

Building Blocks for Computer Vision with Stochastic Partial Differential Equations

Tobias Preusser · Hanno Scharr · Kai Krajsek · Robert M. Kirby

Received: 1 September 2007 / Accepted: 12 May 2008 / Published online: 10 July 2008
© Springer Science+Business Media, LLC 2008

Abstract We discuss the basic concepts of computer vision with stochastic partial differential equations (SPDEs). In typical approaches based on partial differential equations (PDEs), the end result in the best case is usually one value per pixel, the “expected” value. Error estimates or even full probability density functions PDFs are usually not available. This paper provides a framework allowing one to derive such PDFs, rendering computer vision approaches into measurements fulfilling scientific standards due to full error propagation. We identify the image data with random fields in order to model images and image sequences which carry uncertainty in their gray values, e.g. due to noise in the acquisition process. The noisy behaviors of gray values is modeled as 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 approxi-

mation based upon piecewise multi-linear finite elements. We present the 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 PDEs for denoising and optical flow extraction. These models are discretized with the stochastic Galerkin method. Our selection of SPDE models allows us 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

T. Preusser (✉)
Center of Complex Systems and Visualization, Bremen
University, Bremen, Germany
e-mail: tp@mevis.de

H. Scharr · K. Krajsek
Institute for Chemistry and Dynamics of the Geosphere,
Institute 3: Phytosphere, Forschungszentrum Juelich GmbH,
Juelich, Germany

H. Scharr
e-mail: h.scharr@fz-juelich.de

K. Krajsek
e-mail: k.krajsek@fz-juelich.de

R.M. Kirby
School of Computing and Scientific Computing and Imaging
Institute, University of Utah, Salt Lake City, UT, USA
e-mail: kirby@cs.utah.edu

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. Of course it is good scientific practice that proper measurements must be equipped with error estimates (Gauss 1987; de Laplace 1812). Thus, for many applications not only the measured values, but also their errors should be—and more and more are—taken 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. Unfortunately, for realistic models the computation of error propagation is sometimes difficult or cumbersome, and

therefore most contributions dealing with error estimates are restricted in one or more of the following:

- Data is presumed to be Gaussian distributed—the error is then represented by a variance (cf. e.g. Nestares and Fleet 2003; Weber and Malik 1994);
- Bounds on the error are derived only (cf. e.g. Nestares et al. 2000; Nestares and Fleet 2003);
- No true error estimates, confidence measures only are derived (cf. e.g. Bruhn et al. 2005; Haussecker et al. 1998).

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 single values only 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 losing information on the precision due to approximated error bounds.

We look at 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 PDEs with the stochastic setting by replacing classical image functions by the stochastic analog. Special care, however, 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 calculation 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 (Perona and Malik 1990), variational optical flow estimation

by Horn-Schunck (1981), a version with a robust smoothing term (Black and Anandan 1991; Cohen 1993; Weickert 1998) and optical flow computation with a regularized data term (Bruhn et al. 2005). 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. Haussecker and Spies 1999; Papenberg et al. 2006; Amiaz and Kiryati 2006 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 didactical reasons and as demonstrators of the formalism. This enables the reader to draw conclusions and connections to deterministic image processing as well as stochastic image processing not based on PDEs.

Although we present our stochastic framework in combination with a classic finite element approach it is possible to combine the stochastic Galerkin method with finite difference schemes as well. In the present work we have chosen the former approach since there exists a wide variety of functional analytic tools for finite element methods.

Benefit of the novel approach Results of the above mentioned algorithms for the mean (or more precisely expected value) calculated with the new formalism do not significantly differ from results a standard finite element algorithm would give. In fact, the new formalism often boils down to a standard deterministic implementation when we model the input distribution only by its mean. 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.* Under some assumptions concerning the smoothness of the output process, the calculated output distributions converge to the same distributions one would get when running *infinitely many Monte-Carlo (MC) simulations* of the applied algorithm and projecting the resulting probability density function into our stochastic ansatz space. Thus, our approach outperforms MC in terms of accuracy and computational efficiency, as the full knowledge about the input distribution is used to calculate the output distribution, not only (few) samples from the input distribution.

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 denoising, 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 (Wiener 1938), the stochastic Galerkin method has been applied to a range of problems in computational mechanics (Meecham and Jeng 1968; Chorin 1971; Chorin 1974; Maltz and Hitzl 1979; Deb et al. 2001; Le Maître et al. 2002; Xiu and Karniadakis 2002; Xiu and Karniadakis 2003; Lucor et al. 2004). This technique has also recently been introduced into other disciplines such as thermodynamics (Ghanem 1999; Narayanan and Zabarar 2004; Xiu and Karniadakis 2003) and physical chemistry (Reagan et al. 2004; Reagan et al. 2005), 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) (Bao and Krim 2004) or Markov-Random-Field assumptions (Scharf et al. 2003). However our framework yields general extensions to the stochastic interpretation of energy functionals presented in Scharf (2006).

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 multilinear 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 Sect. 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 and Malik 1990) model, the optical flow extraction with the Horn and Schunck approach (Horn and Schunck 1981), a robust smoothing term for the optical flow field (Black and Anandan 1993; Cohen 1993; Weickert 1998), and finally a combined local global (CLG) approach (Bruhn et al. 2005). 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. In the Appendix we summarize all building blocks and stochastic PDE models in order to support the understanding of the material, the reimplementation of the models, and the reproduction of our results.

2 Stochastic Images as Random Fields

In the following sections (see Sects. 2.1–2.3) we give the exact mathematical definition of our ansatz space.

The core idea we put forward is that, in the presence of noise, gray values in each pixel are samples from per pixel distributions. We model these per-pixel distributions or, to be more precise, their Probability Density Functions (PDF) in contrast to standard image processing approaches where only the expected value or single samples from the distributions are modeled. Roughly speaking, each pixel stores a representation of the random field at that point (with corresponding PDF) instead of a single value, i.e. 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. The stochastic finite element space we introduce in the sections below is continuous in space as is typical in FEM approaches. It is not only continuous in the spatial domain, where it consists of a standard bi-linear interpolation scheme, represented by compact interpolation functions P_i , where i indicates pixel position (cf. Fig. 2), but also the stochastic domain. In the stochastic domain per-pixel random variables ξ_i with uniform distribution are applied. The uniform distributions of ξ_i are transformed into proper, continuous PDFs using process functions. These process functions are approximated as weighted sums of polynomial, orthogonal basis functions H^α , where α indicates the degree of the polynomial. The weights f_α^i describing a function in the stochastic finite element space are denoted *modes*, e.g. the first mode of a gray value image u is an image containing the expected gray values, the second mode is an image containing weights belonging to H^2 —using a special selection of H^2 , it is proportional to the standard deviation, but we also use other selections—etc.

2.1 Random Fields and Wiener-Askey Polynomial Chaos

First we 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 Keese (2004).

Let $(\Omega, \mathcal{A}, \mu)$ be a complete probability space, where Ω is the event space, $\mathcal{A} \subset 2^\Omega$ the σ -algebra, and μ the probability measure. Following (Xiu and Karniadakis 2002), we can represent any general second-order *random process* $X(\omega)$, $\omega \in \Omega$ in terms of a collection of random variables $\xi = (\xi_1, \dots, \xi_N)$ with independent components. Let $\rho_i : \Gamma_i \rightarrow \mathbb{R}^+$ be the PDFs of the random variables $\xi_i(\omega)$, $\omega \in \Omega$, and its image $\Gamma_i \equiv \xi_i(\Omega) \in \mathbb{R}$ be intervals in \mathbb{R} for

$i = 1, \dots, N$. Then

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

is the joint probability density of the random vector ξ with the support

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

As commented in Xiu and Karniadakis (2002), 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 (Malliavin 1997) in which the random processes based upon the random variables ξ reside. The inner product (scalar-product) of this Hilbert space is given by:

$$\langle a, b \rangle = \int_{\Omega} (a \cdot b) d\mu = \int_{\Gamma} (a \cdot b) \rho(\xi) d\xi$$

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:

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

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

$$f : D \times \Gamma \rightarrow \mathbb{R}.$$

Such a set of processes is referred to as a *random field* (Keese 2004) which can also be interpreted as a function-valued random variable, because for every event $\omega \in \Omega$ the *realization* $f(\cdot, \xi(\omega)) : D \rightarrow \mathbb{R}$ 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. $\xi \in \Gamma$. Throughout this paper we will use random fields $f \in L^2(D) \otimes L^2(\Gamma)$ such that $f(\cdot, \xi) \in L^2(D)$ for almost all $\xi \in \Gamma$. In this case let us define the norm $\|\cdot\|$ as

$$\begin{aligned} \|f(x, \xi)\|^2 &= \mathbb{E}[\|f(x, \xi)\|_{L^2(D)}^2] \\ &= \int_{\Gamma} \int_D (f(x, \xi))^2 dx \rho(\xi) d\xi, \end{aligned}$$

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

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

with stochastic ansatz functions $H^{\alpha}(\xi)$. Those can be chosen according to the Wiener-Askey Polynomial Chaos Approach (Xiu and Karniadakis 2002) (generalized Polynomial Chaos, gPC) in which the functions $H^{\alpha}(\xi)$ are orthogonal polynomials ranging up to p th order (that is, a polynomial of degree $p - 1$) 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 what is to follow we will denote the coefficients f_{α}^i as *modes* of the stochastic process $f^i(\cdot)$. Furthermore we denote the space spanned by the polynomials with

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

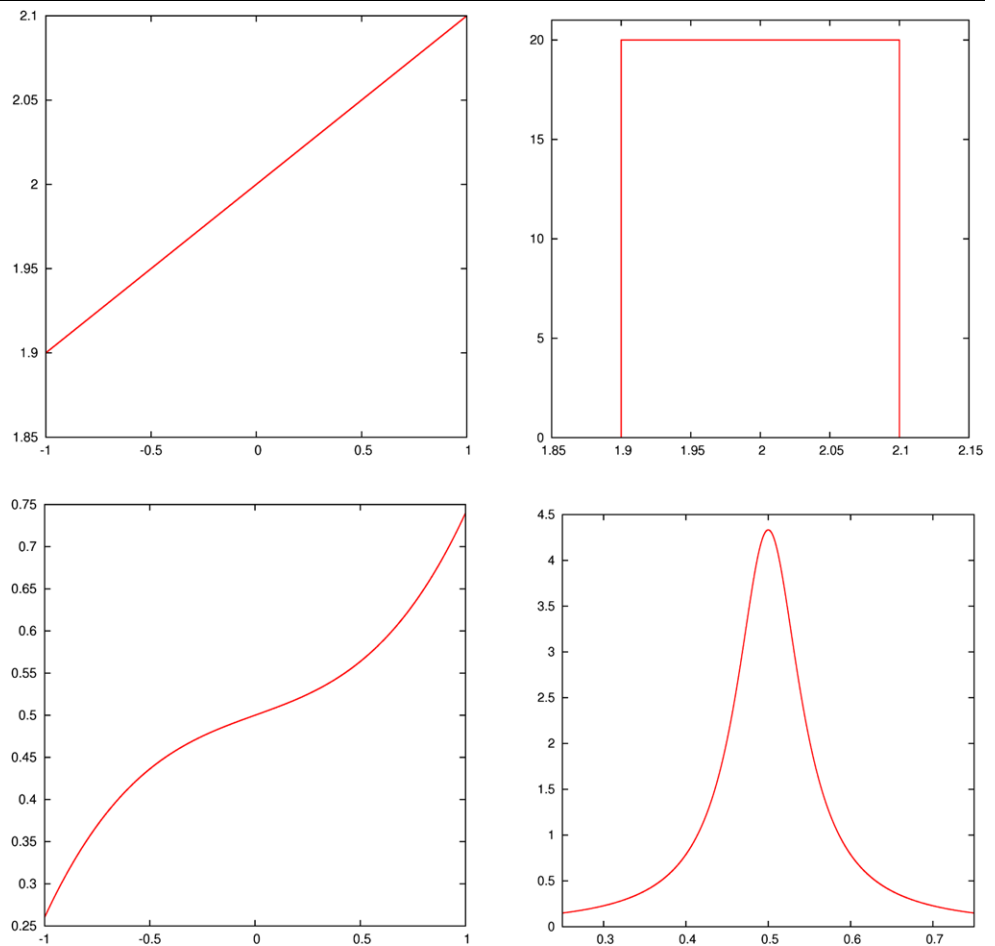
Convergence rates of the system depend on the choice of orthogonal polynomials for underlying probability density functions of a random model parameter. Each probability distribution has a corresponding optimal set of orthogonal polynomials (Xiu and Karniadakis 2002); e.g., for Gaussian random functions, 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 the experiments shown throughout the paper we have selected Legendre polynomials to span the polynomial space \mathcal{P}_p . 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 distributions ranging from uniform to Gaussian shaped processes can be approximated with Legendre polynomials.

Fig. 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



2.2 Stochastic Still Images

Let us assume we are examining still images. An acquisition process of such digital images yields noisy images due to various technical and physical reasons (cf. Forsyth and Ponce 2003, 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 $\{1, \dots, N_1\} \times \{1, \dots, N_2\}$. So the image has $N := N_1 N_2$ 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 (Preusser and Rumpf 1999). This means, we consider the finite-

dimensional space

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

on the domain $D = [1, N_1] \times [1, N_2] \subset \mathbb{R}^2$. Above, 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 at x_i and vanish at every other vertex (see Fig. 2 for their support). 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 $(f^1, \dots, f^N) \in \mathbb{R}^N$.

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

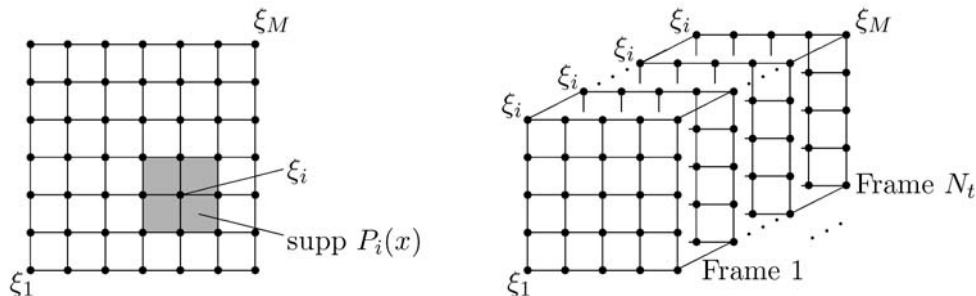


Fig. 2 We model the random behavior of the detector elements by random variables ξ_i each having a spatial extent which is given by finite element shape functions P_i . *Left:* For 2D images or still 3D images the behavior of each pixel is modeled by a separate (independent) random

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

that these random variables are independent and that they are all supported on the same domain $\Gamma_i = \Gamma_* \subset \mathbb{R}$. 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 $\xi = (\xi_1, \dots, \xi_N)$ is a vector of random variables.

