MADCAM – THE MULTISPECTRAL ACTIVE DECOMPOSITION CAMERA

Klaus B. Hilger and Mikkel B. Stegmann

IMM, Informatics and Mathematical Modelling
Technical University of Denmark, Building 321
Richard Petersens Plads, DK-2800 Kgs. Lyngby, Denmark

ABSTRACT

A real-time spectral decomposition of streaming three-band image data is obtained by applying linear transformations. The Principal Components (PC), the Maximum Autocorrelation Factors (MAF), and the Maximum Noise Fraction (MNF) transforms are applied. In the presented case study the PC transform appears to provide the best result for separating signal from noise. The main difficulty for the more advanced methods is to obtain good estimates for the correlation structure of the noise since problems arise due to spatial and temporal coding of the video signal. A new MNF transform is proposed that utilised information drawn from the temporal dimension instead of the traditional spatial approach. Using the CIF format (352×288) frame rates up to 30 Hz are obtained and in VGA mode (640×480) up to 15 Hz.

1. INTRODUCTION

The decomposition of multispectral images is motivated by extracting important, otherwise occluded information on the correlation structures in the data. Furthermore, by means of the purely data driven methods, signal and noise can often be separated, thus providing high quality grey-scale images from e.g. cheap RGB sensors such as web-cams et cetera.

As opposed to the Principal Components (PC) transform, [4], the Maximum Noise Fraction (MNF) traditionally takes the spatial nature of the image into account. The MNF transform was proposed as a transformation for ordering multispectral data in terms of image quality with implications for noise removal. It was introduced by Green et al. in 1988, [3], inspired by earlier work on the Maximum Autocorrelation Factors (MAF) transform by Switzer and Green in 1984, [8]. The MNF/MAF transform is further described in [1, 9, 2, 5].

Whereas the PC transform only requires knowledge or an estimate of the dispersion (covariance) matrix, the MNF transform requires the dispersion matrix of the noise structure as additional information. This has traditionally been solved by using the spatial information in the image. We propose a learning phase prior to the application of the MNF transform on the streaming image data. In the learning phase we utilise the temporal dimension to obtain an estimate of the correlation structure of the noise.

Section 3 describes the hardware used. In section 2 the PC and the MNF/MAF transforms are presented. Section 4 describes the implementation of the proposed MADCam. Section 5 contains performance results, and section 6 contains some concluding remarks.

{kbh,mbs}@imm.dtu.dk, http://www.imm.dtu.dk/∼{kbh,mbs}

2. SIGNAL AND NOISE DECOMPOSITION

In the following the PC and the MNF/MAF transformations are described together with the learning phase in which the covariance structure for the noise is estimated.

2.1. Principal Components Analysis

Consider a multivariate data set of P variables with grey levels $r_i(\boldsymbol{x}), i = 1, \cdots, P$, where $\boldsymbol{x}$ is the coordinate vector denoting the grid point of the sample.

Let

$$\boldsymbol{r}(\boldsymbol{x}) = \begin{bmatrix} r_1(\boldsymbol{x}) \\ \vdots \\ r_P(\boldsymbol{x}) \end{bmatrix} \tag{1}$$

and assume first and second order stationarity such that

$$\mathrm{E}\{\boldsymbol{r}(\boldsymbol{x})\} = \mathbf{0} \tag{2}$$

$$\mathrm{D}\{\boldsymbol{r}(\boldsymbol{x})\} = \boldsymbol{\Sigma}. \tag{3}$$

Determining the direction of maximum variation means finding the direction $\boldsymbol{a}$, with $\boldsymbol{a}^T\boldsymbol{a} = 1$, such that the linear combination $y(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{r}(\boldsymbol{x})$ possesses maximum variance.

The PC transformation thus chooses P linear transformations

$$y_i(\boldsymbol{x}) = \boldsymbol{a}_i^T \boldsymbol{r}(\boldsymbol{x}), \;\; i = 1, \cdots, P \tag{4}$$

such that the variance for $y_i(\boldsymbol{x})$ is maximum among all linear transforms orthogonal to $y_j(\boldsymbol{x}), j = 1, \cdots, i-1$. The variance is given by

$$\lambda_i = \boldsymbol{a}_i^T \boldsymbol{\Sigma} \boldsymbol{a}_i. \tag{5}$$

We see that the basis for the PCs is identified as the conjugate eigenvectors of the dispersion matrix. Let $\lambda_1 \geq \cdots \geq \lambda_P \geq 0$ be the eigenvalues with the corresponding conjugate eigenvectors $\boldsymbol{a}_1, \cdots, \boldsymbol{a}_P$. Then $y_i(\boldsymbol{x})$ is the i'th PC (PCi) with variance λ_i.

