vehicle detection stage.

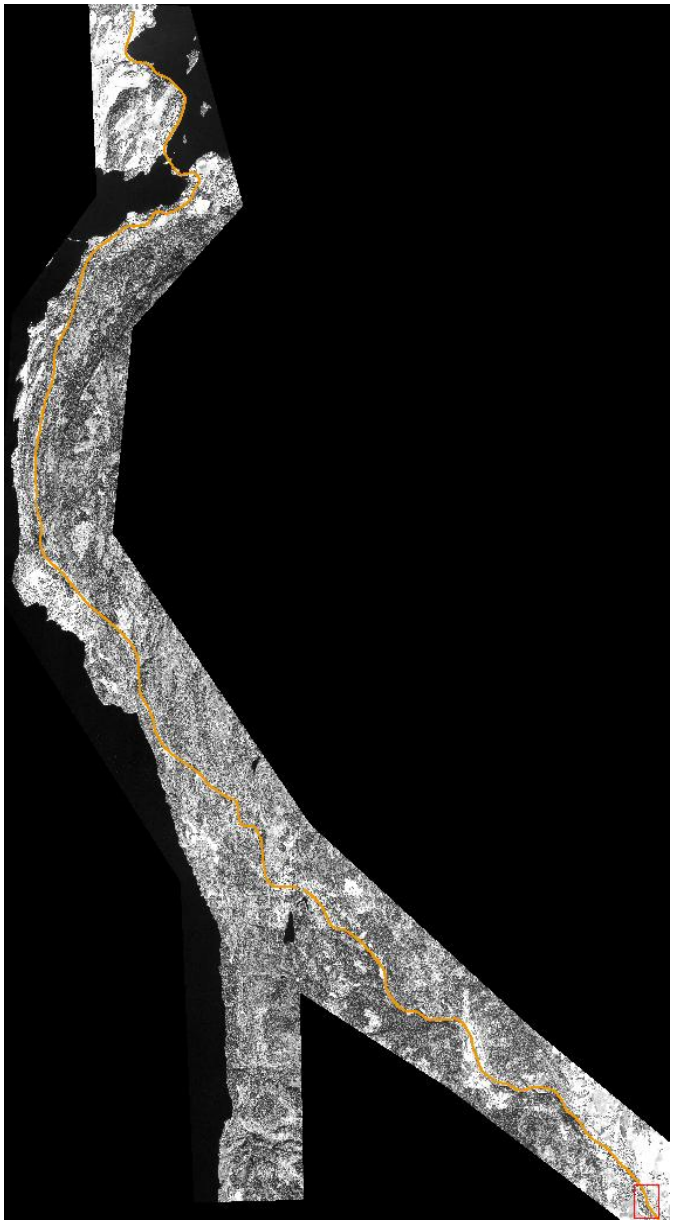


Figure 6-1: Total image and segmented road.

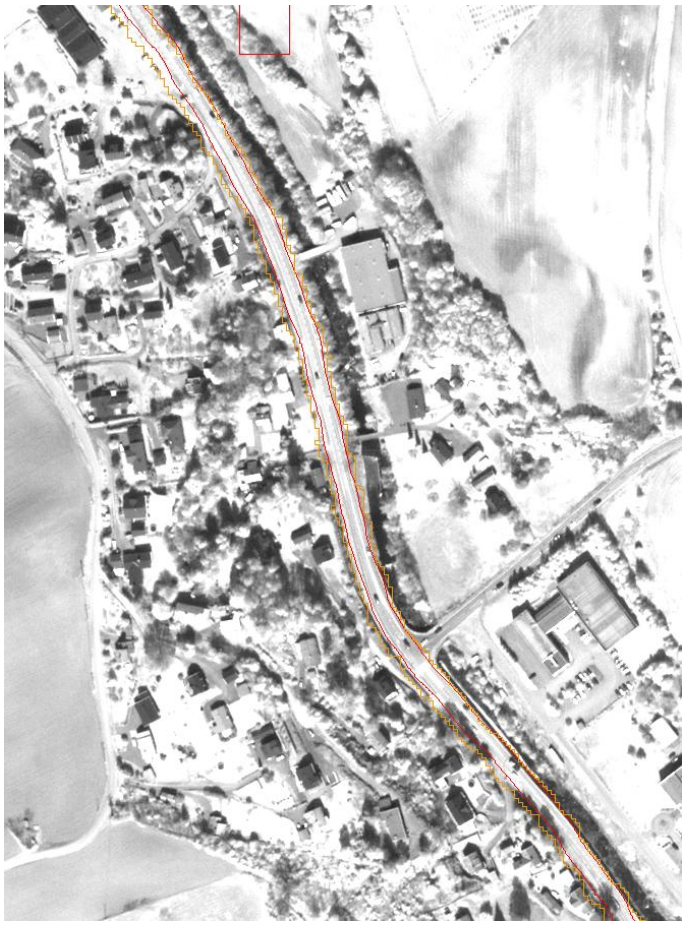


Figure 6-2: Subimage from the above, showing the outline of the coarse road initially found from the multispectral image (orange) and the smoother road found from panchromatic refinement of this (red).

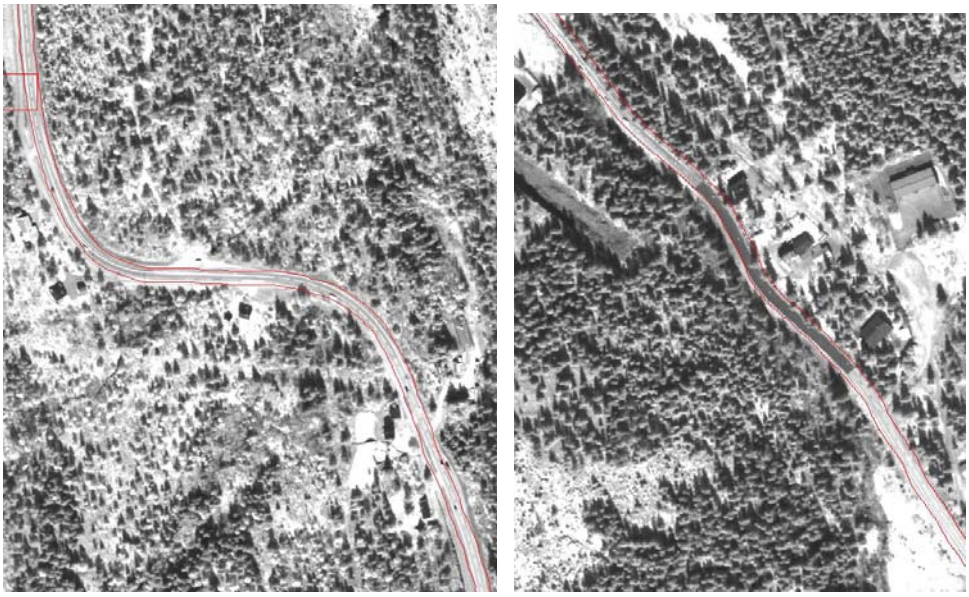


Figure 6-3: Subimages from other areas along E16, showing the outline of the refined panchromatic mask.