As described in the previous section, the stochastic Galerkin method represents any stochastic process, $X(\xi)$, by a weighted sum of orthogonal polynomials (Ghanem and Spanos 1991; Xiu and Karniadakis 2002; Xiu and Karniadakis 2002; Xiu and Karniadakis 2003). These polynomials are functions of a vector of independent random variables, $\xi(\omega)$, $\omega \in \Omega$, of known distribution. In the case of this study, the random processes of interest are the (stochastic) gray values attributed to each pixel. The random variables 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

$$f(x, \xi) = \sum_{i \in \mathcal{I}} f^i(\xi_i) P_i(x). \tag{3}$$

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

Combining (1) and (3), we construct a finite-dimensional space $\mathcal{H}_{\text{still}}^{h,p} := V_2^h \otimes \mathcal{P}_p$ containing discrete random fields

$$f(x, \xi) = \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p f_\alpha^i H^\alpha(\xi_i) P_i(x). \tag{4}$$

With notational abuse we can also write $f(x, \xi) = \sum_{\alpha=1}^p f_\alpha(x) H^\alpha(\xi)$ where $f_\alpha(x) = \sum_{i \in \mathcal{I}} f_\alpha^i P_i(x)$. 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

$$F := (F_1; \dots; F_p) \\ := (f_1^1, \dots, f_1^N; \dots; f_p^1, \dots, f_p^N) \in \mathbb{R}^{pN}. \tag{5}$$

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 typical for FEM methods. Moreover, setting $p = 1$ we reduce to the classical finite element discretization and $\mathcal{H}_{\text{still}}^{h,p} \equiv V_2^h$.

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 $\{1, \dots, N_1\} \times \{1, \dots, N_2\}$. 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 $y_i = y_{(i_x, i_t)}$ 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 (Mikula et al. 2004). 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

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

on the spatio-temporal domain $D \times I = [1, N_1] \times [1, N_2] \times [1, N_t] \subset \mathbb{R}^3$. In the following we will also use the notation $R := D \times I$. In the definition, $\delta_{ij} = \delta_{i_x j_x} \delta_{i_t j_t}$ 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 ξ_i only (cf. Fig. 2). This means that the random variables ξ_i are time independent. However, it does *not* mean that PDFs described with them are time independent. Temporal changes of the PDFs are modeled by temporal changes of the stochastic modes f_α^i . In fact, the acquisition of gray values for pixels y_i and y_j of the image sequence is modeled by the same time-dependent stochastic process if $i_x = j_x$. Consequently we have to modify the expansion (4) such that it involves $H^\alpha(\xi_{i_x})$ instead of $H^\alpha(\xi_i)$. Our ansatz space $\mathcal{H}_{\text{seq}}^{h,p} := V_3^h \otimes \mathcal{P}_p \subset L^2(D \times I) \otimes L^2(\Gamma)$ has dimension pN and the discrete random fields have the expansion

$$f(y, \xi) = \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p f_\alpha^i H^\alpha(\xi_{i_x}) P_i(y) = \sum_{\alpha=1}^p f_\alpha H^\alpha(\xi). \quad (6)$$

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 $f(y, \xi) = \sum_{\alpha=1}^p f_\alpha(y) H^\alpha(\xi)$. Let us finally note that we have again constructed a proper finite dimensional sub-space $\mathcal{H}_{\text{seq}}^{h,p} \subset H^1(D \times I) \otimes L^2(\Gamma)$.

Remark 2 The discretization defined in (6) indeed yields time-dependent processes as can be seen as follows: 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 $\text{span } Q_j^2$. Considering an image sequence the stochastic process must be time-dependent, i.e. the image has the expansion

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

Now we discretize the stochastic processes by gPC and piecewise linear expressions 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

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

which is the same as (6) if we set $P_i(y) = Q_j^2(x) Q_k^1(t)$ 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 tent 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

In the section above we introduced finite dimensional ansatz spaces that model stochastic still images ($\mathcal{H}_{\text{still}}^{h,p}$) as well as stochastic image sequences ($\mathcal{H}_{\text{seq}}^{h,p}$). We now present some tools that are needed for image processing tasks with those ansatz spaces. 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, Legendre, 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 $\langle H^\alpha, H^\beta \rangle = 0$.

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 (4) we can compute the mean of the random field f as

$$\begin{aligned} \mathbb{E}[f](x) &= \int_{\Gamma} f(x, \xi) \rho(\xi) d\xi \\ &= \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p f_{\alpha}^i P_i(x) \int_{\Gamma} H^{\alpha}(\xi_i) \rho(\xi) d\xi \\ &= \sum_{i \in \mathcal{I}} f_1^i P_i(x) \end{aligned} \tag{8}$$

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

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

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

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 to one we have $\langle H^1, H^1 \rangle = \mathbb{E}[H^1] = 1$ and thus

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

This expression is not an element of the physical finite element space V_2^h any longer as we have the square of our linear

finite element basis functions (recall that our finite element space consists of only affine functions). But we can use a standard interpolation (or nodal evaluation) to represent the term in V_2^h . This leads us to

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

We see that the square at $P_i(x)$ vanished due to this approximation.

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 $\text{Cov}[f, g](x) = \mathbb{E}[(f - \mathbb{E}[f])(g - \mathbb{E}[g])]$ we get

$$\begin{aligned} \text{Cov}[f, g](x) &= \int_{\Gamma} \sum_{i,j \in \mathcal{I}} \sum_{\alpha,\beta=2}^p f_{\alpha}^i g_{\beta}^j H^{\alpha}(\xi_i) H^{\beta}(\xi_j) \\ &\quad \times P_i(x) P_j(x) \rho(\xi) d\xi. \end{aligned}$$

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

$$\text{Cov}[f, g](x) = \sum_{i \in \mathcal{I}} \sum_{\alpha=2}^p f_{\alpha}^i g_{\alpha}^i \langle H^{\alpha}, H^{\alpha} \rangle P_i(x). \tag{10}$$

Here again a term $P_i(x)$ vanished due to nodal evaluation. As expected we get $\text{Cov}[f, f] = \text{Var}[f]$ 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 an evaluation with stochastic images. So let us consider a function $g : \mathcal{H}_{\text{still}}^{h,p} \rightarrow L^2(R) \otimes L^2(\Gamma)$ of a discrete stochastic image f having an expansion as in (4). Examples for g are gradient magnitudes and edge-indicator functions like

$$g(u) = \nabla u \cdot \nabla u, \quad \text{and} \quad g(u) = (1 + |\nabla u|^2 / \lambda^2)^{-1}. \tag{11}$$

Both functions are well known and often needed in image processing, e.g. for the Perona-Malik diffusion (Perona and Malik 1990).

In general the result of the application of g on u does not lie in $\mathcal{H}_{\text{still}}^{h,p}$ any longer. 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 (4), we compute a L^2 -projection $g_{h,p}(x, \xi)$ of $g(u)$ onto $\mathcal{H}_{\text{still}}^{h,p}$. The projection is defined by the orthogonality relation

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

for $\beta = 1, \dots, p$ and for all $j \in \mathcal{I}$. Substituting the expansion (4) of $g_{h,p}$ into this orthogonality yields

$$\sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p \langle H^\alpha(\xi_i), H^\beta(\xi_j) \rangle \int_D g_\alpha^i P_i(x) P_j(x) dx = \mathbb{E} \left[\int_D g(u)(x, \xi) H^\beta(\xi_j) P_j(x) dx \right] \quad (12)$$

for $\beta = 1, \dots, p$ and for all $j \in \mathcal{I}$. This is a linear system of equations for the coefficients $(g_\alpha^i)_{i,\alpha}$ of the projection $g_{h,p}$. Denoting the vector of coefficients with $G = (g_\alpha^i)_{i,\alpha}$ the system can be written as

$$MG = R \quad \text{with} \quad R = \left(\mathbb{E} \left[\int_D g(u)(x, \xi) H^\beta(\xi_j) P_j(x) dx \right] \right)_{j,\beta} \quad (13)$$

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

$$(M^{\alpha,\beta})_{ij} = \langle H^\alpha(\xi_i), H^\beta(\xi_j) \rangle \int_D P_i(x) P_j(x) dx \quad (14)$$

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

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 (Thomee 1984) can simplify the effort enormously, since it diagonalizes the stochastic mass matrix. Lumping of masses yields a block diagonal mass matrix \tilde{M} such that

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

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 (13). However, if a direct expansion of g as a product of expansions like (4) is available, we can proceed differently as we shall see in the next paragraph.

3.3 Gradient Magnitude

Let us use the projection discussed above 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 the system of (13). Directly using mass lumping leads us to

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

Using the basis representation (4) of ∇u we can write $\nabla u \cdot \nabla u$ as

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

If we order the DOF of u in the vector U as in (5) we can derive from this the block-system

$$g_\alpha^i = \tilde{M}_{(i,\alpha),(i,\alpha)}^{-1} U \cdot K_{(i,\alpha)} U, \quad (15)$$

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

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

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. Conversely, we also observe that the variance captures the gradients of the mean input image. The spatial resolution of the test image is 257×257 , the gray values range in the interval $[0, 1]$, and we have used a value of $\lambda = 0.01$ (see (11)).

Fig. 3 We show the stochastic modes for the test evaluation of the gradient magnitude and the edge-indicator (cf. Fig. 4) on a test image. One realization of the distribution modeled in this example is shown in the *top left* image. On the *right* the stochastic modes used in the expansion (4) are depicted. According to (8) the mean corresponds to the first mode of the expansion. The variance must be computed from the remaining modes according to (9). We have used a color coding as shown in the color ramp (*bottom left*) to make a differentiation between positive and negative values possible. Note that we have scaled the images to match the full color range. Thus, the colors give a qualitative impression only

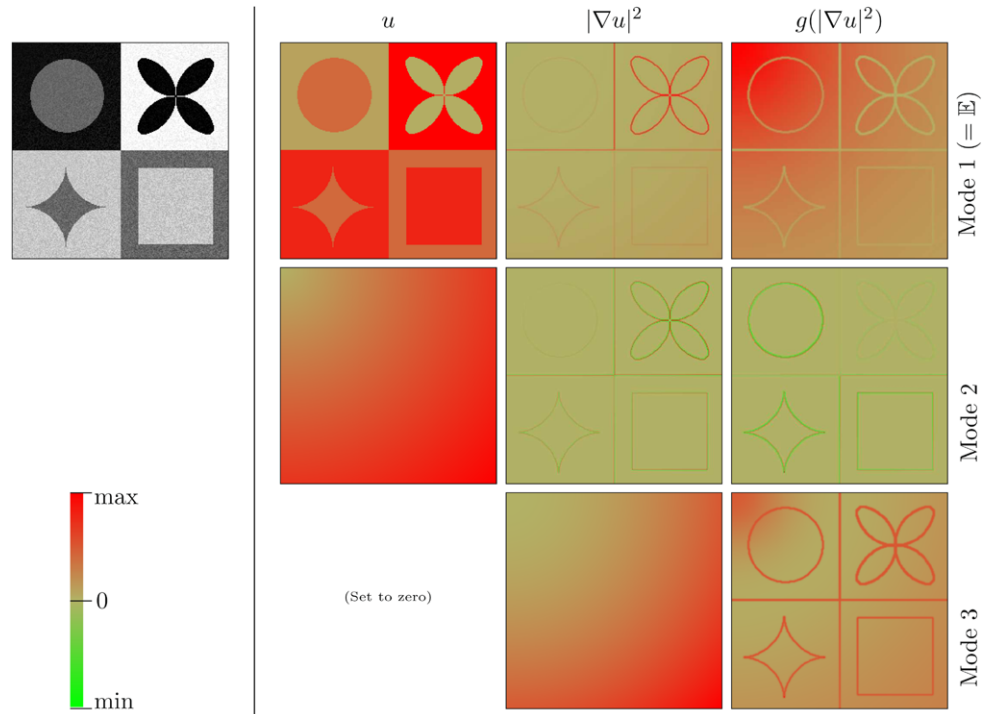
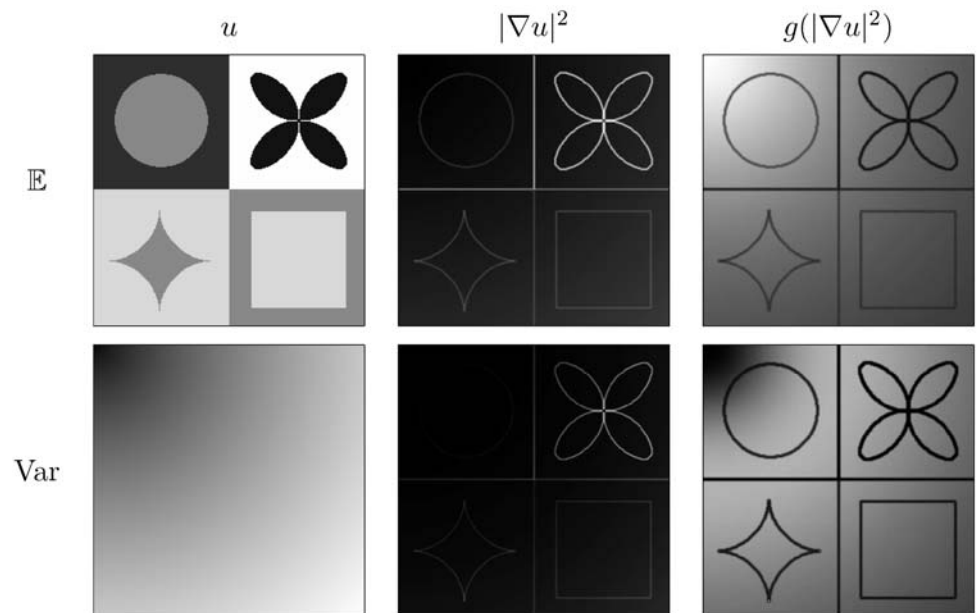


Fig. 4 For a sample image (*left*) we show the gradient magnitude (*middle*) and an edge-indicator (*right*). The expectations are shown in the *top row*, whereas the variances are shown in the *bottom row*. Note that we show the gradient images with a contrast enhancement to better visualize their global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



3.4 Edge-Indicator Function

The computation of the gradient magnitude was simplified by taking mass lumping into account. Furthermore it benefited 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.

$$g(v) = (1 + v/\lambda^2)^{-1}, \tag{16}$$

where v is the representation of $|\nabla u|^2$ whose coefficients have been derived in (15). As above, we insert this function into the right hand side R (cf. (13)) and get for $G = M^{-1}R$

$$\begin{aligned} g_\alpha^i &= (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} \mathbb{E} \left[\int_D g(v)(x, \xi) H^\alpha(\xi_i) P_i(x) dx \right] \\ &= (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} \int_\Gamma \int_D g(v)(x, \xi) H^\alpha(\xi_i) P_i(x) dx \rho(\xi) d\xi. \end{aligned}$$

To be more specific let us substitute the actual edge-indicator function $g(v) = (1 + v/\lambda^2)^{-1}$ and the expansion (4) for v . This leads us to the identity

$$g_{\alpha}^i = (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} \int_{\Gamma} \int_D \frac{H^{\alpha}(\xi_i) P_i(x)}{1 + \lambda^{-2} \sum_{j \in \mathcal{I}} \sum_{\beta=1}^p V_{\beta}^j H^{\beta}(\xi_j) P_j(x)} dx \rho(\xi) d\xi \tag{17}$$

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 $p = 3$. It is again 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

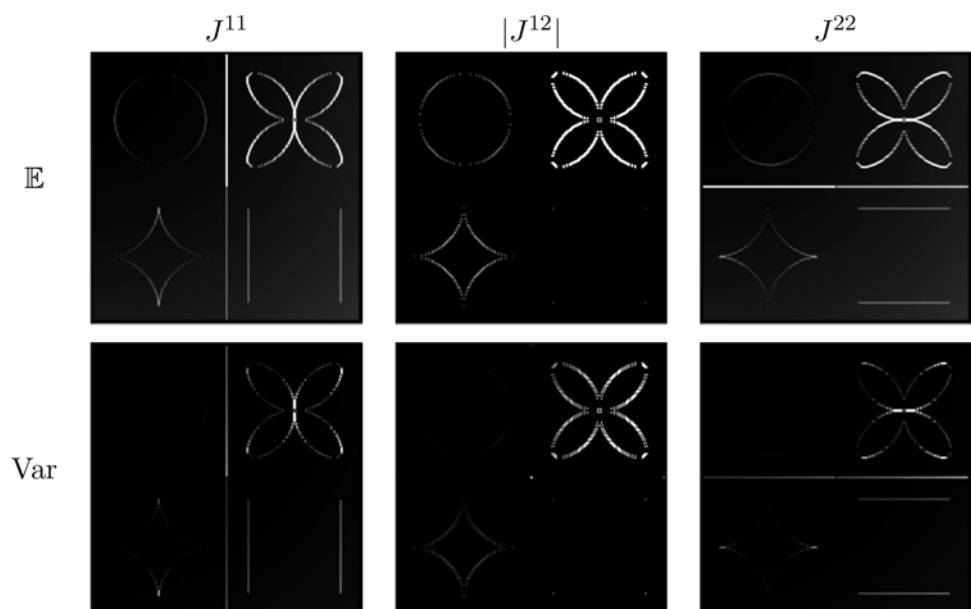
section we describe how to evaluate these high-dimensional integrals and how to use quadrature rules to compute coefficients as in (17). Although Γ is an M -dimensional space, where M is the number of random variables that model the stochastic behavior of the image or image sequence, the computation 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 (14). 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 an 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

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



expectations again. So let us focus on the integral

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

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 reference-element (as is standard in finite element methods (Thomee 1984)) 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.

