Building Blocks for Computer Vision with Stochastic Partial Differential Equations

Tobias Preusser* Hanno Scharr† Kai Krajsek†
Robert M. Kirby‡

September 18, 2007

We discuss the basic concepts of computer vision with stochastic partial differential equations (SPDEs). To model images and image sequences which carry uncertainty in their gray values due to noise in the acquisition process, we identify the image data with random fields. The noisy behavior of gray values is modeled by stochastic processes which are approximated with the method of generalized polynomial chaos (Wiener-Askey-Chaos). The Wiener-Askey polynomial chaos is combined with a standard spatial approximation with piecewise multi-linear finite elements. We present basic building blocks needed for computer vision and image processing in this stochastic setting, *i.e.* we discuss the computation of stochastic moments, projections, gradient magnitudes, edge indicators, structure tensors, etc. Finally we show applications of our framework to derive stochastic analogs of well known partial differential equations (PDEs) for de-noising and optical flow extraction. These models are discretized with the stochastic Galerkin method. Our selection of SPDE models allows to draw connections to the classical deterministic models as well as to stochastic image processing not based on PDEs. Several examples guide the reader through the presentation and show the usefulness of the framework.

Keywords. Image Processing, Error Propagation, Random Fields, Polynomial Chaos, Stochastic Partial Differential Equations, Stochastic Galerkin Method, Stochastic Finite Element Method

1 Introduction

In computer vision applications, *e.g.* medical or scientific image data analysis, as well as in industrial scenarios, images are used as input measurement data. Thus, for many applications not only the measured values, but also their error should be – and more and more is – taken

*Center for Complex Systems and Visualization, Bremen University, Germany, tp@cevis.uni-bremen.de
†Institute for Chemistry and Dynamics of the Geosphere, Institute 3: Phytosphere, Research Center Juelich, Germany, {h.scharr, k.krajsek}@fz-juelich.de
‡School of Computing and Scientific Computing Institute, University of Utah, UT, USA, kirby@cs.utah.edu

into account for further processing. This error propagation must be done for every processing step, such that the final result comes with a reliable precision estimate. But, unfortunately, for realistic models the computation of error propagation is sometimes hard or cumbersome, and therefore most contributions dealing with error estimates are restricted in one or more of the following:

- Data is presumed to have a Gaussian distribution – the error is then represented by a variance (*cf. e.g.* [29, 40]);
- Bounds on the error are derived only (*cf. e.g.* [30, 29]);
- No true error estimates, confidence measures only are derived (*cf. e.g.* [6, 15]).

This paper presents a framework for the treatment of computer vision models on images and image sequences whose gray values are not assumed to be only single values but distributions of values. The methodology is not restricted to Gaussian distributions and the output is not restricted to error bounds. In fact the distributions can be much richer than those that are completely described by the two values mean and variance.

The key concept is an identification of image data with random fields. Thereby we identify the gray values of images with random processes which are supposed to model the behavior of image detector elements and the influence of noise. We approximate those stochastic processes with the method of generalized polynomial chaos (gPC) and supplement this approximation with a spatial discretization by Finite Element shape functions. This leads us to an ansatz-space for stochastic images and image-sequences. Computer vision models and algorithms using this concept of stochastic images as random fields transform the input distributions into output distributions without loosing information on the precision due to approximated error bounds.

For computer vision models, which originate from the minimization of energies through the solution of Euler-Lagrange equations, or which come from other partial differential equations (PDEs) it is straight forward to augment the deterministic equations to the stochastic setting by replacing classical image functions by the stochastic analog. However, special care must be taken for nonlinear operators which lead to a coupling of the stochastic modes and moments. For standard tools in computer vision we discuss the stochastic generalization. With the help of projections and mass lumping in the stochastic space we are able to write down simple equations for the computation of means, variances, and covariances, as well as gradient magnitudes, edge indicators, and structure tensors. Those operations include the computation of stochastic integrals which can be computed in advance and stored in lookup tables.

We demonstrate the usage of the formalism by exemplary implementations of very well known and thus well understood algorithms: Gaussian smoothing via isotropic diffusion, Perona-Malik isotropic nonlinear diffusion [32], variational optical flow estimation by Horn-Schunck [16], a version with a robust smoothing term [3, 9, 41] and optical flow computation with a regularized data term [6]. These algorithms are prototypes of linear and nonlinear, energy-based computer vision approaches for regularization, noise suppression, and parameter estimation with a wealth of practical applications. They are applied to very simple, thus easy to interpret test data in order to show the benefits and limitations of the formalism put forward here. We do not advocate that these algorithms are the 'best' for certain applications (*cf. e.g.* [31, 1] for state of the art optical flow approaches): The advantages and disadvantages of the exemplary algorithms are well known. Here they are only used for pedagogical reasons and as

demonstrators of the formalism which enable the reader to draw conclusions and connections to deterministic image processing as well as stochastic image processing not based on PDEs. Results of these algorithms for the mean (or more precisely expectation value) calculated with the new formalism do not differ from results a standard finite element algorithm would give. In fact, the new formalism boils down to a standard deterministic implementation, when we model the input distribution only by its mean. But the benefit of our new approach lies in the handling of distributions: It allows for precise, local and data depending error estimations beyond the Gaussian assumption.

Related work The use of PDEs in computer vision has been popular during the last decades. Mostly, those PDEs are the necessary conditions (Euler-Lagrange equations) for minima of certain energy functionals. Approaches to de-noising, restoration, in-painting, segmentation, registration, optical flow estimation, *etc.*, and combinations of the latter are too numerous to give a short overview here.

To the best of the authors' knowledge, SPDEs have not yet been applied in computer vision to transform input distributions into output distributions, but they are well established tools in other disciplines. Based on the Wiener-Hermite polynomial chaos expansion [43], the stochastic Galerkin method has been applied to a range of problems in computational mechanics [26, 7, 8, 25, 10, 23, 45, 47, 22]. This technique has also recently been introduced into other disciplines such as thermodynamics [13, 28, 48] and physical chemistry [35, 34], in part because it leads to efficient solutions to stochastic problems of interest, *i.e.* not only parameter sensitivity and uncertainty quantification.

Our contribution shall not be confused with statistical parameter selection methods, *e.g.* via Stochastic Differential Equations (SDEs) [2] or Markov-Random-Field assumptions [37]. But our framework yields general extensions to the stochastic interpretation of energy functionals presented in [36].

Paper organization In Sect. 2 we review some notion from the theory of probability and introduce the reader to the theory of random fields. We derive a way of identifying images and image-sequences with random fields. Thereby we combine the approximation of stochastic processes by the Wiener-Askey Polynomial Chaos with the standard multi-linear interpolation/approximation schemes in space. In the following Sect. 3 we discuss basic building blocks for computer vision with images as random fields. There we also analyze the structure of the resulting block operators and their assembly which involves the computation of integrals in the random space. In Sec. 4 the stochastic generalization of some well known PDEs used in computer vision is presented and discretized with help of the building blocks. We consider a linear and a nonlinear diffusion (Perona Malik [32]) model, the optical flow extraction with the Horn and Schunck approach [16], a robust smoothing term for the optical flow field [4, 9, 41], and finally a combined local global (CLG) approach [6]. An investigation of the bias of the CLG model further underlines the usefulness of our framework. Conclusions are drawn and an outlook is given in Sect. 6.

2 Stochastic Images as Random Fields

Considering noise, it is appropriate to model gray values for each pixel within an image or an image sequence as the realization of a random variable characterized by a Probability Density

Function (PDF). In the following we consider images and image sequences with uncertain gray values as realizations of random fields. Thereby we model the uncertainty of the gray values with random processes, for which we use a certain approximation with orthogonal basis functions.

2.1 Random Fields and Wiener-Askey Polynomial Chaos

First we would like to review some background from the theory of probability, define some notions, and review the Wiener-Askey generalized polynomial chaos (gPC). A good overview of the methodology we use in the following is given in [19].

Let $(\Omega, \mathcal{A}, \mu)$ be a complete probability space, where Ω is the event space, $\mathcal{A} \subset 2^\Omega$ the $\sigma-$algebra, and μ the probability measure. Following [46], we can represent any general second-order *random process* $X(\omega), \omega \in \Omega$ in terms of a collection of random variables $\boldsymbol{\xi} = (\xi_1, \dots \xi_N)$ with independent components. Let $\rho_i : \Gamma_i \to I\!R^+$ be the probability density functions of the random variables $\xi_i(\omega), \omega \in \Omega$, and its image $\Gamma_i \equiv \xi_i(\Omega) \in I\!R$ be intervals in $I\!R$ for $i = 1, \dots, N$. Then

$$\rho(\boldsymbol{\xi}) = \prod_{i=1}^{N} \rho_i(\xi_i), \quad \forall \boldsymbol{\xi} \in \Gamma$$

is the joint probability densisty of the random vector $\boldsymbol{\xi}$ with the support

$$\Gamma = \prod_{i=1}^{N} \Gamma_i \subset I\!R^N.$$

As commented in [46], this allows us to conduct numerical formulations in the finite dimensional (N-dimensional) random space Γ. Let us denote $L^2(\Gamma)$ as the probabilistic Hilbert space [24] in which the random processes based upon the random variables $\boldsymbol{\xi}$ reside. The inner product of this Hilbert space is given by:

$$< a, b >= \int_\Gamma (a \cdot b)\ d\mu = \int_\Gamma (a \cdot b)\ \rho(\boldsymbol{\xi}) d\boldsymbol{\xi}$$

where we have exploited independence of the random variables allowing to write the measure as product of measures in each stochastic direction. For notational simplicity we will often omit the integral in terms of integration against the probability measure as a shorthand for the decomposition implied above. We similarly define the expectation of a random process $X \in L^2(\Gamma)$ as:

$$\mathbb{E}[X(\boldsymbol{\xi})] = \int_\Gamma X(\boldsymbol{\xi})\ d\mu = \int_\Gamma X(\boldsymbol{\xi})\ \rho(\boldsymbol{\xi}) d\boldsymbol{\xi}.$$

Considering a spatial domain $D \subset I\!R^d$ we define a set of random processes which are indexed by the spatial position $x \in D$:

$$f : D \times \Gamma \to I\!R.$$

Such a set of processes is referred to as *random field* [19] which can also be interpreted as a function valued random variable, because for every event $\boldsymbol{\xi}$ the *realization* $f(\cdot, \boldsymbol{\xi}) : D \to I\!R$ is a function on D. For a vector-space Y the class $Y \otimes L^2(\Gamma)$ denotes the space of random fields whose realizations lie in Y for a.e. $\boldsymbol{\xi} \in \Gamma$. Throughout this paper we will use random fields

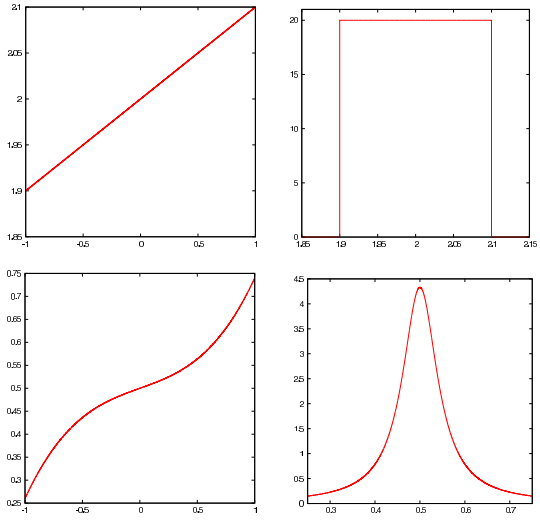


Figure 1: Top row: For a random process (left) that has been modeled with a linear combination of 2 Legendre polynomials we show the resulting probability density function (right). Bottom row: For a random process that has been modeled with a linear combination of 4 Legendre polynomials we show the resulting PDF.

$f \in L^2(D) \otimes L^2(\Gamma)$ such that $f(\cdot, \boldsymbol{\xi}) \in L^2(D)$ for almost all $\boldsymbol{\xi} \in \Gamma$. In this case let us define the norm $\|\!|\cdot|\!\|$ as

$$\|\!| f(x, \boldsymbol{\xi}) |\!\|^2 = \mathbb{E}[\| f(x, \boldsymbol{\xi}) \|^2_{L^2(D)}] = \int_\Gamma \int_D \big(f(x, \boldsymbol{\xi})\big)^2 \, dx \, d\mu,$$

that is, $\|\!|\cdot|\!\|$ denotes the expected value of the L^2-norm of the function f.

A key to the numerical treatment of a stochastic process $f^i \in L^2(\Gamma_i)$ is the expansion

$$f^i(\xi_i) = \sum_{\alpha=1}^{p} f^i_\alpha H^\alpha(\xi_i) \tag{1}$$

with stochastic ansatz functions $H^\alpha(\boldsymbol{\xi})$. Those can be chosen according to the Wiener Askey Polynomial Chaos Approach [46] (generalized polynomial chaos, gPC) in which the functions are orthogonal polynomials ranging up to p^{th} degree and p must be chosen to be large enough so that the solutions will meet the accuracy requirements for the particular system of interest. In the sequel we will denote the coefficients f^i_α as *modes* of the stochastic process $f^i(\cdot)$. Furthermore we denote the space spanned by the polynomials with

$$\mathcal{P}_p = \text{span}\{H^\alpha \,|\, \alpha = 1, \ldots, p\}.$$

Convergence rates of the system depend on the choice of orthogonal polynomials for underlying probability density functions of a random model parameter. Each probability distribution has a corresponding optimal set of orthogonal polynomials [46]; *e.g.*, for Gaussian random functions, Hermite polynomials provide the best convergence, whereas Legendre polynomials are best utilized for functions of uniform distributions, *etc.*

Expansions like (1) exhibit fast convergence rates when the stochastic response of the system is sufficiently smooth in the random space, *e.g.*, bifurcation behavior is absent. Unlike the Monte Carlo method, which is amazingly robust because it uses effectively no information about the underlying process to determine the sampling or reconstruction procedure, the gPC methodology attempts to use as much of the inherent structure (such as smoothness within the stochastic space) as possible to make the methodology computationally tractable.

The corresponding PDF of a random process is obtained through the branches of the derivative of the inversion of its process-function. In many cases the derivative of the inverse can be

obtained with the *inverse function theorem*, but if the inversion is analytically not feasible a binning and histogram of the function values of the process is beneficial.

In our experiments shown throughout the paper we have selected Legendre polynomials. They are simple to use for modeling processes and distributions having finite support, like uniform and truncated Gaussian distributions. In Fig. 1 we show how various distributions ranging from uniform to Gaussian shaped processes can be approximated with Legendre polynomials.

2.2 Stochastic Still Images

Let us assume we are looking at still images. An acquisition process of such digital images yields noisy images due to various technical and physical reasons (*cf.* [12], Chap. 1). In this section our intention is to model the stochastic process of gray-value measurement (*i.e.* the uncertain output of the detector elements of the camera) with the help of the methodology presented in the last section. For the sake of simplicity we restrict our presentation to two-dimensional images. An extension to n-dimensional images is not difficult.

