# Characterizing the Structure Tensor Using Gamma Distributions

Oskarsson, Magnus

*Published in:*
Pattern Recognition (ICPR), 2016 23rd International Conference on

*DOI:*
10.1109/ICPR.2016.7899727

2017

*Document Version:*
Publisher's PDF, also known as Version of record

Link to publication

*Citation for published version (APA):*
Oskarsson, M. (2017). Characterizing the Structure Tensor Using Gamma Distributions. In *Pattern Recognition (ICPR), 2016 23rd International Conference on* (pp. 763-768). IEEE - Institute of Electrical and Electronics Engineers Inc.. https://doi.org/10.1109/ICPR.2016.7899727

*Total number of authors:*
1

# Characterizing the Structure Tensor Using Gamma Distributions

Magnus Oskarsson
Centre for Mathematical Sciences, Lund University, Lund, Sweden
Email: magnuso@maths.lth.se

***Abstract*—The structure tensor is a powerful tool describing the local intensity structure of an image or image sequence. In this paper we give a model for the noise distribution of the components of the tensor. In order to do so we have also investigated some properties of the gamma distribution. We show that, given an input image corrupted with Gaussian noise, the noise in the structure tensor can be modeled well by gamma distributions. We apply our model to automatic contrast enhancement of images taken under poor illumination. We show how our noise model can be used for automatic parameter selection in the filtering process, giving powerful results without the need for cumbersome parameter tuning.**

## I. Introduction

We will in this paper investigate the statistical properties of the so-called structure tensor. The *structure tensor* or *second moment matrix* is used to efficiently capture the local gradient structure in an image or image sequence. Methods using this matrix have been developed and applied in image analysis in numerous papers [1], [2], [3] and was e.g. a central tool in early feature extraction methods, [4], [5]. Our goal in this paper is to characterize some of the statistical properties of the structure tensor. This gives us tools to analyze properties of functions of the structure tensor, such as the determinant and the Eigenvalues. This is important in many applications, and we will especially show how this can be used for automatic parameter selection.

The paper is organized as follows. We will in section II describe a number of known, and some new results relating to the gamma distribution. Then, in section III we will use the properties of the gamma distribution to characterize the noise distribution of the different components of the structure tensor. We will then test our models on an example application – in section IV we investigate how we can automatically set parameters in a structure adaptive image enhancement method. This example is merely used to illustrate the modeling and we do not claim that these methods necessarily increase state-of-the-art. But we argue that given these methods, the modeling of the tensor components gives us a powerful tool for parameter estimation.

## II. The Gamma Distribution

We will in this section describe some well known properties of the Gamma distribution as well as some new, that we will use in the following sections to characterize the statistical properties of the structure tensor. For more information on the gamma distribution see e.g. [6], [7].

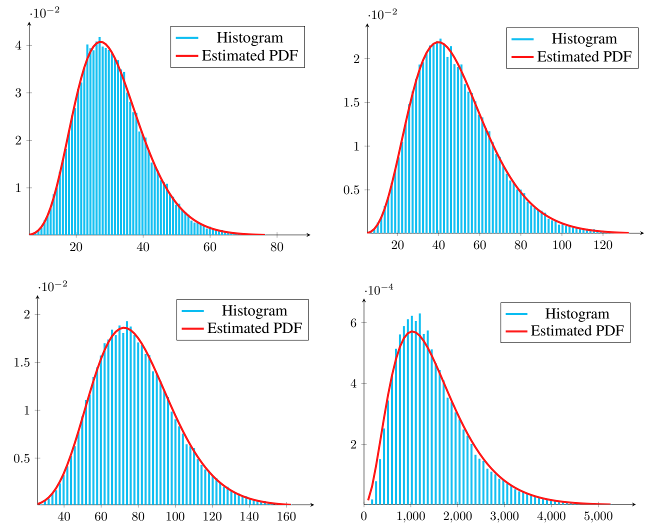


Fig. 1. Top row: The distributions of two input gamma distributed variables. The histograms of the random data as well as the theoretical PDFs are shown. In the bottom the sum and the product of the two datasets are shown to the left and right respectively. The model distribution PDFs are overlaid.