For the later use we define the tensors

$$\begin{aligned} A_{\alpha,\beta,\gamma,\delta}^{i,j,k,l} &= \begin{cases} \int_{\Gamma_1} H^\alpha(\xi_1)H^\beta(\xi_1)H^\gamma(\xi_1)H^\delta(\xi_1)\rho(\xi_1)d\xi_1, & i = j = k = l, \\ \int_{\Gamma_1} \int_{\Gamma_2} H^\alpha(\xi_1)H^\beta(\xi_1)H^\gamma(\xi_1)H^\delta(\xi_2)\rho(\xi_1)d\xi_1\rho(\xi_2)d\xi_2, & i = j = k \neq l, \\ \vdots & \text{permutations of three} \\ & \text{equal latin indices,} \\ \int_{\Gamma_1} \int_{\Gamma_2} \int_{\Gamma_3} H^\alpha(\xi_1)H^\beta(\xi_2)H^\gamma(\xi_3)H^\delta(\xi_3)\rho(\xi_1)d\xi_1 \dots \rho(\xi_3)d\xi_3, & i, j \neq k = l, i \neq j, \\ \vdots & \text{permutations of two} \\ & \text{equal latin indices,} \\ \int_{\Gamma_1} \int_{\Gamma_2} \int_{\Gamma_3} \int_{\Gamma_4} H^\alpha(\xi_1)H^\beta(\xi_2)H^\gamma(\xi_3)H^\delta(\xi_4)\rho(\xi_1)d\xi_1 \dots \rho(\xi_4)d\xi_4 & \text{all latin indices} \\ & \text{are different,} \end{cases} \\ B_{\alpha,\beta,\gamma}^{i,j,k} &= \begin{cases} \int_{\Gamma_1} H^\alpha(\xi_1)H^\beta(\xi_1)H^\gamma(\xi_1)\rho(\xi_1)d\xi_1, & i = j = k, \\ \int_{\Gamma_1} \int_{\Gamma_2} H^\alpha(\xi_1)H^\beta(\xi_1)H^\gamma(\xi_2)\rho(\xi_1)d\xi_1\rho(\xi_2)d\xi_2, & i = j \neq k, \\ \vdots & \text{permutations of two} \\ & \text{equal latin indices,} \\ \int_{\Gamma_1} \int_{\Gamma_2} \int_{\Gamma_3} H^\alpha(\xi_1)H^\beta(\xi_2)H^\gamma(\xi_3)\rho(\xi_1)d\xi_1 \dots \rho(\xi_3)d\xi_3, & i \neq j, i \neq k, j \neq k, \end{cases} \\ C_{\alpha,\beta}^{i,j} &:= \begin{cases} \int_{\Gamma_1} H^\alpha(\xi_1)H^\beta(\xi_1)\rho(\xi_1)d\xi_1, & i = j, \\ \int_{\Gamma_1} \int_{\Gamma_2} H^\alpha(\xi_1)H^\beta(\xi_2)\rho(\xi_1)d\xi_1\rho(\xi_2)d\xi_2, & i \neq j, \end{cases} \end{aligned} \tag{18}$$

which give us the desired lookup-tables for all possible combinations of tuples of basis functions and their locations within the reference-element. We have used the fact that $|\Gamma_i| = \int_{\Gamma_i} \rho(\xi)d\xi_i = \int_{\Gamma_i} \rho_i(\xi_i)d\xi_i = 1$ for any ran-

dom variable ξ_i . Using (7) 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. Preusser and Rumpf 1999). 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 (17), 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 $\{\kappa_1, \dots, \kappa_d\} \in \mathbb{R}^d$ and the quadrature points with $\{z_1, \dots, z_d\} \in \Gamma_*^d$. Thus, we have to compute

$$g_\alpha^i \approx (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} \sum_{l_1, \dots, l_M=1}^d \kappa_{l_1} \cdots \kappa_{l_M} H^\alpha((z_l)_i) \times \int_D g(v)(x, z_l) P_i(x) dy,$$

where we set $z_l = (z_{l_1}, \dots, z_{l_M}) \in \Gamma$ such that $(z_l)_i = z_{l_i}$. 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 those stochastic variables only, whose supporting spatial basis function intersects 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

$$g_\alpha^i \approx (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} \sum_{E \in \mathcal{G}} \sum_{\substack{l_j=1 \\ j=1, \dots, M \\ x_j \in E}}^d \left(\prod_j \kappa_{l_j} \right) H^\alpha((z_l)_i) \times \int_E g(v)(x, z_l) P_i(x) dy.$$

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

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

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

Although it appears 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 this section 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 and insight.

Namely, first we present linear diffusion of an image u via $\partial_t u = \Delta u$, with initial condition $u(t = 0) = f$ and where Δ denotes the spatial Laplacian. This describes the temporal evolution of u , creating the so-called Gaussian Scale-Space (Iijima 1962, 1963; Witkin 1983). And second we present a nonlinear diffusion of u via $\partial_t u = \text{div}(g(|\nabla u|)\nabla u)$, also called Perona-Malik diffusion (Perona and Malik 1990).

Our intention is to enable the reader to gain understanding of computer vision with SPDEs and to see the connections to classical/deterministic approaches with PDEs on the 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. Deb et al. 2001; Keese 2004; Ghanem and Spanos 1991; Xiu and Karniadakis 2002).

Discretization of SPDEs is completely analogous to classical FEM discretizations of PDEs, when having the stochastic ansatz spaces from Sect. 2 at hand:

1. temporal derivatives are exchanged by an Euler scheme,
2. the resulting equations are converted into their respective weak formulation¹ by projection onto test functions z and integrating by parts, and
3. solution images u are represented by their expansions.

This yields a system of equations which can be written in block form. The system is then solved for the coefficients of the expansion of u . We will now examine each of the aforementioned diffusion problems in detail.

¹The weak formulation of a PDE means that it does not have to hold point-wise everywhere (strong form), but only when integrated against a test function. This relaxes the problem: instead of deriving an exact solution everywhere, we derive a solution satisfying the strong form on average over the test-function domain.

4.1 Linear Diffusion Filtering

We begin with the SPDE formulation of the most prominent linear filter, the heat equation which yields the Gaussian Scale-Space (Iijima 1962, 1963; Witkin 1983). We define the classical diffusion PDE

$$\partial_t u = \Delta u \tag{19}$$

as an evolution equation acting on the image gray values u . Instead of the usual intensity image $u(t, x)$, we directly use the gray-value functions $u(t, x, \xi)$ with the random fields introduced in Sect. 2. Thus, the stochastic version of the heat equation reads:

Given initial data $f(x, \xi)$ find a family $\{u(t, x, \xi)\}_{t \in \mathbb{R}^+}$ of filtered versions of this image such that

$$\begin{aligned} \partial_t u(t, x, \xi) - \Delta u(t, x, \xi) &= 0 \quad \text{a.e. in } \mathbb{R}^+ \times D \times \Gamma, \\ \partial_\nu u(t, x, \xi) &= 0 \quad \text{a.e. on } \mathbb{R}^+ \times \partial D \times \Gamma, \\ u(0, x, \xi) &= f(x, \xi) \quad \text{a.e. in } D \times \Gamma. \end{aligned} \tag{20}$$

We will now discretize this equation step-by-step. First let us apply the temporal backwards Euler scheme. This means we replace the temporal derivative by a backward difference quotient $\partial_t u(t, x, \xi) \approx (u(t, x, \xi) - u(t - \tau, x, \xi))/\tau$, introducing the time-step τ . Using the notation $u^n(x, \xi) := u(n\tau, x, \xi)$ we obtain:

For $n = 1, 2, 3, \dots$ find $u^n : D \times \Gamma \rightarrow \mathbb{R}$ such that

$$\begin{aligned} u^n(x, \xi) - \tau \Delta u^n(x, \xi) &= u^{n-1}(x, \xi) \quad \text{a.e. in } D \times \Gamma, \\ \partial_\nu u^n(x, \xi) &= 0 \quad \text{a.e. on } \mathbb{R}^+ \times \partial D \times \Gamma, \end{aligned} \tag{21}$$

where $u^0(x, \xi) = f(x, \xi)$ a.e. in $D \times \Gamma$.

Consequently we have transferred the parabolic SPDE (20) into a sequence of elliptic SPDEs (21). Each of the equations of this sequence must be interpreted in a weak sense: Following the stochastic Galerkin Method (Deb et al. 2001), we first need to project each equation to a set of test functions z in the same way as done for a standard Galerkin Method (Thomee 1984). Second, we integrate the expression by parts. The difference is that in the stochastic case all z reside in the space $H^1(D) \otimes L^2(\Gamma)$ introduced in Sect. 2.2. The projection is done as follows: We multiply each 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

$$\begin{aligned} \mathbb{E} \left[\int_D u^n(x, \xi) z(x, \xi) dx - \tau \int_D \Delta u^n(x, \xi) z(x, \xi) dx \right] \\ = \mathbb{E} \left[\int_D u^{n-1}(x, \xi) z(x, \xi) dx \right] \end{aligned}$$

for all test-functions $z \in H^1(D) \otimes L^2(\Gamma)$. Integrating by parts in D leads to the weak form of the SPDE (21)

$$\begin{aligned} \mathbb{E} \left[\int_D u^n(x, \xi) z(x, \xi) dx - \tau \int_D \nabla u^n(x, \xi) \cdot \nabla z(x, \xi) dx \right] \\ = \mathbb{E} \left[\int_D u^{n-1}(x, \xi) z(x, \xi) dx \right]. \end{aligned} \tag{22}$$

Having derived the weak form, we need to represent the solution u^n in a finite-dimensional sub-space, i.e. we consider $u^n \in \mathcal{H}_{\text{still}}^{p,h}$. To do so, we substitute the expansion (4) into this weak form. In addition to this we plug in the basis functions $H^\beta(\xi_j) P_j(x)$ as test-functions z . Using the linearity of the expectation to pull the coefficients $(u^n)_\alpha^i$ and $(u^{n-1})_\alpha^i$ in front of the integrals we get

$$\begin{aligned} \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p (u^n)_\alpha^i \left(\mathbb{E} \left[\int_D H^\alpha(\xi_i) P_i(x) H^\beta(\xi_j) P_j(x) dx \right] \right. \\ \left. - \tau \mathbb{E} \left[\int_D (H^\alpha(\xi_i) \nabla P_i(x)) \cdot (H^\beta(\xi_j) \nabla P_j(x)) dx \right] \right) \\ = \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p (u^{n-1})_\alpha^i \mathbb{E} \left[\int_D H^\alpha(\xi_i) P_i(x) H^\beta(\xi_j) P_j(x) dx \right] \end{aligned}$$

for all $j \in \mathcal{I}$ and $\beta = 1, \dots, p$. Remember, the coefficients $(u^n)_\alpha^i$ are the stochastic modes of u at pixel positions i , i.e. images containing weights belonging to polynomials H^α of order α .

This is a system of equations which can be written in a block form

$$\begin{aligned} \sum_{\alpha=1}^p (M^{\alpha,\beta} + \tau L^{\alpha,\beta})(U^n)_\alpha = \sum_{\alpha=1}^p M^{\alpha,\beta} (U^{n-1})_\alpha \\ \text{for } \beta = 1, \dots, p \end{aligned} \tag{23}$$

ordering the unknowns of u^n as in (5) to get U^n . The matrices $M^{\alpha,\beta}$ and $L^{\alpha,\beta}$ are stochastic mass- and stiffness-matrices, respectively. They have the entries

$$\begin{aligned} (M^{\alpha,\beta})_{ij} &= \mathbb{E} \left[\int_D H^\alpha(\xi_i) P_i(x) H^\beta(\xi_j) P_j(x) dx \right] \\ &= \left(\int_\Gamma H^\alpha(\xi_i) H^\beta(\xi_j) \rho(\xi) d\xi \right) \\ &\quad \times \left(\int_D P_i(x) P_j(x) dx \right) \\ &= C_{\alpha,\beta}^{i,j} \left(\int_D P_i(x) P_j(x) dx \right), \end{aligned} \tag{24}$$

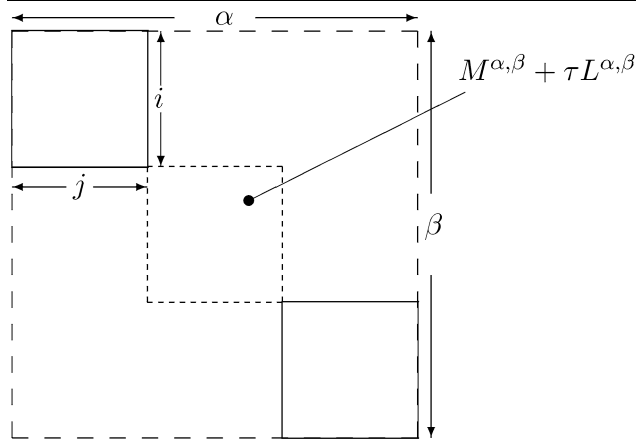


Fig. 6 Structure of the block system of the stochastic heat equation (23)

$$\begin{aligned}
 (L^{\alpha,\beta})_{ij} &= \mathbb{E} \left[\int_D (H^\alpha(\xi_i) \nabla P_i(x)) \cdot (H^\beta(\xi_j) P_j(x)) dx \right] \\
 &= \left(\int_\Gamma H^\alpha(\xi_i) H^\beta(\xi_j) \rho(\xi) d\xi \right) \\
 &\quad \times \left(\int_D \nabla P_i(x) \cdot \nabla P_j(x) dx \right) \\
 &= C_{\alpha,\beta}^{i,j} \left(\int_D \nabla P_i(x) \cdot \nabla P_j(x) dx \right) \quad (25)
 \end{aligned}$$

where we have used the tensor C from (18). The matrices result from the classical mass- and stiffness-matrices by an entry-wise multiplication with the expectation of pair-products of stochastic basis functions. The coefficient $C_{\alpha,\beta}^{i,j}$ 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 system 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 existing deterministic FEM code on each of the stochastic modes separately, provided we have multiplied the deterministic system matrix component-wise with the tensor $C_{\alpha,\beta}^{ij}$. We are going to discuss results of the stochastic linear diffusion in Sect. 4.4.