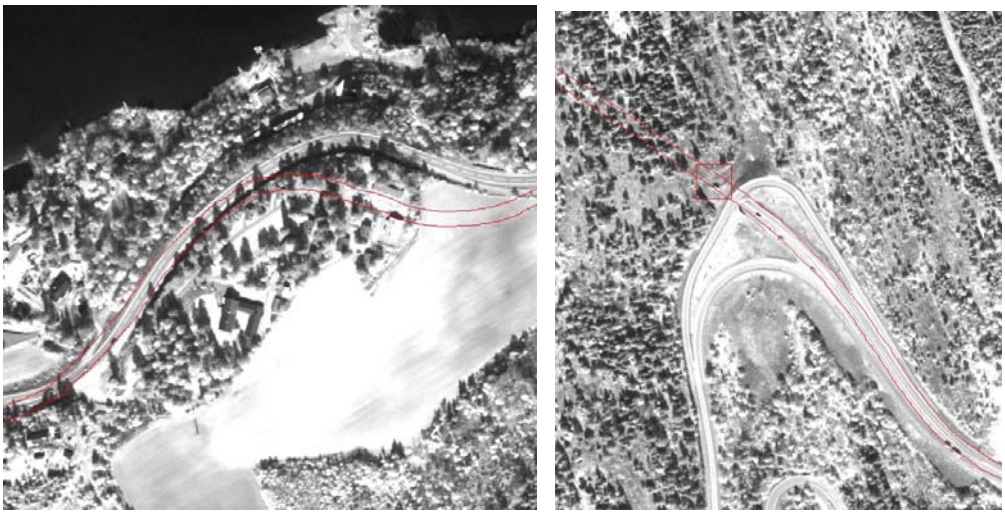


Figure 6-4: E16 showing two types of problems. **Left:** The segmentation has failed due to a dark road surface and bright nearby areas which both happens at a place where there is a large inconsistence between the vector data and the image data in terms of geographic position. Problems like this may be helped through the use of automatic geo-correction based on a digital elevation model prior to the segmentation. **Right**: The road proceeds in to a tunnel. The segmentation is not actually wrong here, but additional information from the GIS telling the vehicle detection that there is a tunnel here and that detection is not possible, will be needed.

6.1.2 RV3, Østerdalen south

The Quickbird image covering Østerdalen south was processed for automatic extraction of the road RV3 (Figure 6-5Figure 6-7) using the multispectral image only. The algorithm is able to successfully detect the road in the image, but the boundary of the road mask is quite rugged (Figure 6-6Figure 6-7). In particular note that the algorithm is robust with respect to the existence of trees and tree shadows occluding parts of the road (Figure 6-6Figure 6-7).

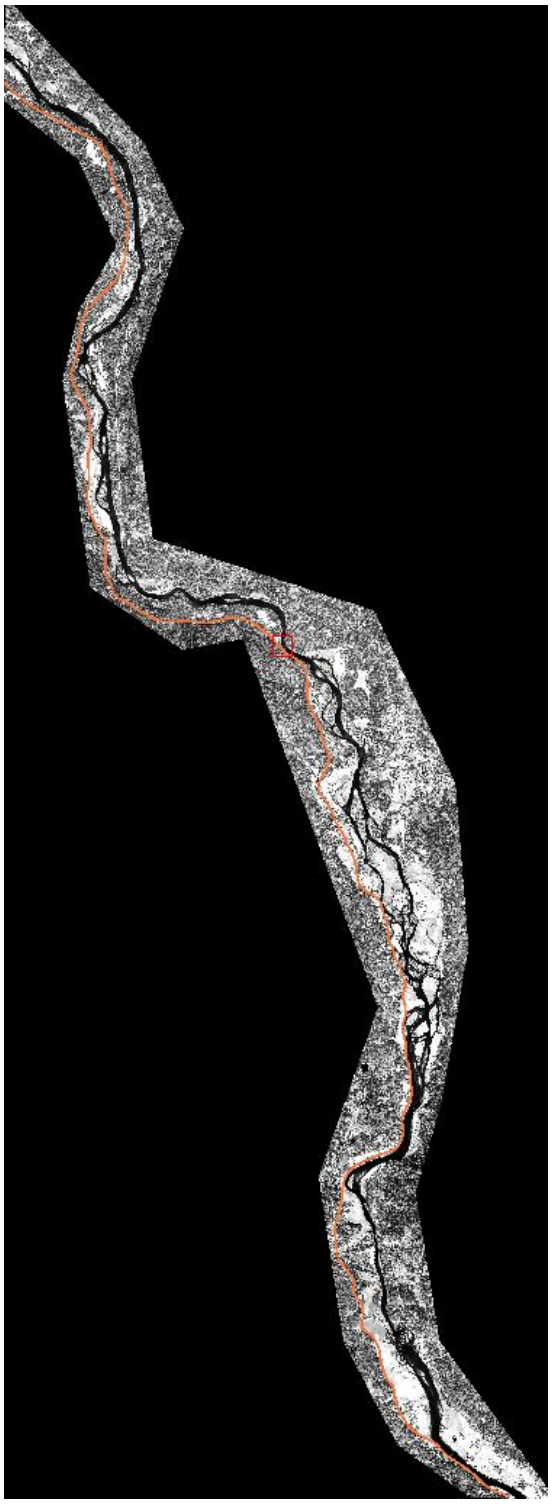


Figure 6-5: Entire road segmented in Østerdalen sør image.

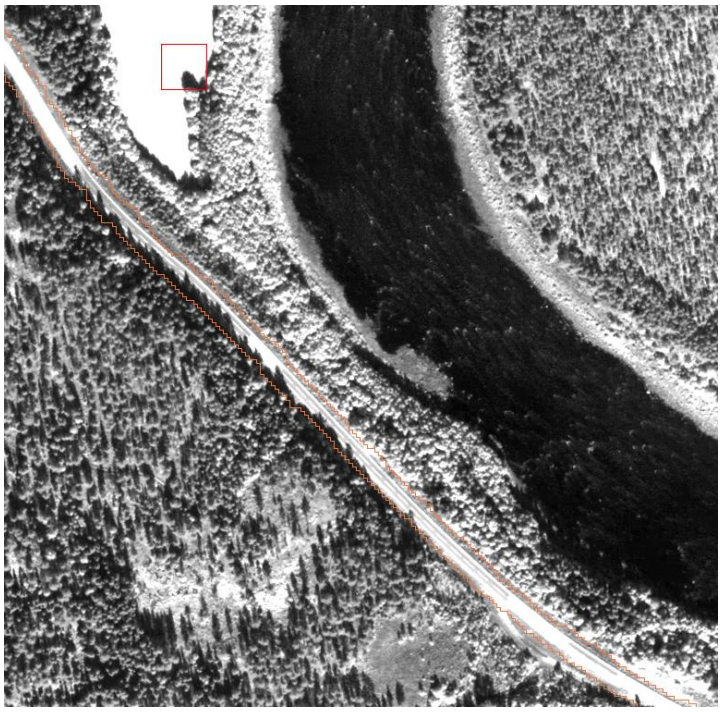


Figure 6-6: Part of segmented road.

Figure 6-7: Part of the segmented road

6.1.3 RV3, Østerdalen north

The Quickbird image covering Østerdalen north was processed for automatic extraction of the road RV3 (Figure 6-8 -Figure 6-10) using both the multispectral and panchromatic image. The algorithm is able to successfully detect the road in the image, and the boundary is precisely located with a smooth boundary (Figure 6-9Figure 6-10). In particular note that the algorithm is robust with respect to the existence of trees and tree shadows occluding parts of the road (Figure 6-9), and also handles road cover with changing spectral response (Figure 6-10, right).

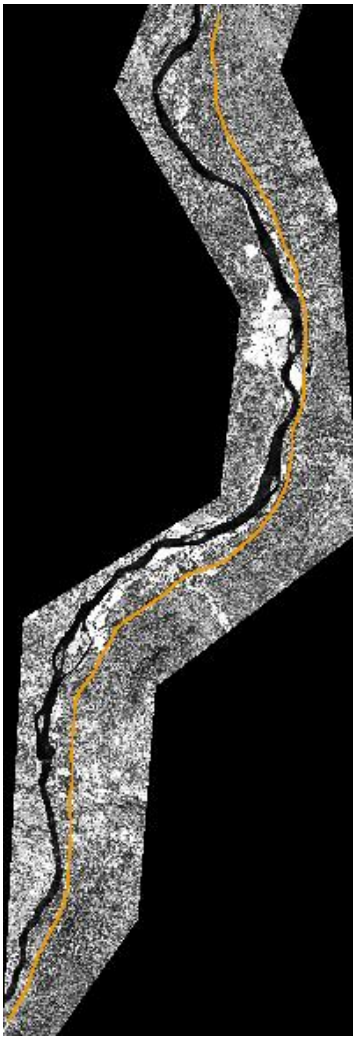


Figure 6-8: Segmented road through whole image.