The probability density function, PDF, of a Gamma distribution, $\mathrm{Gamma}(k, \Theta)$, is given by

$$p(x) = \frac{x^{k-1} e^{-\frac{x}{\Theta}}}{\Theta^k \Gamma(k)}, \quad x \geq 0, \tag{1}$$

where $\Gamma(k)$ is the Gamma function. It has two parameters, the shape parameter $k$ and the scale parameter $\Theta$ that control the properties of the distribution. The mean of the distribution is given by $m = k\Theta$ and the variance is given by $v = k\Theta^2$. The shape parameter and the scale parameter can in turn be calculated from the mean and the variance according to $k = m^2/v$ and $\Theta = v/m$. This means that the mean and variance completely describe the distribution. The distribution has a number of other properties that follow from the definition. Multiplying a gamma distributed stochastic variable with a scalar gives again a gamma distributed stochastic variable:

$$X \in \mathrm{Gamma}(k, \Theta) \Rightarrow cX \in \mathrm{Gamma}(k, c\Theta). \tag{2}$$

If we have a number of independent stochastic variables with the *same scale parameters* the sum is gamma distributed:

$$X_i \in \mathrm{Gamma}(k_i, \Theta), X = \sum_i X_i \Rightarrow \tag{3}$$

$$X \in \mathrm{Gamma}(\sum k_i, \Theta). \tag{4}$$

One may ask what happens if the summed distributions do not have the same scale parameter. In this case the result will not be gamma distributed. In [8], [9] it is shown that the probability distribution is given by an infinite linear combination of gamma distributions. Let

$$X_i \in \operatorname{Gamma}(k_i, \Theta_i), \quad X = \sum_i X_i, \tag{5}$$

then the probability distribution of $X$ is given by

$$p(x) = C \sum_{j=0}^{\infty} \frac{\delta_j x^{\rho+j-1} e^{-x/\Theta_1}}{\Gamma(\rho+j)\Theta_1^{\rho+j}}, \tag{6}$$

with

$$\rho = \sum_{i=1}^{n}, \quad C = \prod_{i=1}^{n} (\frac{\Theta_1}{\Theta_i})^{k_i}, \tag{7}$$

and

$$\delta_j = \frac{1}{j+1} \sum_{i=1}^{j+1} i \gamma_i \delta_{j+1-i}, \quad \delta_0 = 1, \tag{8}$$

$$\gamma_j = \sum_{i=1}^{n} k_i \frac{(1 - \frac{\Theta_1}{\Theta_i})^j}{j}. \tag{9}$$

We will in this paper work with the hypothesis that both the sum and product of two or more gamma distributed stochastic variables (with arbitrary parameters) can be well *approximated* by a gamma distribution.

**Conjecture 1.** *If* $X_1 \in \operatorname{Gamma}(k_1, \Theta_1)$ *and* $X_2 \in \operatorname{Gamma}(k_2, \Theta_2)$ *are independent then* $X = X_1 + X_2$ *is approximately gamma distributed with parameters*

$$k_s = \frac{(k_1\Theta_1 + k_2\Theta_2)^2}{k_1\Theta_1^2 + k_2\Theta_2^2} \quad \text{and} \quad \Theta_s = \frac{k_1\Theta_1^2 + k_2\Theta_2^2}{k_1\Theta_1 + k_2\Theta_2}. \tag{10}$$

If we have a sum of more than two variables we get

**Corollary 1.** *If* $X_i \in \operatorname{Gamma}(k_i, \Theta_i)$ *are independent then* $X = \sum_i X_i$ *is approximately gamma distributed with parameters*

$$k_s = \frac{(\sum_i k_i\Theta_i)^2}{\sum_i k_i\Theta_i^2} \quad \text{and} \quad \Theta_s = \frac{\sum_i k_i\Theta_i^2}{\sum_i k_i\Theta_i}. \tag{11}$$

For two independent variables the mean of their product is given by the product of their means. The variance of the product is given by the formula:

$$\operatorname{Var}(XY) = \operatorname{E}[X]^2 \operatorname{Var}(Y) + \operatorname{E}[Y]^2 \operatorname{Var}(X) + \tag{12}$$

$$\operatorname{Var}(X)\operatorname{Var}(Y). \tag{13}$$

If we assume that the product of two gamma distributed variables is gamma distributed we can use this fact to estimate the parameters of the resulting distribution and we get the following conjecture:

**Conjecture 2.** *If* $X_1 \in \operatorname{Gamma}(k_1, \Theta_1)$ *and* $X_2 \in \operatorname{Gamma}(k_2, \Theta_2)$ *are independent then* $X = X_1X_2$ *is approximately gamma distributed with parameters*

$$k_p = \frac{(k_1\Theta_1 \cdot k_2\Theta_2)^2}{(k_1\Theta_1)^2 \cdot k_2\Theta_2^2 + (k_2\Theta_2)^2 \cdot k_1\Theta_1^2 + k_1\Theta_1^2 \cdot k_2\Theta_2^2}, \tag{14}$$

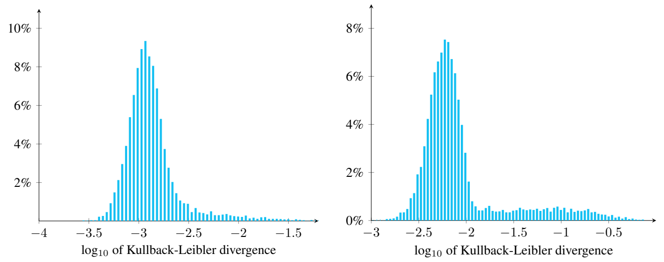


Fig. 2. Histogram of the logarithm of the Kullback-Leibler divergence between data histograms and model distribution. To the left is the result for the sum of two variables and to the right for the product.

*and*

$$\Theta_p = \frac{(k_1\Theta_1)^2 \cdot k_2\Theta_2^2 + (k_2\Theta_2)^2 \cdot k_1\Theta_1^2 + k_1\Theta_1^2 \cdot k_2\Theta_2^2}{k_1\Theta_1 \cdot k_2\Theta_2}. \tag{15}$$

For a product of more than two variables we can write this in the following form:

**Corollary 2.** *If* $X_i \in \operatorname{Gamma}(k_i, \Theta_i)$ *are independent then* $X = \prod_i X_i$ *is approximately gamma distributed with parameters*

$$v_p = \prod_i (\operatorname{Var}(X_i) + \operatorname{E}[X_i]^2) - \prod_i \operatorname{E}[X_i]^2, \tag{16}$$

$$m_p = \prod_i X_i, \quad k_p = \frac{m_p^2}{v_p}, \quad \Theta_p = \frac{v_p}{m_p}. \tag{17}$$

To test our hypotheses we did the following test. We generated a large set of numbers drawn from two different gamma distributions. This gives us two vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$ where $x_i \in \operatorname{Gamma}(k_1, \Theta_1)$ and $y_i \in \operatorname{Gamma}(k_2, \Theta_2)$. We chose $k$ and $\Theta$ randomly and set $n = 100\ 000$ in our experiments. We took the element-wise sum and product of $x$ and $y$ and calculated the histograms of these vectors as an estimate of the probability distribution. We then compared this with the proposed gamma distributions. An example can be seen in figure 1, where the distributions of $x$, $y$, $x+y$ and $xy$ are shown. We repeated the experiment a number of times and calculated the Kullback-Leibler divergence [10], between the estimated histogram and the estimated gamma distribution. The result can be seen in figure 2 where a histogram of the logarithm of the Kullback-Leibler divergence is shown for the sum and product respectively. One can see that we get consistently small deviations, leading to the conclusion that the model distributions follow the data well.

A special case for the sum of two gamma distributed variables is when they have different scale parameters but the *same shape* parameters. This situation occurs when we take a linear combination of two equally distributed variables,

$$X_1 \in \operatorname{Gamma}(k, \Theta), X_2 \in \operatorname{Gamma}(k, \Theta), \tag{18}$$

$$X = aX_1 + bX_2 \Rightarrow X = Y_1 + Y_2, \tag{19}$$

$$Y_1 \in \operatorname{Gamma}(k, a\Theta), Y_2 \in \operatorname{Gamma}(k, b\Theta). \tag{20}$$

In this case we can actually calculate the exact distribution of the sum in a more simple form than the one given by

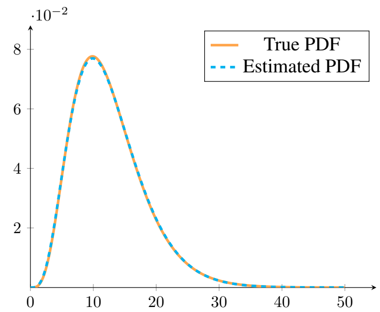


Fig. 3. The figure shows the model distribution and the exact theoretical distribution, for the sum of two input distributions with the same shape parameters.

equation (6). This is simply done by convolving the two gamma probability distribution functions. This leads to an expression that can be written as a sum of two addends involving Whittaker functions. In figure 3 the sum of two gamma distributed variables with the same shape parameter is shown. One can see that the estimated gamma distribution follows the exact distribution closely.