We assume the pixels of the image are located at a regular quadrilateral grid of dimension $\{1,\ldots,N_1\}\times\{1,\ldots,N_2\}$. We define the grid-width h as

$$h = \min\left\{\frac{1}{N_1-1}, \frac{1}{N_2-1}\right\}.$$

So the image has $N := N_1N_2$ pixels and its degrees of freedom (DOF) lie on the vertices of a regular grid $\mathcal{G}$ with $(N_1-1)(N_2-1)$ quadrilateral elements E_j. We denote the set of vertices of $\mathcal{G}$ with $\mathcal{I}$ and order the pixels $x_i \in \mathcal{I}$ lexicographically, *i.e.* from left to right, from top to bottom. Classically, we introduce a finite element space by using a bi-linear interpolation scheme [33]. This means, we consider the finite-dimensional space

$$V_2^h := \operatorname{span}\{P_i \,|\, P_i \in C^0(D), P_i(x_j) = \delta_{ij}, P_i|_E \text{ is bilinear } \forall E \in \mathcal{G}\} \subset H^1(D)$$

where δ is the Kronecker-δ. So the space V_2^h is spanned by the classical piecewise linear tent-functions P_i, which are equal to one in x_i and vanish at every other vertex. The space $H^1(D)$ denotes the Sobolev space $H^{1,2}(D)$ of functions having square integrable weak derivatives. Every image $f \in V_2^h$ has a representation

$$f(x) = \sum_{i\in\mathcal{I}} f^i P_i(x) \tag{2}$$

where the vector of degrees of freedom is $(f^1,\ldots,f^N) \in I\!R^N$.

Let us now assume that gray values of the pixels reveal some uncertainty and thus have a random distribution. In more detail, we assume that the behavior of a pixel at location x_i is determined by a separate stochastic process which depends on a random variable ξ_i. Furthermore we assume that these random variables are independent and that they are all supported on the same domain $\Gamma_i = \Gamma_* \subset I\!R$. The independence is based on the physical assumption of independence of the noise of each detector element or camera pixel. Thus, for a still image with N pixels, the stochastic space $\Gamma = \Gamma_*^N$ is N-dimensional and $\boldsymbol{\xi} = (\xi_1,\ldots,\xi_N)$ is a vector of random variables.

As described in the previous section, he stochastic Galerkin method represents any stochastic process, $X(\boldsymbol{\xi})$, by a weighted sum of orthogonal polynomials [14, 45, 46, 47] which are functions of a vector of independent random variables, $\boldsymbol{\xi}(\omega), \omega \in \Omega$, of known distribution. In the case

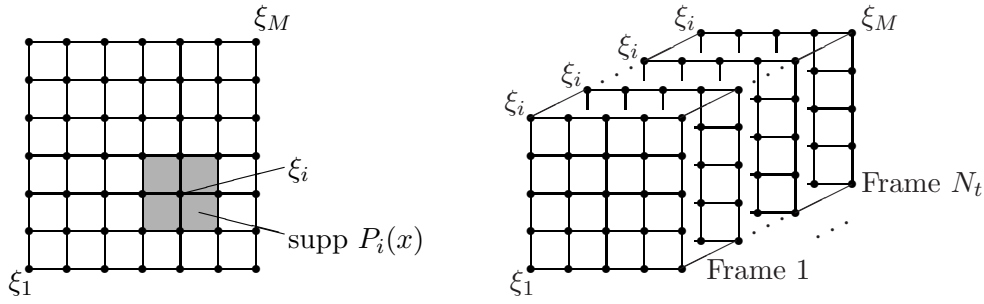


Figure 2: We model the random behavior of the detector elements by random variables ξ_i each having a spatial extent which is given by finite element shape functions P_i. Left: For 2D images or still 3D images the behavior of each pixel is modeled by a separate random variable. Right: For image sequences the behavior of pixels which have the same spatial coordinate (but may be located in different frames) is modeled by the same random variable.

of this study, the random processes of interest are the (stochastic) gray values attributed to each pixel. The random variables of interest will be chosen to represent the distributions from which gray values are sampled.

Consequently, our ansatz space is the tensor product space $V^h \otimes \mathcal{P}_p \subset H^1(D) \otimes L^2(\Gamma)$, *i.e.* we are considering random fields whose realizations are functions in V^h. Since the random variables for the pixels are independent, an image $f \in V_2^h \otimes L^2(\Gamma)$ decomposes into

$$f(x, \boldsymbol{\xi}) = \sum_{i \in \mathcal{I}} f^i(\xi_i) P_i(x).$$

This means that the behavior of a pixel with spatial-extent (support) P_i is modeled by the stochastic process $f^i(\boldsymbol{\xi})$ (*cf.* Fig. 2).

In summary we have constructed a finite-dimensional space $\mathcal{H}^{h,p}_{\text{still}} := V_2^h \otimes \mathcal{P}_p$ containing discrete random fields

$$f(x, \boldsymbol{\xi}) = \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^{p} f^i_\alpha \, H^\alpha(\xi_i) P_i(x). \tag{3}$$

With notational abuse we can also write $f(x, \boldsymbol{\xi}) = \sum_{\alpha=1}^{p} f_\alpha(x) H^\alpha(\boldsymbol{\xi})$ where $f_\alpha(x) = \sum_{i \in \mathcal{I}} f^i_\alpha P_i(x)$. So the f_α are the images that show the stochastic mode α of the pixel's processes. Altogether, the stochastic image has pN DOF and we use the ordering

$$F := (F_1; \ldots; F_p) := (f_1^1, \ldots, f_1^N; \quad \ldots \quad ; f_p^1, \ldots, f_p^N) \in \mathbb{R}^{pN}. \tag{4}$$

Remark 1 *Indeed we have constructed a proper finite dimensional sub-space of $H^1(D) \otimes L^2(\Gamma)$. Consequently the stochastic images (interpreted as discrete random fields) have weak spatial derivatives as usual. Moreover, setting $p = 1$ we fall back to the classical finite element discretization and $\mathcal{H}^{h,p}_{still} \equiv V_2^h$.*

2.3 Stochastic Image Sequences

Let us now consider an image sequence consisting of N_t frames. Each frame has dimension $M := N_1 N_2$, thus its pixels lie on a regular quadrilateral grid of dimension $\{1,\dots,N_1\}\times\{1,\dots,N_2\}$. We define the grid-width h as

$$h = \min\left\{\frac{1}{N_1-1}, \frac{1}{N_2-1}, \frac{1}{N_t-1}\right\}.$$

The complete image sequence has $N := N_1 N_2 N_t$ pixels and its degrees of freedom (DOF) lie on the vertices of a regular grid $\mathcal{G}$ with $(N_1-1)(N_2-1)(N_t-1)$ hexahedral elements E_j. For the later use it is convenient to introduce a special indexing of the vertices of this grid $\mathcal{G}$, which has a spatial and a temporal component. Within each frame we order the vertices lexicographically as we did for still images in the previous section. For the spatio-temporal ordering we use a multi-index $i = (i_x, i_t)$ such that a pixel y_i in the image sequence is then referred to with its index $i_x \in \{1,\dots,M\}$ within the frame $i_t \in \{1,\dots,N_t\}$. In the following we denote the multi-index set with $\mathcal{J} = \{1,\dots,M\}\times\{1,\dots,N_t\}$, moreover we use the abbreviation $y = (t,x)$.

Classically, we identify the image sequence with a three-dimensional image and a trilinear interpolation scheme [27]. This means that we use a bilinear interpolation within the frames and an additional linear interpolation between the frames, *i.e.* we define the space

$$V_3^h := \operatorname{span}\{P_i \,|\, P_i \in C^0(D\times I), P_i(y_j) = \delta_{ij}, P_i|_E \text{ is trilinear } \forall E \in \mathcal{G}\} \subset H^1(D\times I)$$

where $\delta_{ij} = \delta_{i_x j_x}\delta_{i_t j_t}$ is the Kronecker-δ applied to the multi-indices i and j. Of course, every image sequence has a representation analog to (2).

To derive a model for stochastic image sequences we must take into account that each frame is recorded by the same set of detector/camera elements. Thus, a sequence with N_t frames and $N = N_1 N_2 N_t$ degrees of freedom is modeled by $N_1 N_2$ random variables only (*cf.* Fig. 2). In fact, the acquisition of gray values for pixels y_i and y_j of the image sequence is modeled by the same stochastic process if $i_x = j_x$. Consequently we have to modify the expansion (3) such that it involves $H^\alpha(\xi_{i_x})$ instead of $H^\alpha(\xi_i)$. Our ansatz space $\mathcal{H}^{h,p}_{\text{seq}} := V_3^h \otimes \mathcal{P}_p \subset L^2(R)\otimes L^2(\Gamma)$ has dimension pN and the discrete random fields have the expansion

$$f(y,\boldsymbol{\xi}) = \sum_{i\in\mathcal{I}}\sum_{\alpha=1}^{p} f_\alpha^i H^\alpha(\xi_{i_x}) P_i(y) = \sum_{\alpha=1}^{p} f_\alpha H^\alpha(\boldsymbol{\xi}). \tag{5}$$

Again by abusing the notation we can define image sequences $f_\alpha(y) := f_\alpha^i P_i(y)$ showing the stochastic modes and thus $f(y,\boldsymbol{\xi}) = \sum_{\alpha=1}^p f_\alpha(y) H^\alpha(\boldsymbol{\xi})$. Let us finally note that we have again constructed a proper finite dimensional sub-space $\mathcal{H}^{h,p}_{\text{seq}} \subset H^1(R)\otimes L^2(\Gamma)$.

Remark 2 *The discretization presented above is equivalent to the following alternative approach: Assume we have discretized each frame of the sequence by standard 2D tent-functions $Q_j^2(x)$ such that a frame with stochastic data has the representation $\sum_{j=1}^M f_j(\xi_j) Q_j^2(x)$. The stochastic processes $f_j(\xi_j)$ model the random behavior of the pixel j with spatial extent* $\operatorname{span} \mathrm{Q}_\mathrm{j}^2$. *Considering an image sequence the stochastic process must be time-dependent,* i.e. *the image has the expansion*

$$f(t,x,\boldsymbol{\xi}) = \sum_{j=1}^{M} f_j(t,\xi_j) Q_j^2(x).$$

Now we discretize the stochastic processes by the Polynomial Chaos and piecewise linear in time, i.e.

$$f_j(t,\xi_j) = \sum_{\alpha=1}^{p}\sum_{k=1}^{N_t}(f_j)_\alpha^k H^\alpha(\xi_j)Q_k^1(t),$$

where Q_j^1 *are* $1D$ *tent-functions. Putting these discretizations together we obtain*

$$f(y,\boldsymbol{\xi}) = f(t,x,\boldsymbol{\xi}) = \sum_{j=1}^{M}\sum_{\alpha=1}^{p}\sum_{k=1}^{N_t}(f_j)_\alpha^k H^\alpha(\xi_j)Q_k^1(t)Q_j^2(x),$$

which is the same as (5) *if we set* $P_i(y) = Q_j^2(x)Q^1k(t)$ *and* $f_\alpha^i = (f_j)_\alpha^k$ *for the multi-index* $i = (j,k)$*. Here we use the fact that the standard* nD *tent-functions are tensor products of the 1D hat functions. We have chosen to present the discretization as above since it is more consistent with the standard discretization of,* e.g. *the optical flow equations with finite elements, although the notation is slightly more complicated.*

3 Building blocks for computer vision with stochastic finite elements

After having introduced finite dimensional ansatz spaces that model stochastic still images as well as stochastic image sequences we present some tools that are needed for image processing tasks with those. Later we will use these building blocks to discretize the stochastic analog of some well known models for image denoising and optical flow computation. The presentation in this section is based on the ansatz space $\mathcal{H}_{\text{still}}^{h,p}$ for stochastic still images. We emphasize that a modification for the ansatz space for stochastic image sequences $\mathcal{H}_{\text{seq}}^{h,p}$ is very simple.

Throughout this section we are going to use the following properties of the polynomial basis H^α, which in fact are satisfied by possible choices (Hermite-, Lagrangian, etc.) of basis functions for the gPC approach:

- The first basis function is constant, $H^1 \equiv 1$ and such that $\mathbb{E}[H^1] = 1$,
- The basis is orthogonal with respect to the measure, *i.e.* for $\alpha \neq \beta$ we have $\langle H^\alpha, H^\beta\rangle = 0$. (6)

As a simple consequence of those properties we directly get for $\alpha > 1$ that the basis functions H^α have zero mean, *i.e.* $\mathbb{E}[H^\alpha] = \langle H^\alpha, H^1\rangle = 0$.

3.1 Mean, Variance, and Covariance

An analysis of the stochastic images involves the stochastic moments of the images' distributions. From the full discretization (5) we can compute the mean of the random field f as

$$\mathbb{E}[f](x) = \int_\Gamma f(x,\boldsymbol{\xi})\,d\mu = \sum_{i\in\mathcal{I}}\sum_{\alpha=1}^{p} f_\alpha^i P_i(x)\int_\Gamma H^\alpha(\xi_i)\,d\mu = \sum_{i\in\mathcal{I}} f_1^i P_i(x) \tag{7}$$

because only the stochastic integral over $H^1(\xi_i)$ (the mean mode) does not vanish.

From here we can proceed using the identity $\mathrm{Var}[f] = \mathbb{E}[f^2] - (\mathbb{E}[f])^2$ to obtain

$$\begin{aligned}\mathrm{Var}[f](x) &= \mathbb{E}[f^2](x) - (\mathbb{E}[f])^2(x) \\ &= \int_\Gamma \sum_{i,j\in\mathcal{I}} \sum_{\alpha,\beta=1}^{p} f_\alpha^i f_\beta^j H^\alpha(\xi_i) H^\beta(\xi_j) P_i(x) P_j(x)\, d\mu - \sum_{i,j\in\mathcal{I}} f_1^i f_1^j P_i(x) P_j(x).\end{aligned}$$

Because of the orthogonality of the polynomials H^α the stochastic integral $\langle H^\alpha(\xi_i), H^\beta(\xi_j)\rangle$ vanishes if $i = j$ and $\alpha \neq \beta$. For the same reason only the term for $\alpha = \beta = 1$ remains if $i \neq j$. Because the first mode is constant equal one we have $\left\langle H^1, H^1 \right\rangle = \mathbb{E}[H^1] = 1$ and thus

$$\begin{aligned}\mathrm{Var}[f](x) &= \sum_{i\in\mathcal{I}} \sum_{\alpha=1}^{p} (f_\alpha^i)^2 \left\langle H^\alpha, H^\alpha \right\rangle P_i^2(x) + \sum_{\substack{i,j\in\mathcal{I} \\ i\neq j}} f_1^i f_1^j P_i(x) P_j(x) - \sum_{i,j\in\mathcal{I}} f_1^i f_1^j P_i(x) P_j(x) \\ &= \sum_{i\in\mathcal{I}} \sum_{\alpha=1}^{p} (f_\alpha^i)^2 \left\langle H^\alpha, H^\alpha \right\rangle P_i^2(x) - \sum_{i\in\mathcal{I}} (f_1^i)^2 P_i^2(x) \\ &= \sum_{i\in\mathcal{I}} \sum_{\alpha=2}^{p} (f_\alpha^i)^2 \left\langle H^\alpha, H^\alpha \right\rangle P_i^2(x).\end{aligned}$$

This expression is not an element of the physical finite element space V_2^h any more. But we can use a standard interpolation (or nodal evaluation) to represent the term in V_2^h. This leads us to

$$\mathrm{Var}[f](x) = \sum_{i\in\mathcal{I}} \sum_{\alpha=2}^{p} (f_\alpha^i)^2 \langle H^\alpha, H^\alpha \rangle P_i(x). \tag{8}$$

Along the same line formulas for higher stochastic moments like skewness, curtosis, *etc.* can be derived.

