

The Weibull manifold in low-level image processing: an application to automatic image focusing

Vasileios Zografos*, Reiner Lenz, Michael Felsberg

*Linköping University, Department of Electrical Engineering, Computer Vision
Laboratory, SE-581 83 Linköping, SWEDEN*

Abstract

In this paper, we introduce a novel framework for low-level image processing and analysis. First, we process images with very simple, difference-based filter functions. Second, we fit the 2-parameter Weibull distribution to the filtered output. This maps each image to the 2D Weibull manifold. Third, we exploit the information geometry of this manifold and solve low-level image processing tasks as minimisation problems on point sets. For a proof-of-concept example, we examine the image autofocusing task. We propose appropriate cost functions together with a simple implicitly-constrained manifold optimisation algorithm and show that our framework compares very favourably against common autofocus methods from literature. In particular, our approach exhibits the best overall performance in terms of combined speed and accuracy.

1. Introduction

Low-level image processing typically involves the application of some type of filter function onto the image data, which results in the extraction of a number of desired feature descriptors. These descriptors are identified, isolated and subsequently evaluated in the later stages of the image analysis pipeline. The analysis step is often carried out in some representation of the resulting feature space (e.g. histogram), an endeavour which can be

*Corresponding author

Email addresses: zografos@isy.liu.se (Vasileios Zografos),
Reiner.Lenz@itn.liu.se (Reiner Lenz), michael.felsberg@liu.se (Michael Felsberg)

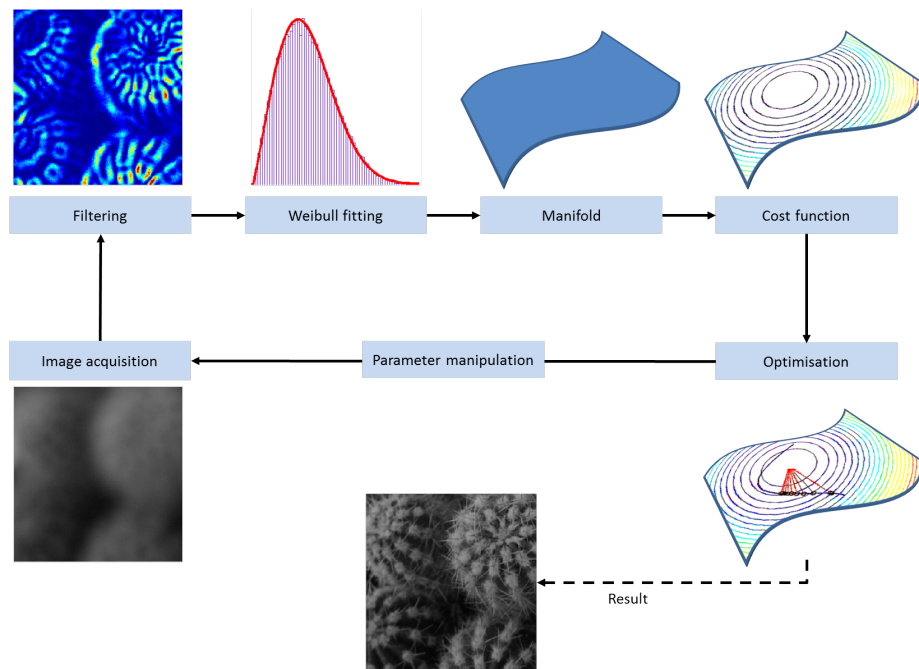


Figure 1: An outline of the main components of our image processing framework, here illustrating the autofocus problem. The starting point is the image acquisition step.

both difficult and inefficient due to the arbitrary-dimensional, discretised nature and inconvenient form of the space. So for example, distances, angles and directions between elements of the feature space may not always be defined or easily and accurately approximated. Furthermore, notions such as mean and dispersion might not be applicable in certain feature spaces. Therefore, explicit information about the geometry of the feature space is usually desirable, if not necessary, for reliable processing and image analysis.

The responses of difference-based filter functions on image data, are known to be Weibull distributed [16, 2, 49]. As such, every filtered image may be represented as a single, unique point on a statistical manifold. We exploit this fact and carry out the analysis in the 2-dimensional Weibull manifold, which is much simpler than the original feature space. The 2D, smooth Weibull manifold has well known properties that have been examined in statistical literature, but this is the first time its geometry has been applied to the area of image processing and analysis. In this new lower-dimensional representation, one may easily make use of the established notions of distance, arc-length, curvature and mean, in order to perform basic image analysis

tasks in a more principled and efficient manner.

We have devised a modular framework (see Fig. 1) that casts the image processing and analysis tasks as an implicitly-constrained optimisation problem, where we exploit the known manifold geometry to drive the optimisation algorithm to a good solution. For the purpose of feature extraction, we will be using simple line and edge filters in the form of the dihedral filters originally proposed by [27]. These filters, constructed with the aid of representation theory of discrete groups, are both fast and simple and allow for very efficient implementation for parallel execution. Their application in tasks such as content-based image retrieval and classification of high-level scene concepts has been described in a previous publication [49]. We demonstrate the efficacy of our framework and in particular the advantage of the Weibull manifold solution over standard methods, in the well known problem of automatic image focusing.