4.2 Perona Malik Diffusion

We will now implement a stochastic version of the well known nonlinear Perona Malik diffusion (Perona and Malik

1990) in order to demonstrate the influence of its non-linear behavior. Proceeding analogously to Sect. 4.1 we formulate this problem as:

Given initial data $f(x, \xi)$ find a family $\{u(t, x, \xi)\}_{t \in \mathbb{R}^+}$ of filtered versions of this image such that

$$\begin{aligned}
 \partial_t u(t, x, \xi) - \operatorname{div}(g(|\nabla u(t, x, \xi)|) \nabla u(t, x, \xi)) &= 0 \\
 \text{a.e. in } \mathbb{R}^+ \times D \times \Gamma, \\
 \partial_\nu u(t, x, \xi) &= 0 \quad \text{a.e. on } \mathbb{R}^+ \times \partial D \times \Gamma, \\
 u(0, x, \xi) &= f(x, \xi) \quad \text{a.e. in } D \times \Gamma.
 \end{aligned}$$

Here g is the edge indicator function we have already worked with in Sect. 3.4.

Remark 4 It is known that the Perona Malik model suffers from an ill-posedness which leads to non-existence of solutions (Kichenassamy 1997). Since numerical schemes unavoidably introduce regularizations, the ill-posedness in general does not lead to problems in practice. However, we emphasize that in our framework it is straight-forward to implement the regularized Perona-Malik model as introduced by Catté et al. (1992), which uses a smoothed version u_ρ of the image u to compute the diffusion tensor $g(|\nabla u_\rho|)$. In the FEM context this smoothing can be obtained by one scale step of the (stochastic) heat equation (cf. Sect. 5.5).

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

$$\begin{aligned}
 &\mathbb{E} \left[\int_D u^n(x, \xi) z(x, \xi) dx \right. \\
 &\quad \left. - \tau \int_D g(|\nabla u^{n-1}(t, x, \xi)|) \nabla u^n(x, \xi) \cdot \nabla z(x, \xi) dx \right] \\
 &= \mathbb{E} \left[\int_D u^{n-1}(x, \xi) z(x, \xi) dx \right]
 \end{aligned}$$

for all test functions $z \in H^1(D) \otimes L^2(\Gamma)$. We observe that only the second term differs from its linear analog (22), and only by the factor $g(|\nabla u^{n-1}(t, x, \xi)|)$, i.e. the nonlinearity which is treated explicitly here. Considering $u^n \in \mathcal{H}_{\text{still}}^{h,p}$ as above leads to a system of equations

$$\begin{aligned}
 &\sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p (u^n)_\alpha^i \left(\mathbb{E} \left[\int_D H^\alpha(\xi_i) P_i(x) H^\beta(\xi_j) P_j(x) dx \right] \right. \\
 &\quad \left. - \tau \mathbb{E} \left[\int_D \left(\sum_{\gamma=1}^p \sum_{k \in \mathcal{I}} (G^n)_\gamma^k H^\gamma(\xi_k) P_k(x) \right) \right. \right. \\
 &\quad \left. \left. \times (H^\alpha(\xi_i) \nabla P_i(x)) \cdot (H^\beta(\xi_j) \nabla P_j(x)) dx \right] \right)
 \end{aligned}$$

$$= \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p (u^{n-1})_{\alpha}^i \mathbb{E} \left[\int_D H^{\alpha}(\xi_i) P_i(x) H^{\beta}(\xi_j) P_j(x) dx \right]$$

for each $\beta = 1, \dots, p$ and $j \in \mathcal{I}$. Here we have substituted the expansion (4) for the edge indicator $g(|\nabla u^n|)$ which has been derived in Sect. 3.4, i.e. $(G^n)_{\gamma}^k$ denotes the stochastic modes of g .

Remark 5 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 analogous condition must hold for the stochastic diffusion equation (Deb et al. 2001), and we have to take care of the positivity of the stochastic process of the diffusion tensor g . In case the positivity is violated, the stochastic process describing the distribution of the diffusion tensor must be truncated such that it stays away from zero. If $f(\xi, x)$ describes the diffusion tensor at a fixed location $x \in D$, a truncated process would be $\tilde{f}(\xi, x) = \max\{c, f(\xi, x)\}$ for some constant $0 < c \ll 1$. Consequently, all integrations in the weak form of the equation must use the truncated process \tilde{f} . However, in our numerical experiments we did not encounter any problems with the projection and quadrature method described in Sects. 3.4 and 3.6.2 and the resulting g was always positive.

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

$$\sum_{\alpha=1}^p (M^{\alpha,\beta} + \tau(L^n)^{\alpha,\beta})(U^n)_{\alpha} = \sum_{\alpha=1}^p M^{\alpha,\beta}(U^{n-1})_{\alpha}$$

for $\beta = 1, \dots, p$

if we order the unknowns of u^n as in (5) to get U^n . The mass-matrices $M^{\alpha,\beta}$ are as before in (24) and $(L^n)^{\alpha,\beta}$ is now

$$\begin{aligned} & ((L^n)^{\alpha,\beta})_{ij} \\ &= \mathbb{E} \left[\int_D \left(\sum_{\gamma=1}^p \sum_{k \in \mathcal{I}} (G^n)_{\gamma}^k H^{\gamma}(\xi_k) P_k(x) \right) \right. \\ & \quad \left. \times (H^{\alpha}(\xi_i) \nabla P_i(x)) \cdot (H^{\beta}(\xi_j) P_j(x)) dx \right] \\ &= \sum_{\gamma=1}^p \sum_{k \in \mathcal{I}} (G^n)_{\gamma}^k \left(\int_{\Gamma} H^{\alpha}(\xi_i) H^{\beta}(\xi_j) H^{\gamma}(\xi_k) \rho(\xi) d\xi \right) \\ & \quad \times \left(\int_D \nabla P_i(x) \cdot \nabla P_j(x) P_k(x) dx \right) \\ &= \sum_{\gamma=1}^p \sum_{k \in \mathcal{I}} (G^n)_{\gamma}^k B_{\alpha,\beta,\gamma}^{i,j,k} \left(\int_D \nabla P_i(x) \cdot \nabla P_j(x) P_k(x) dx \right). \end{aligned} \tag{26}$$

This block system is not block-diagonal any more because of the stochastic diffusion tensor g which leads to an 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

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

where

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

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.

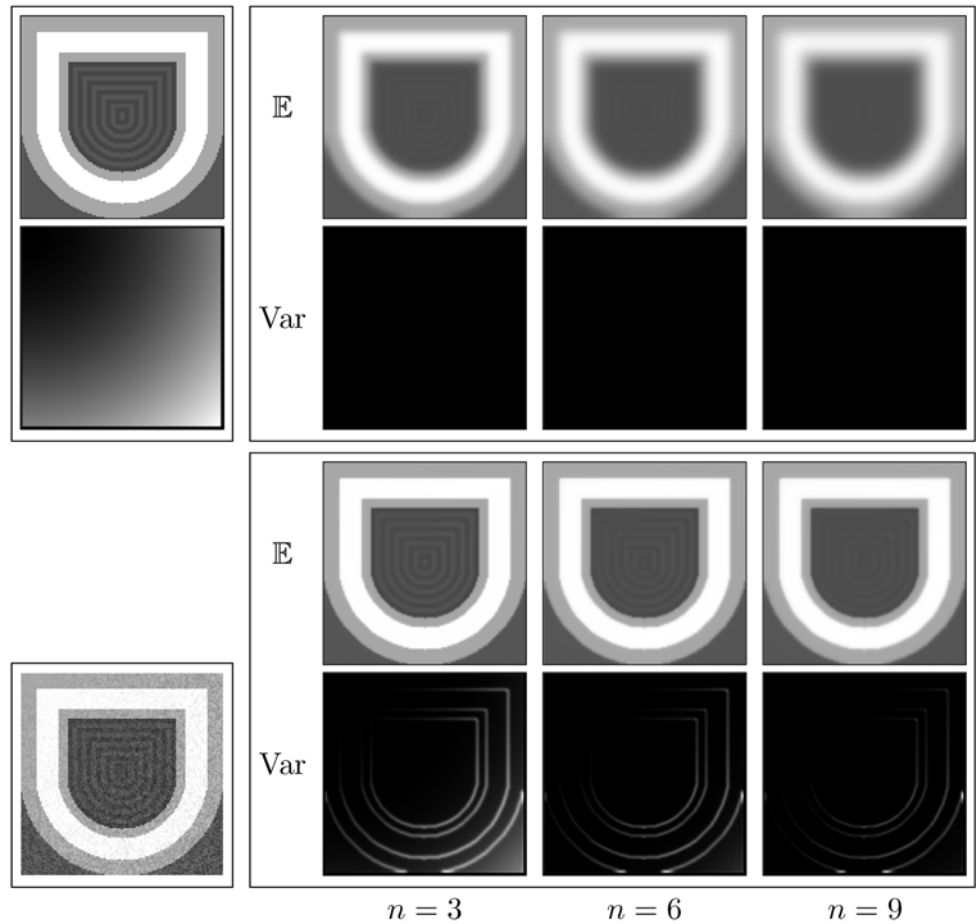
4.3 Numerical Aspects and Efficiency

In (24), (25), and (26) we have seen that for standard diffusion models the local stochastic mass- and stiffness-matrices result from an entry-wise multiplication of the classical mass and stiffness matrices with the coefficients of the tensors A , B , and C from (18). This means that in the standard case the additional effort for the assembly of the local stochastic matrices is just one multiplication with the tensor coefficient per local matrix entry. The tensors A , B , and C as well as local standard mass- and stiffness-matrices can be computed in advance and stored in lookup tables.

For the stochastic heat equation and the stochastic Perona Malik equation the resulting linear system of equations has $p \times p$ blocks (cf. Fig. 10), where p is the number of stochastic modes used in the expansion (4). Each of those blocks has the size of the corresponding classical deterministic problem, i.e. $N \times N$, where N is the number of pixels in the image. Consequently the dimension of the stochastic linear system is $(pN) \times (pN)$. Since the systems are symmetric and positive definite they can be solved by standard iterative linear solvers like e.g. the conjugate gradient (CG) method (Avriel 2003).

As a consequence of the local support of the basis functions P_i the resulting systems are sparse. Therefore, matrix vector multiplications in the iterative CG solver need an effort of $O(p^2N)$ in contrast to $O(N)$ for the deterministic matrices, since the number of bands is multiplied by p and the number of unknowns is multiplied by p as well. Since a CG solver converges after M steps if the number of unknowns is M , we conclude that the effort of the stochastic

Fig. 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. One realization of the random field (i.e. a noisy image) is shown in the *bottom left* image



variant is equal to the deterministic effort multiplied by a factor of p^2 , i.e. $O(p^2N)$ instead of $O(N)$.

The experiments shown in Fig. 8 below show that our framework outperforms a naive approach like Monte-Carlo (MC) sampling. In this MC approach samples are drawn from the input distribution, i.e. different noisy images. To these images the standard, deterministic algorithm is applied. Doing so, the output distribution is sampled. In fact, for two stochastic modes $p = 2$ the effort of our algorithm corresponds to $p^2 = 4$ Monte-Carlo samples for which a good approximation of the distribution in general is not obtained (cf. Fig. 8).

4.4 Results

In Fig. 7 we present 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 $p = 2$. The image is of resolution 129×129 and the gray values range in the interval $[0, 1]$. For the sake of simplicity, the variance of the input data is defined by set-

ting the second mode to

$$f_2(x) = \delta |x|, \quad \delta = \frac{0.3}{129\sqrt{2}}$$

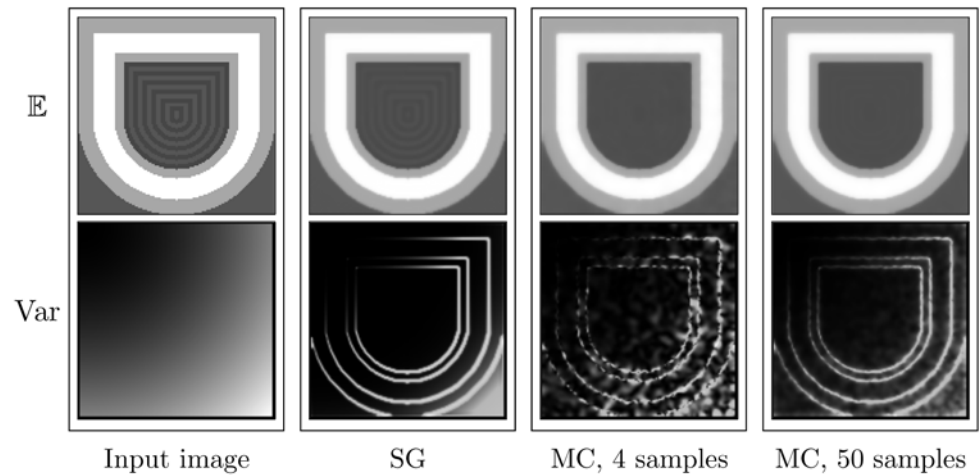
and letting all higher modes be zero. This means that we model a spatially varying uniform distribution of the input gray values (cf. Fig. 1, top). In Sect. 5.7 we will model Gaussian-shaped distributions as well.

Let us note that for uniform distributions over an interval $[c - d, c + d]$ the mean is given by c and the variance is $\sigma^2 = d^2/3$. Conversely, given the variance σ^2 the interval half width is $d = \sqrt{3}\sigma$. Using two modes for the description of the stochastic process and setting the second mode $f_2(x) = a$ means that we have a variance of

$$\begin{aligned} \text{Var}(f) &= a^2 \langle H^2, H^2 \rangle = a^2 \int_{-1}^1 \xi_i^2 \rho_i(\xi_i) d\xi_i \\ &= a^2 \int_{-1}^1 \frac{\xi_i^2}{2} d\xi_i = \frac{1}{3} a^2. \end{aligned}$$

This results from the fact that the second Legendre basis function is $H^2(\xi_i) = \xi_i$ and the PDF is $\rho_i(\xi_i) = 1/2$. We see that the second mode $f_2 = a$ directly represents the half width of the interval of a uniform distribution.

Fig. 8 A comparison of the stochastic Perona-Malik model with Monte-Carlo experiments is shown for the scale step ($n = 6$). As before, the *top row* shows the mean and the *bottom row* the variance. *Left:* Input distribution of a test image (cf. Fig. 7). *Middle left:* Result from the stochastic Perona-Malik model discretized with the stochastic Galerkin (SG) method. *Middle right:* Monte Carlo experiment with 4 samples. *Right:* Monte Carlo Experiment with 50 samples



In our example the variance of the input data ranges from zero to

$$\sigma = \frac{\max_x f_2}{\sqrt{3}} = \frac{\max_x \delta|x|}{\sqrt{3}} = \frac{0.3}{\sqrt{3}}.$$

This means in our example we model an uncertainty of the gray values varying up to ± 0.3 around the mean gray value. For the computations we have chosen a time-step of $\tau = 1$ and we use $\lambda = 0.02$ (see (11)).

Looking at the top frames in Fig. 7 we see, that no structure from the expectation images (top) can be seen in the variances (second row) and vice versa. 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 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, i.e. we see the coupling of the modes. There is also a smoothing of the variance present with increasing scale step n . In the limit $n \rightarrow \infty$ the results of the linear and non-linear diffusion models will be the same, i.e. an image of constant gray value and zero variance.