Let us now assume that we have two random images f and g whose covariance we would like to compute. Using the identity $\mathrm{Cov}[f,g](x) = \mathbb{E}\big[(f - \mathbb{E}[f])(g - \mathbb{E}[g])\big]$ we get

$$\mathrm{Cov}[f,g](x) = \int_\Gamma \sum_{i,j\in\mathcal{I}} \sum_{\alpha,\beta=2}^{p} f_\alpha^i g_\beta^j H^\alpha(\xi_i) H^\beta(\xi_j) P_i(x) P_j(x)\, d\mu.$$

As above, the only non-zero terms of these sums are for $i = j$ and $\alpha = \beta$. Together with the nodal evaluation of the resulting spatial terms this yields

$$\mathrm{Cov}[f,g](x) = \sum_{i\in\mathcal{I}} \sum_{\alpha=2}^{p} f_\alpha^i g_\alpha^i \langle H^\alpha, H^\alpha \rangle P_i(x). \tag{9}$$

As expected we get $\mathrm{Cov}[f,f] = \mathrm{Var}[f]$ from our expressions.

3.2 Projections

Quite often in computer vision (nonlinear) functions of the gray values or their derivatives must be evaluated. In the following we present a recipe for the treatment of such a (nonlinear)

evaluation with stochastic images. So let us consider a function $g : \mathcal{H}^{h,p}_{\text{still}} \to L^2(R) \otimes L^2(\Gamma)$ of a discrete stochastic image f having an expansion as in (3). Examples for g are gradient magnitudes and edge indicator functions

$$g(U) = \nabla u \cdot \nabla u, \qquad \text{and} \qquad g(U) = (1 + |\nabla u|^2/\lambda^2)^{-1}$$

where u is the function defined through (3) given the vector U of degrees of freedom. Both functions are well known and often needed in image processing, *e.g.* for the Perona-Malik diffusion [32].

In general the result of the application of g to U does not lie in $\mathcal{H}^{h,p}_{\text{still}}$ any more. For classical image processing with finite elements this problem arises as well. There, *e.g.* approximations of the gradient with finite differences or inexact quadrature rules are used as a remedy. It seems appealing to use such approximations in the stochastic case as well. However, we emphasize that for nonlinear quantities special care must be taken, since a coupling of the stochastic modes (*cf.* Fig. 4) takes place, which may be difficult to capture with such approximations.

To obtain an expansion of $g(U)$ of the form (3), we compute a L^2-projection $g_{h,p}(x, \boldsymbol{\xi})$ of $g(U)$ onto $\mathcal{H}^{h,p}_{\text{still}}$. The projection is defined by the orthogonality relation

$$\mathbb{E}\left[\int_D g_{h,p}(x,\boldsymbol{\xi})\, H^\beta(\xi_j) P_j(x)\, dx\right] = \mathbb{E}\left[\int_D g(U)(x,\boldsymbol{\xi}) H^\beta(\xi_j) P_j(x)\, dx\right].$$

for $\beta = 1, \dots, p$ and for all $j \in \mathcal{I}$. Substituting the expansion (3) of $g_{h,p}$ into this orthogonality, and denoting the vector of coefficients with $G = (g^i_\alpha)_{i,\alpha}$ yields

$$\sum_{i\in\mathcal{I}} \sum_{\alpha=1}^{p} \left\langle H^\alpha(\xi_i), H^\beta(\xi_j)\right\rangle \int_D g^i_\alpha P_i(x) P_j(x)\, dx = \mathbb{E}\left[\int_D g(U)(x,\boldsymbol{\xi}) H^\beta(\xi_j) P_j(x)\, dx\right] \tag{10}$$

for $\beta = 1, \dots, p$ and for all $j \in \mathcal{I}$. This is a linear system of equations for the coefficients G of the projection $g_{h,p}$. The system can be written as

$$MG = R \qquad \text{with} \qquad R = \left(\mathbb{E}\left[\int_D g(U)(x,\boldsymbol{\xi}) H^\beta(\xi_j) P_j(x)\, dx\right]\right)_{j,\beta} \tag{11}$$

where $M = ((M^{\alpha,\beta})_{ij})$ is the stochastic block-mass matrix

$$(M^{\alpha,\beta})_{ij} = \left\langle H^\alpha(\xi_i), H^\beta(\xi_j)\right\rangle \int_D P_i(x)\, P_j(x)\, dx \tag{12}$$

whose blocks correspond to the modes of u. In Sect. 3.6 we discuss the mass matrix and its assembly in more detail.

So the desired expansion of g is given by the solution of this system which involves the inversion of the stochastic mass matrix: $G = M^{-1}R$. This inversion of M may be computationally intensive. However, using mass lumping [39] can simplify the effort enormously, since it diagonalizes the stochastic mass matrix. Lumping of masses yields a block diagonal mass matrix $\widetilde{M}$ such that

$$(\widetilde{M}^{\alpha,\beta})_{ij} = \delta_{i,j}\delta_{\alpha,\beta} \sum_{k\in\mathcal{I}} \sum_{\gamma=1}^{p} (M^{\alpha,\gamma})_{ik} = \delta_{i,j}\delta_{\alpha,\beta} \sum_{k\in\mathcal{I}} \sum_{\gamma=1}^{p} \langle H^\alpha(\xi_i), H^\gamma(\xi_k)\rangle \int_D P_i(x)\, P_k(x)\, dx$$

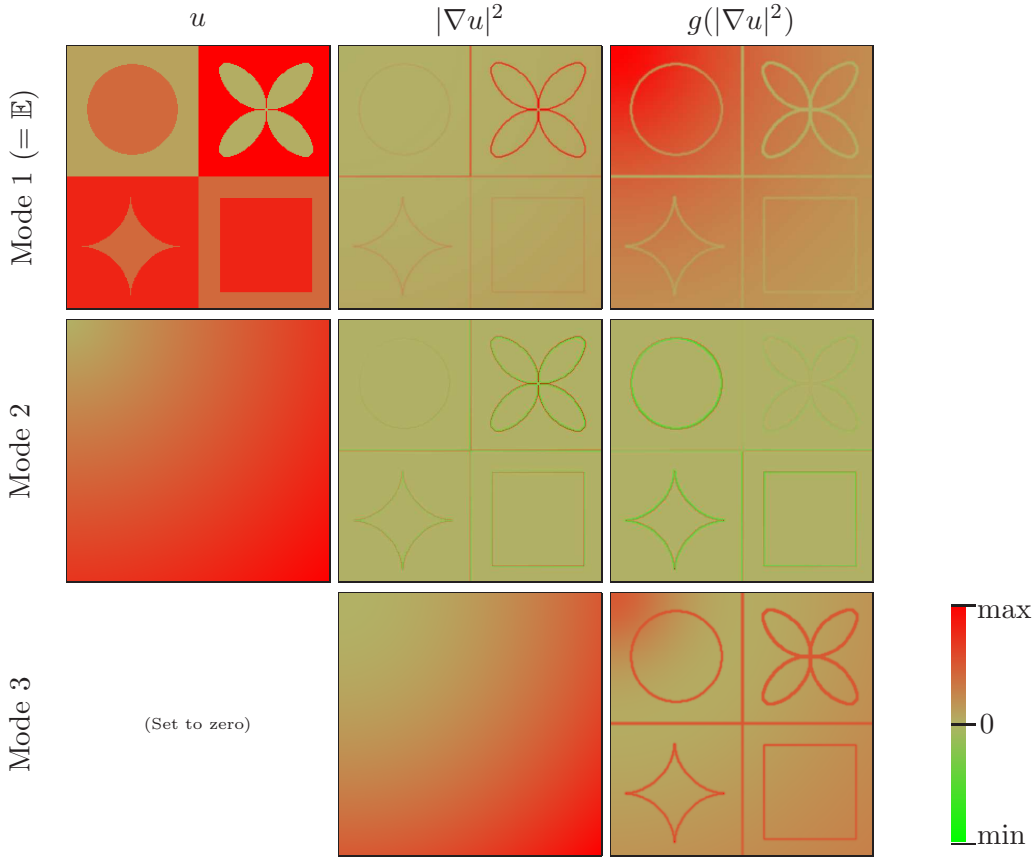


Figure 3: For the test evaluation of the gradient magnitude and the edge indicator (*cf.* Fig. 4) on a test-image we show the stochastic modes. According to (7) the mean corresponds to the first mode of the expansion. The variance must be computed from the remaining modes according to (8). We have used a color coding as shown in the color ramp (bottom right) to make a differentiation between positive and negative values possible. Please note that we have scaled the images to match the full color range. Thus, the colors give a qualitative impression only.

In Sect. 3.6.1 we describe how to compute this mass matrix. For the assembly of the right hand side R we have different options. Usually one would use a stochastic quadrature rule to evaluate the expectation in the right hand side. However, if a direct expansion of g as a product of expansions as (3) is available, we can proceed differently as we shall see in the next paragraph.

3.3 Gradient magnitude

Let us use the projection discussed in the last paragraph to derive an expression for the gradient magnitude of a stochastic image. We consider $g(U) = |\nabla u|^2 \ = \nabla u \cdot \nabla u$ and insert this into

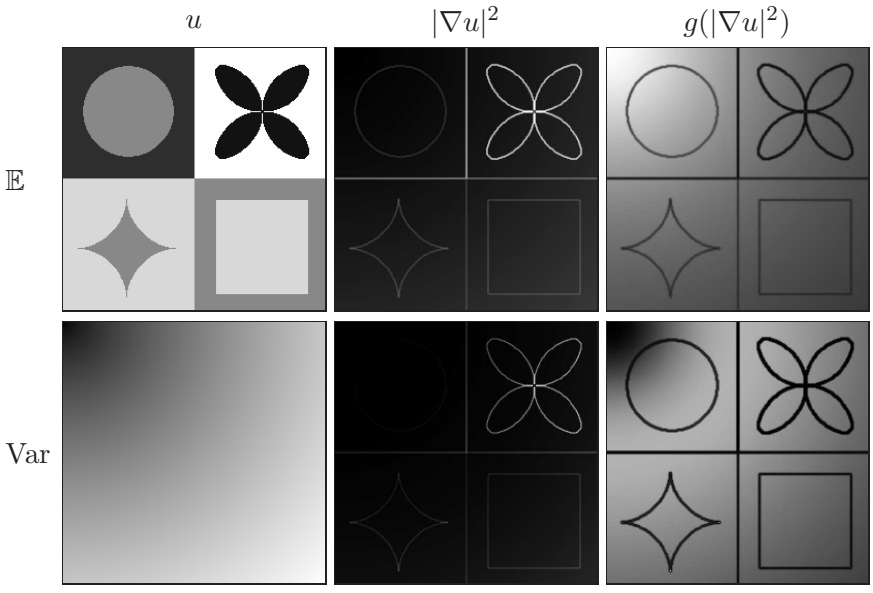


Figure 4: For a sample image (left) we show the gradient magnitude (middle) and an edge indicator (right). The expectations are shown in the top row, whereas the variances are shown in the bottom row. Please note that we show the gradient images with a contrast enhancement to better visualize their global variance.

the orthogonality relation (10). Directly using mass lumping leads us to

$$g_\alpha^i = (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1}\, \mathbb{E}\left[\int_D (\nabla u \cdot \nabla u)(x,\boldsymbol{\xi}) H^\alpha(\xi_i) P_i(x)\, dx\right].$$

But using the basis representation (3) of ∇u we can write $\nabla u \cdot \nabla u$ as

$$(\nabla u \cdot \nabla u)(x,\boldsymbol{\xi}) = \sum_{j,k\in\mathcal{I}} \sum_{\beta,\gamma=1}^{p} u_\beta^j u_\gamma^k H^\beta(\xi_j) H^\gamma(\xi_k) \nabla P_j(x) \cdot \nabla P_k(x).$$

If we order the DOF of u in the vector U as in (4) we can derive from this the block-system

$$g_\alpha^i = \widetilde{M}^{-1}_{(i,\alpha),(i,\alpha)} U \cdot K_{(i,\alpha)}\, U, \tag{13}$$

where the block-matrix $K_{(i,\alpha)}$ is defined by

$$\left(\left(K_{(i,\alpha)}\right)^{\beta,\gamma}\right)_{j,k} = \left\langle H^\alpha(\xi_i)\, H^\beta(\xi_j), H^\gamma(\xi_k)\right\rangle \int_D P_i(x)\, \nabla P_j(x) \cdot \nabla P_k(x)\, dx.$$

Again, the blocks of this matrix correspond to the modes of u. Here, $K_{(i,\alpha)}$ is not a block-diagonal matrix, thus there is a coupling between the modes of U. This is the reason why standard approximations on the modes (like finite differences for each mode) do not yield the correct result. In Sect. 3.6.1 we discuss the assembly and the structure of this matrix in more detail.

In Figs. 3 and 4 we show the computation of the gradient magnitude on a test-image. For the computations we have used $p = 3$ and Legendre-Polynomials as basis functions for the stochastic processes on $\Gamma_* = [-1, 1]$. From the images it is clearly visible how the modes are coupled through the nonlinear-operators, *i.e.* the spatial variation of the variance is visible in the mean of the gradient-magnitude. And vice versa we see, that the variance captures the gradients of the mean input image. The spatial resolution of the test image is 257×257, and the grid width is scaled to $h = 1$. The gray values range in the interval $[0, 1]$ and we have used a value of $\lambda = 0.01$.

3.4 Edge indicator function

The computation of the gradient magnitude could be simplified by taking mass lumping into account. Furthermore it benefitted from the fact that we can directly write down the expansion of $\nabla u \cdot \nabla u$ as a product of sums. The setting is more complicated if we consider a nonlinear edge indicator function, *e.g.*

$$g(V) = (1 + V/\lambda^2)^{-1}, \tag{14}$$

where V is the representation of $|\nabla u|^2$ whose coefficients have been derived in (13). As above, we insert this function into the right hand side (11) and get

$$g_\alpha^i = (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1} \, \mathbb{E}\left[\int_D g(V)(x,\boldsymbol{\xi}) H^\alpha(\xi_i) P_i(x)\, dx\right] = (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1} \int_\Gamma \int_R g(V)(x,\boldsymbol{\xi}) H^\alpha(\xi_i) P_i(x)\, dx\, d\mu.$$

To be more specific let us substitute the actual edge indicator function $g(V) = (1 + V/\lambda^2)^{-1}$ and the expansion for V. This leads us to the identity

$$g_\alpha^i = (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1} \int_\Gamma \int_R \frac{H^\alpha(\xi_i) P_i(x)}{1 + \lambda^{-2} \sum_{j\in\mathcal{I}} \sum_{\beta=1}^p V_\beta^j H^\beta(\xi_j) P_j(x)}\, dx\, d\mu \tag{15}$$

which is only computable with a quadrature rule in the physical and the stochastic space. This will be discussed in Sect. 3.6.1.