This paper is organised as follows. In Section 2 we describe in more detail the individual components of our framework. We then continue onto Section 3 with a brief introduction to automatic image focusing and include some of the common approaches from literature. In the same section, we describe how our proposed framework may be applied to the solution of the autofocus problem. Section 4 follows with a description of the comparative experiments run and a description of the datasets used, as well as an analysis of the results. We conclude in Section 6.

2. Basic framework

The image processing framework we are introducing in this paper consists of a number of individual components that work together in a loop. First is the *image acquisition* component. This is the module which captures the image to be processed and makes them available to the rest of the system. This could be as simple as a look-up table of a stack of images or some interface to an imaging device. In general, the rest of the system does not have an explicit model of the imaging process, the contents of and the conditions in the imaged scene. However, modification of some parameters or physical properties of the acquisition system is carried out and is controlled by the rest of the framework.

The next step, is the *filtering* component. This component isolates important characteristic features from the image, such as edges or corners and generates a new representation of the image data in this new feature space.

The output of the filtering stage is a vector of magnitudes of filter responses, which is known to be Weibull distributed. The natural next step therefore is to fit a Weibull distribution to the data and describe the filtered image with just two numbers, the Weibull *scale* and *shape*. Note that this is of course a global application of the filters to the whole image. One may subdivide the image in patches and apply the filters at each patch. The per-patch filtered results are still Weibull distributed, but now they capture the local image properties instead.

Every such scale-shape pair may be considered as a point on the 2D Weibull *manifold*, and the manifold can be given a Riemannian metric, by the Fisher matrix, at every such point. Abstraction of images as manifold coordinates, allows us to turn the image processing task into equivalent operations on points such as curve fitting or optimisation. Using this line of thought, the next component is a *cost function* defined on the manifold, which should obtain an appropriate minimum related to the image processing task at hand. We may recover such a minimum using a manifold-based *optimisation* algorithm.

The image acquisition modification parameters are adjusted, driven by the optimisation component and the main loop (Fig. 1) is repeated until convergence to a good solution. The main framework components are described in more detail in the rest of this section.

2.1. Image filtering with dihedral filters

In the proposed processing scheme we characterize the visual properties of images with the help of low-level filter systems. We use the so-called dihedral filter systems that meet the requirements of both, fast execution times and simplicity. These filters are constructed with the help of the representation theory of discrete groups, which is a generalization of the theory of the discrete Fourier transform. Their usage is based on three observations: almost all digital images consist of pixels located on a square or hexagonal grid; The transformations that map these grids into themselves are the so-called dihedral groups; The filter functions should be adapted to these transformations. In this paper we will use filters defined on a 5×5 window and on grayscale images. The filters are constructed as follows. First we define two positions in the window as equivalent if there is a group element in the dihedral group (i.e. a rotation or a reflection) that maps one point into the other. As a result we get the six equivalence classes c_1, c_2, a_1, a_2, i and o shown in the

matrix

$$\begin{pmatrix} c_1 & i & a_1 & i & c_1 \\ i & c_2 & a_2 & c_2 & i \\ a_1 & a_2 & o & a_2 & a_1 \\ i & c_2 & a_2 & c_2 & i \\ c_1 & i & a_1 & i & c_1 \end{pmatrix} \quad (1)$$

The value at the origin o will be ignored. For the remaining 24 points we use the tools from the representation theory of the dihedral groups to divide them into six subsets of 4 points each. The first four subsets are the points in the equivalence classes c_1, c_2, a_1, a_2 . The other two contain the points in the class i . From the general theory it follows that for each of these quadruples of points there are two linear combinations g_x, g_y which transform like the traditional 2×2 edge detectors. For the four points (a, b, c, d) defined by the corners c_1 or c_2 we get the following

$$c_i = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad (2)$$

and $g_x = (a + c) - (b + d), g_y = (a + b) - (c + d)$. We combine them to obtain the magnitude value given by the squared norm $\|e\|^2 = g_x^2 + g_y^2$ which corresponds to the output of a simple edge detector on the 2×2 pixels. A similar construction is used for all quadruples resulting in a vector $E = (e_1^2, \dots, e_6^2)$ containing all magnitude values from the six quadruples. Finally we compute the euclidean length $\|E\|$ of the vector and use this as a measure of the “edge-strength” of the pixel distribution in this 5×5 window. The important properties of this construction is that the value $\|E\|$ does not change when the underlying 5×5 pixel pattern undergoes one of the dihedral transformations (rotation and reflection) and that there are an equal number of $+1$ and -1 coefficients involved in the filtering.