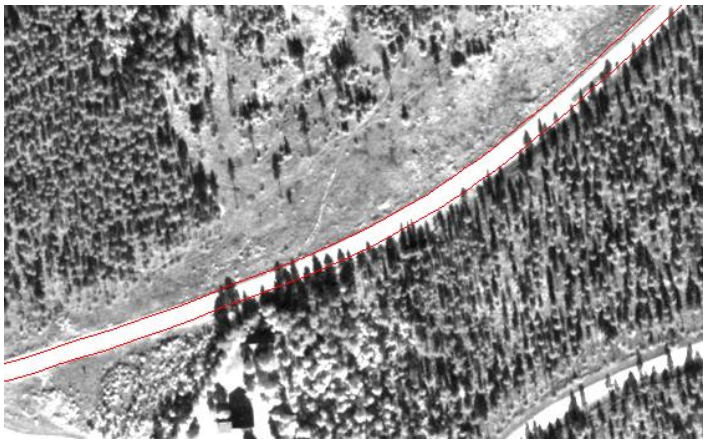


Figure 6-9: Section of segmented road with outline of refined panchromatic mask.

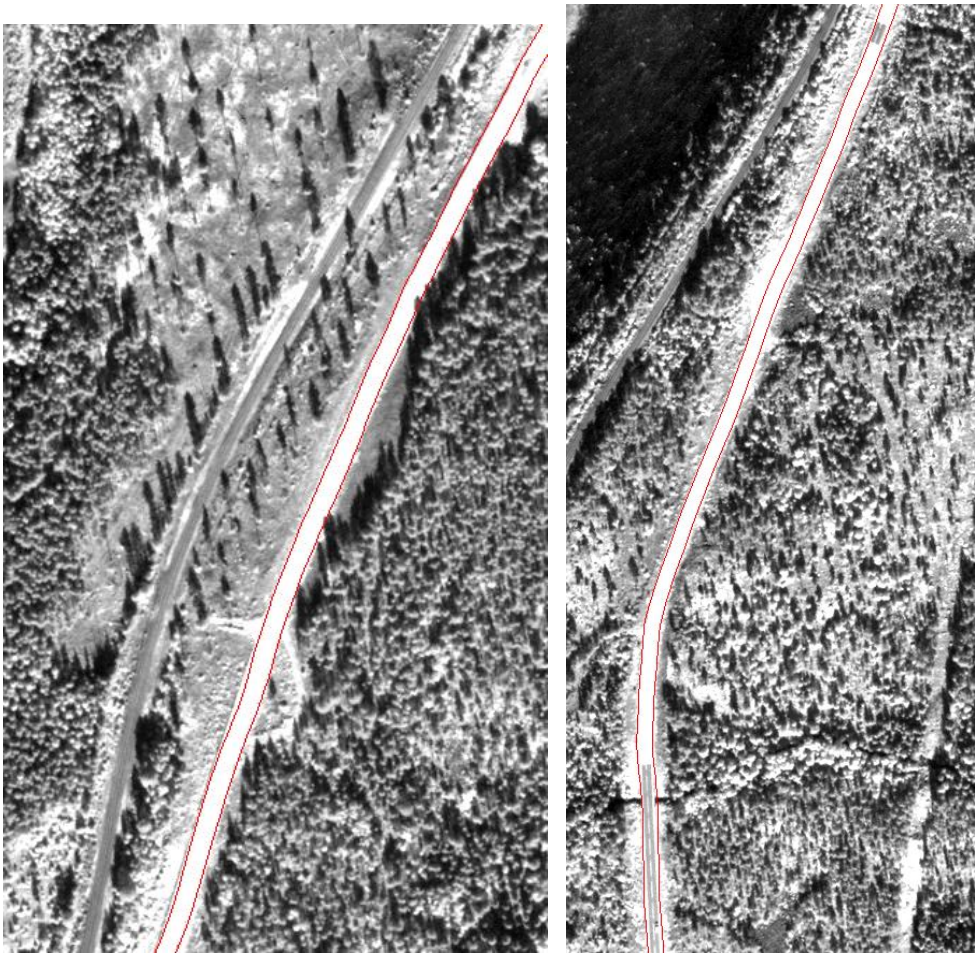


Figure 6-10: Outline of refined panchromatic mask.

6.1.4 Eiker

The Quickbird image covering Eiker was processed for automatic extraction of the road E134/RV35 (Figure 6-11Figure 6-14) using the multispectral image only. The road segmentation is designed for single trajectories, and complex road structures like roundabouts can confuse

the algorithm resulting in a failure in the segmentation (Figure 6-12). However, as long as the road vector describes single trajectories, the segmentation may work successfully even when the image contain complex areas (Figure 6-13). A problem with urban areas is that many objects may appear bright in the multispectral image (blue band) (Figure 6-14). Since the algorithm is designed for rural areas, it may produce erroneous results in these cases.

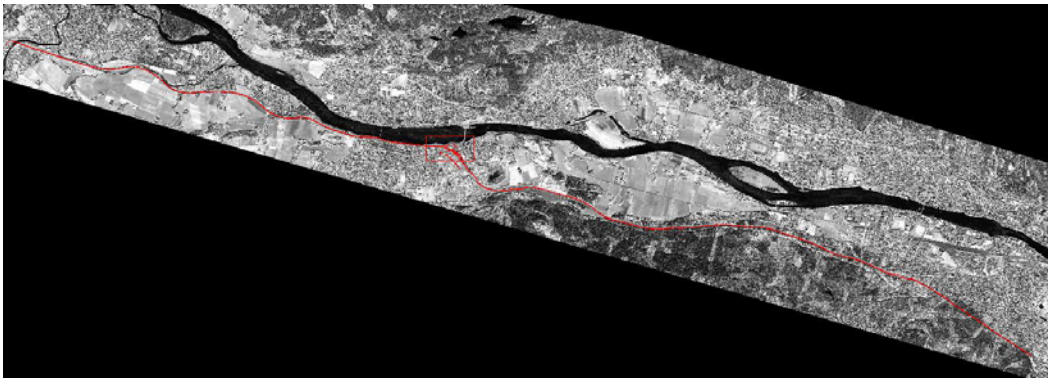


Figure 6-11: Segmented road.

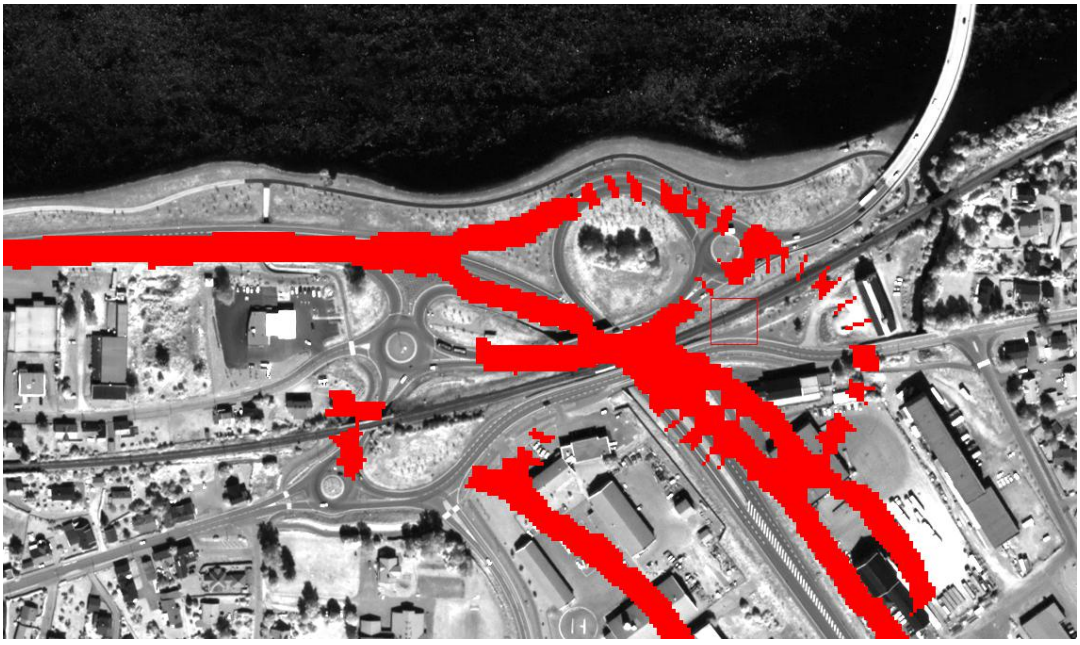


Figure 6-12: The road segmentation is designed for single trajectories. Complex structures resulting in more than one trajectory along the road can lead to failures as seen here where there is both a complex road vector and complex surroundings.