In Figs. 3 and 4 we show the evaluation of the edge-indicator function, where we have used the quadrature rules described in the next section. Again we have used Legendre-polynomials and $p = 3$. And again it is clearly visible from the images how the stochastic modes are coupled through the nonlinear edge indicator function.

3.5 Diffusion- and Structure-tensors

The concepts presented in the last paragraph can easily be generalized to tensor valued functions. If we consider *e.g.* the structure tensor $J = (\nabla f)^T \nabla f$ of a stochastic image f we need to compute the stochastic representation of derivatives $(\partial_m f)(\partial_n f)$. Those quantities, however, can be obtained with the projection technique from Sect. 3.3. We just have to replace the expansion of $\nabla u \cdot \nabla u$ with the expansion of the desired product of derivatives $(\partial_m f)(\partial_n f)$. In Fig. 5 we show the three components of the structure tensor of the test image from Fig. 4.

3.6 Stochastic Integrals

In the last sections we have defined multiple quantities which involve integration over the random space Γ. In this section we describe how to evaluate these high-dimensional integrals and

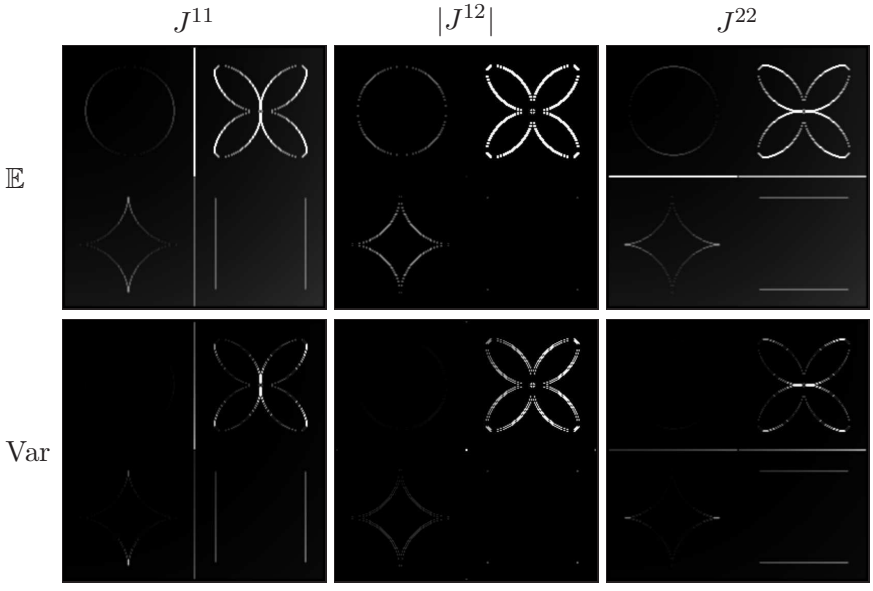


Figure 5: For the test image shown in Fig. 4 we depict the stochastic moments of the structure tensor. Please note that again we have enhanced the contrast of the gray values for the presentation.

how to use quadrature rules to compute coefficients as in (15). Although Γ is an M-dimensional space, where M is the number of random variables that model the stochastic behavior of the image or image sequence, the computation of the integrals is not very complicated. Let us first focus on the matrices and tensors that appeared in the previous sections. Then we discuss the numerical stochastic quadrature that is used to compute the coefficients of the edge indicator.

3.6.1 Stochastic Matrices

During the last sections we have encountered the inner product on the random space multiple times, *i.e.* integrals over pairs or triples of stochastic basis functions. Those integrals are the coefficients that are multiplied with integrals in the physical space *e.g.* like for coefficients of the stochastic mass matrix (12). In general it is possible to separate the stochastic and the spatial integration such that in most cases the computation of integrals reduces to a component-wise multiplication of an integral over the random space with a integral over the physical space.

The expectations of products of tuples of stochastic basis functions play a central role in the concept presented in this work. In the section on a stochastic Galerkin method for diffusion and optical flow models, we will need those expectations again. So let us focus on the integral

$$\begin{aligned}\left\langle H^\alpha(\xi_i), H^\beta(\xi_j)\right\rangle &= \mathbb{E}\left[H^\alpha(\xi_i)H^\beta(\xi_j)\right] = \int_\Gamma H^\alpha(\xi_i)H^\beta(\xi_j)\,d\mu \\ &= \int_{\Gamma_1}\cdots\int_{\Gamma_M} H^\alpha(\xi_i)H^\beta(\xi_j)\rho(\xi_1)\,d\xi_1\cdots\rho(\xi_M)\,d\xi_M\end{aligned}$$

Already for images of moderate size M a computation of this high dimensional integral seems not feasible. But fortunately, we have to integrate over a small number (here 2) stochastic

coordinates (random variables) only. Moreover, the values of the integrals do not depend on the actual physical location of the corresponding pixel (here the values of i and j). Only we have to decide whether the locations coincide, if the pixels are neighbors, or if their spatial extent does not overlap. So without loss of generality, we can assume that the spatial locations of the random variables lie within a master-element and thus attain the values $\{1, \dots, 4\}$.

Having this simplification in mind, we can easily compute the expectation of products of pairs, triples and quadruples of stochastic basis functions, which are the quantities that we need for the concept presented in this work. For every choice of polynomial basis (Legendre, Hermite, Laguerre, *etc.*) we can store the values in a lookup table.

Using the notation $d\mu_i = \rho_i(\xi_i)\, d\xi_i$ we define the tensors

$$A^{i,j,m,n}_{\alpha,\beta,\gamma,\delta} = \begin{cases} \displaystyle\int_{\Gamma_1} H_\alpha(\xi_1)H_\beta(\xi_1)H_\gamma(\xi_1)H_\delta(\xi_1)\, d\mu_1, & i=j=k=l, \\ \displaystyle\int_{\Gamma_1}\int_{\Gamma_2} H_\alpha(\xi_1)H_\beta(\xi_1)H_\gamma(\xi_1)H_\delta(\xi_2)\, d\mu_1 d\mu_2, & i=j=k\neq l, \\ \vdots & \text{permutations of three equal latin indices}, \\ \displaystyle\int_{\Gamma_1}\int_{\Gamma_2}\int_{\Gamma_3} H_\alpha(\xi_1)H_\beta(\xi_2)H_\gamma(\xi_3)H_\delta(\xi_3)\, d\mu_1 d\mu_2 d\mu_3, & i,j\neq k=l, i\neq j, \\ \vdots & \text{permutations of two equal latin indices}, \\ \displaystyle\int_{\Gamma_1}\int_{\Gamma_2}\int_{\Gamma_3}\int_{\Gamma_4} H_\alpha(\xi_1)H_\beta(\xi_2)H_\gamma(\xi_3)H_\delta(\xi_4)\, d\mu_1 d\mu_2 d\mu_3 d\mu_4 & \text{all latin indices are different}, \end{cases} \tag{16}$$

$$B^{i,j,k}_{\alpha,\beta,\gamma} = \begin{cases} \displaystyle\int_{\Gamma_1} H_\alpha(\xi_1)H_\beta(\xi_1)H_\gamma(\xi_1)\, d\mu_1, & i=j=k, \\ \displaystyle\int_{\Gamma_1}\int_{\Gamma_2} H_\alpha(\xi_1)H_\beta(\xi_1)H_\gamma(\xi_2)\, d\mu_1 d\mu_2, & i=j\neq k, \\ \vdots & \text{permutation of two equal latin indices}, \\ \displaystyle\int_{\Gamma_1}\int_{\Gamma_2}\int_{\Gamma_3} H_\alpha(\xi_1)H_\beta(\xi_2)H_\gamma(\xi_3)\, d\mu_1 d\mu_2 d\mu_3, & i\neq j, i\neq k, j\neq k, \end{cases}$$

$$C^{i,j}_{\alpha,\beta} := \begin{cases} \displaystyle\int_{\Gamma_1} H_\alpha(\xi_1)H_\beta(\xi_1)\, d\mu_1, & i=j, \\ \displaystyle\int_{\Gamma_1}\int_{\Gamma_2} H_\alpha(\xi_1)H_\beta(\xi_2)\, d\mu_1 d\mu_2, & i\neq j, \end{cases}$$

which give us the desired lookup-tables for all possible combinations of tuples of basis functions and their locations within the master element. We have used the fact that $|\Gamma_i| = \int_{\Gamma_i} d\mu_i = \int_{\Gamma_i} \rho_i(\xi_i) d\xi_i = 1$ for any random variable ξ_i. Using (6) we can easily derive some properties of the tensors A, B, and C, *e.g.* we can check whether an entry is zero due to reasons of orthogonality.

Let us finally note that the storage of the integrals over tuples of spatial basis functions or their derivatives in lookup tables is a standard approach in the numerical treatment of finite element methods (*cf. e.g.* [33]). In our setting many of the matrix entries and integrals can be computed by pairwise multiplication of an entry from the stochastic lookup table with an entry from the spatial lookup table.

3.6.2 Stochastic Quadrature

To concretize the quadrature announced in Sect. 3.4 for the computation of the edge indicator function we use an M-dimensional quadrature rule for the stochastic integral. This quadrature shall be the tensor product of a d-point one-dimensional quadrature rule. Thereby we use the fact (*cf.* Sect. 2.2) that all random processes are defined on the same domain $\Gamma_i = \Gamma_*$. Let us denote the weights of the one-dimensional quadrature by $\{\kappa_1, \ldots, \kappa_d\} \in I\!R^d$ and the quadrature points with $\{z_0, \ldots, z_d\} \in \Gamma_*^d$. Thus, we have to compute

$$g_\alpha^i \approx (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1} \sum_{l_1.,\ldots,l_M=1}^{d} \kappa_{l_1} \cdots \kappa_{l_M} H^\alpha((\boldsymbol{z}_l)_i) \int_R g(V)(x, \boldsymbol{z}_l) P_i(x)\, dy,$$

where we use $\boldsymbol{z}_l = (z_{l_1}, \ldots, z_{l_M}) \in \Gamma$ such that $(\boldsymbol{z}_l)_i = z_{l_i}$. Fortunately, the spatial basis function P_i has compact support and the random variables are coupled to the support of spatial basis functions. Thus we must integrate over the stochastic variables only, whose supporting spatial basis function intersect the support of P_i. Splitting the integral over D into a sum of integrals over the elements E of the grid $\mathcal{G}$ yields

$$g_\alpha^i \approx (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1} \sum_{E \in \mathcal{G}} \sum_{\substack{l_j=1 \\ x_j \in E}}^{d} (\prod_j \kappa_{l_j}) H^\alpha((\boldsymbol{z}_l)_i) \int_E g(V)(x, \boldsymbol{z}_l) P_i(x)\, dy.$$

So on each element $E \in \mathcal{G}$ the M-dimensional stochastic integral reduces to an 4 dimensional stochastic integral, because every element E has 4 vertices.

For the particular edge indicator function from Sect. 3.4 this means

$$g_\alpha^i \approx (\widetilde{M}^{\alpha,\alpha})_{ii}^{-1} \sum_{E \in \mathcal{G}} \sum_{\substack{l_j=1 \\ x_j \in E}}^{d} (\prod_j \kappa_{l_j}) H^\alpha(\boldsymbol{z}_{l_i}) \int_E \frac{P_i(x)}{1 + \lambda^{-2} \sum\limits_{\substack{k \in \mathcal{I} \\ x_k \in E}} \sum_{\beta=1}^{p} V_\beta^k H^\beta(z_{l_k}) P_k(x)} dy.$$

Although it looks still complicated, this integral can easily be computed with a traversal of the grid and local operations on the elements E.

4 Stochastic Galerkin Method for Diffusion Filtering

In the following sections we show the usage of the stochastic finite element discretization and the building blocks presented in the preceding sections. To do so, we apply them to stochastic versions of well known and simple partial differential equations frequently used in computer vision. We have chosen a selection of very basic and simple diffusion equations for which the computer vision community has gained a wide understanding of and insight. Our intention is to enable the reader to gain understanding of computer vision with SPDEs and to see relations to classical/deterministic approaches with PDEs on basis of those simple models. We are neither going to address issues of existence and uniqueness of solutions of the continuous SPDEs, nor are we going to discuss the spaces in which solutions reside. Instead, we point the interested reader to the ongoing research in the area of Galerkin FEM for SPDEs (*cf. e.g.* [10, 19, 14, 46]).

4.1 Linear Diffusion Filtering

Let us start with the SPDE formulation of the most prominent linear filter, the heat equation which yields the Gaussian Scale-Space [44]. Starting from the well known classical form we can directly substitute the gray-value functions with the random fields introduced in Sect. 2. Thus, the stochastic version of the heat equation reads:

> Given initial data $f(x, \boldsymbol{\xi})$ find a family $\{u(t, x, \boldsymbol{\xi})\}_{t \in \mathbb{R}^+}$ of filtered versions of this image such that
>
> $$\begin{aligned} \partial_t u(t, x, \boldsymbol{\xi}) - \Delta u(t, x, \boldsymbol{\xi}) &= 0 && \text{a.e. in } \mathbb{R}^+ \times D \times \Gamma, \\ \partial_\nu u(t, x, \boldsymbol{\xi}) &= 0 && \text{a.e. on } \mathbb{R}^+ \times \partial D \times \Gamma, \\ u(0, x, \boldsymbol{\xi}) &= f(x, \boldsymbol{\xi}) && \text{a.e. in } D \times \Gamma. \end{aligned}$$

In a first step towards discretization let us apply the usual temporal backwards Euler scheme and replace the temporal derivative by a backward difference quotient, thus introducing the time-step τ and the notation $u^n(x, \boldsymbol{\xi}) := u(n\tau, x, \boldsymbol{\xi})$. Consequently we have transferred the parabolic SPDE into a sequence of elliptic SPDEs:

> For $n = 1, 2, 3, \ldots$ find $u^n : D \times \Gamma \to \mathbb{R}$ such that
>
> $$\begin{aligned} u^n(x, \boldsymbol{\xi}) - \tau \Delta u^n(x, \boldsymbol{\xi}) &= u^{n-1}(x, \boldsymbol{\xi}) && \text{a.e. in } D \times \Gamma, \\ \partial_\nu u^n(x, \boldsymbol{\xi}) &= 0 && \text{a.e. on } \mathbb{R}^+ \times \partial D \times \Gamma, \end{aligned} \tag{17}$$
>
> where $u^0(x, \boldsymbol{\xi}) = f(x, \boldsymbol{\xi})$ a.e. in $D \times \Gamma$.