These filters used here represent one special type of edge detectors and in the framework of the representation theory, they transform like the two-dimensional representation of the group. For a detailed description of the group theoretical filter systems we refer the interested reader to the references [27, 28, 31]. The choice for using the dihedral filters was dictated by their simplicity of representation, their execution speed and their ease of parallelisation. In principle, it is possible to utilise other “edge-type” filters instead, which are similarly based on the calculation of pixel differences. This is because such types of filters naturally lead to the same distribution

families (Weibull, discussed next) that are fundamental to our framework, meaning that we do not need to make any alterations to the rest of the framework when switching to a different filter function. Notable choices with a Weibull connection are the Gaussian-based derivative filters [16], and the Gabor filters ([3] and see Fig. 2(a)), and possibly many others.

2.2. The Weibull distribution

The Weibull distribution is a member of the family of extreme value distributions, and arises as a natural limiting distribution of the extrema in a random sample with size $n \rightarrow \infty$ [21]. The probability density function (pdf) of a Weibull random variable x is defined as

$$p(x|\mu, \sigma, k) = \frac{k}{\sigma} \left(\frac{x - \mu}{\sigma} \right)^{k-1} \exp \left[- \left(\frac{x - \mu}{\sigma} \right)^k \right], \quad (3)$$

for $x \geq \mu$, $\mu \in \mathbb{R}$, $k, \sigma \in \mathbb{R}^+$ and where $\theta = (\mu, \sigma, k)$ are the location, scale and shape parameters of the distribution respectively. For $\mu = 0$, we obtain the 2-parameter Weibull distribution.

The Weibull distribution has a large domain of applications and has been traditionally used in survival analysis, reliability engineering, extreme event modeling, and material sciences. Quite recently [18, 16] have established a link between the Weibull distribution and difference filters on images. In [18] the authors suggested the connection to extreme value theory [21], via the properties of sums of correlated variables [2], whereas [16] follows the alternative path of fragmentation theory [7]. We have also subsequently exploited these ideas and presented a connection between the dihedral filters from Section 2.1 and extreme value theory in [49]. More specifically, for dihedral filter systems whose kernels consist of an identical number of +1 and -1 coefficients, we have found that the statistical distribution of their norm $\|E\|$ follows the Weibull model, for a vast class of natural images.

2.3. Basic properties and fitting

The Weibull distribution is quite versatile in that, for different parameter values it can obtain the form of other commonly known distributions. For example, when $k=1$ its mode vanishes and becomes the exponential distribution, and for $k=2$, we obtain the Rayleigh distribution (see Fig. 2 (a)). In addition, the Weibull pdf is roughly symmetrical for k near 3.6, when $k < 3.6$ the distribution is left-skewed, and right-skewed otherwise (See Fig. 2 (b)).

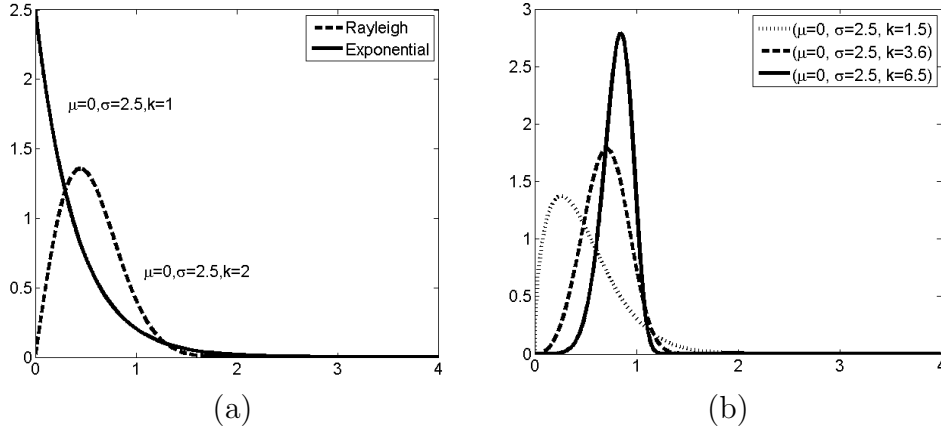


Figure 2: The Weibull distribution can take forms of other common distributions. Here the exponential and Rayleigh models (a). The skewness of the Weibull distribution is a function of its shape parameter k only (b).

The maximum likelihood estimation (MLE) of the Weibull distribution

$$\hat{\theta} = \operatorname{argmin}_{\theta} \sum_i \ln p(x_i|\theta) \quad (4)$$

from a data sample x , generally involves solving a numerical optimisation problem, since there are no closed-form solutions for the estimates of its 3 parameters. In this paper, we provide a standard steepest-descent fitting scheme for estimating the Weibull parameters from a data sample. However, for the interested reader we suggest the book by [38], which offers an extensive discussion on more advanced iterative schemes and effective initial estimates for θ .