Figure 6-13: Subparts of successfully segmented road.

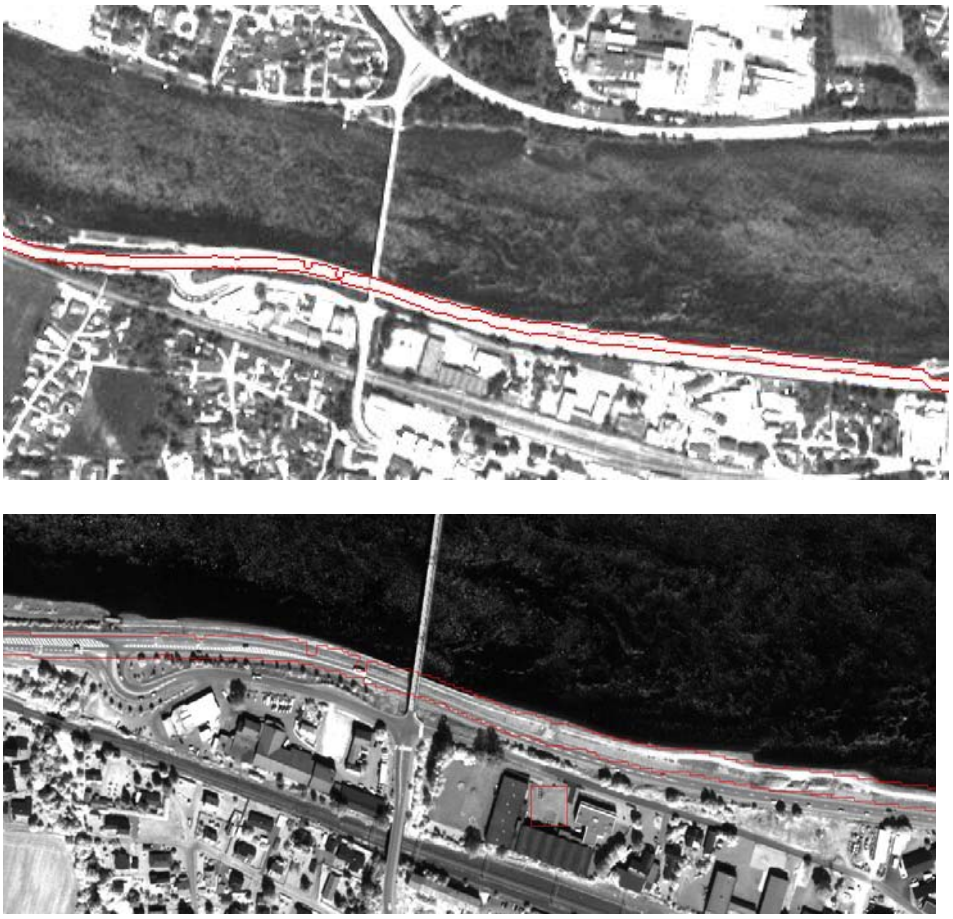


Figure 6-14: These two images show the same area in multispectral (top) and panchromatic resolution (bottom) with the road mask superimposed. This shows that in an urban area like this several objects can appear as bright areas in the multispectral image and lead to an erroneous trace. The approach is however primarily designed rural areas. For urban areas like this an extension of the approach where several multispectral bands are used could be considered.

6.1.5 Nordkjosbotn

The Quickbird image covering the area south of Nordkjosbotn was processed using both the multispectral and panchromatic image for automatic extraction of the road E6 (Figure 6-15, Figure 6-18). For many parts of this image the panchromatic refinement method using both the multispectral and panchromatic image provided good results (Figure 6-16, Figure 6-17). However, bright areas close to the road may confuse the algorithm (Figure 6-18).

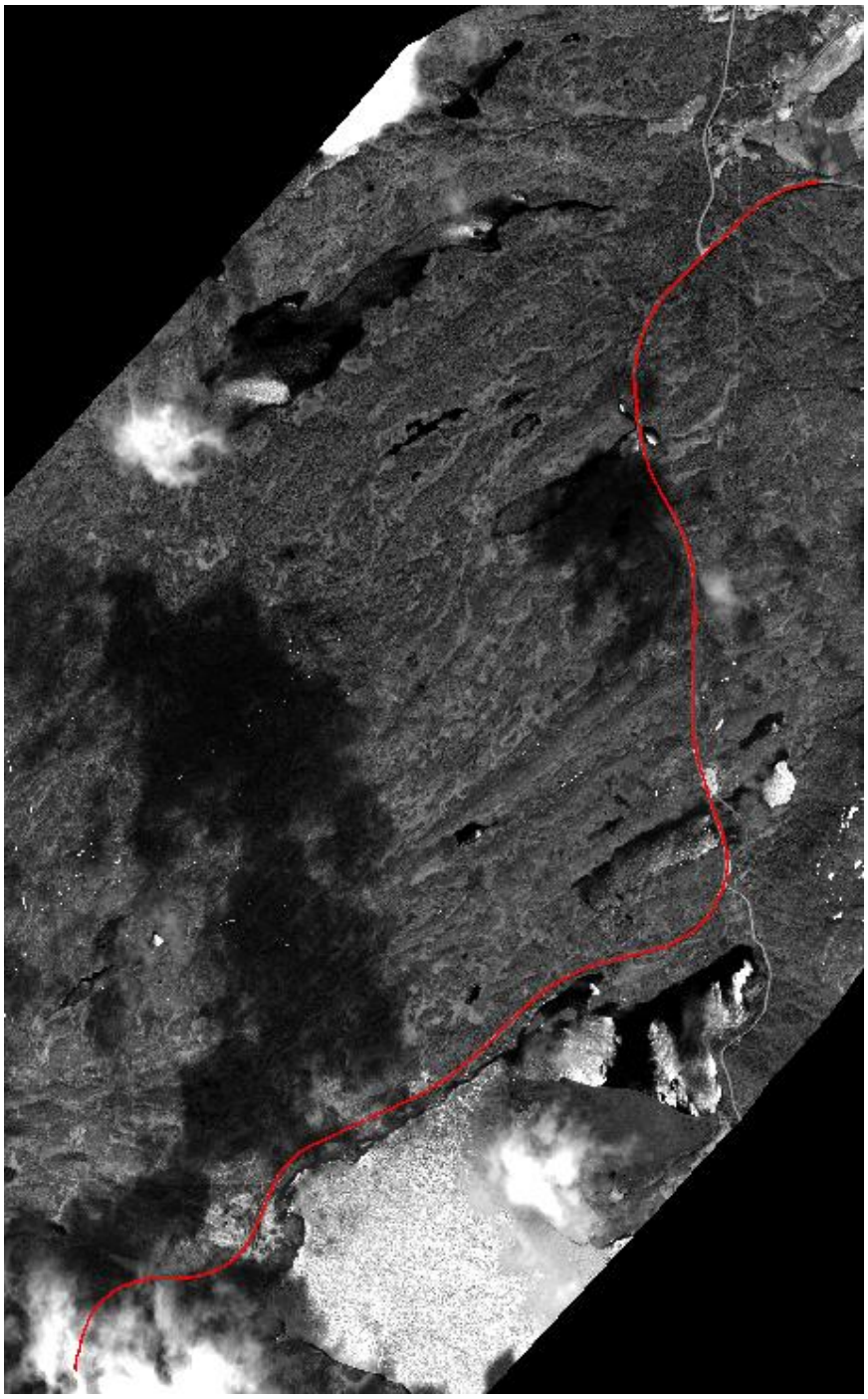


Figure 6-15: Road segmented in the image.

