Analysis of Multispectral Reconnaissance Imagery for Target Detection and Operator Support

Oskar Brattberg and Jörgen Ahlberg
Dept. of IR Systems, Div. of Sensor Tecnology
Swedish Defence Research Agency (FOI)
P.O Box 1165, SE-581 11 Linköping, Sweden.
{seekoman,jorahl}@foi.se

Abstract

This paper describes a method to estimate motion in an image sequence acquired using a multispectral airborne sensor. The purpose of the motion estimation is to align the sequentually acquired spectral bands and fuse them into multispectral images. These multispectral images are then analysed and presented in order to support an operator in an air-to-ground reconnaissance scenario.

1 Introduction

Depending on the scenario, methods for detection of targets in imagery typically rely on spatial and spectral analysis. Spatial analysis can be used for extended targets, i.e., targets covering a large number of pixles. In multi- or hyperspectral imagery, spectral analysis can be used to detect extended as well as small, or even subpixel, targets, since each pixel contains intensity levels for several spectral bands. Moreover, using spectral analysis, targets where the spatial characteristics are distorted (targets in semi-hide or camouflage) can be detected. Naturally, spatial and spectral methods can be combined to produce more robust detections.

While spatial detection methods require a large spatial resolution (with respect to the target size and distance) spectral detection methods require spectral resolution, that is, a sensor that captures several spectral bands. A regular consumer color camera have three bands (red, green, and blue), but there exist cameras within the entire optical range from ultraviolet to longwave infrared and with a large variation in spatial, temporal, and spectral resolution.

This paper presents a system for target detection and recognition using an airborne multispectral sensor. The purpose of the system is operator support, i.e., semi-automatic image analysis where a user can select which detections are intresting and should be remembered as targets and which detections should be incorporated in the background model.

The paper contains two parts; first the preprocessing of the image data from the sensor, and, second, the image analysis. The sensor specific preprocessing consists of alignment and fusion of the spectral bands. The image analysis consists of creating models of target and backgrounds and classifying new pixels.

2 Sensor and sensor data

The sensor used here is MultimIR [2], a multispectral midwave infrared camera with sensitivity in the 1.5–5.2 μm band. The sensor is a cooled MCT-detector (78 K) and has a spatial resolution of 384×288 pixels. The camera is equipped with a rotating filter wheel, which contains four filters enabling the sensor to register four spectral bands. The filter wheel rotates 25 revolutions/second and the sensor captures 100 images/second. Thus, there is a time lapse of 0.01 seconds between each band, and 0.04 seconds between each frame of four bands.

In the scenario treated here, the camera is mounted in a Cessna aircraft, looking straight down on the ground acquiring a sequence of images. The measurement is done during a flight over a military exercise area containing ground targets (vehicles) in forest and in open terrain.

When using the camera for aerial reconnaissance the delay of 0.01 seconds between the bands give rise to a large spatial displacement that need to be corrected to achieve correct spectral data. Moreover, the difference between the bands is affected by some dynamic factors like side sweeping winds, variation in speed, altitude, and attitude, and vibrations in the airplane transported to the camera rig. Static factors like optics and sensor array size need to be considered as well.

3 Motion correction

In order to perform spectral analysis, we need spectral vectors (pixels) where each component (spectral band) originates from the same point in the scene. Thus, we must align the captured images to a common coordinate system, that is, estimate the motion between consecutive images.

The simplest conceiveable motion model is a translation-only model. Here, we use affine transformations that better copes with changes in orientation and altitude of the aircraft. We select a set of patches (templates) in the current image, and search for the patches in the next image. From the set of correspondences we calculate the coordinate transformation between the images.

For this scheme to work, we need to consider the following:

- How do we select the templates?
- How do we match the templates with the next image?
- How do we match different spectral bands?

Selecting template patches

Due to time constraints, we do not want to use all pixels for motion estimation. Additionally, there are better and worse points in the scene to use for motion estimation. The best points would be small, single objects with homogeneous background, but since the images rarely contain such perfect points, we search for as good points as possible. Also, the points should be distributed over a large part of the image.

We thus divide the image into a number of blocks, and in each block, we select the pixel with the highest Harris corner measure [4]

$$\frac{(g * I_x^2) \cdot (g * I_y^2) - (g * (I_x \cdot I_y))^2}{g * I_x^2 + g * I_y^2}. \qquad (1)$$

I_x, I_y are derivatives estimated by simple 3 × 3 filters and g is a Gaussian smoothing kernel, and $*$ means convolution.

We use a blocksize of 40 × 40 pixels and use 48 blocks corresponding to the 320 × 240 pixels in the centre of the image. Patches centered around the the selected points are extracted, and the number of selected patches thus equals the number of blocks.