Therefore, given the Weibull log-likelihood function

$$L(x|\theta) = n(\log k - k \log \sigma) + (k-1) \sum_i \log(x_i - \mu) - \sum_i \left(\frac{x_i - \mu}{\sigma} \right)^k, \quad (5)$$

with n being the size of the data sample x , and a current estimate of the Weibull parameters $\hat{\theta}_t$, we may obtain the next-best estimate as $\hat{\theta}_{t+1} = \hat{\theta}_t - \lambda \nabla_{\theta} L(x|\hat{\theta}_t)$, where $\lambda > 0$ is the step size parameter. The gradient components

of the log-likelihood $\nabla_{\theta} L(x|\hat{\theta}_t)$ are given by:

$$\begin{aligned}\frac{\partial L(x|\theta)}{\partial \mu} &= -(k-1) \sum_i \frac{1}{x_i - \mu} + \frac{k}{\sigma} \sum_i \left(\frac{x_i - \mu}{\sigma}\right)^{k-1}, \\ \frac{\partial L(x|\theta)}{\partial \sigma} &= \frac{k}{\sigma} \left[-n + \sum_i \left(\frac{x_i - \mu}{\sigma}\right)^k\right], \\ \frac{\partial L(x|\theta)}{\partial k} &= \frac{n}{k} - n \log \sigma + \sum_i \log(x_i - \mu) - \sum_i \left(\frac{x_i - \mu}{\sigma}\right)^k \log\left(\frac{x_i - \mu}{\sigma}\right).\end{aligned}\tag{6}$$

In this work we restrict ourselves to the 2-parameter Weibull distribution, since it has a simpler form and various analytic properties of its manifold are known. We still fit the 3-parameter model $p(x|\mu, \sigma, k)$ as explained above, but we work with the 2-parameter equivalent $p(x|\sigma, k) \approx p(x|0, \sigma, k)$. This has the same effect as fitting the 2-parameter model to the normalised sample $x - \mu$, and as a result no additional fitting is required. Since in practice the parameter μ accounts for very little variation in the data, we can work on the simple 2-parameter Weibull instead, without much loss of generality.

2.4. The Weibull statistical manifold

A set of probability densities can, under certain smoothness conditions, be equipped with a metric that makes them into a Riemannian manifold. A Riemannian manifold may be thought of as a geometrical object that locally looks like a flat Euclidean space. This means that at every point on the manifold we have a tangent space and on this tangent space a positive definite matrix defining distances and angles between elements on the tangent space. If these matrices vary smoothly along the manifold then they define a Riemannian metric on that manifold. In the case where the points on the manifold are given by probability distributions these manifolds are known as statistical manifolds. Statistical manifolds allow for differential geometry tools to be applied to information theoretic and statistical problems. The study of such statistical manifolds has led to the development of *information geometry* [1] as a novel research field.

From information geometry, one can show [15, 37] that the Fisher information

$$g_{i,j} = E \left(\frac{\partial^2 L(x|\theta)}{\partial \theta^i \partial \theta^j} \right),\tag{7}$$

induces a Riemannian metric on the parameter space of the distribution family $p(\theta)$, using for coordinates the parameters θ . For the particular case

of the 2-parameter Weibull distribution, the *Fisher information matrix* is given by

$$g_{i,j} = \begin{bmatrix} \frac{\sigma^2}{k^2} & \frac{\xi-1}{k} \\ \frac{\xi-1}{k} & \frac{\xi^2-2\xi+\pi^2/6+1}{\sigma^2} \end{bmatrix}, \quad (8)$$

where $\xi = \lim_{n \rightarrow \infty} (\sum_{k=1}^n \frac{1}{k} - \ln(n))$ is the Euler-Mascheroni constant.

A large range of image processes, such as automatic focusing and colour mapping, when examined through our framework, will produce point sequences (along some implicit curves), lying on the Weibull manifold. There, we can analyse and characterise these processes using information geometric tools and the specific properties of the manifold. Working on this manifold, is significantly more convenient than some arbitrary-dimensional discretised histogram space that is produced from the filtered image data. Additional properties of the Weibull 2-manifold are presented in detail in [34, 9].

2.5. The cost function

Every image that has been processed by the filtering and Weibull fitting steps of our framework, will generate one point on the manifold, which encapsulates the statistical properties of that filtered image. The next step is to assign some type of “quality” to these points, that in a way represents the suitability of the extracted features in relation to the problem which we wish to solve.

This is achieved with the introduction of a cost function F , defined on the Weibull manifold. One is free to construct a wide array of cost functions that incorporate some knowledge about the problem at hand, but as long as they are expressed in the Weibull coordinate system (σ, k) , we can use the same basic methodology in every case to obtain the best solution.

Since we are exploring the problem of automatic image focusing in this paper, we suggest two different cost functions, which are suited for but not necessarily restricted to, this particular problem. The first, and simplest cost function, is the variance of the Weibull distribution [14], and is defined as

$$F_v(\sigma, k) = \sigma^2 \left[\Gamma \left(\frac{k+2}{k} \right) - \Gamma^2 \left(\frac{k+1}{k} \right) \right], \quad (9)$$

with Γ being the gamma function. The Weibull variance is a smooth, scalar valued function which increases quadratically in σ , while decreasing as a power-law in k . It behaves much in the same way as the sample variance,

describing the frequency content of a signal. Variance-type cost functions are widely used in autofocus research due to their connection with Fourier optics [19]. There, the imaging system is modelled as a linear system that can be characterised by its point spread function. One can show that increased focus results in an increased high-frequency content in the image. As such, on average, the difference between neighbouring pixels will increase and so will the pixel sample variance. In addition, the variance will attain a maximum value at the best focused image of the sequence.