2.2. Maximum Noise Fraction Transform

Let us again consider the random signal variable $\boldsymbol{r}(\boldsymbol{x})$ and assume first and second order stationarity by imposing Eq. 2 and 3. When we assume that an additive noise structure applies

$$\boldsymbol{r}(\boldsymbol{x}) = \boldsymbol{s}(\boldsymbol{x}) + \boldsymbol{n}(\boldsymbol{x}), \tag{6}$$

the dispersion structure can then be separated into

$$\mathrm{D}\{\boldsymbol{r}(\boldsymbol{x})\} = \boldsymbol{\Sigma} = \boldsymbol{\Sigma}_s + \boldsymbol{\Sigma}_n. \tag{7}$$

The MNF transformation chooses P linear transformations

$$z_i(\boldsymbol{x}) = \boldsymbol{b}_i^T \boldsymbol{r}(\boldsymbol{x}), \;\; i = 1, \cdots, P \tag{8}$$

that maximise the signal-to-noise ratio (SNR) for the i'th component defined by

$$\mathrm{SNR}_i = \frac{\mathrm{V}\{\boldsymbol{b}_i^T s_i(\boldsymbol{x})\}}{\mathrm{V}\{\boldsymbol{b}_i^T n_i(\boldsymbol{x})\}}. \tag{9}$$

Combining Eq. 7 and Eq. 9 we find

$$\mathrm{SNR}_i = \frac{\boldsymbol{b}_i^T \boldsymbol{\Sigma} \boldsymbol{b}_i}{\boldsymbol{b}_i^T \boldsymbol{\Sigma_n} \boldsymbol{b}_i} - 1, \tag{10}$$

and the problem is reduced to solving a generalized eigenproblem, say

$$\boldsymbol{\Sigma_n} \boldsymbol{b}_i = \chi_i \boldsymbol{\Sigma} \boldsymbol{b}_i. \tag{11}$$

Let $\chi_1 \leq \cdots \leq \chi_P$ be the eigenvalues of $\boldsymbol{\Sigma_n}$ with respect to $\boldsymbol{\Sigma}$ with the corresponding conjugate eigenvectors $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_P$. Then $z_i(\boldsymbol{x})$ is the i'th MNF (MNFi). A high order component has a high noise fraction, hence the name Maximum Noise Fraction transform.

The central issue in obtaining good MNF components is the estimation of the dispersion matrix for the noise. In [7, 6] several models are presented for estimating noise in images based on spatial characteristics. Using the difference between the current pixel and its neighbours, the MNF reduces to the MAF transform. When the covariance structure for the noise is proportional to the identity matrix, the MNF transform reduces to the PC transform. Unlike the PC the MNF transform is invariant to linear rank preserving transformations of the original data.

A comparative study of the PC and the MNF/MAF transforms applied to high dimensional data can be found in [5]. The MNF transform does a much better job of separating signal from noise than the PC transform. It produces a nice ordering of the new components, which can often be perceived as a decomposition of spatial frequency.

By applying the logarithm transform to the signal $\boldsymbol{r}(\boldsymbol{x})$, the MNF transform is able to handle multiplicative noise, since signal and noise are split into an additive structure.

2.3. The Learning Phase

In order to obtain a good estimate of the dispersion of the noise we propose the following learning phase. The central issue is to learn the correlation structure of the noise across the frequency bands. The noise models proposed in the literature may not be useful when handling streaming video, since the data can be corrupted by radical spatial/temporal compression coding. The coding may cause the noise in the image to be very different from the traditional – e.g. salt & pepper noise.

Instead of looking at the spatial autocorrelations in the image, we look in the temporal dimension of the data. For every three-band frame-pair recorded in the learning phase, the covariance structure for the noise is estimated as the dispersion of the corresponding image differences. As more frame-pairs are available, the dispersions of the camera noise are pooled together.