Figure 6-16: Part of segmented road after panchromatic refinement.

Figure 6-17: Part of segmented road after panchromatic refinement.

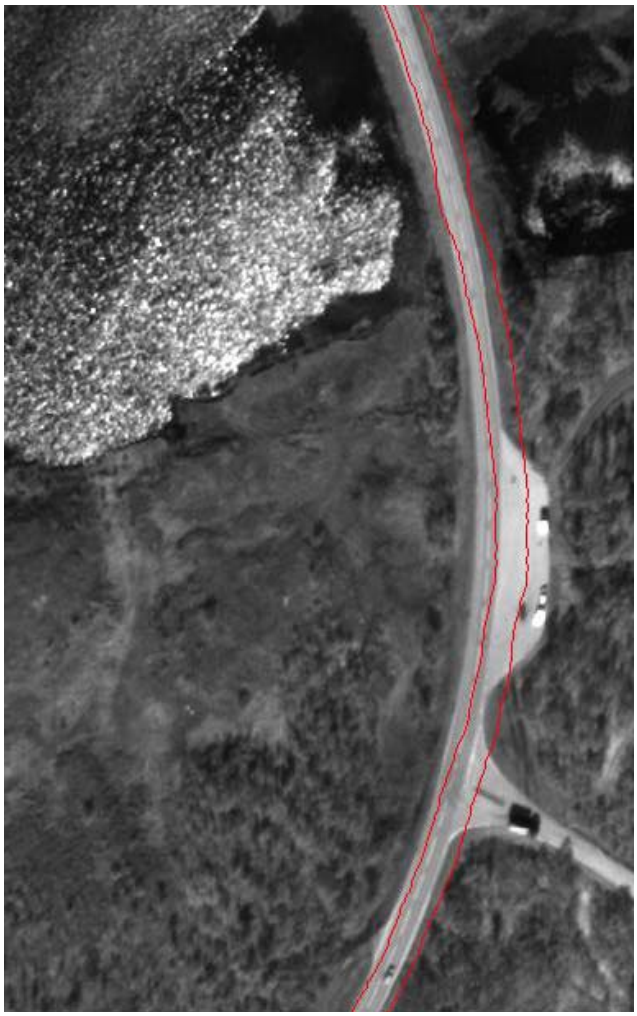


Figure 6-18: Here a problem occurs in the segmentation due to a bright area very close to the road.

6.2 Automatic cloud and cloud shadow detection results

The training data is collected from 4 different images, 2 Quickbird images from Hedrum and Tønsberg and 2 WorldView-2 images from Nordkjosbothn. For each training image the mean vector and covariance matrix of each class was estimated.

We evaluated the performance of the proposed cloud and cloud shadow detection procedure by visually inspecting the resulting cloud/shadow maps. The algorithm was applied to a large set of Quickbird and WorldView-2 images (Figure 6-19, Figure 6-25), covering many different areas in Norway. The images covering Tjølling and Stavern (Figure 6-19 and Figure 6-21) contain a lot of haze. However, the algorithm was able to successfully identify clouded areas. Due to the haze, no cloud shadowed areas were present in the images. The image covering Riksvei 3 in Østerdalen (Figure 6-22) is particularly interesting since we were successfully able to delineate the shadowed areas covering the road. The images covering the Mosjøen area (Figure 6-24 and Figure 6-25) did not contain any clouds, but a lot of haze. Also here the algorithm was able to distinguish clouds from haze. However, some snow covered areas were misclassified as clouds. Many of the Nordkjosbotn images (Figure 6-27) also contained a lot of haze, but no clouds. Also for these images the algorithm worked well. One Nordkjosbotn image (Figure 6-26) contained

both clouds and cloud shadows. The algorithm was able to detect both clouds and cloud shadows, and in particular, was able to indicate shadowed areas covering E6 in the image.

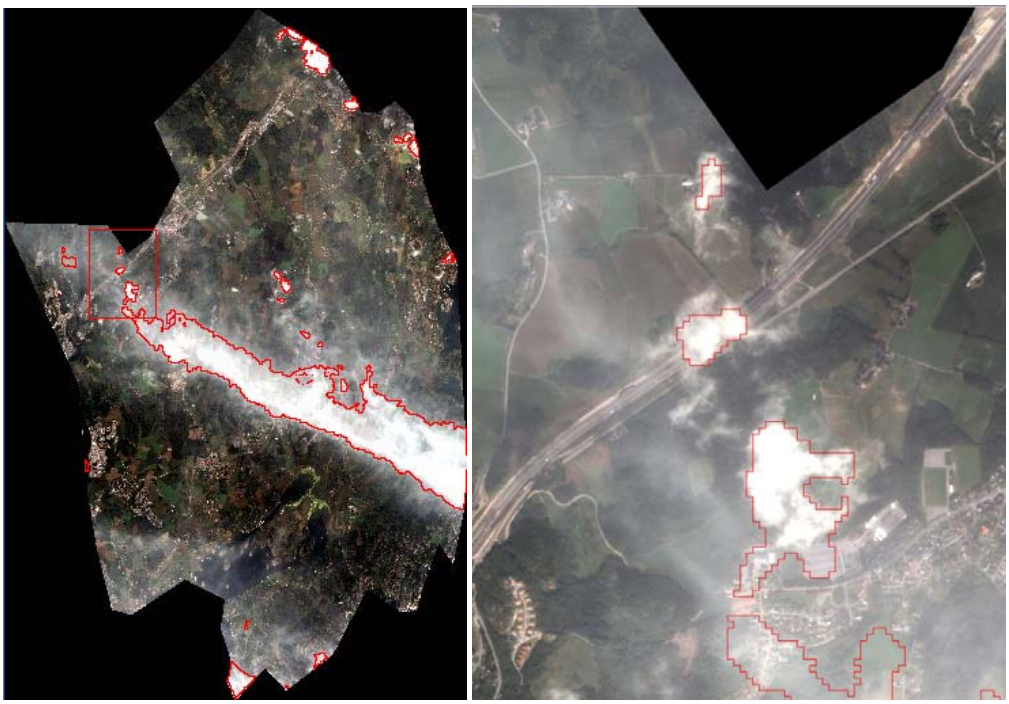


Figure 6-19: Left image: Cloud map of the Tjølling July 24, 2009 Quickbird image. Red lines indicate detected clouds. Right image: Zoom indicated by the rectangle. No detected cloud shadows in the Tjølling image.

Figure 6-20: Left Image: Cloud/shadow map of the Ørland August 7, 2004 Quickbird image. Red lines indicate detected clouds, and blue lines indicate detected cloud shadows. Right image: Zoom indicated by the rectangle.