## III. The Probability Distribution of the Structure Tensor

We will now turn our attention to the statistical properties of the structure tensor, building upon the discussion in the previous section. Given an input image $I$ the structure tensor at a point $\mathbf{x_0}$ is defined in the following way:

$$J_\rho(\mathbf{x_0}) = G_\rho \star (\nabla I(\mathbf{x_0}) \nabla I(\mathbf{x_0})^T) \equiv \begin{bmatrix} j_{11}(\mathbf{x_0}) & j_{12}(\mathbf{x_0}) \\ j_{12}(\mathbf{x_0}) & j_{22}(\mathbf{x_0}) \end{bmatrix}, \quad (21)$$

where

$$\nabla I(\mathbf{x_0}) = \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{bmatrix}_{\mathbf{x}=\mathbf{x_0}}, \quad (22)$$

is the intensity gradient of $I$ at the point $\mathbf{x_0}$. Basically it is the outer product of the gradient, but additionally filtered. Here $G_\rho$ is the Gaussian kernel function

$$G_\rho(x,y) = \frac{1}{2\pi\rho^2} e^{-\frac{1}{2}(\frac{x^2+y^2}{\rho^2})} \quad . \quad (23)$$

The notation $\star$ means element-wise convolution of the matrix $\nabla I(\mathbf{x_0}) \nabla I(\mathbf{x_0})^T$ in a neighborhood centered at $\mathbf{x_0}$. This convolution gives a smoothing in the space of gradients which leads to much of the robustness of the representation. We will now try to characterize the noise distribution of the different elements of $J_\rho$. The calculation of the elements involve three basic steps. First we differentiate the image in the $x$ and $y-$direction giving us the gradient. This is done by linearly filtering the image with a differentiation kernel. Then we calculate the outer product of the gradient. This is a non-linear operation on the gradient components. In the third and last step we smooth the components, and this is again a linear filtering, in this case with a Gaussian smoothing kernel.

We will assume that the image is corrupted by normally distributed noise, i.e. the measured image $I$ is given by

$$I(\mathbf{x}) = I_0(\mathbf{x}) + e(\mathbf{x}), \quad e(\mathbf{x}) \in N(0, \sigma^2). \quad (24)$$

We will further look at points where $I_0$ is varying slowly so that $\nabla I(\mathbf{x}) \approx \nabla e(\mathbf{x})$.

*a) Step one:* The noise of the entries of the gradient will be normally distributed. This is due to the fact that we apply a linear filter. The mean of the gradient will be zero, and the variance will be given by the sum of the squared entries of the filter times the initial error variance $\sigma^2$, i.e. $\in N(0, d \cdot \sigma^2)$.

*b) Step two:* Here we want to characterize the noise of $\frac{\partial I}{\partial x}^2, \frac{\partial I}{\partial y}^2$ and $\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$. The first two are squares of something that is normally distributed. This gives $\chi^2$-distributed error, which also can be described as a gamma distribution with shape parameter equal to 0.5 and scale parameter $2d \cdot \sigma^2$. The noise in $\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$ will follow a normal product distribution, with zero mean and variance $d^2\sigma^4$. The PDF of a normal product distribution is given by a modified Bessel function. In the top of figure 4 these errors are shown for a synthetically generated example. The estimated PDFs follow the histograms of the noise well.

*c) Step three:* This step in the calculation of the structure tensor is the Gaussian smoothing of the tensor components. This is a linear filtering and we will make use of our model hypothesis from the previous section. This means that the noise in $j_{11}$ and $j_{22}$ will follow gamma distributions. We will have to be a bit careful when we calculate the parameters of these distributions, since they are not entirely independent. This is due to the previous differential filtering. The mean will not change, $(d\sigma^2)$, and the variance will be equal to $d_{11}\sigma^4$, where $d_{11}$ only depends on the differential filter and the Gaussian smoothing filter. We will model the noise in $j_{12}$ with a Gaussian distribution with zero mean. Again we have to be a bit careful when we estimate the variance, due to the covariance between pixels. However the variance will again be proportional to $\sigma^4$, i.e. $\in N(0, d_{12}\sigma^4)$. In Figure 4 the estimated distributions of the structure tensor is shown for a synthetic example. In figure 5 the same is shown for a real image with added Gaussian noise. One can see that in this case the error distributions follow the model.