Each of the equations of this sequence must be interpreted in a weak sense: Following the stochastic Galerkin Method [10], we multiply this equation by a test-function $z \in H^1(D) \otimes L^2(\Gamma)$, we integrate over the physical domain R and consider the expectation of the resulting integrals. This yields

$$\mathbb{E}\left[\int_R u^n(x, \boldsymbol{\xi})\, z(x, \boldsymbol{\xi})\, dx\, d\mu - \tau \int_R \Delta u^n(x, \boldsymbol{\xi})\, z(x, \boldsymbol{\xi})\, dx\, d\mu\right] = \mathbb{E}\left[\int_R u^{n-1}(x, \boldsymbol{\xi})\, z(x, \boldsymbol{\xi})\, dx\, d\mu\right]$$

for all test-functions $z \in H^1(D) \otimes L^2(\Gamma)$. Integrating by parts in R leads to the weak form of the SPDE (17)

$$\mathbb{E}\left[\int_R u^n(x, \boldsymbol{\xi})\, z(x, \boldsymbol{\xi})\, dx\, d\mu - \tau \int_R \nabla u^n(x, \boldsymbol{\xi}) \cdot \nabla z(x, \boldsymbol{\xi})\, dx\, d\mu\right] = \mathbb{E}\left[\int_R u^{n-1}(x, \boldsymbol{\xi})\, z(x, \boldsymbol{\xi})\, dx\, d\mu\right]$$

Now, approximating the solution in the finite-dimensional sub-space, *i.e.* considering $u^n \in \mathcal{H}^{p,h}_{\text{still}}$, we can substitute the expansion (5) into this weak form. Moreover we plug in the basis functions $H^\beta(\xi_j)P_j(x)$ as test-functions z. Using the linearity of the expectation we get

$$\begin{aligned} \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^{p} (u^n)^i_\alpha \Bigg(& \mathbb{E}\left[\int_R H^\alpha(\xi_i) P_i(x)\, H^\beta(\xi_j) P_j(x)\, dx\, d\mu\right] \\ & - \tau \mathbb{E}\left[\int_R \big(H^\alpha(\xi_i) \nabla P_i(x)\big) \cdot \big(H^\beta(\xi_j) \nabla P_j(x)\big)\, dx\, d\mu\right]\Bigg) \\ & = \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^{p} (u^{n-1})^i_\alpha \mathbb{E}\left[\int_R H^\alpha(\xi_i) P_i(x)\, H^\beta(\xi_j) P_j(x)\, dx\, d\mu\right] \end{aligned}$$

for all $j \in \mathcal{I}$ and $\beta = 1, \ldots, p$. This is a system of equations which can be written in a block form

$$\sum_{\alpha=1}^{p} (M^{\alpha,\beta} + \tau L^{\alpha,\beta})(U^n)_\alpha = \sum_{\alpha=1}^{p} M^{\alpha,\beta}(U^{n-1})_\alpha \qquad \text{for } \beta = 1, \ldots, p \tag{18}$$

ordering the unknowns of u^n as in (4). The matrices $M^{\alpha,\beta}$ and $L^{\alpha,\beta}$ are stochastic mass- and stiffness-matrices, respectively. They have the entries

$$\begin{aligned}
(M^{\alpha,\beta})_{ij} &= \mathbb{E}\left[\int_R H^\alpha(\xi_i) P_i(x)\, H^\beta(\xi_j) P_j(x)\, dx\, d\mu\right] \\
&= \left(\int_\Gamma H^\alpha(\xi_i)\, H^\beta(\xi_j)\, d\mu\right)\left(\int_R P_i(x)\, P_j(x)\, dx\right) \qquad (19) \\
&= C^{i,j}_{\alpha,\beta}\left(\int_R P_i(x)\, P_j(x)\, dx\right) \\
(L^{\alpha,\beta})_{ij} &= \mathbb{E}\left[\int_R \big(H^\alpha(\xi_i)\nabla P_i(x)\big)\cdot\big(H^\beta(\xi_j) P_j(x)\big)\, dx\, d\mu\right] \\
&= \left(\int_\Gamma H^\alpha(\xi_i)\, H^\beta(\xi_j)\, d\mu\right)\left(\int_R \nabla P_i(x)\cdot\nabla P_j(x)\, dx\right) \qquad (20) \\
&= C^{i,j}_{\alpha,\beta}\left(\int_R \nabla P_i(x)\cdot\nabla P_j(x)\, dx\right)
\end{aligned}$$

where we have used the tensor C from (16). The matrices result from the classical mass- and stiffness-matrices by an entry-wise multiplication with the expectance of pair-products of stochastic basis functions. The coefficient $C^{i,j}_{\alpha,\beta}$ is zero for $\alpha \neq \beta$ due to the orthogonality of the basis functions H^α. Consequently the resulting system is block-diagonal (*cf.* Fig. 6) where each diagonal block corresponds to the smoothing of one stochastic mode of the image.

Remark 3 *The fact that the stochastic heat equation leads to a block diagonal is due to the linearity of the heat equation. Already in Sect. 3.3 we have seen that a nonlinear operator couples the stochastic modes and thus results in a dense system (in the stochastic space).*

Due to the block structure we can implement the solution of the stochastic heat equation very efficiently, because we can use our existing deterministic FEM code on each of the stochastic modes separately, provided we have multiplied the deterministic system matrix componentwise with the tensor $C^{ij}_{\alpha,\beta}$. We are going to discuss results of the stochastic linear diffusion in Sect. 4.3.

4.2 Perona Malik Diffusion

In a second step we would like to implement a stochastic version of the well known nonlinear Perona Malik diffusion [32]. Proceeding completely analog to Sect. 4.1 we formulate this problem as:

> Given initial data $f(x, \boldsymbol{\xi})$ find a family $\{u(t, x, \boldsymbol{\xi})\}_{t \in \mathbb{R}^+}$ of filtered versions of this image such that
>
> $$\begin{aligned}
> \partial_t u(t, x, \boldsymbol{\xi}) - \operatorname{div}(g(|\nabla u(t, x, \boldsymbol{\xi})|)\nabla u(t, x, \boldsymbol{\xi}) = 0 \qquad & \text{a.e. in } \mathbb{R}^+ \times D \times \Gamma, \\
> \partial_\nu u(t, x, \boldsymbol{\xi}) = 0 \qquad & \text{a.e. on } \mathbb{R}^+ \times \partial D \times \Gamma, \\
> u(0, x, \boldsymbol{\xi}) = f(x, \boldsymbol{\xi}) \qquad & \text{a.e. in } D \times \Gamma.
> \end{aligned}$$

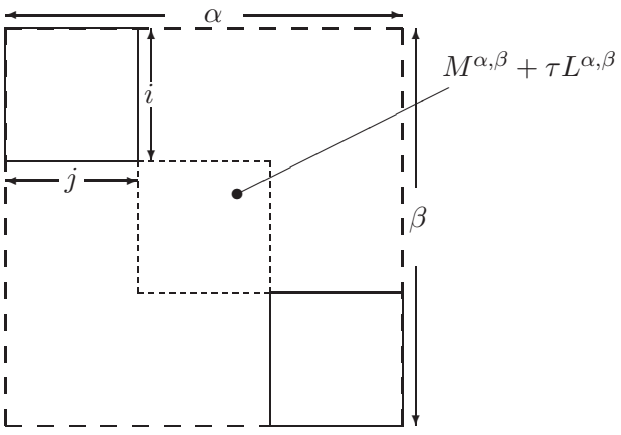


Figure 6: Structure of the block system of the stochastic heat equation (18).

Here g is the edge indicator function we have already worked with in Sect. 3.4. Again we apply the backward Euler approximation of the temporal derivative and evaluate the non-linearity at the old time-step leading to a semi-implicit scheme. Interpreting the equation in a weak sense yields

$$\mathbb{E}\left[\int_R u^n(x,\boldsymbol{\xi})\, z(x,\boldsymbol{\xi})\, dx\, d\mu - \tau \int_R g(|\nabla u^{n-1}(t,x,\boldsymbol{\xi})|)\nabla u^n(x,\boldsymbol{\xi}) \cdot \nabla z(x,\boldsymbol{\xi})\, dx\, d\mu\right] = \mathbb{E}\left[\int_R u^{n-1}(x,\boldsymbol{\xi})\, z(x,\boldsymbol{\xi})\, dx\, d\mu\right]$$

for all test functions $z \in H^1(D) \otimes L^2(\Gamma)$. Considering $u^n \in \mathcal{H}^{h,p}_{\text{still}}$ leads to a system of equations

$$\sum_{i\in\mathcal{I}}\sum_{\alpha=1}^{p}(u^n)^i_\alpha \left(\mathbb{E}\left[\int_R H^\alpha(\xi_i)P_i(x)\, H^\beta(\xi_j)P_j(x)\, dx\, d\mu\right] - \tau\mathbb{E}\left[\int_R \left(\sum_{\gamma=1}^{p}\sum_{k\in\mathcal{I}}(G^n)^k_\gamma H^\gamma(\xi_k)P_k(x)\right)\left(H^\alpha(\xi_i)\nabla P_i(x)\right)\cdot\left(H^\beta(\xi_j)\nabla P_j(x)\right) dx\, d\mu\right]\right) = \sum_{i\in\mathcal{I}}\sum_{\alpha=1}^{p}(u^{n-1})^i_\alpha \mathbb{E}\left[\int_R H^\alpha(\xi_i)P_i(x)\, H^\beta(\xi_j)P_j(x)\, dx\, d\mu\right]$$

for each $\beta = 1,\dots,p$ and $j \in \mathcal{I}$. Here we have substituted the expansion (5) for the edge indicator $g(|\nabla u^n|)$ which has been derived in Sect. 3.4.

Remark 4 *For the classical deterministic diffusion equation we have to assert that the diffusion tensor is positive definite such that the resulting equation is elliptic and the resulting bilinear form coercive. An analog condition must hold for the stochastic diffusion equation [10] and we have to take care of the positivity of the stochastic process of the diffusion tensor g. In our numerical experiments we did not encounter any problems with the projection and quadrature method described in Sections 3.4 and 3.6.2.*

The above system of equations is again a block-system which can be written as

$$\sum_{\alpha=1}^{p}\left(M^{\alpha,\beta} + \tau(L^n)^{\alpha,\beta}\right)(U^n)_\alpha = \sum_{\alpha=1}^{p} M^{\alpha,\beta}(U^{n-1})_\alpha \qquad \text{for } \beta = 1,\dots,p$$

if we order the unknowns of u^n as in (4). The mass matrices $M^{\alpha,\beta}$ are as before in (19) and $(L^n)^{\alpha,\beta}$ is now

$$\begin{aligned}\left((L^n)^{\alpha,\beta}\right)_{ij} &= \mathbb{E}\left[\int_R \left(\sum_{\gamma=1}^{p}\sum_{k\in\mathcal{I}} (G^n)_\gamma^k H^\gamma(\xi_k)P_k(x)\right)\left(H^\alpha(\xi_i)\nabla P_i(x)\right)\cdot\left(H^\beta(\xi_j)P_j(x)\right)dx\,d\mu\right] \\ &= \sum_{\gamma=1}^{p}\sum_{k\in\mathcal{I}} (G^n)_\gamma^k \left(\int_\Gamma H^\alpha(\xi_i)\,H^\beta(\xi_j)\,H^\gamma(\xi_k)\,d\mu\right)\left(\int_R \nabla P_i(x)\cdot\nabla P_j(x)\,P_k(x)\,dx\right) \\ &= \sum_{\gamma=1}^{p}\sum_{k\in\mathcal{I}} (G^n)_\gamma^k\, B_{\alpha,\beta,\gamma}^{i,j,k}\left(\int_R \nabla P_i(x)\cdot\nabla P_j(x)\,P_k(x)\,dx\right).\end{aligned} \tag{21}$$

This block system is not block-diagonal any more, because of the stochastic diffusion tensor g which leads to a entry-wise multiplication and summation with the tensor B in the block system.

Integration of the entries of $(L^n)^{\alpha,\beta}$ can be simplified by using an inexact integration scheme. If E denotes one hexahedral element of the grid, we can use

$$\sum_{\gamma=1}^{p}\sum_{k\in\mathcal{I}} (G^n)_\gamma^k\, B_{\alpha,\beta,\gamma}^{i,j,k}\left(\int_E \nabla P_i(x)\cdot\nabla P_j(x)\,P_k(x)\,dx\right) \approx \widetilde{G}\,{}_{\alpha,\beta}^{i,j}\int_E \nabla P_i(x)\cdot\nabla P_j(x)\,dx$$

where

$$\widetilde{G}\,{}_{\alpha,\beta}^{i,j} = \frac{1}{4}\sum_{\gamma=1}^{p}\sum_{k\in\mathcal{I}\cap E} (G^n)_\gamma^k\, B_{\alpha,\beta,\gamma}^{i,j,k}$$

is the evaluation of the diffusion coefficient at the center-point of the element E. Using this approximation simplifies implementation of the matrix assembly, since existing FEM code for the Perona-Malik model can be reused.

4.3 Results

In Fig. 7 we show results of the linear and nonlinear diffusion applied to a stochastic test-image. We have chosen to work with a stochastic basis containing Legendre polynomials up to order $p = 4$. The image is of resolution 129×129 and the gray values range in the interval $[0, 1]$. For sake of simplicity, the variance of the input data is defined by setting the second mode to

$$f_2(x) = \delta\,|x|, \qquad \delta = 10^{-8}$$

and letting all higher modes be zero. This means that we model a uniform distribution of the input gray values (*cf.* Fig. 1). In Sect. 5.6 we will model Gaussian-shaped distributions as well.

In our example the variance of the input data ranges from zero to $\max_x \sqrt{\delta|x|} \approx 0.009$. Let us note that for uniform distributions over an interval $[c-d, c+d]$ the mean is given by c and the variance is $\sigma^2 = d^2/6$. Vice versa, given the variance σ^2 the interval half width is $d = \sqrt{6}\sigma$. This means in our example we model an uncertainty of the gray values varying up to ± 0.23 around the mean gray value. For the computations we have chosen a time-step of $\tau = 1$, the grid-width is scaled to $h = 1$, and we use $\lambda = 0.02$.