For a sequence of N frames we obtain $N-1$ estimates

$$\boldsymbol{\Sigma_n}^i = \mathrm{D}\{\boldsymbol{r}^i(\boldsymbol{x}) - \boldsymbol{r}^{i+1}(\boldsymbol{x})\}, \quad i = 1, \cdots, N-1 \tag{12}$$

where $\boldsymbol{r}^i(\boldsymbol{x})$ is the signal at position $\boldsymbol{x}$ in the i'th frame. The pooled estimate of the dispersion structure of the noise is found by

$$\boldsymbol{\Sigma_n} = \frac{1}{N-1} \sum_{i=1}^{N-1} \boldsymbol{\Sigma_n}^i. \tag{13}$$

In order to have a meaningful learning phase, the recorded image sequence should be with minimal motion. The learning phase can be terminated when the change in the pooled

Fig. 1. The MADCam basis: A Philips PCVC690K Camera.

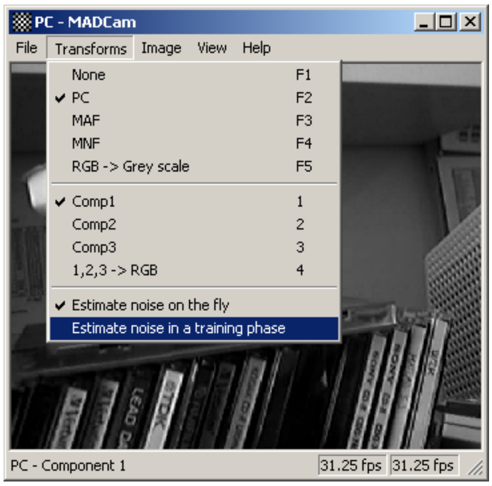

Fig. 2. A screen shot of the MADCam user interface.

estimate is sufficiently small. The length of the phase is therefore closely linked to the frame-rate of the camera.

Hybrid methods can also be constructed combining both spatial and temporal information about the noise structure by simple pooling of covariance structures.

3. HARDWARE

The basis of the MADCam is a low-cost consumer quality Philips Video Camera Scanner, PCVC690K, capable of providing 30 (I420 coded) frames per second with a maximum resolution of VGA (640$\times$480) via an USB interface, see figure 1. The optical sensor consists of an 1/4" CCD, pixels 640(H)x480(V). Figure 3 contains a CIF (352$\times$288) RGB image obtained by the MADCam with no transformation applied. Each band is stretched linearly between its mean ± 3 standard deviations (std). The transformations presented in the following are being carried out by an Athlon 1200 MHz desktop PC connected to the camera.

Although this MADCam features a specific camera, we stress that any Video for Windows (VFW) or TWAIN compliant camera could be used in the implementation described below (without code modifications).

4. IMPLEMENTATION

The MADCam is implemented as a C++ application running under Windows 2000. The program was augmented with real-time video input capabilities by the Microsoft VisionSDK. All core image processing was coded in platform-independent standard C++ with no use of libraries to obtain optimal performance w.r.t. to the special task of computing the PC/MAF/MNF transform. An exception to this is the LAPACK eigenproblem solver. In a preliminary version of the implementation parts of the Intel Image Processing Library was used but due to lack of flexibility, this actually gave a performance penalty.

No cache analysis has been carried out in the current implementation. However, due to the massive memory transfer we anticipate a substantial performance increase if the memory layout is being optimised. Furthermore, due to the inherent parallelism large portions of the code could be converted to processor specific MMX/SSE/3DNow! optimised code causing an additional performance increase.

Refer to figure 2 for a screen shot of the working application.

5. RESULTS

The image shown in figure 3 is transformed using the PC, the MAF and the MNF mode of the MADCam. The results are shown in the figures 4 to 6. All images are stretched between their mean ± 3 std.

Looking at the Principal Components in figure 4 we see that a nice decomposition of the RGB signal. It appears as if most of the interesting signal is compressed into one component i.e. PC1. The covariance structure for the image was estimated by

$$\mathbf{\Sigma} = \begin{bmatrix} 4106 & 3501 & 1616 \\ 3501 & 3880 & 2404 \\ 1616 & 2404 & 2751 \end{bmatrix}. \tag{14}$$

The MAF transform seems successful in obtaining high spatial autocorrelation in the first component but a lot of interesting signal is still remaining in the higher MAF components, see figure 5. This can be explained by the difficulties encountered when handling the noise due to the MPEG-like

Fig. 3. The red, green and blue band of a CIF (352×288) image obtained by the Philips Camera ordered top-down, each band is stretched linearly between its mean ±3 standard deviations (std).