An alternative cost function, is the arctangent of the principal eigen-vector of the Fisher information matrix g_{ij}

$$F_a(\sigma, k) = \arctan\left(\frac{V_{21}}{V_{11}}\right), \quad (10)$$

where $V_{[2 \times 2]}$ is the symmetric eigen-vector matrix of g_{ij} . (10) can be rewritten in closed form as

$$F_a(\sigma, k) = \arctan\left(\frac{-6k^4 + \sigma^2\gamma - \sqrt{\omega}}{-12(\xi - 1)\sigma k^2}\right), \quad (11)$$

where

$$\begin{aligned} \gamma &= 6(\xi - 1)^2 + \pi^2 & \text{and} \\ \omega &= -24\pi^2\sigma^2k^4 + (\sigma^2\gamma + 6k^4)^2. \end{aligned} \quad (12)$$

F_a has similar properties to F_v but it is smoother. It is based on the idea that during autofocusing, most of the data variance is explained by σ^2 , so an optimum path is one of increasing σ . When at maximum focus, there is very little variance left to be explained by σ , so the principal eigen-vector will align with the k -coordinate axis, meaning that $\min(F_a(\sigma, k)) = \pi/2$.

Both of these cost functions will be evaluated in a number of focusing experiments and compared against other standard cost functions. They are illustrated in Fig. 3.

2.6. The optimisation method

We are restricting ourselves to image processing tasks where only a single parameter is modified. This is because such processes will define an implicit curve on the manifold, meaning a pre-defined search orientation along the curve. Here we present a basic approach for solving these very specific types of optimisation problems defined on manifolds. Note that for the remainder

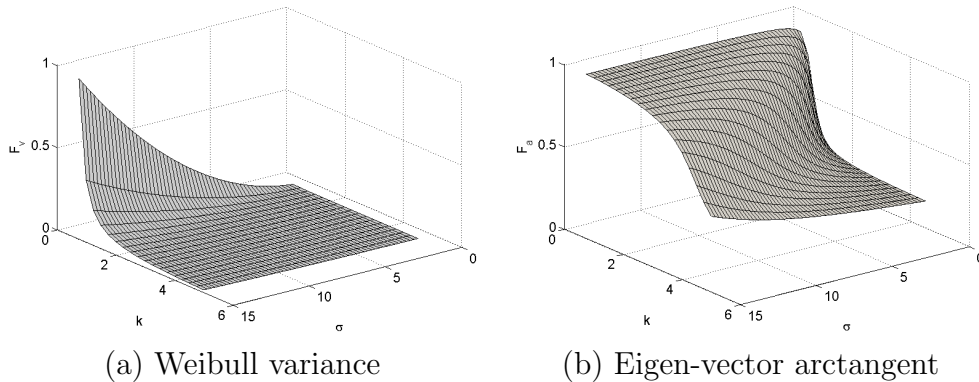


Figure 3: An illustration of the two proposed (normalised) cost functions defined on (σ, k) . We see that both attain a maximum on the same region but their shape (i.e. rate of increase) differs.

of this paper we shall consider the optimisation problem in terms of minimisation.

Therefore, given a modification parameter t defined in some arbitrary interval, and some non-trivial mapping $m : \mathbb{R} \rightarrow \mathbb{R}^2$ from the domain of t onto an implicit curve x_t on the Weibull manifold, we seek to minimise the expression

$$\operatorname{argmin}_t F(m(t)) = \operatorname{argmin}_t F(x_t) = \operatorname{argmin}_t F(\sigma(t), k(t)), \quad (13)$$

where F is some cost function defined on the Weibull coordinate system. The mapping m includes the image generation, dihedral filtering and Weibull fitting steps and maps each image onto a point on the manifold (see Fig.4). Thus for a time sequence of t values, we obtain a sequence of manifold points that determine an implicit curve.

As (13) suggests, our approach will be to work exclusively on the manifold and constrained to lie on the curve, without explicit knowledge of the curve or the mappings involved. In short, we deal with an *implicitly constrained optimisation problem on a manifold*.

We propose an iterative estimate-update stepping solution, where given an existing sample x_t we estimate the next point \hat{x}_{t+1} on the curve and map back onto the modification space to obtain $t+1$. We then generate the actual sample x_{t+1} , update our observations and proceed with the optimisation until convergence. We do not employ an explicit correction step between \hat{x}_{t+1} and x_{t+1} , although such a functionality can be introduced if desired.