From the images shown in Fig. 7 we see that as expected from theory the heat equation damps the modes separately without coupling them. In fact the damping is very fast, and the

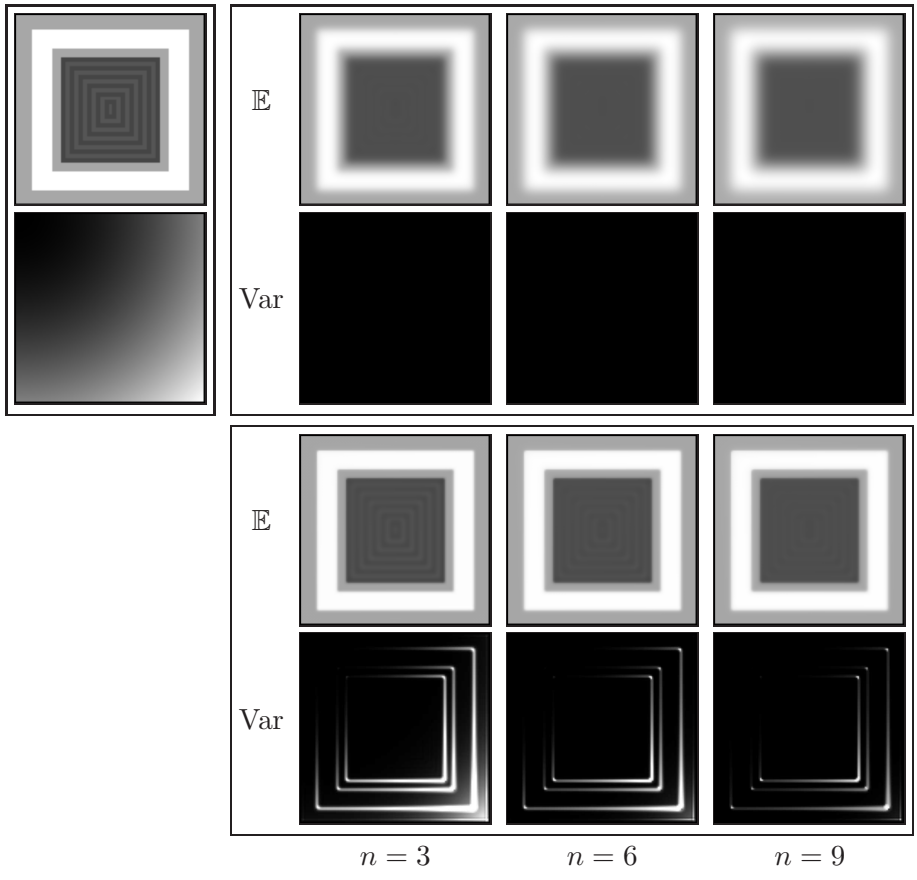


Figure 7: For a stochastic input image (mean and variance depicted in the top left frame) we show several scale steps from the linear diffusion (top frame) and the nonlinear Perona-Malik diffusion (bottom frame). In each frame the top row shows the mean and the bottom row shows the variance. For the linear diffusion the variance of the image drops by more than one order of magnitude per scale step. Thus the variance-images appear black. In contrast to that, the coupling of the image gradient onto the variance is nicely visible from the Perona-Malik images.

maximum value of the variance drops by more than one order of magnitude per scale-step. Thus the images depicted in the corresponding row of Fig. 7 are black. If we adjusted the contrast, the variances of the smoothed image would have the same structure as the input variance.

In contrast to that, the variance of the images from Perona-Malik diffusion show the structure of the gradient. But there also a smoothing of the variance is present with increasing scale step n. In the limit $n \to \infty$ the results of the linear and nonlinear diffusion models will be the same, *i.e.* an image of constant gray value and zero variance.

5 Optical Flow Computations

In the following sections the stochastic finite element method is applied to slightly more complex models for the well known optical flow problem. Intentionally we have chosen very simple models like the Horn and Schunck (HS) model [16], a discontinuity preserving optical flow model [3, 9, 41], and the combined local global (CLG) model [5, 6].

5.1 A stochastic optic flow equation

Let us first derive an optical flow equation for stochastic images. We consider a given noisy image sequence $f : D \times I \to I\!R$ on a spatial domain $D \subset I\!R^2$ and a time interval $I := [0, T]$ for $T > 0$. To distinguish spatial from spatio-temporal derivatives we introduce the notation $\nabla_{x,t}$ for the space-time gradient and ∇_x for the purely spatial gradient. The partial derivatives are denoted with ∂_1, ∂_2, and ∂_t and, finally, we write $R := D \times I$.

We search for a vector field (optical flow field) $w : D \times I \to I\!R^2$ such that $w = (u, v)$ describes the motion of structures inside the image sequence f. As usual, we consider the gray values to be conserved – here along stochastic trajectories $x(t, \boldsymbol{\xi})$ of objects through the image sequence

$$f(x(t, \boldsymbol{\xi}), t, \boldsymbol{\xi}) = \text{const.}$$

The stochastic trajectories yield a stochastic optical flow field $w(x, t, \boldsymbol{\xi}) = \dot{x}(t, \boldsymbol{\xi})$. As for the classical optical flow equation, we differentiate the brightness constancy with respect to time t:

$$\begin{aligned} 0 &= \partial_t (f(x(t, \boldsymbol{\xi}), t, \boldsymbol{\xi})) \\ &= \dot{x}(t, \boldsymbol{\xi}) \cdot \nabla_x f(x(t, \boldsymbol{\xi}), t, \boldsymbol{\xi}) + \partial_t f(x(t, \boldsymbol{\xi}), t, \boldsymbol{\xi}) + \dot{\boldsymbol{\xi}} \partial_{\boldsymbol{\xi}} f(x(t, \boldsymbol{\xi}), t, \boldsymbol{\xi}). \end{aligned}$$

The temporal behavior of the image sequence is modeled by the stochastic process f and the vector of random variables $\boldsymbol{\xi}$ is fixed over the whole sequence for one realization of the image sequence. This means that $\boldsymbol{\xi}$ does not change over time t (*i.e.* $\dot{\boldsymbol{\xi}} = 0$), thus we can omit the last term. So we can formulate our stochastic optical flow constraint as

$$0 = w(x, t, \boldsymbol{\xi}) \cdot \nabla_x f(x, t, \boldsymbol{\xi}) + \partial_t f(x, t, \boldsymbol{\xi}) \quad \text{a.e. } \Gamma. \tag{22}$$

This equation is completely analog to the classical optical flow equation and we observe the ill-posedness also known as the aperture problem.

5.2 Stochastic Horn and Schunck Model

The stochastic analog of the HS model regularizes the ill-posed optic flow equation by requiring the gradient of the flow field $\nabla_{xt} w(x, t, \boldsymbol{\xi})$ to be small. Here we use a regularization in space and time which usually yields better results [42]. Similar to the classical approach [16] we formulate an energy to fulfill both the stochastic optical flow equation and the smoothness constraints simultaneously. The peculiarity of the stochastic approach is that we take the $\|\!|\cdot|\!\|$ norm as it has been introduced in Sect. 2 instead of the L^2-norm. Thus, our energy involves the expectance

$$\begin{aligned} E(w) &= \|\!| w \cdot \nabla_x f + \partial_t f |\!\|^2 + \frac{1}{2} \mu \|\!| \nabla_{x,t} w |\!\|^2 \qquad (23) \\ &= \mathbb{E}\left[\int_R (w(y, \boldsymbol{\xi}) \cdot \nabla_x f(y, \boldsymbol{\xi}) + \partial_t f(y, \boldsymbol{\xi}))^2 \, dy + \frac{1}{2} \mu \int_R \nabla_{x,t} w(y, \boldsymbol{\xi}) \cdot \nabla_{x,t} w(y, \boldsymbol{\xi}) \, dy \right] \end{aligned}$$

which in fact is the expectance of the classical HS energy E_{HS} applied to stochastic image sequences, *i.e.*

$$E(w) = \mathbb{E}[E_{\mathrm{HS}}(w(y,\boldsymbol{\xi}))]. \tag{24}$$

Remark 5 *Equation* (24) *can be interpreted as follows: If we insert a stochastic optical flow field into the classical HS energy E_{HS} we obtain a stochastic process that yields the distribution of E_{HS} with respect to $\boldsymbol{\xi}$. Taking the expectance in* (24) *means that we consider the expected energy from this distribution and minimize it.*

The computation of the Euler equations as a necessary condition for a minimum of this energy is straight forward. For a function $z \in \mathcal{H}$ we compute a variation of the energy in direction ze_k for $k = 1, 2$, where $e_{1,2}$ are the Euclidean basis vectors of $\mathbb{R}^2$. *I.e.* we vary the components u and v of the flow field w by setting $k = 1$ and 2, respectively. Using the observation (24) we get

$$0 = \frac{d}{d\epsilon} E(w + \epsilon z e_k)\Big|_{\epsilon=0} = \frac{d}{d\epsilon}\mathbb{E}[E_{\mathrm{HS}}(w + \epsilon z e_k)]\Big|_{\epsilon=0} = \mathbb{E}\left[\frac{d}{d\epsilon} E_{\mathrm{HS}}(w + \epsilon z e_k)\Big|_{\epsilon=0}\right]$$

were we have assumed the energies involved to be finite, such that we can interchange differentiation and integration. The above equation means that the Euler equations for the stochastic energy (23) are just the expectance of the Euler equations of the classical energy E_{HS} and read

$$\begin{aligned}
0 &= \mathbb{E}\left[\int_R z(y,\boldsymbol{\xi})\,\partial_1 f(y,\boldsymbol{\xi})\left(w(y,\boldsymbol{\xi})\cdot\nabla_x f(y,\boldsymbol{\xi}) + \partial_t f(y,\boldsymbol{\xi})\right) dy + \mu\int_R \nabla_{x,t} u(y,\boldsymbol{\xi})\cdot\nabla_{x,t} z(y,\boldsymbol{\xi})\,dy\right],\\
0 &= \mathbb{E}\left[\int_R z(y,\boldsymbol{\xi})\,\partial_2 f(y,\boldsymbol{\xi})\left(w(y,\boldsymbol{\xi})\cdot\nabla_x f(y,\boldsymbol{\xi}) + \partial_t f(y,\boldsymbol{\xi})\right) dy + \mu\int_R \nabla_{x,t} v(y,\boldsymbol{\xi})\cdot\nabla_{x,t} z(y,\boldsymbol{\xi})\,dy\right].
\end{aligned} \tag{25}$$

Thus, the stochastic optical flow w is a solution of the SPDE-system

$$\begin{aligned}
\partial_1 f(y,\boldsymbol{\xi})(w(y,\boldsymbol{\xi})\cdot\nabla f(y,\boldsymbol{\xi}) + \partial_t f(y,\boldsymbol{\xi})) - \mu\Delta u(y,\boldsymbol{\xi}) = 0 \quad &\text{a.e. in } R\times\Gamma,\\
\partial_2 f(y,\boldsymbol{\xi})(w(y,\boldsymbol{\xi})\cdot\nabla f(y,\boldsymbol{\xi}) + \partial_t f(y,\boldsymbol{\xi})) - \mu\Delta v(y,\boldsymbol{\xi}) = 0 \quad &\text{a.e. in } R\times\Gamma
\end{aligned} \tag{26}$$

in the weak sense defined by (25). We note that this SPDE system is completely analogue to the classical system which results from a minimization of the HS Energy.

5.3 Discretization

Let us derive the linear systems of equations which result from the discretization of the Euler equations of the optical flow energies (26). In the following we consider only one equation of the system, since the derivation for the other equation is completely analogue.

Substituting the Galerkin expansions (5) for stochastic image sequences into the weak form (*cf.* (25))

$$0 = \mathbb{E}\left[\int_R z\,\partial_1 f\,(w\cdot\nabla f + \partial_t f)\,dy + \mu\int_R \nabla u\cdot\nabla z\,dy\right] =: \mathbb{E}[I] + \mu\mathbb{E}[II]$$

yields for $z = H^\beta(\xi_{j_x})P_j(y)$

$$\begin{aligned}\mathbb{E}[I] = \mathbb{E}\Bigg[& H^\beta(\xi_{j_x})P_j(y)\sum_{k\in\mathcal{J}}\sum_{\gamma=1}^{p} f_\gamma^k H^\gamma(\xi_{k_x})\partial_1 P_k(y) \times \\ & \times \begin{pmatrix}\sum_{i\in\mathcal{J}}\sum_{\alpha=1}^{p} u_\alpha^i H^\alpha(\xi_{i_x})P_i(x) \\ \sum_{i\in\mathcal{J}}\sum_{\alpha=1}^{p} v_\alpha^i H^\alpha(\xi_{i_x})P_i(x)\end{pmatrix}\cdot \sum_{l\in\mathcal{J}}\sum_{\delta=1}^{p} f_\delta^l H^\delta(\xi_{l_x})\nabla P_l(y)\Bigg] \\ & + \mathbb{E}\left[H^\beta(\xi_{j_x})P_j(y)\sum_{k\in\mathcal{J}}\sum_{\gamma=1}^{p} f_\gamma^k H^\gamma(\xi_{k_x})\partial_1 P_k(y)\sum_{l\in\mathcal{J}}\sum_{\delta=1}^{p} f_\delta^l H^\delta(\xi_{l_x})\partial_t P_l(y)\right]\end{aligned}$$

which can be collapsed to

$$\begin{aligned}\mathbb{E}[I] = & \sum_{i\in\mathcal{J}}\sum_{\alpha=1}^{p}\Bigg(\sum_{k,l\in\mathcal{J}}\sum_{\gamma,\delta=1}^{p} f_\gamma^k f_\delta^l \int_\Gamma H^\alpha(\xi_{i_x})H^\beta(\xi_{j_x})H^\gamma(\xi_{k_x})H^\delta(\xi_{l_x})\,d\mu \quad \times \\ & \times \int_R P_i(y)P_j(y)\partial_1 P_k(y)\Big(\partial_1 P_l(y)u_\alpha^i + \partial_2 P_l(y)v_\alpha^i\Big)\,dy\Bigg) \\ & + \sum_{k,l\in\mathcal{J}}\sum_{\gamma,\delta=1}^{p} f_\gamma^k f_\delta^l \int_\Gamma H^\beta(\xi_{j_x})H^\gamma(\xi_{k_x})H^\delta(\xi_{l_x})\,d\mu \int_R P_j(y)\partial_1 P_k(y)\partial_t P_l(y)\,dy.\end{aligned} \tag{27}$$

The second term becomes

$$\mathbb{E}[II] = \sum_{i\in\mathcal{J}}\sum_{\alpha=1}^{p} u_\alpha^i \int_\Gamma H^\alpha(\xi_{i_x})H^\beta(\xi_{j_x})\,d\mu \int_R \nabla P_i(y)\cdot\nabla P_j(y)\,dy. \tag{28}$$

This identity leads to the stiffness matrix $L^{\alpha,\beta}$ which we already defined for the heat equation in (20). Moreover, using the tensors A and B from Sect. 3.6.2 we can define the matrices

$$(S_{mn}^{\alpha,\beta})_{ij} = \sum_{k,l\in\mathcal{J}}\sum_{\gamma,\delta=1}^{p} f_\gamma^k f_\delta^l A_{\alpha,\beta,\gamma,\delta}^{i_x,j_x,k_x,l_x}\int_R P_i(y)P_j(y)\partial_m P_k(y)\partial_n P_l(y)\,dy, \tag{29}$$