## IV. Application: Structure Adaptive Image Enhancement

The structure tensor has been used in a number of denoising methods. We will in this section investigate how we can use the noise modeling of the structure tensor, in order to set parameters for image enhancement in an automatic way. We will specifically test our parameter setting for contrast enhancement of images taken under dimlight conditions. In order to do this we need to increase the contrast in the image, which will inevitably increase the visible noise in the image. This means that we need to denoise the image. There are many ways to increase the contrast of an image; a classic way is to do histogram equalization. We will rather look at the problem as a tone mapping problem. Tone mapping is used to map a high dynamic range of gray values to a low dynamic range. This is often needed to display high dynamic range images on low dynamic range displays. For an overview of tone mapping methods see [11]. We will increase the dynamic range of

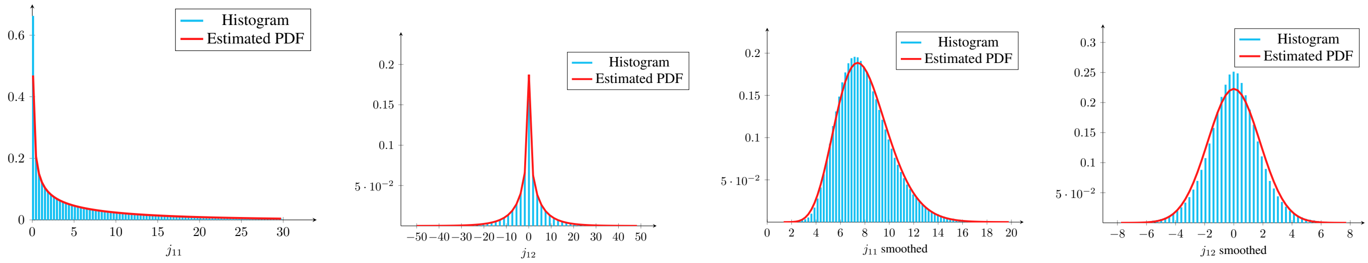


Fig. 4. The figure shows histograms of data and model distributions. From left to rigth: the unsmoothed $j_{11}$ and $j_{12}$, the smoothed $j_{11}$ and $j_{12}$.

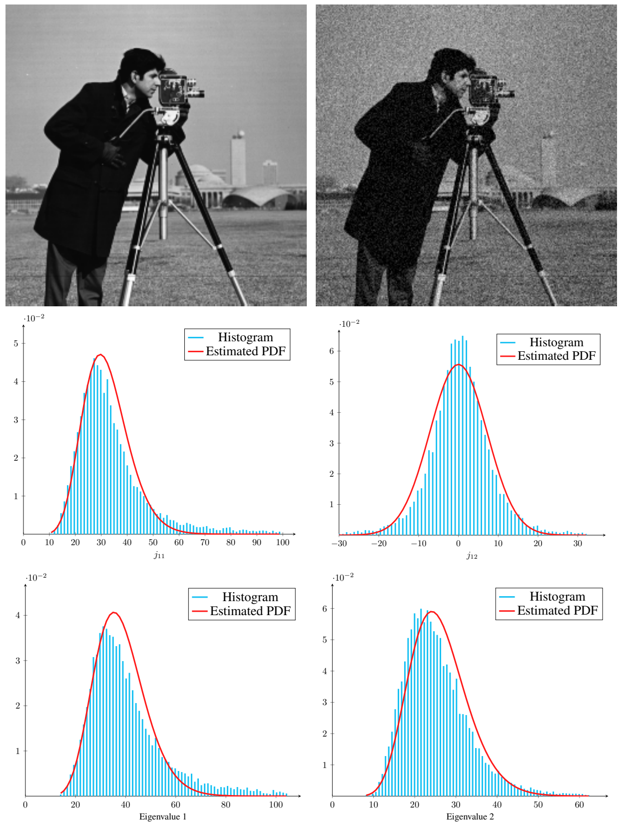


Fig. 5. Top: The original image and the input image with added Gaussian noise is shown. Middle: To the left the error distribution for $j_{11}$ is shown, and to the right the error distribution for $j_{12}$ is shown. Bottom: The error distributions of the two Eigenvalues of the structure tensor is shown.