Matching and motion estimation

To align the first band with the following acquisition of the same spectral band (not the following image), we use patches of the size 7×7 pixels. The position of the patch is predicted from the previous motion estimate, except for the first time when the prediction is made from approximate knowledge of the aircraft's altitude and speed. To avoid a full search, the displacement from the prediction is maximized to 17 pixels in each direction. Within the search space, the coordinates $(x'_i\ y'_i)$ of the ith patch (originating at $(x_i\ y_i)$ in the previous image) are found using normalized zero-mean cross-correlation.

We use an affine motion model

$$\begin{pmatrix} a & b \\ d & e \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} c \\ f \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix} \qquad (2)$$

and find the parameters (a, b, c, d, e, f) by least squares using all 48 correspondences.

Visual inspection of the individual motion vectors gave that even though the majority of the indivdual motion vectors agreed, there were always a few outliers that would disturb the least squares solution. We thus use Random Sampling and Consensus (RANSAC) [3] to fit the model disregarding outliers. An example of how RANSAC compares to ordinary least squares in the presence of outliers is given in Figure 1. RANSAC selects a subset of the data points as inliers, and, in our case, it typically selects 40–45 points as inliers, which is enough to motivate the use of RANSAC (see Figure 2 (solid line)).

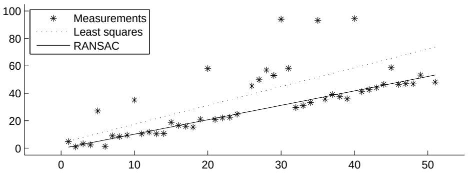

Figure 1: RANSAC compared to regular least squares.

Matching different spectral bands

The affine transformation is also used by splitting it up into four sequential transformation steps to predict the motion between the different spectral bands. Since the bands have quite different appearence, see Figure 3, regular template search is not useful. However, the transitions between materials appear at the same point in all bands, so instead of matching the bands directly, we match

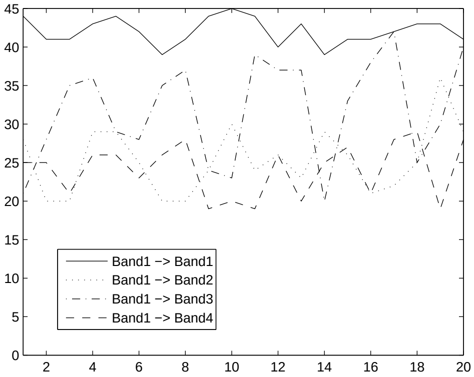

Figure 2: The number of inliers selected by RANSAC for the different bands. The solid line shows the number of inliers when aligning a band with the next image of the same spectral band. The dotted, dashed, and dash-dotted lines show the number of inliers when doing cross-band matching. The number of inliers are, as expected, much lower in the latter case.

the absolute values of the derivatives of the bands. This gives a much more uniform look of the bands. Now we can conduct template search and RANSAC approximation as before, however, since we have a better prediction we use a smaller search space (5×5 pixels). We also use larger patches (37×37 pixels). As can be seen in Figure 2, we get much fewer inliers in the cross-band motion estimation.

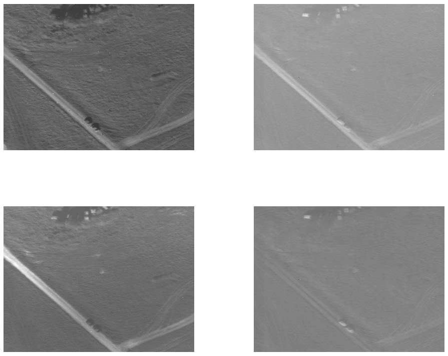

Figure 3: The four bands before processing.

Interpolation and fusion

Using the obtained transformations we create a multispectral image by fusing the four bands using bilinear interpolation. Additionally, for many pixels in the image, we have several measurements for each spectral band (since the images are largely overlapping). This can be used in two ways, one is to create an average in each pixel to give high redundance in the mesurement the other is to do the image analysis in each image alone and to improve the chance of detecting a subpixel target.

4 Spectral image analysis

When the motion correction and fusion is done, we have a sequence of multispectral images to analyse. Each pixel is a four-dimensional vector, and we seek to characterize those vectors. Our main tools are *anomaly detection*, *signature-based target detection*, and *spectral clustering*.

Anomaly detection is used to detect previously unseen targets by using a (spectral) background model and a distance function to measure the distance from the model to a sample (pixel). If the distance exceeds a threshold the pixel is considered an anomaly.

When you instead want to detect a known target (i.e., with known spectral signature) the distance from a target model to a sample should be below a threshold.

Spectral clustering refers to unsupervised segmentation based on the spectral vectors of a scene, typically using variants of K-Means or Expectation-Maximization.