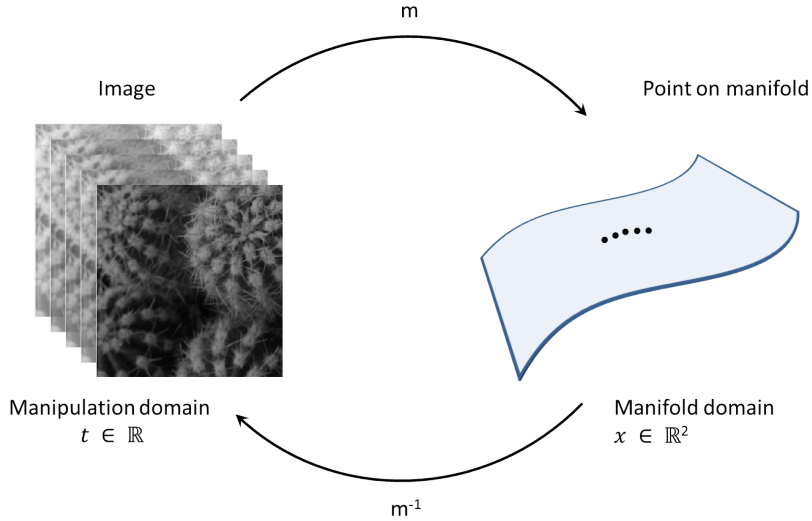


Figure 4: The mapping m and its inverse m^{-1} from the image (or modification) space to the manifold. m is explicit and involves the image generation, the filtering and Weibull fitting steps and in general does not have a tractable closed form. The inverse mapping from the manifold to the image space is implicit and usually a surjection.

Euclidean space

First we explore this problem in its simpler form, defined in the Euclidean space. From standard numerical optimisation theory, given a smooth cost function $F(x)$, its gradient field $\nabla F(x)$ and a current evaluation x_i of the location of the solution, we can obtain the next best, first order, estimate of the solution along the direction of the steepest descent as

$$\hat{x}_{i+1} = x_i - \lambda_i \nabla F(x_i), \quad (14)$$

where λ_i is some positive step-length. In the case where we are restricted to lie on an implicit curve, then the orientation of the search is pre-determined and (14) can be re-written as

$$\hat{x}_{i+1} = x_i + \lambda_i T_i, \quad (15)$$

where T_i represents the (normalised) tangent vector at the point x_i . The direction along T_i represents our best (first order) estimate of the form of the curve beyond the point x_i . In addition, T_i is numerically approximated via finite differences as $T_i = \frac{x_i - x_{i-1}}{\|x_i - x_{i-1}\|}$.

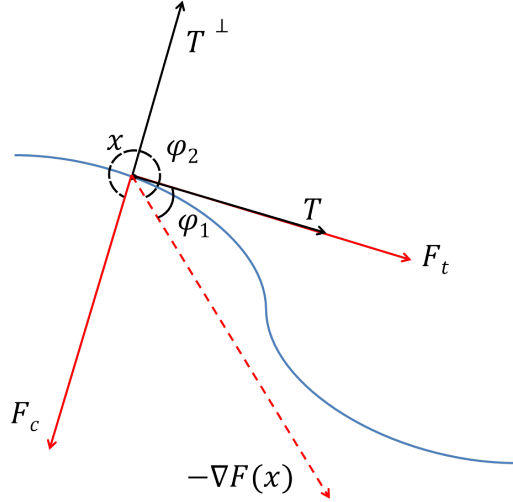


Figure 5: The component vectors at each point (particle) along the curve, the interaction of which determine the direction and acceleration step-length λ of the optimisation.

The step-length λ_i , is chosen using the very simple analogy of the planar motion of a particle along a curve. At each location x_i , we may define 3 vectors. The gradient vector $\nabla F(x_i)$, T_i and its orthogonal complement $T_i^\perp = [-T_i(2), T_i(1)]$, using a right-hand orientation convention (see Fig. 5). We already have analytic expressions for $F(x_i)$ and $\nabla F(x_i)$ and T_i^\perp is easily calculated given T_i . We may further decompose $\nabla F(x_i)$ into its two components. The *tangential* part $F_t = \langle -\nabla F(x_i) | T_i \rangle$, which is the scalar projection of the gradient vector on the curve tangent, and the *centripetal* part $F_c = \langle -\nabla F(x_i) | T_i^\perp \rangle$, which is the projection of the gradient vector along the tangent normal. The interaction between F_t and F_c is what determines the increase or decrease of the next step-length. In particular, we may formulate the step-length as

$$\lambda_i = \text{sgn}(F_c) K \left(\frac{F_t}{F_c} \right) \lambda_{i-1}, \quad (16)$$

where λ_{i-1} is the step-length at the previous iteration, $K(\cdot)$ is some damping kernel that maps from $[-\infty, \infty]$ to $[-1, 1]$ and $\text{sgn}(F_c)$ is used to determine the correct sign change for all possible configurations of T_i^\perp in space, relative to the gradient field. For the damping kernel we chose the logistic function

$$K(x) = \frac{2a}{1 + \exp(-2x)} - a, \quad (17)$$

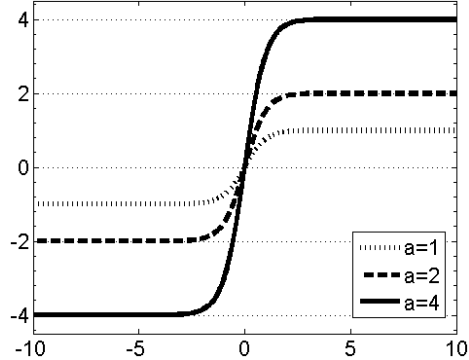


Figure 6: Illustration of the damping kernel (logistic function) from (17) for different values of the acceleration factor a .

where $a \geq 1$ is some acceleration scaling factor (Fig. 6). This formulation essentially suppresses infinitely large increases in acceleration when $F_c \rightarrow 0$.