our input image by doing structure adaptive image smoothing using the structure tensor. This filtering will give us a higher bit range than the input 24-bit color input. This is the equivalent of doing super resolution, but on the grayscale as opposed to the pixel resolution. We will then map this high dynamic range image to a standard 24-bit color image using tone mapping. Here we will use the method described in [12], which will automatically increase the contrast. We have used the standard implementation available at [13]. The structure tensor has been used for image denoising in a number of previous work. We will closely follow the methods described in [14], [15], [16] and will here give the outline of their methods for completeness. One main difficulty of these methods is the parameter estimation and we will here show how our error modelling can be used to set parameters automatically. This denoising method is based on constructing Gaussian smoothing kernels that are adapted to the local edge and gradient structure. This is done by looking at the Eigenvalues and Eigenvectors of the structure tensor. So we will start by looking at the noise distribution of the two Eigenvalues of the structure tensor.

Since the structure tensor is a $2\times2$-matrix we can calculate the Eigenvalues explicitly. They are given by,

$$\lambda_1 = 0.5\cdot(j_{11}+j_{22}+\sqrt{j_{11}^2-2j_{11}j_{22}+4j_{12}^2+j_{22}^2}), \quad (25)$$

$$\lambda_2 = 0.5\cdot(j_{11}+j_{22}-\sqrt{j_{11}^2-2j_{11}j_{22}+4j_{12}^2+j_{22}^2}). \quad (26)$$

This involves taking square roots, which we want to avoid in our modeling, since this will make it untractable to estimate the following error distributions. So instead, for the estimation of variance and mean of $\lambda_1$ and $\lambda_2$, we make the following approximation,

$$\hat{\lambda}_1 = 0.5\cdot(j_{11}+j_{22}+|j_{12}|+|j_{11}-j_{22}|), \quad (27)$$

$$\hat{\lambda}_2 = 0.5\cdot(j_{11}+j_{22}-|j_{12}|-|j_{11}-j_{22}|). \quad (28)$$

We will again model the noise as gamma distributed, and hence we want to estimate the mean and variance of $\hat{\lambda}_1$ and $\hat{\lambda}_2$. We will model $j_{12}$ and $j_{11}-j_{22}$ with normal distributions so taking the absolute value of these is straight-forward, yielding folded normal distributions. This will give means $A_1+B_1\cdot\sigma^2$ and $A_2+B_2\cdot\sigma^2$, where $A_i$ and $B_i$ only depend on the filters used in the calculation of the structure tensor. The variances are given by $D_1\cdot\sigma^4$ and $D_2\cdot\sigma^4$, where again $D_i$ only depends on the filters, but where some care has to be taken in order to handle the covariances. In Figure 5 the estimated distributions for the noise in the Eigenvalues is shown, for a real image with added Gaussian noise.

We will now see how we can use this information in the denoising of an image. The filtering is done by applying a kernel at each pixel neighbourhood,

$$I_{out}(\mathbf{x_0}) = \frac{1}{\mu(\mathbf{x_0})}\iint_{\Omega} k(\mathbf{x_0},\mathbf{x})I_{in}(\mathbf{x})dxdy \ , \quad (29)$$

where

$$\mu(\mathbf{x_0}) = \iint_{\Omega} k(\mathbf{x_0},\mathbf{x})dxdy \quad (30)$$

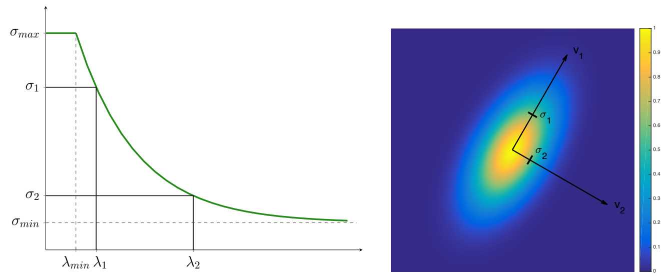


Fig. 6. Depiction of the smoothing kernel construction. Left: the function from Eigenvalues to kernel standard deviation.Right: the heat-map of the resulting kernel is shown, with the Eigenvector directions also shown.