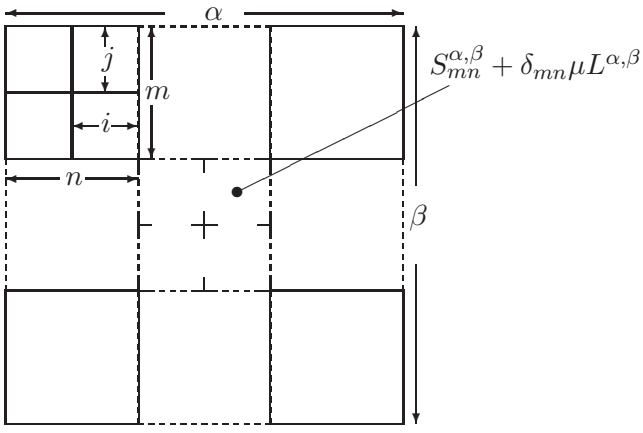
as well as the vector

$$(R_m^\beta)_j = \sum_{k,l\in\mathcal{J}}\sum_{\gamma,\delta=1}^{p} f_\gamma^k f_\delta^l B_{\beta,\gamma,\delta}^{j_x,k_x,l_x}\int_R P_j(y)\partial_m P_k(y)\partial_t P_l(y)\,dy, \tag{30}$$

for $m,n = 1,2$, $i,j = 1,\dots,N$ and $\alpha,\beta = 1,\dots,p$. Now, we can write the discretized Euler equations as

$$\sum_{\alpha=1}^{p}\left(\begin{pmatrix}S_{11}^{\alpha,\beta} & S_{12}^{\alpha,\beta} \\ S_{21}^{\alpha,\beta} & S_{22}^{\alpha,\beta}\end{pmatrix} + \mu\begin{pmatrix}L^{\alpha,\beta} & 0 \\ 0 & L^{\alpha,\beta}\end{pmatrix}\right)\begin{pmatrix}U^\alpha \\ V^\alpha\end{pmatrix} = -\begin{pmatrix}R_1^\beta \\ R_2^\beta\end{pmatrix}, \quad \text{for } \beta = 1,\dots,p. \tag{31}$$

This equation describes a block system of dimension $2pN \times 2pN$ and consists of p blocks corresponding to the stochastic modes. The blocks themselves contain 2×2 sub-blocks corresponding to the two coordinate directions. Each block is similar to the deterministic system



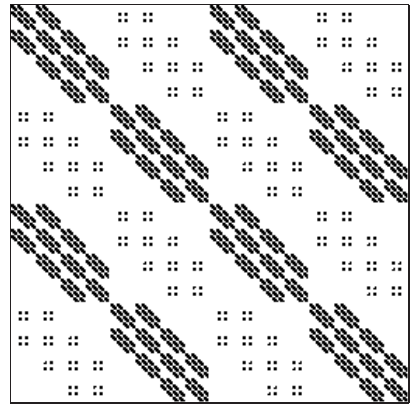


Figure 8: Left: Structure and sub-structure of the block system (31). Right: Non-zero pattern of a block matrix for two stochastic moments (i.e. $p = 2$), a spatial size of 4×4 pixels and 4 frames.

and each sub-block is a $N \times N$ matrix. In Figure 8 we show the structure of this block-system. Again it is dense in the stochastic space (*i.e.* all blocks are non-zero), since the integrals involve multiplications with more than two factors. However, the stiffness-matrix $L^{\alpha,\beta}$ (corresponding to the smoothness term) appears on the diagonals of the block-diagonals only as we have already seen for the heat equation.

5.4 Discontinuity Preserving Optical Flow Computation

Obviously it is now straight forward to generalize discontinuity preserving optical flow models [4, 9, 41] to a stochastic setting as well. This yields the SPDE system

$$\begin{aligned} \partial_1 f(y,\boldsymbol{\xi})(w(y,\boldsymbol{\xi}) \cdot \nabla f(y,\boldsymbol{\xi}) + \partial_t f(y,\boldsymbol{\xi})) - \mu \operatorname{div}(g(|\nabla f(y,\boldsymbol{\xi})|)\nabla u(y,\boldsymbol{\xi})) = 0, \\ \partial_2 f(y,\boldsymbol{\xi})(w(y,\boldsymbol{\xi}) \cdot \nabla f(y,\boldsymbol{\xi}) + \partial_t f(y,\boldsymbol{\xi})) - \mu \operatorname{div}(g(|\nabla f(y,\boldsymbol{\xi})|)\nabla v(y,\boldsymbol{\xi})) = 0, \end{aligned} \tag{32}$$

in R and a.e. Γ, which is interpreted in a sense analogue to (25). To discretize the model we only have to replace the homogeneous stiffness matrices $L^{\alpha,\beta}$ by the inhomogeneous one defined in (21). As for the HS energy we get a system of equations like (31) which has a structure as shown in Fig. 8.

5.5 Combined Local Global (CLG) Method

We now extend our discretization to a stochastic version of the combined local global (CLG) method [5, 6]. Let $J_\rho = (\nabla_{x,t} f)^T (\nabla_{x,t} f)$ denote a smoothed version of the structure tensor of the input sequence f. Denoting the homogeneous motion vector with $\bar{w} = (u, v, 1)$, the stochastic version of the CLG energy is

$$E(w) := \|\bar{w} \cdot J_\rho \bar{w}\|^2 + \frac{1}{2}\mu \|\nabla_{x,t} w\|^2.$$

As above Euler Lagrange equations are given by the expectation of the Euler-Lagrange equations of the deterministic model, thus the solution obeys

$$0 = \mathbb{E}\left[\int_R z(y,\boldsymbol{\xi})\left(J_\rho^{11}(y,\boldsymbol{\xi})u(y,\boldsymbol{\xi}) + J_\rho^{12}(y,\boldsymbol{\xi})v(y,\boldsymbol{\xi}) + J_\rho^{1t}(y,\boldsymbol{\xi})\right)dx + \mu\int_R \nabla_{x,t}u(y,\boldsymbol{\xi})\cdot\nabla_{x,t}z(y,\boldsymbol{\xi})\,d$$

$$0 = \mathbb{E}\left[\int_R z(y,\boldsymbol{\xi})\left(J_\rho^{21}(y,\boldsymbol{\xi})u(y,\boldsymbol{\xi}) + J_\rho^{22}(y,\boldsymbol{\xi})v(y,\boldsymbol{\xi}) + J_\rho^{2t}(y,\boldsymbol{\xi})\right)\right)dx + \mu\int_R \nabla_{x,t}v(y,\boldsymbol{\xi})\cdot\nabla_{x,t}z(y,\boldsymbol{\xi})$$

The discretization is completely analog to the discretization of the stochastic HS model from Sect. 5.2. The structure tensor J of the noisy image data can be computed as in Sect. 3.5. Then we consider J_ρ as the solution of the heat equation for a small single time step of length $\rho^2/2$ with initial data J. To get matrices similar to (29) let us denote the coefficients of the expansion (5) of the smoothed structure tensor with $(J_\rho^{mn})_\gamma^k$ for $m,n \in \{1,2,t\}$, $k \in \mathcal{J}$ and $\gamma = 1,\dots,p$ and define

$$(\widetilde{S}_{mn}^{\alpha,\beta})_{ij} = \sum_{k\in\mathcal{J}}\sum_{\gamma=1}^{p}(J_\rho^{mn})_\gamma^k B_{\alpha,\beta,\gamma}^{i_x,j_x,k_x}\int_R P_i(y)\,P_j(y)\,P_k(y)\,dy,$$

and a right hand side

$$(\widetilde{R}_m^\beta)_j = \sum_{k\in\mathcal{J}}\sum_{\gamma=1}^{p}(J_\rho^{m,t})_\gamma^k C_{\beta,\gamma}^{j_x,k_x}\int_R P_j(y)\,P_k(y)\,dy.$$

The block system is the same as (31) where S and R have been replaced with the ones from above.

Using an inexact quadrature rule as in Section 4.2 we can simplify the computation of the entries of $\widetilde{S}$ and $\widetilde{R}$

$$\sum_{\gamma=1}^{p}\sum_{k\in\mathcal{J}}(J_\rho^{mn})_\gamma^k\,B_{\alpha,\beta,\gamma}^{i_x,j_x,k_x}\left(\int_E P_i(x)\,P_j(x)\,P_k(x)\,dx\right) \approx (\widetilde{J}_\rho^{mn})_{\alpha,\beta}^{i,j}\int_E P_i(x)\,P_j(x)\,dx$$

where

$$(\widetilde{J}_\rho^{mn})_{\alpha,\beta}^{i,j} = \frac{1}{8}\sum_{\gamma=1}^{p}\sum_{k\in\mathcal{J}\cap E}(J_\rho^{mn})_\gamma^k\,B_{\alpha,\beta,\gamma}^{i_x,j_x,k_x}$$

and similarly for the right hand side.

5.6 Results

Let us start with the computation of the optical flow field of two test-sequences. First we consider a square which is filled with a $\sin(c_x x)\sin(c_y y)$ pattern and which moves to the right in front of a background that has a slight gradient in x-direction. For this first example we consider only two stochastic modes, thus approximating the distributions of the input gray-values with a uniform distribution. In Sect. 3.1 we have seen how the variance can be computed from the modes of the polynomial expansion. Setting the second mode $f_2(y) \equiv a$ means that we have a variance of

$$\mathrm{Var}(f) = a^2\left\langle H^2, H^2\right\rangle = a^2\int_{-1}^{1}\xi_i^2\,d\mu_i = a^2\int_{-1}^{1}\frac{\xi_i^2}{2}\,d\xi_i = \frac{1}{3}a^2$$

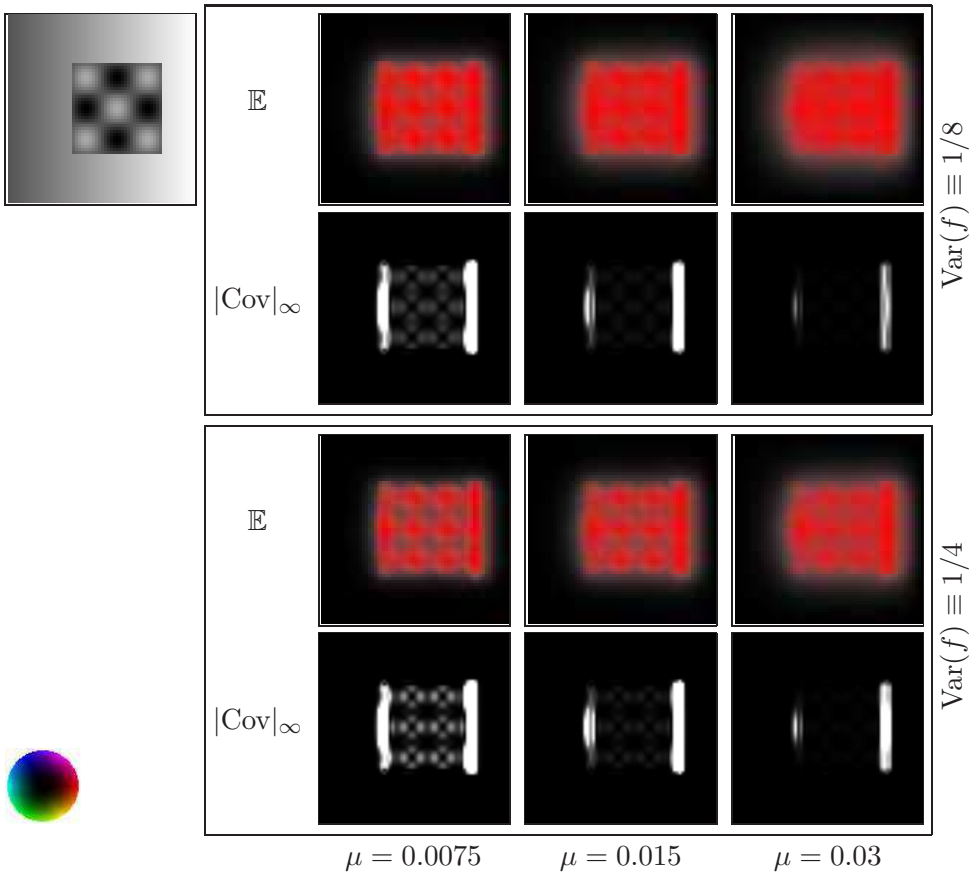


Figure 9: We consider a test-sequence of a textured square moving to the right (frame 5 shown in the top left picture). The gray values are uniformly distributed and from top to bottom we increase the variance of the input data and show the color coded optical flow field. It is clearly visible how the mean and the variance of the flow field capture the gradients of the input data. Moreover we see that with increasing smoothness of the flow field the variance decreases. The color wheel on the lower left indicates the color coding of the flow directions.

because the second Legendre basis function is $H^2(\xi_i) = \xi_i$ and the PDF is $\rho_i(\xi_i) = 1/2$. So by setting $a = \sqrt{3}\sigma$ we model an input process that has a variance of σ^2.

In Fig. 9 we show results of our computations with the stochastic HS model. The mean gray values of the image sequence are in the range $[0, 1]$. In our experiment we have considered variances of the input data f to be $1/8$ and $1/4$. Moreover we have studied the influence of the smoothing parameter μ.

We see that in general there is a high variance of the flow field in the vicinity of edges which are orthogonal to the direction of motion. This behavior can be interpreted as the uncertainty of the location of edges of the moving object.

Fixing the smoothness parameter μ, we see from the images that with increasing variance of the input data f the mean of the flow field becomes more inhomogeneous inside the moving square. The same behavior can be observed for the variance of the flow field. In fact the mean and variance capture the structure of the texture, because the model is non-linear in the derivatives of f. In Sect. 3.3 and Fig. 4 we have already seen that a nonlinear function

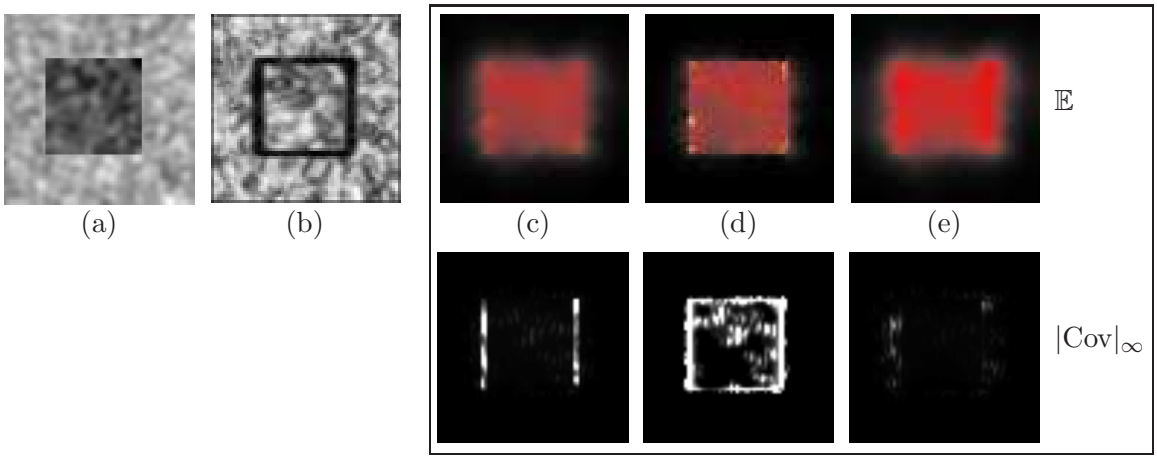


Figure 10: From a sequence showing a moving textured square (mean shown in (a), variance see text) we extract the optical flow with the HS model (c), with the discontinuity preserving model (d) and with the CLG approach (e). The mean of the edge indicator function is depicted in (b). For the flow fields we show the maximum component of the covariance matrix in the bottom row.

couples the stochastic modes, thus a constant variance in the input data leads to output which is sensitive to the gradients of the mean of the input data. For fixed variance of the input data and increasing smoothness μ the influence of the gradients in the input mean is weakened and thus the amplitude of the variance of the flow field is damped. Still there is uncertainty about the edges of the moving objects, thus the variance remains high in those regions.