In Fig. 8 we show a comparison of the stochastic Perona-Malik diffusion results with a Monte-Carlo experiment. To this end we have created samples of the input image distribution shown in Fig. 7, applied our deterministic Perona-Malik diffusion solver, and computed the mean and the variance of the resulting images through the standard formulas from statistics. This experiment can be seen as a validation of the Galerkin approach for the stochastic Perona Malik equation. Indeed the Monte-Carlo experiment seems to converge to the result from our new approach. However, we emphasize that the new approach is much more efficient than the

Monte-Carlo method. To compute the mean and the variance of the output the stochastic Galerkin method needs the effort of about 4 deterministic runs. From the figures we see that with just 4 samples the results from the Monte-Carlo method is quite bad. This underlines the power of our framework.

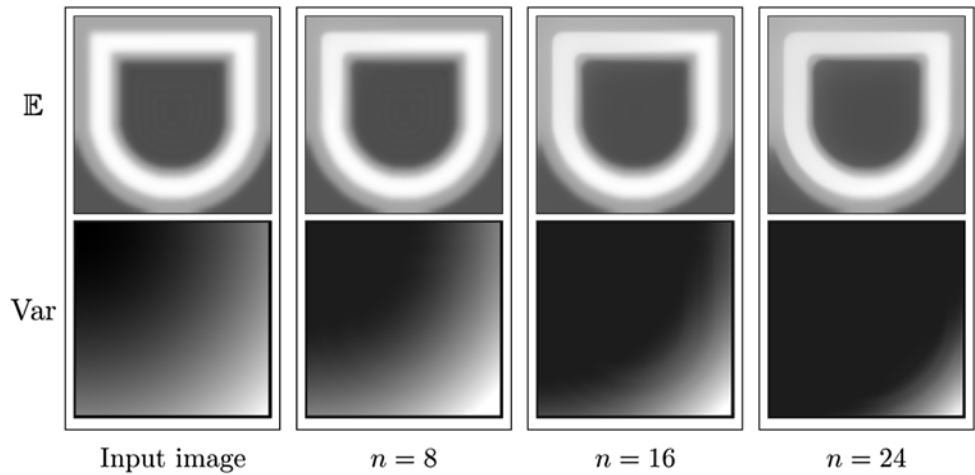
Finally, in Fig. 9 we investigate the edge enhancing character of the stochastic Perona Malik equation. Starting the evolution with a blurred version of the test-pattern already used before we indeed get an enhancement of edges, which however depends on the variance of the input data. From the figure we see that the edge enhancement occurs first in regions with small variance (i.e. top left corner of the test image). For later time steps the enhancement also affects areas with larger variance. This is visible in the figures by an enhancement which progresses from the top left corner of the image to the lower right. In this example the coupling of the image gradients on the variance of the output is very mild. In fact, the structure of the image can be seen just very weakly in the variance images. We use a time-step $\tau = 5$ and an edge-indicator parameter $\lambda = 0.002$.

5 Optical Flow Computations

In the following section 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 such as:

- the Horn and Schunck (HS) model (Horn and Schunck 1981),
- a discontinuity preserving optical flow model (Black and Anandan 1991; Cohen 1993; Weickert 1998), and
- the combined local global (CLG) model (Bruhn et al. 2002; Bruhn et al. 2005).

Fig. 9 The edge enhancing property of the stochastic Perona Malik model is investigated experimentally. From *left to right*: Input image distribution and three scale steps of the stochastic Perona Malik model. Note the edge enhancement, which occurs in regions of low variance, and which is progressing from the *top left* corner of the image with increasing scale



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 $f : D \times I \rightarrow \mathbb{R}$ on a spatial domain $D \subset \mathbb{R}^2$ 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) $w : R \times \Gamma \rightarrow \mathbb{R}^2$ such that $w = (u, v)$ describes the motion of structures inside the image sequence f . Classically, gray values are preserved along trajectories $x(t)$ of objects through the image sequence. Here gray values are preserved along stochastic trajectories $x(t, \xi)$, i.e.

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

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

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

The temporal behavior of the image sequence is modeled by the stochastic process f , but the vector of random variables ξ is time independent (cf. Sect. 2.3). This means that ξ does not change over time t (i.e. $\dot{\xi} = 0$), thus we can omit the last term. So we can formulate our stochastic optical flow constraint as

$$0 = w(x, t, \xi) \cdot \nabla_x f(x, t, \xi) + \partial_t f(x, t, \xi) \quad \text{a.e. } \Gamma. \quad (27)$$

This equation is completely analog to the classical optical flow equation, and as such we observe the ill-posedness also known as the aperture problem (see e.g. Haussecker and Spies 1999).

5.2 Stochastic Horn and Schunck Model

The standard HS model overcomes the aperture problem by selecting as the solution the flow field with minimal overall gradient. By this we mean that the classical HS solution minimizes the energy

$$\begin{aligned} E_{\text{HS}}(w) &= \|w \cdot \nabla_x f + \partial_t f\|^2 + \frac{1}{2} \kappa \|\nabla_{x,t} w\|^2 \\ &= \int_R (w(y) \cdot \nabla_x f(y) + \partial_t f(y))^2 dy \\ &\quad + \frac{1}{2} \kappa \int_R \nabla_{x,t} w(y) \cdot \nabla_{x,t} w(y) dy \end{aligned} \quad (28)$$

where the first term usually is called a data term and the second a smoothness or regularization term. Here we use a regularization in space and time which usually yields better results (Weickert and Schnörr 2001). Also the stochastic analog of the HS model regularizes the ill-posed optic flow equation by requiring the gradient of the flow field $\nabla_{x,t} w(x, t, \xi)$ to be small. However, 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 in (28). Thus, our energy involves the expectation

$$\begin{aligned} E(w) &= \mathbb{E} \|w \cdot \nabla_x f + \partial_t f\|^2 + \frac{1}{2} \kappa \mathbb{E} \|\nabla_{x,t} w\|^2 \\ &= \mathbb{E} \left[\int_R (w(y, \xi) \cdot \nabla_x f(y, \xi) + \partial_t f(y, \xi))^2 dy \right. \\ &\quad \left. + \frac{1}{2} \kappa \int_R \nabla_{x,t} w(y, \xi) \cdot \nabla_{x,t} w(y, \xi) dy \right] \end{aligned} \quad (29)$$

which in fact is the expectation of the classical HS energy E_{HS} from (28) applied to stochastic image sequences, i.e.

$$E(w) = \mathbb{E}[E_{HS}(w(y, \xi))]. \tag{30}$$

Remark 6 Equation (30) 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 ξ . Taking the expectation within (30) means that we consider the expected energy from this distribution and minimize it.

As in the classical approach, we derive the Euler equations as a necessary condition for a minimum of this energy. As there, we vary the components u and v of the flow field w independently. To do so, we select a function $z : D \times I \times \Gamma \rightarrow \mathbb{R}^2$ and 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 . This means $k = 1$ and 2 belong to u and v , respectively. Using the observation (30) we get

$$\begin{aligned} 0 &= \frac{d}{d\epsilon} E(w + \epsilon ze_k) \Big|_{\epsilon=0} = \frac{d}{d\epsilon} \mathbb{E}[E_{HS}(w + \epsilon ze_k)] \Big|_{\epsilon=0} \\ &= \mathbb{E} \left[\frac{d}{d\epsilon} E_{HS}(w + \epsilon ze_k) \Big|_{\epsilon=0} \right] \end{aligned}$$

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 (29) are just the expectation of the Euler equations of the classical energy E_{HS} and read

$$\begin{aligned} 0 &= \mathbb{E} \left[\int_R z(y, \xi) \partial_1 f(y, \xi) \right. \\ &\quad \times (w(y, \xi) \cdot \nabla_x f(y, \xi) + \partial_t f(y, \xi)) dy \\ &\quad \left. + \kappa \int_R \nabla_{x,t} u(y, \xi) \cdot \nabla_{x,t} z(y, \xi) dy \right], \tag{31} \end{aligned}$$

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

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

$$\begin{aligned} \partial_1 f(y, \xi)(w(y, \xi) \cdot \nabla f(y, \xi) + \partial_t f(y, \xi)) - \kappa \Delta u(y, \xi) &= 0 \\ \text{a.e. in } R \times \Gamma, \\ \partial_2 f(y, \xi)(w(y, \xi) \cdot \nabla f(y, \xi) + \partial_t f(y, \xi)) - \kappa \Delta v(y, \xi) &= 0 \\ \text{a.e. in } R \times \Gamma \end{aligned} \tag{32}$$

in the weak sense defined by (31). We note that this SPDE system is completely analog 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 (32). In the following we consider only one equation of the system, since the derivation for the other equation is completely analog.

We start by substituting $f(y, \xi)$, $u(y, \xi)$, and $v(y, \xi)$ with their respective Galerkin expansions (6) for stochastic image sequences into the weak form (cf. (31))

$$\begin{aligned} 0 &= \mathbb{E} \left[\int_R z \partial_1 f(w \cdot \nabla f + \partial_t f) dy + \kappa \int_R \nabla u \cdot \nabla z dy \right] \\ &=: \mathbb{E}[I] + \kappa \mathbb{E}[II]. \end{aligned}$$

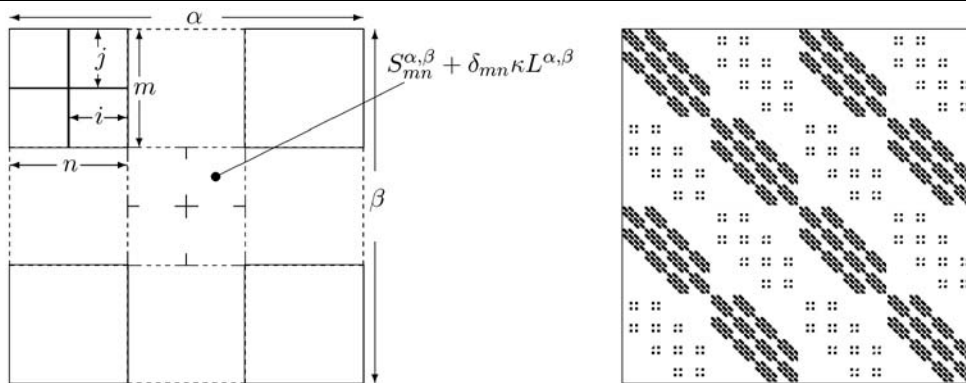
Together with the test function $z = H^\beta(\xi_{j_x}) P_j(y)$ we get

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

which can be collapsed to

$$\begin{aligned} \mathbb{E}[I] &= \sum_{i \in \mathcal{J}} \sum_{\alpha=1}^p \left(\sum_{k,l \in \mathcal{J}} \sum_{\gamma,\delta=1}^p f_\gamma^k f_\delta^l \right. \\ &\quad \times \int_\Gamma H^\alpha(\xi_{i_x}) H^\beta(\xi_{j_x}) H^\gamma(\xi_{k_x}) H^\delta(\xi_{l_x}) \rho(\xi) d\xi \\ &\quad \times \int_R P_i(y) P_j(y) \partial_1 P_k(y) \\ &\quad \times \left. (\partial_1 P_l(y) u_\alpha^i + \partial_2 P_l(y) v_\alpha^i) dy \right) \\ &+ \sum_{k,l \in \mathcal{J}} \sum_{\gamma,\delta=1}^p f_\gamma^k f_\delta^l \\ &\quad \times \int_\Gamma H^\beta(\xi_{j_x}) H^\gamma(\xi_{k_x}) H^\delta(\xi_{l_x}) \rho(\xi) d\xi \\ &\quad \times \int_R P_j(y) \partial_1 P_k(y) \partial_t P_l(y) dy. \tag{33} \end{aligned}$$

Fig. 10 *Left:* Structure and sub-structure of the block system (37). *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



The second term becomes

$$\mathbb{E}[II] = \sum_{i \in \mathcal{J}} \sum_{\alpha=1}^p u_{\alpha}^i \int_{\Gamma} H^{\alpha}(\xi_{i_x}) H^{\beta}(\xi_{j_x}) \rho(\xi) d\xi \times \int_R \nabla P_i(y) \cdot \nabla P_j(y) dy. \tag{34}$$

This identity leads to the stiffness matrix $L^{\alpha,\beta}$ which we already defined for the heat equation in (25). Moreover, using the tensors A and B from Sect. 3.6.2 we can define the matrices

$$(S_{mn}^{\alpha,\beta})_{ij} = \sum_{k,l \in \mathcal{J}} \sum_{\gamma,\delta=1}^p f_{\gamma}^k f_{\delta}^l A_{\alpha,\beta,\gamma,\delta}^{i_x,j_x,k_x,l_x} \times \int_R P_i(y) P_j(y) \partial_m P_k(y) \partial_n P_l(y) dy, \tag{35}$$

as well as the vector

$$(R_m^{\beta})_j = \sum_{k,l \in \mathcal{J}} \sum_{\gamma,\delta=1}^p f_{\gamma}^k f_{\delta}^l B_{\beta,\gamma,\delta}^{j_x,k_x,l_x} \times \int_R P_j(y) \partial_m P_k(y) \partial_l P_l(y) dy, \tag{36}$$

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

$$\sum_{\alpha=1}^p \left(\begin{pmatrix} S_{11}^{\alpha,\beta} & S_{12}^{\alpha,\beta} \\ S_{21}^{\alpha,\beta} & S_{22}^{\alpha,\beta} \end{pmatrix} + \kappa \begin{pmatrix} L^{\alpha,\beta} & 0 \\ 0 & L^{\alpha,\beta} \end{pmatrix} \right) \begin{pmatrix} U^{\alpha} \\ V^{\alpha} \end{pmatrix} = - \begin{pmatrix} R_1^{\beta} \\ R_2^{\beta} \end{pmatrix}, \quad \text{for } \beta = 1, \dots, p. \tag{37}$$

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 and each sub-block is a $N \times N$ matrix. In Fig. 10 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 block-diagonals only as we have already seen for the heat equation.

5.4 Discontinuity-Preserving Optical Flow Computation

As seen above, the HS energy E_{HS} from (28) consists of a data term and a regularization term. The regularization term consists of a linear diffusion (cf. Sect. 4.1) on the velocity components u and v . Discontinuity-preserving optical flow (Black and Anandan 1993; Cohen 1993; Weickert 1998) contains the same data term as in E_{HS} , but it uses a nonlinear regularization term instead of a linear one. This nonlinear regularization boils down to Perona Malik nonlinear diffusion as in Sect. 4.2. Having a discretization for nonlinear diffusion at hand, it is now straight forward to generalize the discontinuity-preserving optical flow model to a stochastic setting as well. This yields the SPDE system

$$\begin{aligned} \partial_1 f(y, \xi) (w(y, \xi) \cdot \nabla f(y, \xi) + \partial_t f(y, \xi)) \\ - \kappa \operatorname{div}(g(|\nabla f(y, \xi)|) \nabla u(y, \xi)) &= 0, \\ \partial_2 f(y, \xi) (w(y, \xi) \cdot \nabla f(y, \xi) + \partial_t f(y, \xi)) \\ - \kappa \operatorname{div}(g(|\nabla f(y, \xi)|) \nabla v(y, \xi)) &= 0, \end{aligned} \tag{38}$$

in R and a.e. Γ , which is interpreted in a sense analog to (31). To discretize the model we only have to replace the homogeneous stiffness matrices $L^{\alpha,\beta}$ from (37) by the inhomogeneous one defined in (26). As for the HS energy we get a system of equations like (37) which has a structure as shown in Fig. 10.