is a normalizing factor. The area $\Omega$ over which the summation is taken is chosen as a finite neighborhood centered around $\mathbf{x_0}$. The kernel $k(\mathbf{x_0}, \mathbf{x})$ is calculated for each point $\mathbf{x_0}$, and adapts to the local gradient structure. The goal is to not smooth over edges. The kernels should be wide in directions of homogeneous intensity and narrow in directions with important structural edges. To find these directions, we calculate the structure tensor. The Eigenvalues and Eigenvectors of $J_\rho$ will give us the structural information that we seek. The Eigenvector $\mathbf{v_2}$ corresponding to the smallest Eigenvalue $\lambda_2$ will be approximately parallel to the direction of minimum intensity variation while the other Eigenvector is orthogonal to this direction. The magnitude of each Eigenvalue will be a measure of the amount of intensity variation in the direction of the corresponding Eigenvector. For a deeper discussion on Eigenvalue analysis of the structure tensor see [17].

The summation kernels $k(\mathbf{x_0}, \mathbf{x})$ are constructed at each point $\mathbf{x_0}$ as Gaussian functions,

$$k(\mathbf{x_0}, \mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\mathbf{x_0})^T \mathbf{R}\boldsymbol{\Sigma}^2\mathbf{R}^T(\mathbf{x}-\mathbf{x_0})} \quad . \tag{31}$$

These include a rotation matrix $\mathbf{R}$ and a scaling matrix $\boldsymbol{\Sigma}$. The rotation matrix is constructed from the eigenvectors $\mathbf{v_i}$ of $J_\rho$, $\mathbf{R} = \begin{bmatrix}\mathbf{v_1} & \mathbf{v_2}\end{bmatrix}$, while the scaling matrix has the following form,

$$\boldsymbol{\Sigma} = \begin{bmatrix} \frac{1}{\sigma(\lambda_1)} & 0 \\ 0 & \frac{1}{\sigma(\lambda_2)} \end{bmatrix} \quad . \tag{32}$$

The function $\sigma(\lambda_i)$ is a decreasing function that sets the width of the kernel along each Eigenvalue direction, and we use the following function

$$\sigma(\lambda) = \begin{cases} \sigma_{max} & \text{if } \lambda \leq \lambda_{min}, \\ \Delta\sigma e^{-d\frac{(\lambda-\lambda_{min})}{\lambda_{min}}} + \sigma_{min} & \text{if } \lambda > \lambda_{min}, \end{cases} \tag{33}$$

with $\Delta\sigma = (\sigma_{max} - \sigma_{min})$. The two parameters $\sigma_{min}$ and $\sigma_{max}$ control the maximum span of the kernel widths. We have used fixed values for these in our experiments, namely $\sigma_{min} = 0.4$ and $\sigma_{max} = 5$. The two parameters $d$ and $\lambda_{min}$ control the slope of the function, and these are set according to our analysis of the gamma distribution. In Figure 6 a graphical representation of how the kernel is constructued is shown, for a specific example.

In order to test our parameter estimation we used a number of real images taken under poor illumination. The input images are shown in Figure 7, where the variety of the input scene, illumination and contrast can be seen. We used an estimate of the noise in the input images (with an error mean of $1.5 - 2.0$ for the input gray value range of $0 - 255$). We then calculated the corresponding estimates of mean ($\mathrm{E}(\lambda_2)$) and variance ($\mathrm{Var}(\lambda_2)$) for the smallest Eigenvalue of the structure tensor. According to Figure 6 we want the kernels in directions where the Eigenvalues suggest only noise. We further want the kernel function to be invariant to intensity scale changes. This leads us to set the parameters for the kernel widths according to

$$\lambda_{min} = \mathrm{E}(\lambda_2), \tag{34}$$

$$d = \frac{\sqrt{\mathrm{Var}(\lambda_2)}}{\lambda_2^2}. \tag{35}$$

Using these settings, the low-contrast input images were filtered using the anisotropic kernels, channel-wise for the red, green and blue channels. One could also work with luminance and chrominance, but we follow the approach taken in [14]. The denoised image were then tone mapped using the democratic tone mapping method of [12], [13]. This method optimally uses the output discretization based on the gray value distribution of the input. For all our experiments we used 2000 bins in the input histogram discretization (see [18] for a discussion on bin size). The results can be viewed in Figure 7. In Figure 8 a close-up of the toy crocodile from Figure 7 is shown. In order to illustrate the denoising effect we have here only linearly amplified the signal for the original and the denoised image. One can see that the edges are well preserved in the denoised image.