In the second numerical test we consider the textured square shown in Fig. 10 which moves to the right with unit speed $w = (u, v) = (1, 0)$ in front of a textured background. The spatial resolution is of size 65×65 and the sequence has 11 frames. We consider 4 stochastic modes (*i.e.* $p = 4$) and set the image shown in Fig. 10 to be the mean. The gray values of the mean image range from 0 to 1. Furthermore we set

$$f_2(y) \equiv 2.96281 \cdot 10^{-2}, \qquad f_3(y) \equiv 0, \qquad f_4(y) \equiv 9.87604 \cdot 10^{-2},$$

such that we model stochastic processes as the one shown in the bottom row of Fig. 1.

In Fig. 10 we show the result of the optical flow computation with the HS model, with the discontinuity preserving model and the CLG approach. We have used a smoothing parameter $\mu = 0.03$, set $\lambda = 1/\sqrt{3}$ and $\rho = 0.0075$. Obviously, for the edge preserving model the covariance is high in the vicinity of the edges, whereas the HS model only yields high variances for edges which are perpendicular to the motion direction. The CLG model shows a similar behavior although the covariance is smaller since the smoothing is larger with the chosen set of parameters.

Finally, we use our framework to examine the bias of the CLG approach. The bias $\mathcal{B}$ is defined by the difference between the true optical flow components $w_0 = (u_0, v_0)$ and expectation $\mathbb{E}[\hat{w}]$ of the estimated optical flow components $\hat{w}$, *i.e.*

$$\mathcal{B} = w_0 - \mathbb{E}[\hat{w}] \ .$$

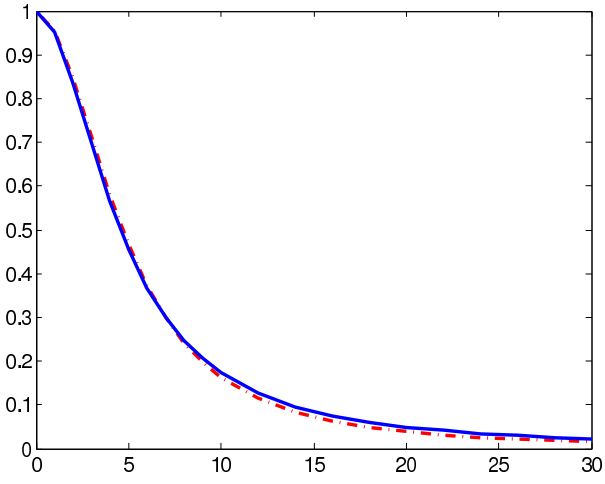


Figure 11: We plot the value of the x-component u of the extracted velocity versus the standard deviation σ of the gray values (red dotted curve) to investigate the bias of the CLG approach. Furthermore, the theoretical devolution of the bias depending on standard deviation is shown (blue solid line). Please note that the values on the horizontal axis are given in percent.

The bias occurring in optical flow estimators based on least squares estimation has been extensively examined (*cf. e.g.* [11, 17, 18]) showing an underestimation of the optical flow components u, v. For local optical flow estimators, *e.g.* of Lucas-Kanade type [21], the functional relationship between the estimated optical flow components $\hat{u}_0, \hat{v}_0$ for noise free input signals (variance $\sigma^2 = 0$) and the estimated optical flow components u, v for noisy input signals ($\sigma^2 > 0$) has been derived in [38]. This relationship can be extended to the CLG-approach which leads to the expectation value of the estimated optical flow components and the bias, respectively [20]

$$\mathbb{E}[\hat{w}] = \left(\frac{\tau_u}{\tau_u + \sigma^2} u_0, \frac{\tau_v}{\tau_v + \sigma^2} v_0 \right) \quad , \qquad \mathcal{B} = \left(\frac{\sigma^2}{\tau_u + \sigma^2} u_0, \frac{\sigma^2}{\tau_v + \sigma^2} v_0 \right) \quad ,$$

where τ_u, τ_v denote real valued parameters that depend on the input signal.

Our framework allows for an easy examination of the dependency of the bias on the variance of the input signal. In Fig. 11 we plot our estimate of the expectation $\mathbb{E}[\hat{u}]$ of the x component of the optical flow (the true underlying flow vector is $w = (u, v) = (1, 0)$) versus the standard deviation σ (in percent) of the input data. The optical flow estimate decreases for increasing noise in the input image and the bias increases. In order to demonstrate the quantitative performance of our framework we plot also the theoretical relationships for the expectation value of the estimated optical flow component. The data dependent parameter τ_u has been determined by fitting to our estimated parameter values. The estimated expectation value and the bias correspond very well to the theoretical curves.

Note that in theory we need only the expectation value and the variance of the input signal in order to estimate the expectation value of the optical flow component. In order to obtain the same result with the deterministic CLG-approach, we would have to perform a Monte-Carlo simulation for a large number of different realizations for each noise level. With our framework only one solution has to be performed per noise level. This involves the potential to analyze the bias also for estimators for which it is unknown yet.

6 Conclusions and Future Work

We have presented a model for the interpretation of images and image sequences with uncertain gray-values as random fields. The distribution of gray values for pixels is modeled by random processes for which we use an approximation according to the Wiener-Askey polynomial chaos

approach. Moreover, we have presented algorithmic building blocks for image processing based on the notion of random fields. These building blocks and the stochastic Galerkin finite element method are the key ingredients for a treatment of stochastic energies and stochastic partial differential equations in computer vision and image processing. We have guided the reader through the derivation and discretization of stochastic analogs of well known partial differential equations frequently used in computer vision. The resulting discretizational schemes are simple, since existing deterministic FEM code can be reused. Moreover the extended models reduce to the deterministic ones if all stochastic modes except the mean mode vanish.

Our framework allows for the efficient study of error propagation through computer vision models. In contrast to existing research, our approach does neither a priori assume Gaussian distributions nor does it deliver error bounds or confidence measures only. In fact the input can be approximations of arbitrary random processes and distributions which by our framework are transformed into output processes and distributions. Previously such results in general could be obtained by computationally very expensive Monte-Carlo simulations only.

We have demonstrated the usefulness of the framework with numerical experiments. For nonlinear models we have shown how the stochastic modes are coupled. Thus, variances of the output images are influenced by gradients of the mean of the input data. For optical flow equations we have shown the computation of the bias of the CLG method.

For the modeling of images as random fields we have assumed independence of the random variables. In the future we plan to combine our approach with statistic/stochastic data analysis, performing Karhunen Loewe expansions of the input data. This is going to yield the minimal set of independent random variables describing the uncertain behavior of the input data. We thus expect to improve our model of random images.

Future research directions also include a closer analysis of the bias computation and a correction for optical flow estimation, potentially leading to a higher precision of the estimation result. For optical flow estimation, most of the current estimation schemes are formulated in finite differences rather than finite elements, profiting from well adapted convolution filters. Thus making this approach compatible with finite difference discretizations is another future goal.

References

[1] T. Amiaz and N. Kiryati. Piecewise-smooth dense optical flow via level sets. *Int. J. Comput. Vision*, 68(2):111–124, 2006.

[2] Y. Bao and H. Krim. Smart nonlinear diffusion: A probabilistic approach. *Pattern Analysis and Machine Intelligence*, 26(1):63–72, January 2004.

[3] M.J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proc. Computer Vision and Pattern Recognition, CVPR-91*, pages 296–302, June 1991.

[4] M.J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Proc. ICCV93*, pages 231–236, 1993.

[5] A. Bruhn, J. Weickert, and C. Schnörr. Combining the advantages of local and global optic flow methods. In *Proc. DAGM*, pages 454–462, 2002.

[6] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods. *Int. J. Comput. Vision*, 61(3):211–231, 2005.

[7] A.J. Chorin. Hermite expansions in Monte Carlo computation. *J. Comput. Phys.*, 8:471–482, 1971.

[8] A.J. Chorin. Gaussian fields and random flow. *J. Fluid Mech.*, 63:21–32, 1974.

[9] I. Cohen. Nonlinear variational method for optical flow computation. In *SCIA93*, pages 523–530, 1993.

[10] M.K. Deb, I.M. Babuška, and J.T. Oden. Solutions of stochastic partial differential equations using Galerkin finite element techniques. *Comput. Methods Appl. Mech. Eng.*, 190:6359–6372, 2001.

[11] C. Fermüller, D. Shulman, and Y. Aloimonos. The statistics of optical flow. *Comput. Vis. Image Underst.*, 82(1):1–32, 2001.

[12] D.A. Forsyth and J. Ponce. *Computer Vision A Modern Approach*. Prentics Hall, 2003.

[13] R.G. Ghanem. Higher order sensitivity of heat conduction problems to random data using the spectral stochastic finite element method. *ASME J. Heat Transfer*, 121:290–299, 1999.

[14] R.G. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Springer-Verlag, New York, NY, 1991.

[15] H. Haussecker, H. Spies, and B. Jähne. Tensor-based image sequence processing techniques for the study of dynamical processes. In *Proceedings of the International Symposium on Real-Time Imaging and Dynamic Analysis, ISPRS, Commission V, Working Group IC V/III*, Hakodate, Japan, June 1998.

[16] B.K.P. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.

[17] S. Van Huffel and J. Vandewalle. The total least squares problem: Computational aspects and analysis. *Frontiers in Applied Mathematics*, 9, 1991.

[18] J.K. Kearney, W.B. Thompson, and D.L. Boley. Optical flow estimation: An error analysis of gradient-based methods with local optimization. *PAMI*, 9(2):229–244, March 1987.

[19] A. Keese. A review of recent developments in the numerical solution of stochastic partial differential equations (stochastic finite elements). Technical Report 2003-06, Technical University Braunschweig, 2003.

[20] K. Krajsek, H. Scharr, R. M. Kirby, and T. Preusser. An approach for analysis and correction of bias in statistical estimators. *Technical Report*, 2007.

[21] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *DARPA Image Understanding Workshop*, pages 121–130, 1981.

[22] D. Lucor, C.-H. Su, and G.E. Karniadakis. Generalized polynomial chaos and random oscillators. *International Journal for Numerical Methods in Engineering*, 60:571–596, 2004.

[23] O.P. Le Maître, M. . Reagan, H.N. Najm, R.G. Ghanem, and O.M. Knio. A stochastic projection method for fluid flow II.: random process. *Journal of Computational Physics*, 181(1):9–44, 2002.

[24] P. Malliavin. *Stochastic Analysis*. Springer, 1997.

[25] F.H. Maltz and D.L. Hitzl. Variance reduction in Monte Carlo computations using multi-dimensional Hermite polynomials. *J. Comput. Phys.*, 32:345–376, 1979.

[26] W.C. Meecham and D.T. Jeng. Use of Wiener-Hermite expansion for nearly normal turbulence. *J. Fluid Mech.*, (32):225–249, 1968.

[27] K. Mikula, T. Preusser, and M. Rumpf. Morphological image sequence processing. *Computing and Visualization in Science*, 6(4):197–209, 2004.

[28] V.A. Narayanan and N. Zabaras. Stochastic inverse heat conduction using a spectral approach. *Int. J. Numer. Meth. Engng.*, 60:1569–1593, 2004.

[29] O. Nestares and D.J. Fleet. Error-in-variables likelihood functions for motion estimation. In *IEEE Intenational Conference on Image Processing (ICIP)*, volume III, pages 77–80, Barcelona, 2003.

[30] O. Nestares, D.J. Fleet, and D. Heeger. Likelihood functions and confidence bounds for total-least-squares problems. In *CVPR'00*, volume 1, 2000.

[31] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *Int. J. Comput. Vision*, 67(2):141–158, 2006.

[32] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12:629–639, 1990.

[33] T. Preusser and M. Rumpf. An adaptive finite element method for large scale image processing. In *Proceedings Scale-Space '99, Scale Space Theories in Computer Vision, Second International Conference*, pages 223–234, 1999.

[34] M. T. Reagan, H.N. Najm, P.P. Pebay, O.M. Knio, and R.G. Ghanem. Quantifying uncertainty in chemical systems modeling. *International Journal of Chemical Kinetics*, 37:386–382, 2005.

[35] M.T. Reagan, H.N. Najm, B.J. Debusschere, O.P. Le Maître, O.M. Knio, and R.G. Ghanem. Spectral stochastic uncertainty quantification in chemical systems. *Combustion Theory and Modelling*, 8:607–632, 2004.

[36] H. Scharr. Diffusion-like reconstruction schemes from linear data models. In *Pattern Recognition 2006*, Lecture Notes in Computer Science 4174, pages 51–60, Berlin, 2006. Springer Verlag.

[37] H. Scharr, M.J. Black, and H.W. Haussecker. Image statistics and anisotropic diffusion. In *Int. Conf. on Computer Vision, ICCV 2003*, pages 840–847, Nice, France, 2003.

[38] M. Sühling. *Myocardial Motion and Deformation Analysis from Echocardiograms*. PhD thesis, Swiss Federal Institute of Technology Lausanne (EPFL), July 2006.

[39] V. Thomee. *Galerkin - Finite Element Methods for Parabolic Problems*. Springer, 1984.

[40] J. Weber and J. Malik. Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14(1):5–19, 1994.

[41] J. Weickert. On discontinuity-preserving optic flow. In *Proc. Computer Vision and Mobile Robotics Workshop*, pages 115–122, 1998.

[42] J. Weickert and C. Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3):245–255, 2001.

[43] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938.

[44] A.P. Witkin. Scale-space filtering. In *Proc. Eighth Int. Joint Conf. on Artificial Intelligence (IJCAI)*, volume 2, pages 1019–1022, 1983.

[45] D.B. Xiu and G.E. Karniadakis. Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos. *Comput. Methods Appl. Mech. Eng.*, 191:4927–4948, 2002.

[46] D.B. Xiu and G.E. Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24:619–644, 2002.

[47] D.B. Xiu and G.E. Karniadakis. Modeling uncertainty in flow simulations via generalized polynomial chaos. *J. Comput. Phys.*, 187:137–167, 2003.

[48] D.B. Xiu and G.E. Karniadakis. A new stochastic approach to transient heat conduction modeling with uncertainty. *International Journal of Heat and Mass Transfer*, 46:4681–4693, 2003.