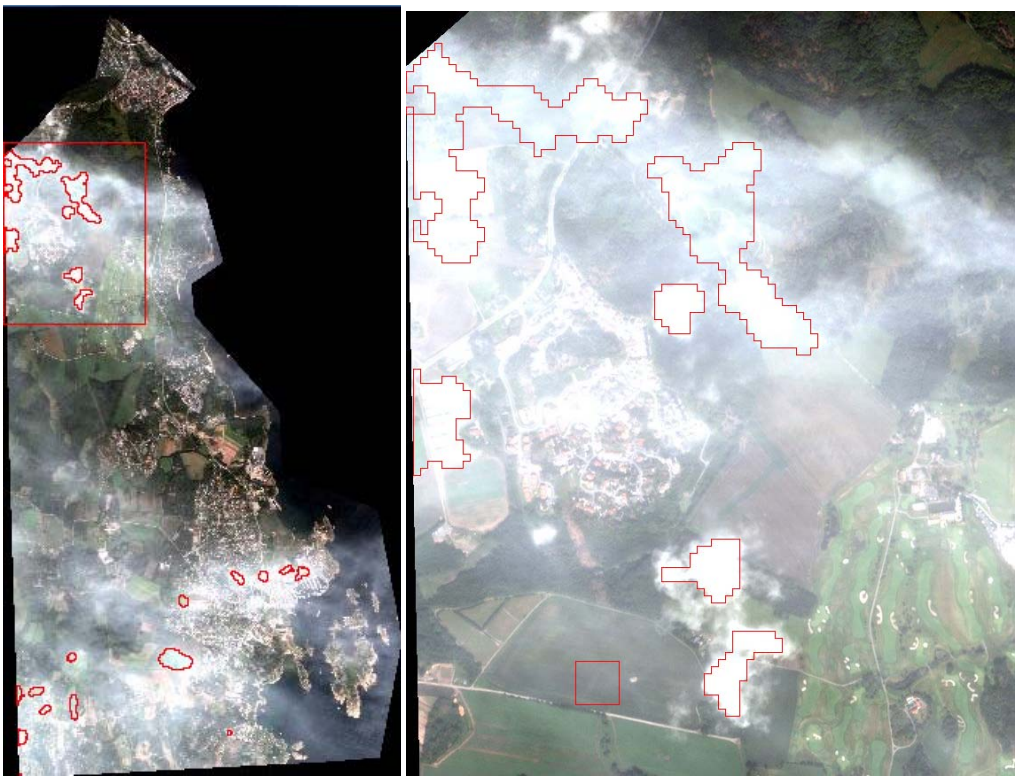


Figure 6-21: Left Image: Cloud map of the July 24, 2009 Stavern Quickbird image. Red lines indicate detected clouds. No detected cloud shadows. Right image: Zoom indicated by the rectangle.

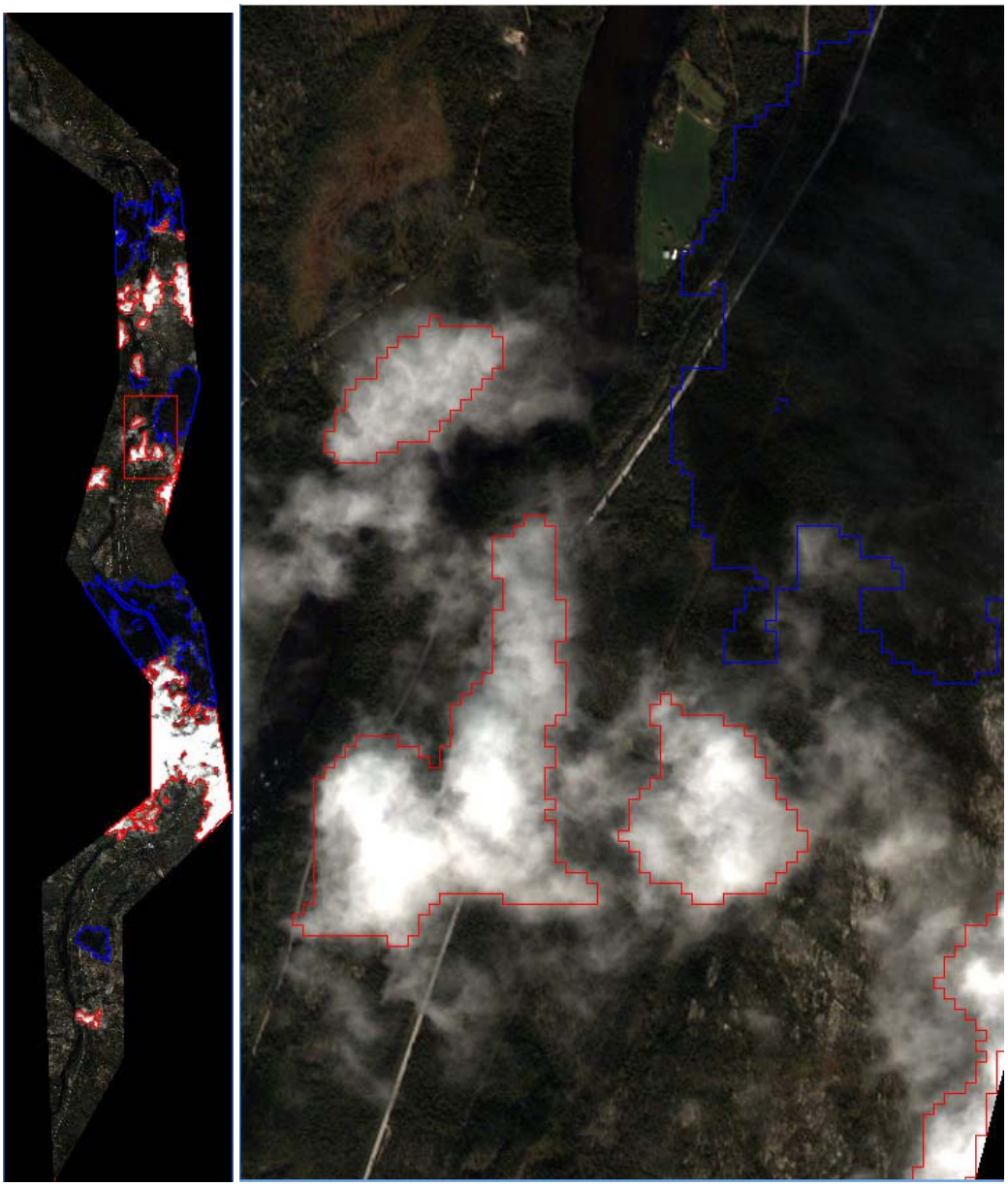


Figure 6-22: Left Image: Cloud/shadow map of the Østerdalen Sep. 6, 2009 Quickbird image. Red lines indicate detected clouds, and blue lines indicate detected cloud shadows. Right image: Zoom indicated by the rectangle.

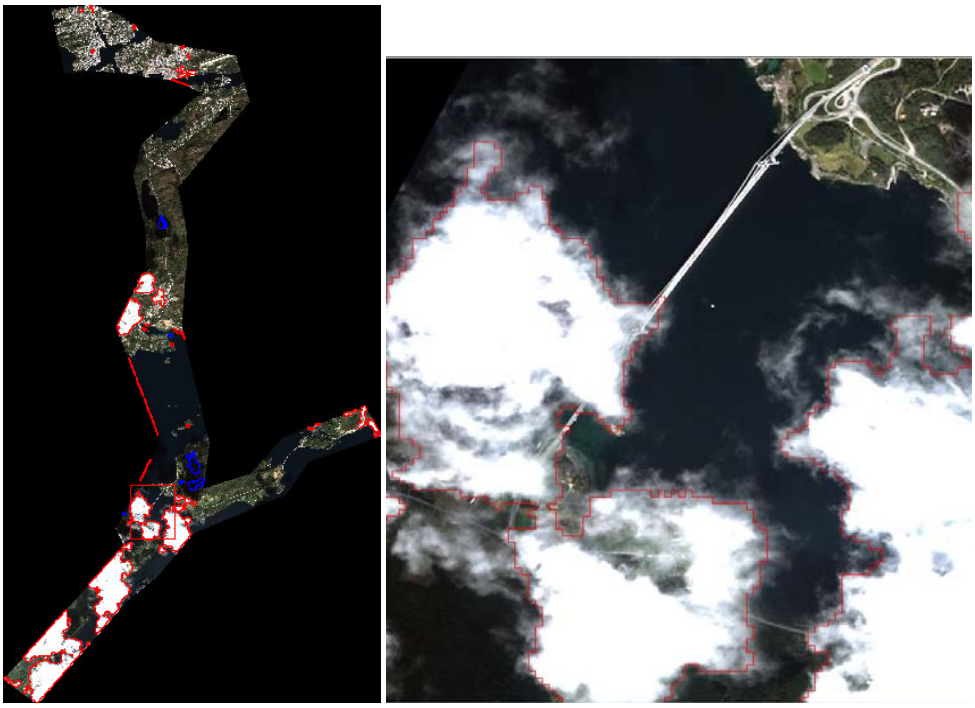


Figure 6-23: Left image: Cloud/shadow map of the Kristiansund July 8, 2008 Quickbird image. Red lines indicate detected clouds, and blue lines indicate detected cloud shadows. Right image: Zoom indicated by the rectangle