Models can be trained from measurements or constructed from knowledge about the material. A background model can be constructed online under the (very reasonable) assumption that the amount of samples containing targets is small compared to true background samples, and thus use the entire image to construct a background model.

Methods for detection in and clustering of multi- and hyperspectral data are described in [1].

Cluster models

We create clusters of weighted multivariate Gaussians, where each cluster typically corresponds to one material type. In practice, several clusters are often needed to represent what we intuitively would call one material, for example grass. The weights correspond to the a priori probability (estimated from data) that a sample from the scene belongs to that cluster. Samples that do not seem to belong to any cluster are regarded as anomalies, and can be used for creating new clusters.

When the first multispectral image is processed, the Stochastic Expectation-Maximization (SEM) algorithm is run to create the initial clusters, and the image is segmented accordingly, as illustrated in Figure 4.

Pixels that have a large distance to all clusters are regarded as anomalies, and can be visualized as in Figure 5.

Updating the models

When new pixels arrive (i.e., the camera moves), they are classified using the cluster model, and the cluster model is updated according to the new pixels. A Gaussian cluster model with mean μ and covariance Σ estimated from N samples can be updated as samples are classified to belong to that cluster. The update

$$\mu' = \frac{N\mu + x}{N+1} \tag{3}$$

$$\Sigma' = \frac{(N-1)\Sigma + (x-\mu')(x-\mu')^T}{N} \tag{4}$$

where N is the number of old samples or a specified learning rate, can be rewritten to allow multiple samples to update the model at the same time. In a real system this updating can help to correct for slow changes in the atmospheric contribution to the image.

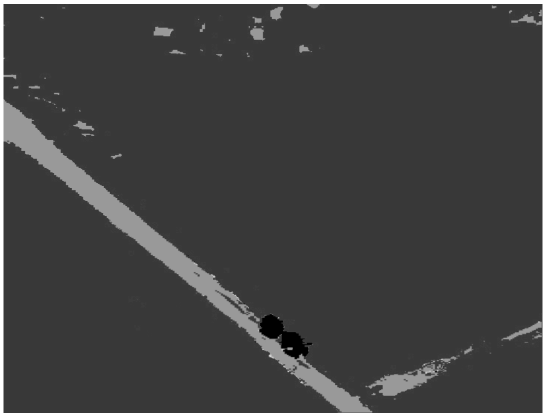

Figure 4: Segmented image using clusters for road, vehicle, and background.

User interface

The modelling is to a large extent controlled by the user, especially in the initial stage. The user can chose whether several clusters should be grouped to one cluster (for example, different types of vegetation) and if detected anomalies should form a new class (target or background cluster). Additionally, the user can mark any group of pixels and create a new cluster.

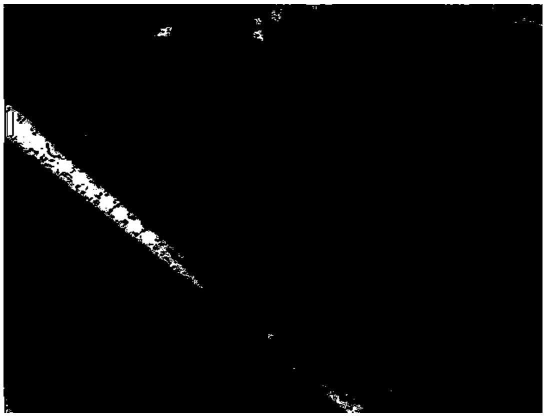

Figure 5: Anomaly image using the same background model as in the cluster image.

5 Conclusion

We have implemented and demonstrated a system for operator support in an air-to-ground reconnaissance scenario using a multispectral midwave infrared camera. Due to the camera design with a rotating filter wheel, it is necessary to align images from different spectral bands, and we have proposed a solution using RANSAC. This gives us a sequence of multispectral images instead of a sequence of images from different spectral bands.

To let an operator analyse the multispectral images we use target and anomaly detections schemes based on Gaussian cluster models. The initial background model is created automatically, and can then be controlled by the user.

References

[1] J. Ahlberg and I. Renhorn, *Multi- and Hyperspectral Target and Anomaly Detection*, Scientific report FOI-R–1526–SE, Swedish Defence Research Agency, 2005.

[2] T. Chevalier et al., *Optroniska system 2004*, User report FOI-R–1422–SE, Swedish Defence Research Agency, 2004.

[3] M. A. Fischler, R. C. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. of the ACM*, Vol 24, pp. 381-395, 1981.

[4] J.A. Noble, *Descriptions of Image Surfaces*, PhD thesis, Dept. of Engineering Science, Oxford Univ., Sept. 1989.