The behaviour of these acceleration components is easily understood if we isolate the two extremal cases in Fig. 7. In the first case, when $F_c \rightarrow 0$ then (16) reduces to $\lambda_i = a\lambda_{i-1}$ and the new step-length represents an acceleration by a . Furthermore, since $\nabla F(x_i) \parallel T_i$, (15) becomes the standard gradient descent in (14). In the second case, when the tangential component vanishes $F_t \rightarrow 0$ then also $\lambda_i \rightarrow 0$, because we are approaching the point of equilibrium, where the particle stops. This is also the stationary point where the cost function is at a minimum along the curve.

The final component is the mapping from the cost function domain, where the optimisation is carried out, onto the modification space of t . We chose the simple inverse mapping

$$\begin{aligned} \Delta t_i &= \lambda_i \quad \text{where} \\ t_{i+1} &= t_i + \Delta t_i, \end{aligned} \tag{18}$$

which is sufficient and both Δt_i and λ_i will converge to 0 without any oscillations, provided that the initial step-length λ_0 is not too large. The rate of convergence depends on the difference in scale between the two quantities and is adjusted by the acceleration scaling factor a in (17).

As a result, we present the main optimisation routine in Algorithm 1 in Appendix B. We also include a Wolfe-type estimation step *getWolfe()* in Algorithm 2, to avoid unnecessary step evaluations during overshoots (see Fig. 8). For the termination conditions, we typically look at the number

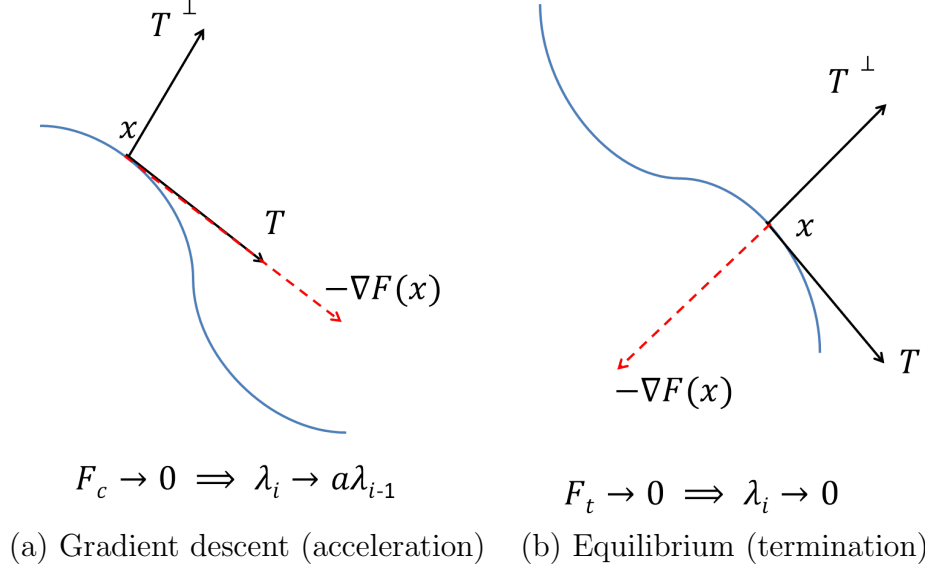


Figure 7: The two extreme behaviours of the particle motion analogue. In (a) we have the configuration of maximum acceleration which simplifies to the usual gradient descent step. In (b) we get the configuration of the equilibrium where the cost function is at a local minimum and the particle stops.

of maximum iterations $iter$, a tolerance on the decrease of the cost function F_{tol} , a tolerance on the change of the modification parameter Δt_{tol} and the distance between samples in the cost function space x_{tol} .

Manifold

The algorithms that we have presented for the Euclidean space, have a simple and natural extension to the Weibull manifold, since at each point x_i , both T_i and $\nabla F(x_i)$ lie on the tangent space of x_i . Thus the scalar products in (16) are now defined in the same way as the dual maps $F_t = -T_{ik} \nabla F_l g_{lk}$ and $F_c = -T_{ik}^\perp \nabla F_l g_{lk}$, and the orthogonal complement T_i^\perp now satisfies the two equations $\sum_{ij} T_i T_j^\perp g_{ij} = 0$ and $\|T_i^\perp\| = \|T_i\|$, giving:

$$T^\perp(2) = \sqrt{1/(1 + \rho^2)}, \quad T^\perp(1) = -T^\perp(2)\rho, \quad (19)$$

with

$$\rho = \frac{T(2)g(2,2) + T(1)g(2,1)}{T(1)g(1,1) + T(2)g(1,2)}. \quad (20)$$

Finally, the x_{tol} termination criterion in Algorithm 1 now changes from the Euclidean distance to the Rao distance between two points (σ_1, k_1) and