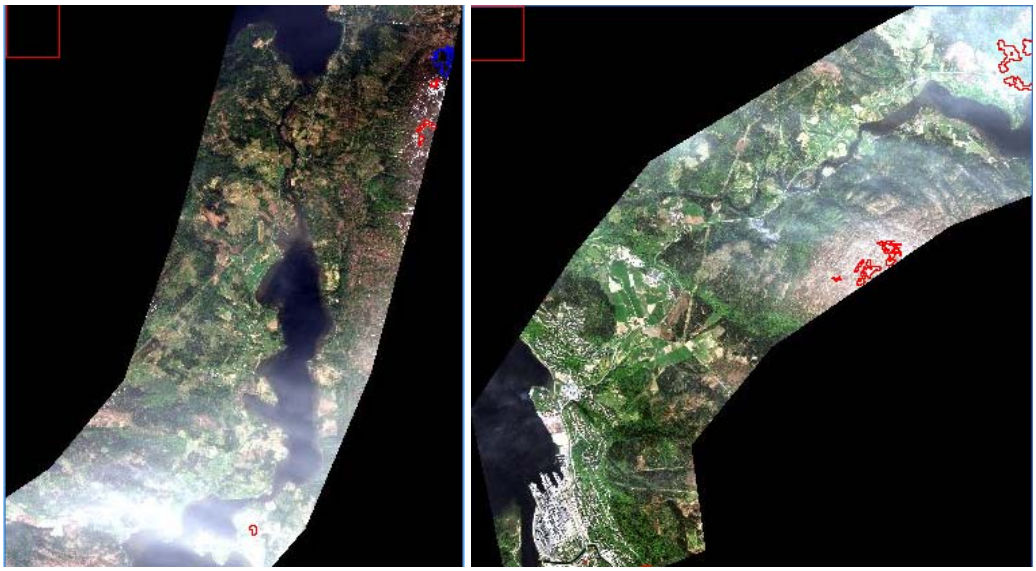


Figure 6-24: Cloud maps of two Mosjøen WorldView-2 images acquired on the June 10, 2010. Redlines indicate detected clouds.

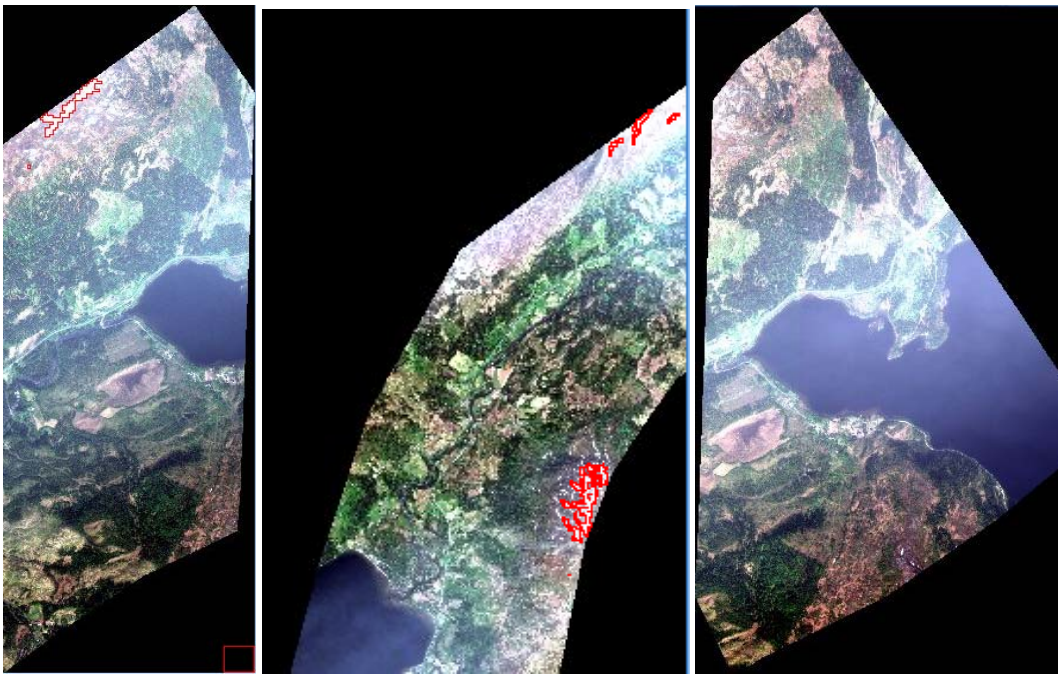


Figure 6-25: Cloud maps of three Mosjøen WolrdView-2 images acquired on June 10, 2010. Red lines indicated detected clouds.

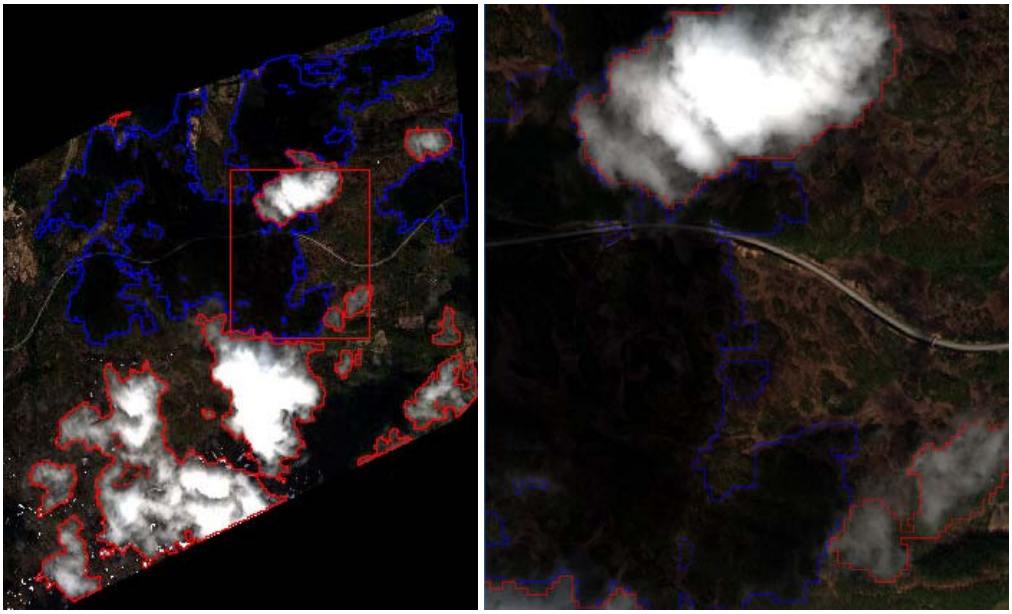


Figure 6-26: Left: Cloud/shadow map of the June 11, 2010 Nordkjosbotn WorldView-2 image. Redlines indicated clouds whereas blue lines indicate detected cloud shadows. Right image: Zoom indicated by the rectangle.