Note that for this model, one can also use a regularized version f_{ρ} of the input image sequence inside the edge indicator function, i.e. $g(|\nabla f_{\rho}(y, \xi)|)$. We emphasize that as for the Perona-Malik model from Sect. 4.2 this extension can

straight forwardly be incorporated within our framework. Here, in our FEM setting, the smoothing can be obtained by e.g. one small scale step of length $\rho^2/2$ the stochastic heat equation, which takes the image sequence f as initial data.

5.5 Combined Local Global (CLG) Method

In Sect. 5.4, above, we exchanged the linear regularization term in the HS energy E_{HS} by a nonlinear one. Here we keep the linear regularization term, but exchange the data term. Let us denote the homogeneous motion vector with $\bar{w} := (u, v, 1)$. The classical HS energy from (28) then reads

$$E_{HS}(w) = \|\bar{w} \cdot J \bar{w}\| + \frac{1}{2} \kappa \|\nabla_{x,t} w\|^2 \tag{39}$$

where we used the notation $J = (\nabla_{x,t} f)^T (\nabla_{x,t} f)$ for the outer product of the spatio-temporal gradient $\nabla_{x,t} f$ of the image sequence f . Smoothing this 3×3 tensor J component-wise yields the so-called structure-tensor J_ρ (Jähne 1993). In a finite difference setting ρ indicates the variance of a Gaussian kernel used to smooth J . Again we consider J_ρ as the solution of the heat equation (as in Sect. 4.1) for a small single time step of length $\rho^2/2$ with initial data J . Such a diffusion has to be applied independently to each structure tensor component J^{mn} , $m, n \in \{1, 2, t\}$. Using J_ρ instead of J in the HS energy E_{HS} yields the stochastic version of the CLG energy

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

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

$$\begin{aligned} 0 &= \mathbb{E} \left[\int_R z(y, \xi) (J_\rho^{11}(y, \xi) u(y, \xi) + J_\rho^{12}(y, \xi) v(y, \xi) \right. \\ &\quad \left. + J_\rho^{1t}(y, \xi)) dx \right. \\ &\quad \left. + \kappa \int_R \nabla_{x,t} u(y, \xi) \cdot \nabla_{x,t} z(y, \xi) dx \right], \\ 0 &= \mathbb{E} \left[\int_R z(y, \xi) (J_\rho^{21}(y, \xi) u(y, \xi) + J_\rho^{22}(y, \xi) v(y, \xi) \right. \\ &\quad \left. + J_\rho^{2t}(y, \xi)) dx \right. \\ &\quad \left. + \kappa \int_R \nabla_{x,t} v(y, \xi) \cdot \nabla_{x,t} z(y, \xi) dx \right]. \end{aligned}$$

The discretization is completely analog to the discretization of the stochastic HS model from Sect. 5.2. To obtain matrices similar to (35) let us denote the coefficients of the expansion (6) of the smoothed structure tensor with $(J_\rho^{mn})_\gamma^k$ for $m, n \in \{1, 2, t\}$, pixel position $k \in \mathcal{J}$ and expansion order $\gamma = 1, \dots, p$ and define

$$\begin{aligned} (\tilde{S}_{mn}^{\alpha,\beta})_{ij} &= \sum_{k \in \mathcal{J}} \sum_{\gamma=1}^p (J_\rho^{mn})_\gamma^k B_{\alpha,\beta,\gamma}^{i_x, j_x, k_x} \\ &\quad \times \int_R P_i(y) P_j(y) P_k(y) dy, \end{aligned} \tag{40}$$

and a right hand side

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

The block system is the same as (37) where S and R have been replaced with the ones from above.

Using an inexact quadrature rule as in Sect. 4.2 we can simplify the computation of the entries of \tilde{S}

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

where

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

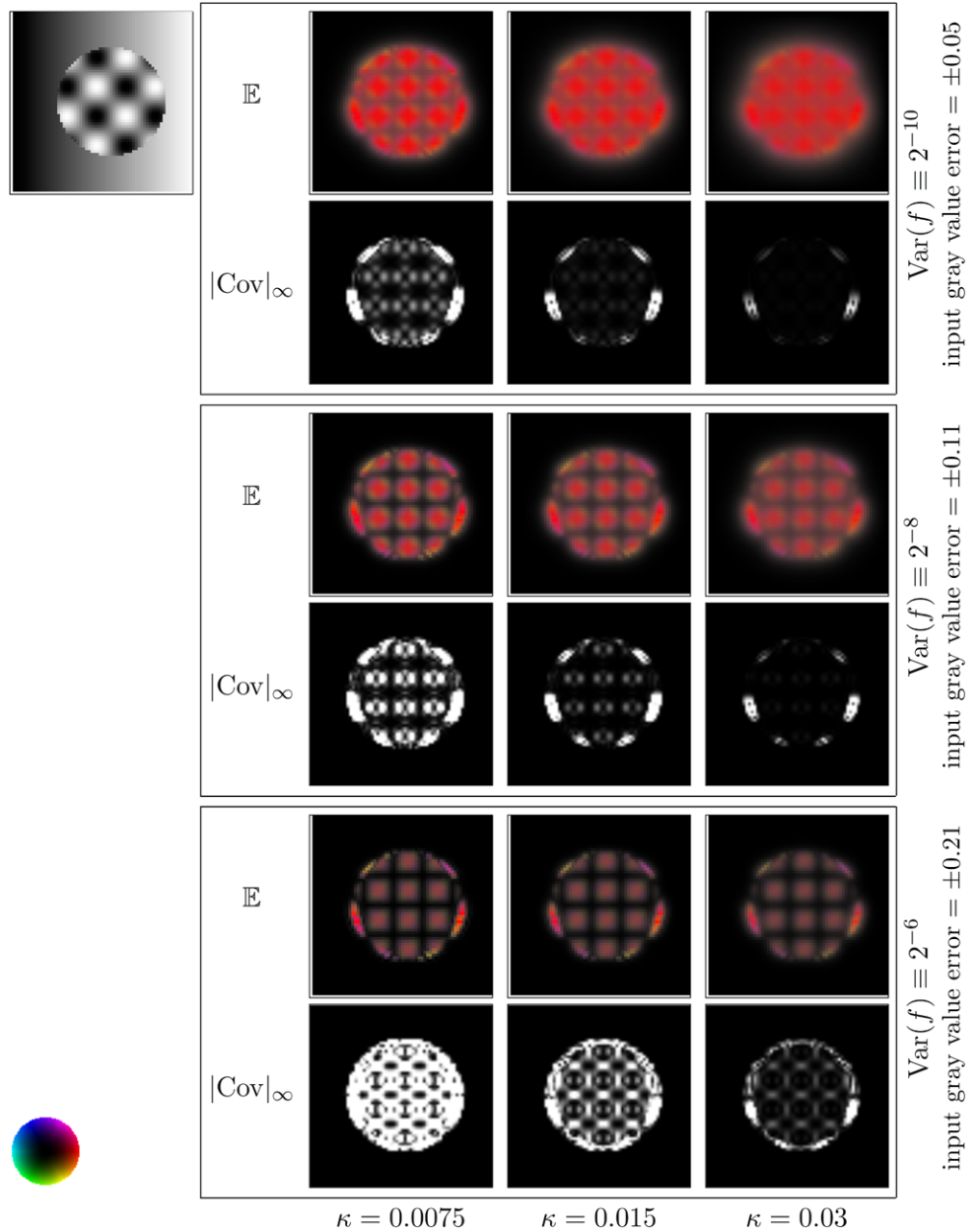
and similarly for the right-hand-side \tilde{R} .

5.6 Numerical Aspects and Efficiency

As for the standard diffusion models the local stochastic matrices result from entry-wise multiplications with the coefficients of the tensors A , B and C from (18). However, in the present case of nonlinearities in (35), (36), and (40) there is also a summation over the stochastic modes $\gamma, \delta = 1, \dots, p$ used in the expansion (4). As for the diffusion models, many components of the deterministic local matrices can be computed in advance and stored in lookup tables. Due to the summation over the stochastic modes the assembly effort is multiplied by a factor of p^2 for the stochastic Horn and Schunck model (35), and by a factor of p for the CLG approach (40). However in the case of the CLG model we need additional effort for the computation and the smoothing of the structure tensor.

Again the resulting linear systems of equations are sparse, symmetric, and positive definite. Thus we can use

Fig. 11 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



a CG method for the solution of the systems. Since we are dealing with vector valued problems, the dimension of the stochastic system is $(2pN) \times (2pN)$. As above for the diffusion models, the effort for the solution is multiplied by a factor of p^2 in comparison to the deterministic system.

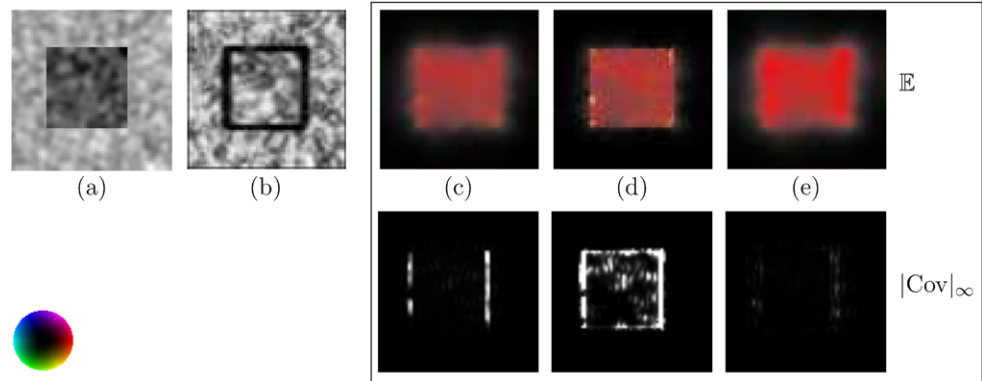
In the following section we will present an experiment that shows how our framework outperforms a Monte-Carlo sampling approach. For the computation of mean and variance of the output we would need $p = 2$ modes. Thus the effort of our framework is approximately equal to 4 Monte-Carlo samples for which a convergence can not be expected.

5.7 Results

Let us start with the computation of the optical flow field of two test-sequences. First we consider a disc 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.

In Fig. 11 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

Fig. 12 From a sequence showing a moving textured square (spatial resolution 65×65 , 11 frames, mean shown in (a), for the variance see the 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*



considered variances of the input data f to be 2^{-6} , 2^{-8} and 2^{-10} . This corresponds to errors in the input gray values of 21%, 11%, and 5%, respectively. Moreover we have studied the influence of the smoothing parameter κ .

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 κ , 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 couples the stochastic modes, thus an output variance is sensitive to gradients of the mean of the input data. For fixed variance of the input data and increasing smoothness κ the influence of the gradients in the input mean is weakened and thus the amplitude of the variance of the flow field is damped. This behavior we have already seen from the heat equation in Sect. 4.1 (cf. also Sect. 4.4). 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. 12 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. 12 to be the mean. The gray values of the mean image range from 0 to 1. Furthermore we set

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

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

In Fig. 12 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 $\kappa = 0.03$, set $\lambda = 1/300$ 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.

Again, we emphasize the benefit of our approach: In contrast to existing work our Ansatz does not only yield bounds for errors or just confidence measures. Indeed we transform distributions of the input data into distributions of the output data. Such distributions of the velocity are depicted in Fig. 13 for two different locations in the moving square sequence from the previous experiment. For a pixel inside the moving object (32/32/5) and for a pixel at its upper border (32/14/5) the figure shows the resulting random processes of the u and v velocity component and their PDFs.

Let us 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 value $\mathbb{E}[\hat{w}]$ of the estimated optical flow components

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

The bias occurring in optical flow estimators based on least squares estimation leading to an underestimation of the optical flow components u, v has been extensively examined (cf. e.g. Van Huffel and Vandewalle 1991; Kearney et al. 1987; Fermüller et al. 2001). For local optical flow estimators, e.g. of Lucas-Kanade type (Lucas and Kanade 1981), 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

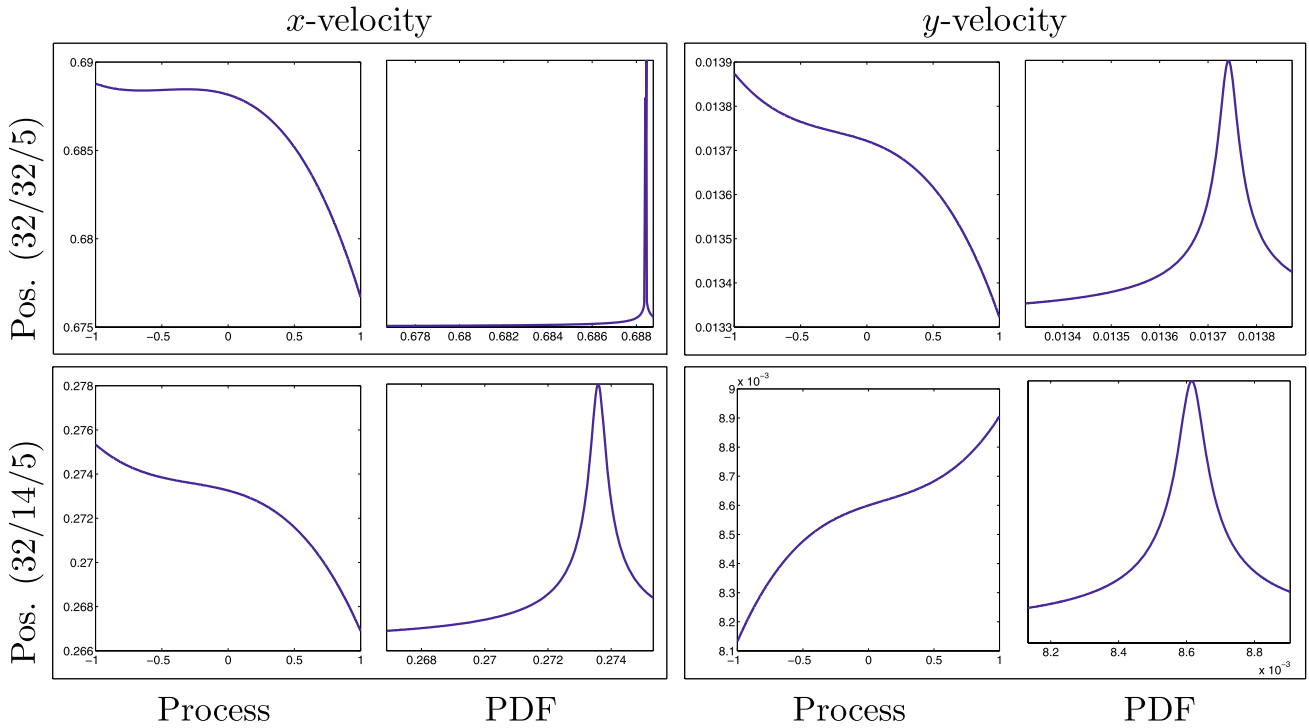


Fig. 13 For two different locations in the moving square image sequence we plot the stochastic processes of the x - and y -component of the velocity as well as the corresponding PDFs. In the *top row* we consider a location in the center frame of the sequence and in the center of the moving object. In the *bottom row* a location in the center frame

and at the boundary of the moving square is considered. The process is approximated using polynomials of degree three ($p = 4$). As such polynomials are antisymmetric around their inflexion point, the PDF is approximated to be symmetric around its maximum