## V. Conclusions

We have in this paper shown how we can model the noise distribution of the structure tensor, primarily using the gamma distribution. In order to estimate the distributions we have also given some new insights into the sum and product of gamma distributed random variables. Finally we have shown how the proposed model can be used for automatic parameter setting.

## Acknowledgment

This work was supported by ELLIIT and WASP.

## References


[1] J. Bigun and G. Granlund, "Optimal orientation detection of linear symmetry," in *1st Int. Conf. on Computer Vision*. IEEE, 1987.

[2] H. Knutsson, "Representing local structure using tensors," in *6th Scandinavian Conference on Image Analysis, Oulu, Finland*, 1989.

[3] B. Jähne, *Spatio-temporal image processing*. Springer, 1993.

[4] W. Förstner, "A feature based correspondence algorithm for image matching," *International Archives of Photogrammetry and Remote Sensing*, vol. 26, no. 3, pp. 150–166, 1986.

[5] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15. Manchester, UK, 1988, p. 50.

[6] H. Stark and J. W. Woods, *Probability, random processes, and estimation theory for engineers*. Prentice-Hall Englewood Cliffs (NJ), 1986.

[7] M. Jambunathan *et al.*, "Some properties of beta and gamma distributions," *The annals of math. stat.*, vol. 25, no. 2, pp. 401–405, 1954.

[8] A. Mathai, "Storage capacity of a dam with gamma type inputs," *Annals of the Inst. of Stat. Math.*, vol. 34, no. 1, pp. 591–597, 1982.

[9] P. Moschopoulos, "The distribution of the sum of independent gamma random variables," *Annals of the Inst. of Stat. Math.*, vol. 37, no. 1, pp. 541–544, 1985.

Fig. 7. The figure shows the output of our system for contrast enhancement of dark images. Beneath every low-contrast image the enhanced output image is shown. We get a high-contrast output without noise or discretization artifacts. Results are best viewed on-screen.

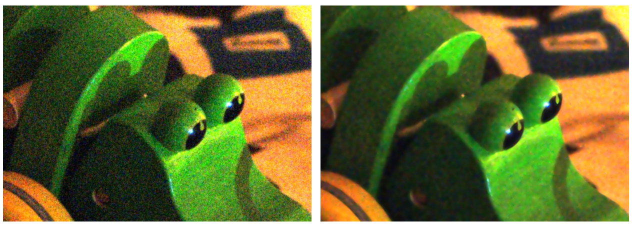

Fig. 8. A close-up of the toy crocodile from Figure 7. Left shows the original image linearly amplified and right shows the denoised amplified version. Notice that the edges are well preserved.


[10] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of math. stat.*, vol. 22, no. 1, pp. 79–86, 1951.

[11] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[12] M. Oskarsson, "Democratic tone mapping using optimal k-means clustering," in *Scandinavian Conf. on Image Analysis*, 2015.

[13] https://github.com/hamburgerlady/democratic-tonemap/.

[14] H. Malm, M. Oskarsson, E. Warrant, P. Clarberg, J. Hasselgren, and C. Lejdfors, "Adaptive enhancement and noise reduction in very low light-level video," in *ICCV*, 2007.

[15] G.-Z. Yang, P. Burger, D. N. Firmin, and S. Underwood, "Structure adaptive anisotropic image filtering," *Image and Vision Computing*, vol. 14, no. 2, pp. 135–145, 1996.

[16] E. Warrant, M. Oskarsson, and H. Malm, "The remarkable visual abilities of nocturnal insects: Neural principles and bioinspired night-vision algorithms," *Proc. of the IEEE*, vol. 102, no. 10, pp. 1411–1426, 2014.

[17] H. Haussecker and H. Spies, *Handbook of Computer Vision and Applications*. Academic Press, 1999, vol. 2, ch. Motion, pp. 125–151.

[18] M. Oskarsson, "Temporally consistent tone mapping of images and video using optimal k-means clustering," *Journal of mathematical imaging and vision*, 2016.