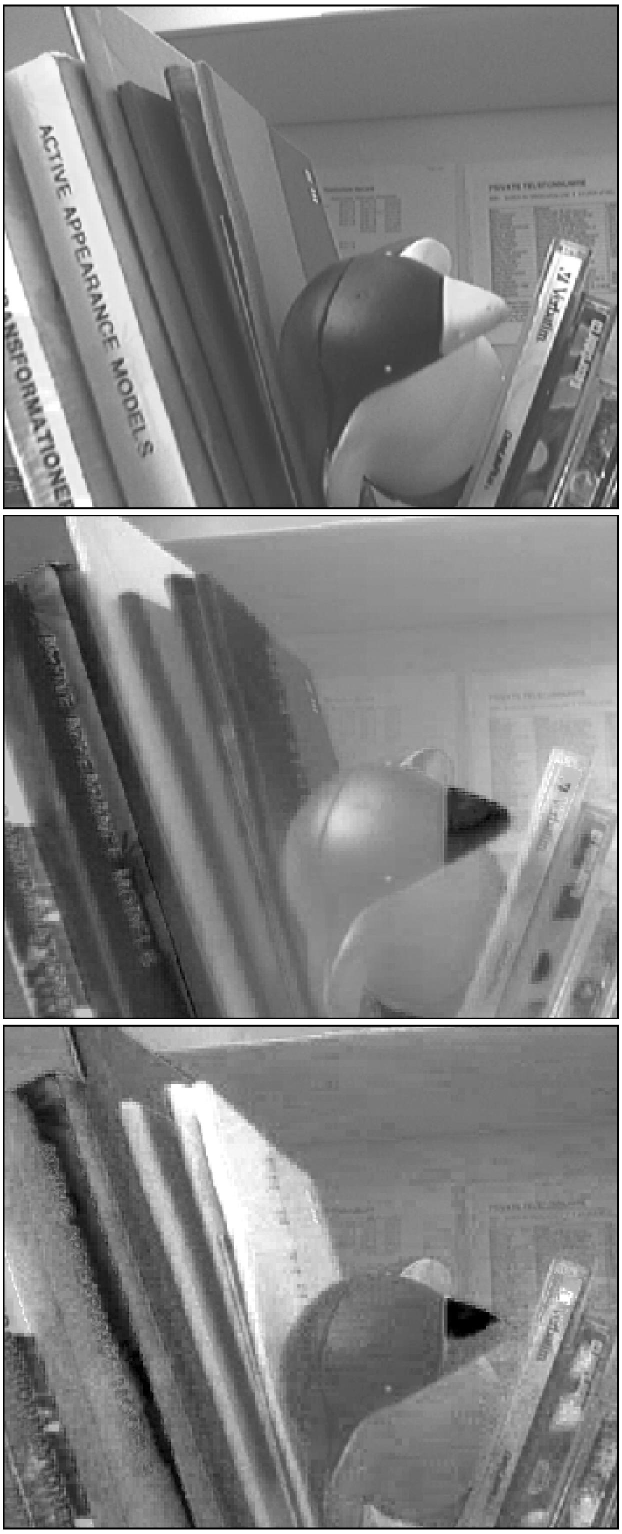

Fig. 4. Principal Components 1-3 ordered top-down. Each component is stretched linearly between its mean ±3 std.