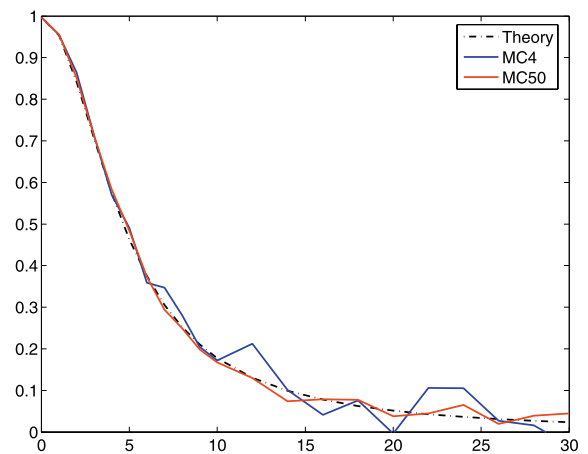
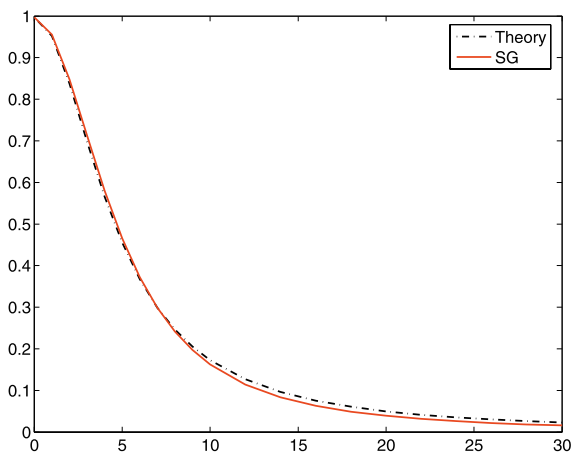
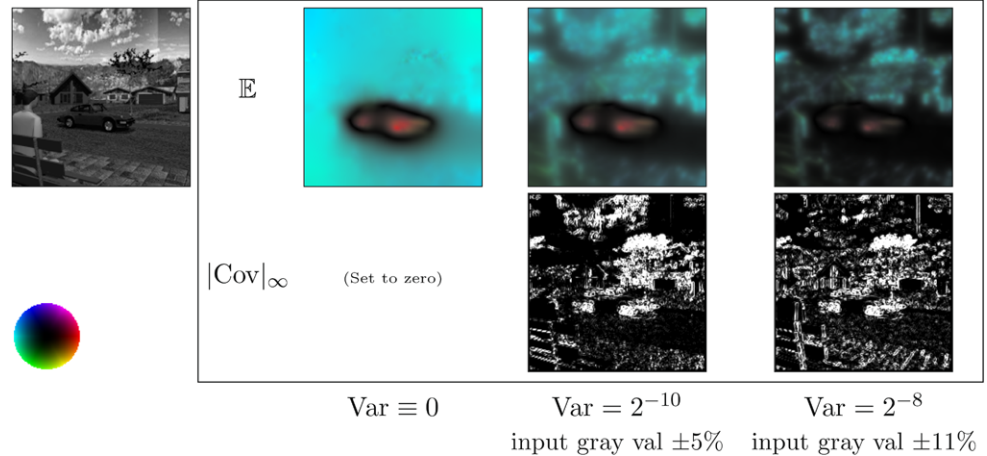


Fig. 14 *Left:* We plot (red curve) the value of the x -component u of the extracted velocity versus the standard deviation σ of the gray values to investigate the bias of the CLG approach. The theoretical devolution of the bias (black curve) depending on standard deviation matches our result very well. *Right:* The sample mean for two Monte Carlo exper-

iments, one with $N = 4$ (denoted as MC4, blue curve) and the other with $N = 50$ (denoted as MC50, red curve) is depicted. Again, the theoretical devolution of the bias depending on standard deviation is shown (black curve). Note that the values on the horizontal axis are given in percent

Fig. 15 The estimation of optical flow from the “street sequence” is shown. *Left:* Frame 110 of the sequence. *Middle left:* Optical flow of the deterministic sequence. *Right:* Optical flow for the sequence, which has been supplied with two different variances of the input gray values. Again, the mean of the optical flow is depicted in the *top row*, and the maximum entry of the covariance matrix is depicted in the *bottom row*



for noisy input signals ($\sigma^2 > 0$) has been derived in Sühling (2006). This relationship can be extended to the CLG-approach leading to a function of the expectation value depending on the variance of the input signal

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

$$\mathcal{B} = \left(\frac{\sigma^2}{\tau_u + \sigma^2} u_0, \frac{\sigma^2}{\tau_v + \sigma^2} v_0 \right).$$

Here, τ_u, τ_v denote real valued parameters that depend on the input signal. Our framework allows to easily examine the dependency of the bias on the variance of the input signal. In Fig. 14 (left) we plot the estimate of the expectation value $\mathbb{E}[\hat{u}]$ of the x component of the optical flow, using our stochastic Galerkin (SG) method, (the true underlying flow vector is $w = (u, v) = (1, 0)$) versus the standard deviation σ (in percent) of the input data. 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 theoretical curve plotted in both graphs is the same, thus enabling for a comparison of the SG results with the MC experiments.

In Fig. 14 (right) the sample mean $\bar{u} = \frac{1}{N} \sum_{j=1}^N u^{(j)}$ is depicted resulting from two Monte Carlo experiments, one with $N = 4$ samples (denoted as MC4) and the other with $N = 50$ samples (denoted as MC50). We see from the figures that the optical flow estimate decreases for increasing noise in the input image. Consequently, the bias increases for increasing noise in the input image. We can observe that the results from our approach correspond very well to the theoretical curve. In our framework it suffices to use two stochastic modes $p = 2$. In that case the effort of our approach is about $p^2 = 4$ times the effort of a deterministic CLG approach (cf. Sect. 5.6). For the

Monte Carlo approach ($N = 4$) with the same computational complexity, the sample mean shows significant fluctuations and cannot cope with the precision of our approach.

As soon as we significantly increase the number of samples ($N = 50$), we obtain a comparable result also for the Monte Carlo approach. But the price we have to pay for a likewise performance is an enormous increase of the number of estimates from 84 (21 different noise levels times an effort equivalent to 4 estimates per noise level) to 1050 (21 different noise levels times 50 estimates per noise level) estimates. With our framework the system of (37) with entries (40) must be solved only once per noise level. Thus, this approach outperforms the naive MC simulation by a very large factor. We conclude that our framework involves the potential to analyze the bias of estimators for which it is not yet known and for which the analytic derivation is cumbersome or even unfeasible.

Our final numerical experiment deals with the estimation of optical flow from the “street sequence”. Since for this artificially generated sequence we do not have any measurement errors, we set the variance of the input data homogeneously to three different values. In Fig. 15 we show the resulting optical flow field as well as the maximum entries of the covariance matrices of the distributions. The sequence is of resolution 200×200 and its gray values are scaled to the range $[0, 1]$. We use two stochastic modes $p = 2$ and set the parameters to $\kappa = 0.1, \rho = 0.1$. For a better comparison we depict the estimated optical flow from the deterministic noise free sequence ($\text{Var} \equiv 0$) as well. As in our synthetic example we see a high variance of the optic flow in regions of high gradient of the input image. Again, this is due to the nonlinear dependence of the model on the input data (cf. Sects. 4.2 and 4.4). Moreover, we see two additional effects from the images: First, due to the large smoothness coefficient κ the variance is damped in regions of nearly homogeneous gray value. Second, because of the above mentioned

bias of the optical flow estimation, the magnitude of the optical flow field decreases in particular in regions with low gradients.

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 discretization 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 various numerical experiments. For linear models there is no coupling between the modes. For the heat equation the mean mode is smoothed and the variance mode is damped. However, for nonlinear operators (gradient magnitude, edge indicator, Perona-Malik diffusion) 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 the Perona-Malik diffusion we have successfully validated our new framework against a naive Monte-Carlo simulation. Moreover, we could show experimentally that the stochastic Perona-Malik equation also has an edge enhancement property. However, this edge enhancement occurs in regions of low variance only.

As more complicated demonstrator applications we have considered various models for the estimation of optical flow. On several test sequences we have demonstrated the performance of our framework. These numerical tests give

some interesting insights and show how the various building blocks act together within the flow estimators. Moreover, we have considered the computation of the bias of the CLG method. Our framework was able to reproduce the theoretically predicted curve. In addition to that we have compared our results with a Monte-Carlo simulation for the bias estimation.

The application of the building blocks for stochastic image processing to existing well known PDE models offer some very interesting insights and raise many new questions about the modeling and the propagation of errors. For the modeling of images as random fields we have assumed independence of the random variables that steer the behavior of different pixels. In the future we will investigate how this assumption affects the various existing models in computer vision. 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 ansatz space for random images and to provide enhanced models for computer vision tasks on noisy 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 difference schemes rather than finite element schemes, profiting from well adapted convolution filters. The implementation of our framework in a stochastic Galerkin/finite difference scheme is already under investigation. This work is going to make our approach compatible with state-of-the-art finite difference discretizations.

Appendix 1: Summary of Building Blocks and Stochastic Models

We summarize all ansatz spaces, building blocks and stochastic variants of classical PDE models in computer vision. Our goal is to provide the reader with a dense and compact summary of the key ingredients of our framework in order to support an implementation and a reproduction of our results.

In the first part, we list the key notation for the ansatz spaces for static images and image sequences. Moreover, we tabulate the main formulas for the building blocks. In the second part, we give all formulas that are relevant for the implementation of the stochastic analogs of the computer vision models we have discussed. There, we summarize per model, which building blocks are used, and how the resulting linear systems of equations are constituted.

A.1 Building Blocks

Ansatz Spaces

<p>Static images Sect. 2.2</p>	<ul style="list-style-type: none"> • Spatial domain $D = [1, N_1] \times [1, N_2]$ • Nodes/pixels of static image $x_i, i \in \mathcal{I} = \{1, \dots, M\}$ • Physical basis functions $P_i \in C^0(D)$ with $P_i(x_j) = \delta_{ij}$, for all $i, j \in \mathcal{I}$ and such that $P_i _E$ is bilinear on each grid cell • Stochastic domain $\Gamma_* = [-1, 1]$ • Random variables $\xi = (\xi_1, \dots, \xi_M)$. PDFs $\rho_i(\xi_i) = \frac{1}{2}$ • Indexing of stochastic modes $\alpha = 1, \dots, p$ • Stochastic basis functions $H^\alpha : [-1, 1] \rightarrow \mathbb{R}$, e.g. Legendre Polynomials on $[-1, 1]$ • Random image $f(x, \xi) = \sum_{i \in \mathcal{I}} \sum_{\alpha=1}^p f_\alpha^i H^\alpha(\xi_i) P_i(x)$
<p>Image sequences Sect. 2.3</p>	<ul style="list-style-type: none"> • Spatio-temporal domain $R := D \times I := ([1, N_1] \times [1, N_2]) \times [1, N_t]$ • Nodes/pixels of image sequence $y_i, i = (i_x, i_t) \in \mathcal{J} = \{1, \dots, M\} \times \{1, \dots, N_t\}$ • Physical basis functions $P_i \in C^0(R)$ with $P_i(y_j) = \delta_{ij} = \delta_{i_x j_x} \delta_{i_t j_t}$, for all $i, j \in \mathcal{J}$ and such that $P_i _E$ is trilinear on each grid cell • Stochastic domain $\Gamma_* = [-1, 1]$ • Random variables $\xi = (\xi_1, \dots, \xi_M)$. PDFs $\rho_i(\xi_i) = \frac{1}{2}$ • Indexing of stochastic modes $\alpha = 1, \dots, p$ • Stochastic basis functions $H^\alpha : [-1, 1] \rightarrow \mathbb{R}$, e.g. Legendre Polynomials on $[-1, 1]$ • Random image sequence $f(y, \xi) = \sum_{i \in \mathcal{J}} \sum_{\alpha=1}^p f_\alpha^i H^\alpha(\xi_{i_x}) P_i(y)$

For image sequences the temporal behavior of the stochastic processes is modeled through a time-dependence of the coefficients $f_\alpha^i = f_\alpha^{i_x}(t)$

Moment Evaluation

<p>Expectations of products of stochastic basis functions (18), Sect. 3.6.1</p>	<ul style="list-style-type: none"> • $A_{\alpha,\beta,\gamma,\delta}^{i,j,k,l} = \int_\Gamma H^\alpha(\xi_i) H^\beta(\xi_j) H^\gamma(\xi_k) H^\delta(\xi_l) \rho(\xi) d\xi$ • $B_{\alpha,\beta,\gamma}^{i,j,k} = \int_\Gamma H^\alpha(\xi_i) H^\beta(\xi_j) H^\gamma(\xi_k) \rho(\xi) d\xi$ • $C_{\alpha,\beta}^{i,j} = \int_\Gamma H^\alpha(\xi_i) H^\beta(\xi_j) \rho(\xi) d\xi$
<p>Mean (8), Sect. 3.1</p>	$E[f](x) = \sum_{i \in \mathcal{I}} f_1^i P_i(x) \quad (\text{first mode})$
<p>Variance (9), Sect. 3.1</p>	$\text{Var}[f](x) = \sum_{i \in \mathcal{I}} \sum_{\alpha=2}^p C_{i,i}^{\alpha,\alpha} (f_\alpha^i)^2 P_i(x)$
<p>Covariance (10), Sect. 3.1</p>	$\text{Cov}[f, g](x) = \sum_{i \in \mathcal{I}} \sum_{\alpha=2}^p C_{i,i}^{\alpha,\alpha} f_\alpha^i g_\alpha^i P_i(x)$

Formulas for image sequences: replace index \mathcal{I} with \mathcal{J} , and x with y

Stochastic Galerkin Method

<p>Projection (evaluation of nonlinear function), Sect. 3.2</p>	<p>Coefficients $G = (g_\alpha^i)_{\alpha,i}$, result from matrix-vector multiplication $G = \tilde{M}^{-1} R$ with</p> $R = \left(\mathbb{E} \left[\int_D g(u)(x, \xi) H^\beta(\xi_j) P_j(x) dx \right] \right)_{j,\beta} \quad \text{and} \quad (\tilde{M}^{\alpha,\beta})_{ij} = \delta_{i,j} \delta_{\alpha,\beta} \sum_{k \in \mathcal{I}} \sum_{\gamma=1}^p C_{i,k}^{\alpha,\gamma} \int_D P_i(x) P_k(x) dx$
<p>Gradient magnitude, Sect. 3.3</p>	<p>Coefficients $g_\alpha^i = (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} U \cdot K_{(i,\alpha)} U$ where $((K_{(i,\alpha)})^{\beta,\gamma})_{j,k} = B_{i,j,k}^{\alpha,\beta,\gamma} \int_D P_i(x) \nabla P_j(x) \cdot \nabla P_k(x) dx$</p>
<p>Edge indicator function, Sect. 3.4</p>	<p>Coefficients $g_\alpha^i = (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} \int_\Gamma \int_D \frac{H^\alpha(\xi_i) P_i(x)}{1 + \lambda^{-2} \sum_{j \in \mathcal{I}} \sum_{\beta=1}^p V_\beta^j H^\beta(\xi_j) P_j(x)} dx \rho(\xi) d\xi$</p>
<p>Structure tensor (component (a, b)), Sect. 3.5</p>	<p>Coefficients $g_\alpha^i = (\tilde{M}^{\alpha,\alpha})_{ii}^{-1} U \cdot K_{(i,\alpha)} U$ where $((K_{(i,\alpha)})^{\beta,\gamma})_{j,k} = B_{i,j,k}^{\alpha,\beta,\gamma} \int_D P_i(x) \partial_a P_j(x) \partial_b P_k(x) dx$</p>