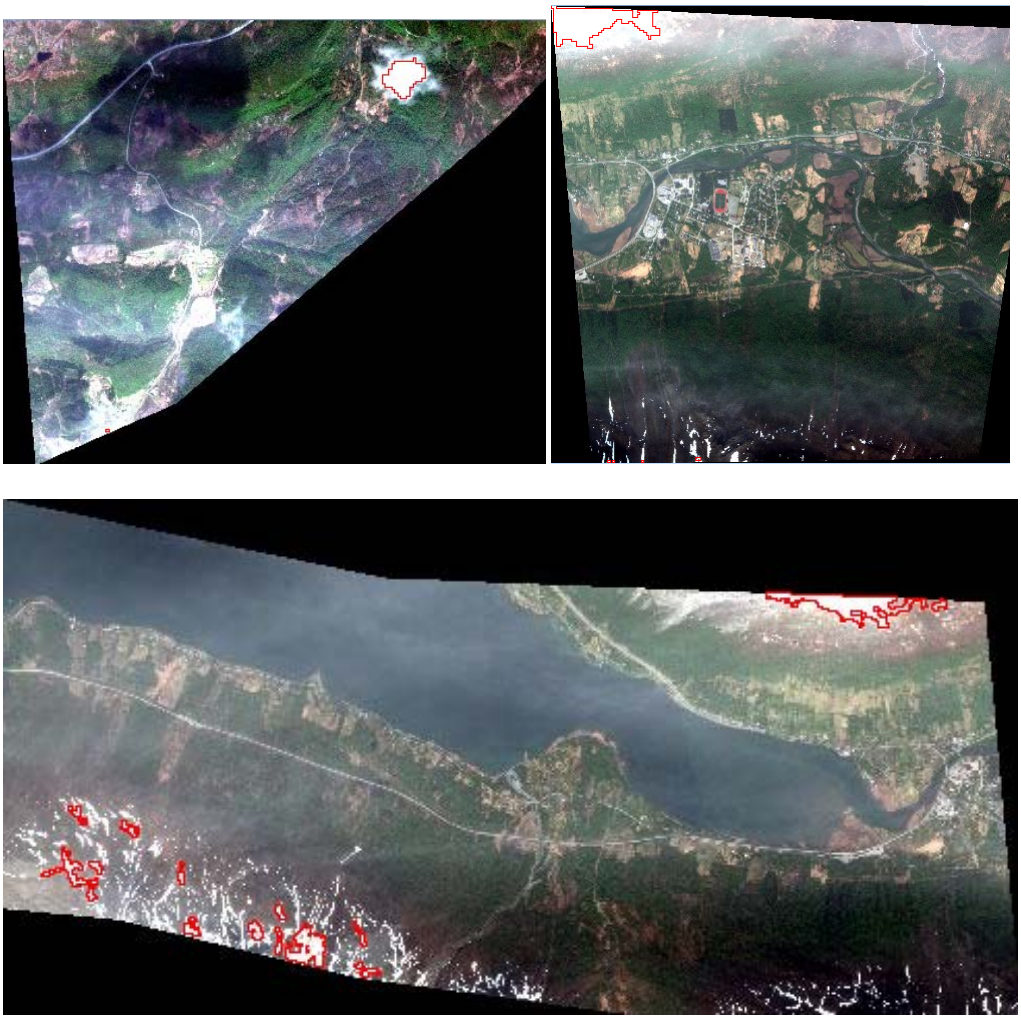


Figure 6-27: Cloud maps of three Nordkjosbotn WorldView-2 images acquired on June 11, 2010. Red lines indicate detected clouds.

6.3 Discussion and conclusion

During 2010 we have developed methods for road segmentation and cloud and cloud shadow detection, steps necessary to automate a process for extracting satellite based traffic statistics.

6.3.1 Road segmentation

The current road segmentation algorithm is currently designed to work for cases were there are few disturbing elements in or near the road. Some constraints of the algorithm are:

- It assumes that the vector data describes a single road trace. Hence, roundabouts and similar structures may cause problems and reduce the performance of the algorithm.
- It is designed and tested for summer conditions
- Tunnels are not handled. Hence, the algorithm may be confused if the road suddenly disappears. This might be solved by using GIS data that label tunnel segments.

- For increased performance, geo-referenced images are preferred, in particular if the altitude of the road is varying abruptly.

The proposed algorithm is a "proof-of-concept" method for road segmentation, and is not evaluated in combination with the vehicle detection and classification algorithms. This issue is important and need to be done since it would reveal the algorithm's performance using the final performance metric, i.e. the classification accuracy.

6.3.2 Cloud/shadow detection

The method for detection of clouds and cloud shadows is divided into two stages. In the first stage we detect the clouds, and in the second we detect the cloud shadows. The reason for this is that the features NIR/green and NDVI results in a poor cloud classification, whereas in order to resolve the shadowed areas from other areas (such as green vegetation), the two last features are necessary.

We also observed that modelling the dataset shift using the simple eigenvector offset was sufficient for cloud detection; however, it did not provide any good detection of cloud shadows. Please note that for cloud detection the classification performance of other land cover types is of poor quality. This is however no problem for the shadow detection.

In this study we did only considered a Gaussian classifier. However, the two-stage algorithm may be applied to any parametric classifiers, but the estimation of the unknown parameters may be more challenging. Another solution may be to improve the definitions of the classes, e.g. by selecting classes that possess a high goodness-of-fit to the multivariate Gaussian distribution. Inclusion of a digital elevation model may improve the classification of shadowed areas. E.g. the slope may help us to distinguish water regions from cloud shadows since water regions always have a slope equal to zero.

We evaluated the performance of the cloud/shadow detection algorithm visually. However, a more precise assessment may be obtained by defining a set of test samples and evaluating the classification accuracy in terms of a confusion matrix or kappa statistics.

Please note that the algorithm is designed for summer images. Including a winter image may have some undesirable effects since the spectral signature of snow is similar to clouds.

Further refinement of the cloud and cloud shadow classification is possible by investigating the problem at several scales. Here we have down-sampled the resolution to 19.2 m (16m for WorldView-2). By including images with high resolution, finer edge details of clouds and cloud shadows may be obtained.

References

Besag, J. "On the statistical analysis of dirty pictures." *J. Royal Statist. Soc. B.* 48, no. 3 (1986): 259-302.

Bruzzone, L., and D. F. Prieto. "Unsupervised retraining of a maximum likelihood classifier for the analysis of multitemporal remote sensing images." *IEEE Trans. Geosci. Remote Sensing* 39, no. 2 (2001): 456-460.

Chong, E. K. P. , and S. H. Zak. *An Introduction to Optimization.* Second. New York: Wiley, 2001.

Fukunaga, K. *Introduction to Statistical Pattern Recognition.* Second. Boston, Mass.: Academic Press, 1990.

Irish, R. R., J. L. Barker, S. N. Goward, and T. Arvidson. "Characterization of the Landsat-7 ETM+ automated cloud-cover assessment (ACCA) algorithm." *Phototramm. Eng. & Remote Sensing* 72, no. 10 (2006): 1179-1188.

Lawson, C. L., and R. J. Hanson. *Solving Least Square Problems.* Englewood Cliffs, N. J.: Prentice Hall, 1974.

Le Hégarat-Mascle, S., and C. André. "Use of Markov random fields for automatic cloud/shadow detection on high resolution optical images." *ISPRS J. Phogramm. Remote Sesning* 64 (2009): 351-361.

Liang , S., H. Fang, and M. Chen. "Atmospheric correction of Landsat ETM+ land surface imagery - Part I: Methods." *IEEE Trans. Geosci. Remote Sensing.* 39, no. 1 (2001): 2490-2498.

Lu, D. "Detection and substitution of clouds/hazes and their cast shadows on IKONOS images." *Int. J. Remote Sensing* 28, no. 17-18 (2007): 4027-4035.

Quiñonero-Candela, J., M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. *Dataset Shfit in Machine Learning.* Edited by J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer and N. D. Lawrence. Cambridge, Mass.: MIT Press, 2009.

Solberg, A. H. S. "Flexible nonlinear contextual classification." *Pattern Recog.* 25 (2004): 1501-1508.