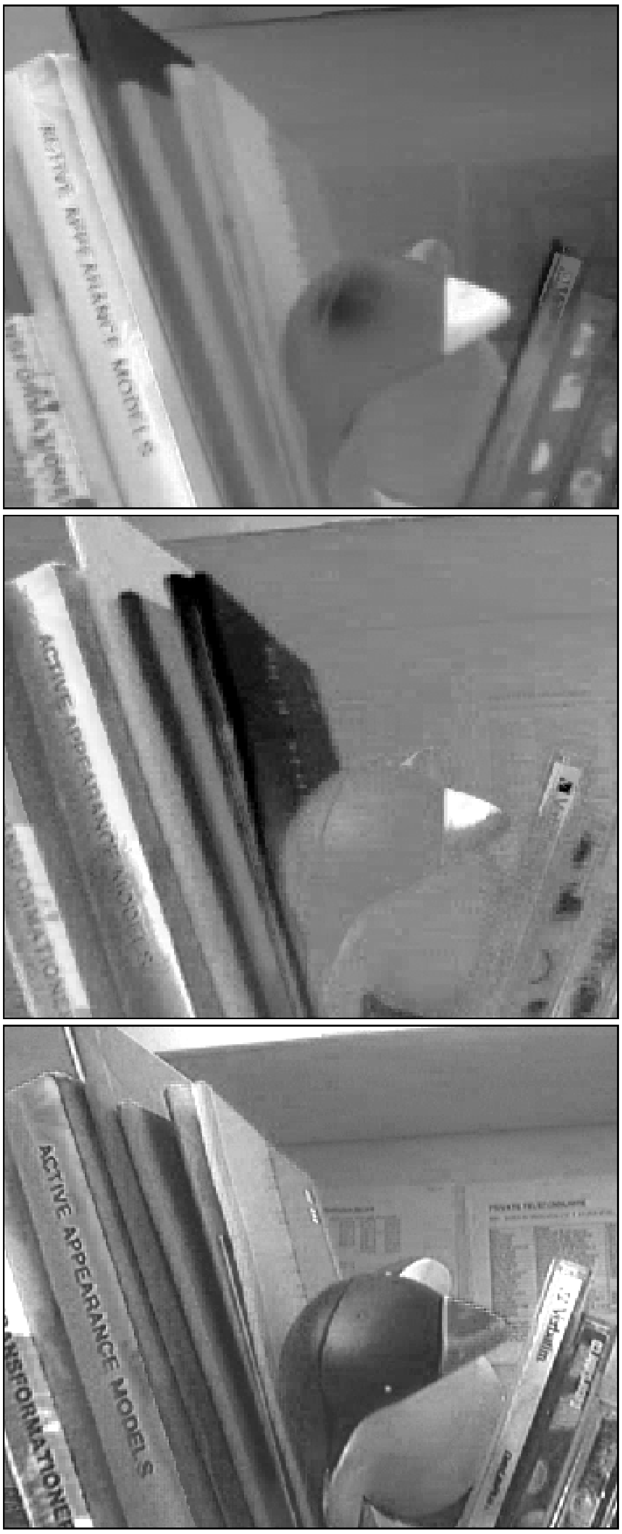

Fig. 5. Maximum Autocorrelation Factors 1-3 ordered top-down. Each component is stretched linearly between its mean ±3 std.

Fig. 6. Maximum Noise Fractions 1-3 ordered top-down. Each component is stretched linearly between its mean ±3 std.

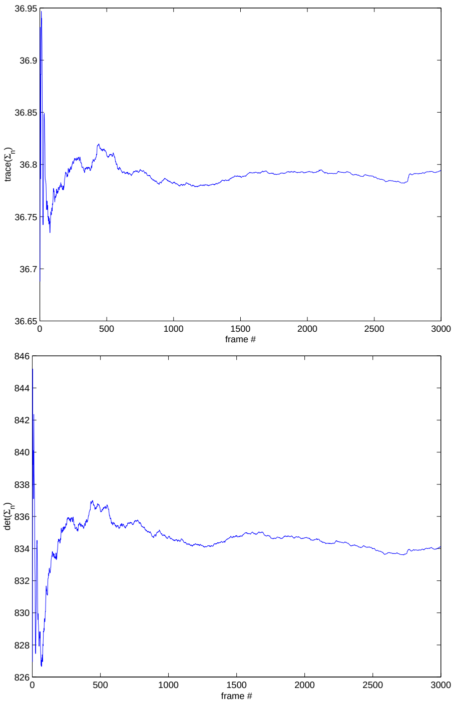

Fig. 7. Learning phase plots. Top-row: the evolution of the trace($\mathbf{\Sigma}_n$) as a function of time. Bottom-row: the evolution of the det($\mathbf{\Sigma}_n$). Notice that the estimate of the covariance matrix for the noise stabilises when integrating over time.

coding of the streaming video signal. Useful MAFs have however been observed when applying images from cameras that utilize less radical spatial/temporal compression. The dispersion structure for the noise was estimated by

$$\mathbf{\Sigma}_n^{\mathbf{MAF}} = \begin{bmatrix} 273 & 253 & 209 \\ 253 & 271 & 228 \\ 209 & 228 & 236 \end{bmatrix}. \qquad (15)$$