Integrals resulting from the projection must be evaluated numerically. Only in very few cases analytical expressions for coefficients can be derived

A.2 Diffusion and Optical Flow Estimation

Stochastic Heat Equation (Linear Diffusion), Sect. 4.1

Required building blocks	<ul style="list-style-type: none"> • Random Image Ansatz Space • Tensor components $C_{\alpha,\beta}^{i,j}$ for $\alpha = \beta$
Input	<ul style="list-style-type: none"> • p modes of the distribution of initial image $u^0(x, \xi)$
Linear system of equations $\alpha, \beta = 1, \dots, p$	$\sum_{\alpha=1}^p (M^{\alpha,\beta} + \tau L^{\alpha,\beta})(U^n)_\alpha = \sum_{\alpha=1}^p M^{\alpha,\beta} (U^{n-1})_\alpha$ $(M^{\alpha,\beta})_{ij} = C_{\alpha,\beta}^{i,j} \left(\int_D P_i(x) P_j(x) dx \right)$ $(L^{\alpha,\beta})_{ij} = C_{\alpha,\beta}^{i,j} \left(\int_D \nabla P_i(x) \cdot \nabla P_j(x) dx \right)$
Output	<ul style="list-style-type: none"> • p modes of the distribution of smoothed image $u^n(x, \xi)$ for each scale step n

Stochastic Perona Malik Model (Nonlinear Diffusion), Sect. 4.2

Required building blocks	<ul style="list-style-type: none"> • Random Image Ansatz Space • Tensor components $B_{\alpha,\beta,\gamma}^{i,j,k}$ • Edge Indicator Function $G = (g_\alpha^i)$ • (Stochastic Heat Equation for Regularization of Gradient Image for Edge Indicator)
Input	<ul style="list-style-type: none"> • p modes of the distribution of initial image $u^0(x, \xi)$
Linear system of equations $\alpha, \beta = 1, \dots, p$	$\sum_{\alpha=1}^p (M^{\alpha,\beta} + \tau (L^n)^{\alpha,\beta})(U^n)_\alpha = \sum_{\alpha=1}^p M^{\alpha,\beta} (U^{n-1})_\alpha$ $(M^{\alpha,\beta})_{ij} = C_{\alpha,\beta}^{i,j} \left(\int_D P_i(x) P_j(x) dx \right)$ $(L^n)^{\alpha,\beta}_{ij} = \sum_{\gamma=1}^p \sum_{k \in \mathcal{I}} (G^n)_\gamma^k B_{\alpha,\beta,\gamma}^{i,j,k} \left(\int_D \nabla P_i(x) \cdot \nabla P_j(x) P_k(x) dx \right)$
Output	<ul style="list-style-type: none"> • p modes of the distribution of smoothed image $u^n(x, \xi)$ for each scale step n

Stochastic Optical Flow Estimation (Stochastic Horn & Schunck Model), Sect. 5.2

Required building blocks	<ul style="list-style-type: none"> • Random Image Sequence Ansatz Space • Tensor components $A_{\alpha,\beta,\gamma,\delta}^{i_x,j_x,k_x,l_x}$ and $B_{\alpha,\beta,\gamma}^{i_x,j_x,k_x}$ • (Edge Indicator Function $G = (g_\alpha^i)$)
Input	<ul style="list-style-type: none"> • p modes of the distribution of the image sequence $f(y, \xi)$
Linear system of equations $\alpha, \beta = 1, \dots, p$	$\begin{pmatrix} S_{11}^{\alpha,\beta} & S_{12}^{\alpha,\beta} \\ S_{21}^{\alpha,\beta} & S_{22}^{\alpha,\beta} \end{pmatrix} + \kappa \begin{pmatrix} L^{\alpha,\beta} & 0 \\ 0 & L^{\alpha,\beta} \end{pmatrix} \begin{pmatrix} U^\alpha \\ V^\alpha \end{pmatrix} = - \begin{pmatrix} R_1^\beta \\ R_2^\beta \end{pmatrix}$ $(S_{mn}^{\alpha,\beta})_{ij} = \sum_{k,l \in \mathcal{J}} \sum_{\gamma,\delta=1}^p f_\gamma^k f_\delta^l A_{\alpha,\beta,\gamma,\delta}^{i_x,j_x,k_x,l_x} \int_R P_i(y) P_j(y) \partial_m P_k(y) \partial_n P_l(y) dy$ $(R_m^\beta)_j = \sum_{k,l \in \mathcal{J}} \sum_{\gamma,\delta=1}^p f_\gamma^k f_\delta^l B_{\beta,\gamma,\delta}^{j_x,k_x,l_x} \int_R P_j(y) \partial_m P_k(y) \partial_l P_l(y) dy,$ <p>$L^{\alpha,\beta}$ as for stochastic heat equation</p>
Output	<ul style="list-style-type: none"> • p modes of the distribution of the x- and y-component of the optical flow field $w(y, \xi)$

Stochastic Optical Flow Estimation (CLG Approach), Sect. 5.5

Required building blocks	<ul style="list-style-type: none"> • Random Image Sequence Ansatz Space • Tensor components $A_{\alpha,\beta,\gamma,\delta}^{i_x,j_x,k_x,l_x}$ and $B_{\alpha,\beta,\gamma}^{i_x,j_x,k_x}$ • Structure Tensor Components $J^{m,t}$ • Stochastic Heat Equation for smoothing of structure tensor $J_{\rho}^{m,t}$
Input	<ul style="list-style-type: none"> • p modes of the distribution of the image sequence $f(y, \xi)$
Linear system of equations $\alpha, \beta = 1, \dots, p$	<p>Linear System as for Horn and Schunck model</p> $(S_{mn}^{\alpha,\beta})_{ij} = \sum_{k \in \mathcal{J}} \sum_{\gamma=1}^p (J_{\rho}^{mn})_{\gamma}^k B_{\alpha,\beta,\gamma}^{i_x,j_x,k_x} \int_R P_i(y) P_j(y) P_k(y) dy,$ $(R_m^{\beta})_j = \sum_{k \in \mathcal{J}} \sum_{\gamma=1}^p (J_{\rho}^{m,t})_{\gamma}^k C_{\beta,\gamma}^{j_x,k_x} \int_R P_j(y) P_k(y) dy$ <p>$L^{\alpha,\beta}$ as for stochastic heat equation</p>
Output	<ul style="list-style-type: none"> • p modes of the distribution of the x- and y-component of the optical flow field $w(y, \xi)$

References

- Amiaz, T., & Kiryati, N. (2006). Piecewise-smooth dense optical flow via level sets. *International Journal of Computer Vision*, 68(2), 111–124.
- Avriel, M. (2003). *Nonlinear programming: Analysis and methods*. New York: Dover.
- Bao, Y., & Krim, H. (2004). Smart nonlinear diffusion: A probabilistic approach. *Pattern Analysis and Machine Intelligence*, 26(1), 63–72.
- Black, M. J., & Anandan, P. (1991). Robust dynamic motion estimation over time. In *Proc. computer vision and pattern recognition, CVPR-91* (pp. 296–302), June 1991.
- Black, M. J., & Anandan, P. (1993). A framework for the robust estimation of optical flow. In *Proc. ICCV93* (pp. 231–236).
- Bruhn, A., Weickert, J., & Schnörr, C. (2002). Combining the advantages of local and global optic flow methods. In *Proc. DAGM* (pp. 454–462).
- Bruhn, A., Weickert, J., & Schnörr, C. (2005). Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3), 211–231.
- Catté, F., Lions, P.-L., Morel, J.-M., & Coll, T. (1992). Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical Analysis*, 29(1), 182–193.
- Chorin, A. J. (1971). Hermite expansions in Monte Carlo computation. *Journal of Computational Physics*, 8, 471–482.
- Chorin, A. J. (1974). Gaussian fields and random flow. *Journal of Fluid Mechanics*, 63, 21–32.
- Cohen, I. (1993). Nonlinear variational method for optical flow computation. In *SCIA93* (pp. 523–530).
- de Laplace, P. S. (1812). *Théorie analytique des probabilités*. Paris: Courcier Imprimeur.
- Deb, M. K., Babuška, I. M., & Oden, J. T. (2001). Solutions of stochastic partial differential equations using Galerkin finite element techniques. *Computer Methods in Applied Mechanics Engineering*, 190, 6359–6372.
- Fermüller, C., Shulman, D., & Aloimonos, Y. (2001). The statistics of optical flow. *Computer Vision and Image Understanding*, 82(1), 1–32.
- Forsyth, D. A., & Ponce, J. (2003). *Computer vision: a modern approach*. Englewood Cliffs: Prentice Hall.
- Ghanem, R. G. (1999). Higher order sensitivity of heat conduction problems to random data using the spectral stochastic finite element method. *ASME Journal of Heat Transfer*, 121, 290–299.
- Ghanem, R. G., & Spanos, P. (1991). *Stochastic finite elements: a spectral approach*. New York: Springer.
- Haussecker, H., & Spies, H. (1999). Motion. In B. Jähne, H. Haußecker, & P. Geißler (Eds.), *Handbook of computer vision and applications* (pp. 309–396). San Diego: Academic Press.
- Haussecker, H., Spies, H., & Jähne, B. (1998). 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 VIII*, Hakodate, Japan, June 1998.
- Horn, B. K. P., & Schunck, B. (1981). Determining optical flow. *Artificial Intelligence*, 17, 185–204.
- Van Huffel, S., & Vandewalle, J. (1991). *Frontiers in applied mathematics: Vol. 9. The total least squares problem: Computational aspects and analysis*. Philadelphia: SIAM.
- Iijima, T. (1962). Basic theory on normalization of pattern (in case of typical one-dimensional pattern). *Bulletin of the Electrotechnical Laboratory*, 26, 368–388 (in Japanese).
- Iijima, T. (1963). Theory of pattern recognition. *Electronics and Communications in Japan* (pp. 123–134).
- Jähne, B. (1993). *Spatio-temporal image processing: Theory and scientific applications. Lecture notes in computer science*. Berlin: Springer.
- Kearney, J. K., Thompson, W. B., & Boley, D. L. (1987). Optical flow estimation: An error analysis of gradient-based methods with local optimization. *PAMI*, 9(2), 229–244.
- Keese, A. (2004). *Numerical solution of systems with stochastic uncertainties: A general purpose framework for stochastic finite elements*. Ph.D. thesis, Technical University Braunschweig.
- Kichenassamy, S. (1997). The Perona-Malik paradox. *SIAM Journal on Applied Mathematics*, 57(5), 1328–1342.
- Lucas, B., & Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *DARPA image understanding workshop* (pp. 121–130).
- Lucor, D., Su, C.-H., & Karniadakis, G. E. (2004). Generalized polynomial chaos and random oscillators. *International Journal for Numerical Methods in Engineering*, 60, 571–596.
- Le Maître, O. P., Reagan, M., Najm, H. N., Ghanem, R. G., & Knio, O. M. (2002). A stochastic projection method for fluid flow

- II: random process. *Journal of Computational Physics*, 181(1), 9–44.
- Malliavin, P. (1997). *Stochastic analysis*. New York: Springer.
- Maltz, F. H., & Hitzl, D. L. (1979). Variance reduction in Monte Carlo computations using multi-dimensional Hermite polynomials. *Journal of Computational Physics*, 32, 345–376.
- Meecham, W. C., & Jeng, D. T. (1968). Use of Wiener-Hermite expansion for nearly normal turbulence. *Journal of Fluid Mechanics*, 32, 225–249.
- Mikula, K., Preusser, T., & Rumpf, M. (2004). Morphological image sequence processing. *Computing and Visualization in Science*, 6(4), 197–209.
- Narayanan, V. A., & Zabaras, N. (2004). Stochastic inverse heat conduction using a spectral approach. *International Journal for Numerical Methods in Engineering*, 60, 1569–1593.
- Nestares, O., & Fleet, D. J. (2003). Error-in-variables likelihood functions for motion estimation. In *IEEE international conference on image processing (ICIP)* (Vol. III, pp. 77–80). Barcelona.
- Nestares, O., Fleet, D. J., & Heeger, D. (2000). Likelihood functions and confidence bounds for total-least-squares problems. In *CVPR'00* (Vol. 1).
- Papenberg, N., Bruhn, A., Brox, T., Didas, S., & Weickert, J. (2006). Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision*, 67(2), 141–158.
- Perona, P., & Malik, J. (1990). Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, 629–639.
- Preusser, T., & Rumpf, M. (1999). An adaptive finite element method for large scale image processing. In *Proceedings scale-space '99, scale space theories in computer vision, second international conference* (pp. 223–234).
- Reagan, M. T., Najm, H. N., Debusschere, B. J., Le Maître, O. P., Knio, O. M., & Ghanem, R. G. (2004). Spectral stochastic uncertainty quantification in chemical systems. *Combustion Theory and Modelling*, 8, 607–632.
- Reagan, M. T., Najm, H. N., Pebay, P. P., Knio, O. M., & Ghanem, R. G. (2005). Quantifying uncertainty in chemical systems modeling. *International Journal of Chemical Kinetics*, 37, 386–382.
- Scharr, H. (2006). Diffusion-like reconstruction schemes from linear data models. In *Lecture notes in computer science: Vol. 4174. Pattern recognition 2006* (pp. 51–60). Berlin: Springer.
- Scharr, H., Black, M. J., & Haussecker, H. W. (2003). Image statistics and anisotropic diffusion. In *Int. conf. on computer vision, ICCV 2003* (pp. 840–847), Nice, France.
- Sühling, M. (2006). *Myocardial motion and deformation analysis from echocardiograms*. Ph.D. thesis, Swiss Federal Institute of Technology Lausanne (EPFL), July 2006.
- Thomee, V. (1984). *Galerkin—finite element methods for parabolic problems*. New York: Springer.
- Gauss, C. F. (1987). *Theory of the combination of observations least subject to errors, part one and part two. Supplement (Classics in Applied Mathematics 11)* (trans: Stewart, G. W.). Society for Industrial Mathematics, Facsimile edition. English version in 1987. Original version in Latin in 1820s.
- Weber, J., & Malik, J. (1994). Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14(1), 5–19.
- Weickert, J. (1998). On discontinuity-preserving optic flow. In *Proc. computer vision and mobile robotics workshop* (pp. 115–122).
- Weickert, J., & Schnörr, C. (2001). Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3), 245–255.
- Wiener, N. (1938). The homogeneous chaos. *American Journal of Mathematics*, 60(4), 897–936.
- Witkin, A. P. (1983). Scale-space filtering. In *Proc. eighth int. joint conf. on artificial intelligence (IJCAI)* (Vol. 2, pp. 1019–1022).
- Xiu, D. B., & Karniadakis, G. E. (2002a). Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos. *Computational Methods in Applied Mechanics and Engineering*, 191, 4927–4948.
- Xiu, D. B., & Karniadakis, G. E. (2002b). The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24, 619–644.
- Xiu, D. B., & Karniadakis, G. E. (2003a). Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of Computational Physics*, 187, 137–167.
- Xiu, D. B., & Karniadakis, G. E. (2003b). A new stochastic approach to transient heat conduction modeling with uncertainty. *International Journal of Heat and Mass Transfer*, 46, 4681–4693.