The MNF in figure 6 seems to perform similar to the PC transform. Figure 7 shows the evolution of the trace($\mathbf{\Sigma}_n$) and the det($\mathbf{\Sigma}_n$) as a function of time. The figure illustrates that the covariance matrix for the noise stabilizes when integrating over time. The still image sequence applied contained the image presented in figure 6. After completing the learning phase the MNF estimate of the dispersion of the noise was

$$\mathbf{\Sigma}_n^{\mathbf{MNF}} = \begin{bmatrix} 12 & 5 & 7 \\ 5 & 9 & 6 \\ 7 & 6 & 14 \end{bmatrix}. \qquad (16)$$

Performance analyses were performed in both the CIF and the VGA mode for a still sequence. The resulting frame rates are shown in table 1 and 2. Using pre-learnt covariance structures for the noise gives higher performances as expected. All controls were fixed for the camera before running the analyses.

Transform	**Format**	**Frame rate**	**CPU usage**
None	CIF	30 Hz	35%
PC	CIF	30 Hz	68%
MAF	CIF	26 Hz	100%
MNF	CIF	30 Hz	99%
None	VGA	30 Hz	97%
PC	VGA	15 Hz	100%
MAF	VGA	9 Hz	100%
MNF	VGA	10 Hz	100%

Table 1. Performance of MADCam on the fly. All covariance structures are reestimated in each frame.

Transform	**Format**	**Frame rate**	**CPU usage**
MAF	CIF	30 Hz	68%
MNF	CIF	30 Hz	68%
MAF	VGA	15 Hz	100%
MNF	VGA	15 Hz	100%

Table 2. Performance of the MADCam when pre-learnt covariance structures for the noise are applied.

6. SUMMARY

A real-time decomposition of streaming multivariate image data is obtained using the MADCam. In CIF mode frame rates up to 30 Hz can be obtained and VGA mode up to 15 Hz. The proposed application is based on Plug'n'Play off-the-shelf technology. A new MNF transformation is proposed in which information about the correlation structure of the noise is extracted in a learning phase.

The decomposition of multivariate image data can often reveal important information about the data. New information may be found that is occluded when viewing the data in the original RGB colour-space. Moreover, by applying the MADCam framework cheap RGB cameras can be used to generate high quality grey-scale video.

The largest difficulty to overcome is how to handle the spatial/temporal coding of the streaming video. If little coding is applied, we expect the MAF and the MNF transforms to be superior to the PC transform in decomposing the signal. Thus, future work may include the application of a consumer quality video camera with a FireWire (IEEE 1394) interface and minimal or none coding of the streaming imagery.

7. REFERENCES

[1] Knut Conradsen, Bjarne Kjær Nielsen, and Tage Thyrsted. A comparison of min/max autocorrelation factor analysis and ordinary factor analysis. In *Proceedings from Symposium in Applied Statistics*, pages 47–56, Lyngby, Denmark, January 1985.

[2] Bjarne K. Ersbøll. *Transformations and Classifications of Remotely Sensed Data: Theory and Geological Cases*. PhD thesis, Institute of Mathematical Statistics and Operations Research, Technical University of Denmark, Lyngby, 1989. 297 pp.

[3] A. A. Green, M. Berman, P. Switzer, and M. D. Craig. Transformation for ordering multispectral data in terms of image quality with implications for noise removal. *IEEE Transactions on Geoscience and Remote Sensing*, 26(1):65–74, 1988.

[4] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *J. Educ. Psych.*, 24:417–441, 1933.

[5] A. A. Nielsen. *Analysis of Regularly and Irregularly Sampled Spatial, Multivariate, and Multi-temporal Data*. PhD thesis, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, 1994. Internet http://www.imm.dtu.dk/∼aa/phd/.

[6] A. A. Nielsen. An extension to a filter implementation of local quadratic surface for image noise estimation. In *Proceedings of the 10th International Conference on Image Analysis and Processing (ICIAP'99)*, pages 119–124, Venice, Italy, 27-29 September 1999.

[7] Søren Ingvor Olsen. Estimation of noise in images: An evaluation. *Graphical Models and Image Processing*, 55(4):319–323, 1993.

[8] P. Switzer and A. A. Green. Min/max autocorrelation factors for multivariate spatial imagery. Technical Report 6, Dept. of statistics, Stanford University, 1984.

[9] P. Switzer and S. E. Ingebritsen. Ordering of time-difference data from multispectral imagery. *Remote Sensing of Environment*, 20:85–94, 1986.