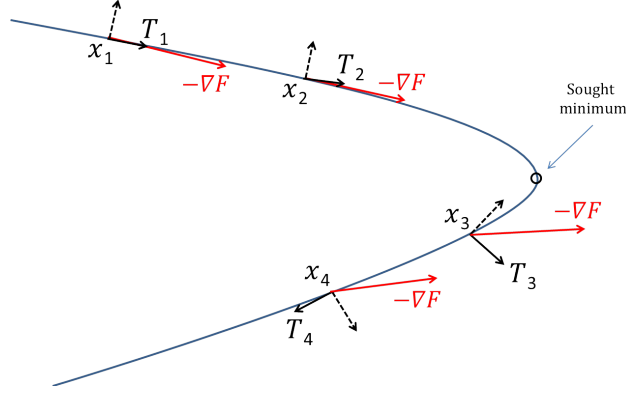


Figure 8: An illustration of a typical overshoot with a reduction of the cost function $F(x_3) < F(x_2)$. Notice that due to the numerical approximation of the tangent vector at x_3 , the step-length has a positive value (acceleration).

(σ_2, k_2) on the Weibull manifold, and which was given by [34] as

$$s = 2b \tanh^{-1} \left(\frac{[\log(\sigma_2/\sigma_1) - a(k_2 - k_1)/(k_2 k_1)]^2 + b^2(k_2 - k_1)/(k_2^2 k_1^2)}{[\log(\sigma_2/\sigma_1) - a(k_2 - k_1)/(k_2 k_1)]^2 + b^2(k_2 + k_1)/(k_2^2 k_1^2)} \right)^{1/2}, \quad (21)$$

where $b = \pi/\sqrt{6}$ and $a = 1 - \xi$.

3. Application to autofocusing

In the previous sections, we have presented all the individual components of our framework. In the next few sections, we will investigate the application of the complete framework to the solution of a typical image processing problem, and that is automatic image focusing. Automatic image focusing, or *autofocusing* is a interesting problem to explore since it belongs to the family of single-parameter image processing tasks, the modification of the parameter is done in some arbitrary (lens) space outside the image domain, and also additional image samples are very expensive, which makes for a good test case for our optimisation approach. Finally, this problem has been explored by various authors in the past and so comparative baseline methods are available for further evaluation.

3.1. Review of passive image autofocus

Passive autofocus systems, determine the correct image focus by analysing the data that enters the optical device. Such methods are usually based on the calculation of a “sharpness function” (SF), which is a real-valued estimate of the image focus. Commonly used sharpness functions in literature have been based on image derivatives [45, 6, 32, 12, 47, 40], statistics [13, 20, 46, 35, 39] and Fourier transforms [24, 44].

The most desirable property that an ideal SF should possess, is that of *unimodality*. In other words, for a through focus series of images (i.e. images sampled between the closest and furthest possible focal points) the SF must obtain a unique global optimum at the position of best focus. Beyond this basic behaviour, there are no widely agreed requirements for an SF. In [20], the authors proposed a set of criteria that a useful SF should fulfill. From those we note the two most important, *accuracy*: the global optimum must be present when the image is in focus, and *range*: the range over which an in-focus image can be obtained must be as large as possible. This means that the SF around the global optimum must have a wide basin of attraction so that any numerical optimisation algorithm has a good chance of locating the point of best focus. The same authors also mention *reproducibility*, that is, the existence of a sharp peak at the global optimum as a desirable property. However, we argue that this might not always be advantageous, especially if we employ a gradient-based numerical optimisation method, or if we wish to model the region around the optimum with a low degree polynomial model (e.g. quadratic).

Besides the type of SF used, passive autofocusing methods may be categorised based on their strategy for locating the global optimum (focus point). The simplest strategy [4], involves capturing a small number of images at some distance apart, thereby coarsely sampling the SF. This process is repeated at finer scales, around the location of the current optimum, until convergence. More advanced methods, such as [23, 25], start by evaluating the SF at a random focus position. The camera lens is moved around the current position and additional SF samples are computed. The focus search continues along the lens direction of improving sharpness with preferably an adaptive step-length.

These methods sample the SF directly and as a result they can be quite slow and expensive, since they typically require a large number of images to be captured. For faster autofocusing, other strategies have been proposed where they try to model the SF based on a limited number of coarse samples.

Chen et al. [10] use a second order discrete difference equation prediction model and [43, 17, 41] use a second order polynomial model.

Even more recently, novel methods have been introduced, which are trained on specific objects, lens positions and imaging conditions, and which can recover the optimal focus very fast and efficiently, by means of matching and a lookup table query. Most notable are those by [11] and [22]. The obvious drawback, is that these approaches are highly specific to the device used, the object of interest and the imaging conditions, and as such, do not scale very well as the number of possible objects increases.

3.2. Weibull for image autofocus

Our framework may be used on the autofocus problem by direct application of the Algorithms 1 and 2, together with either of the cost functions presented in Section 2.5. Furthermore, in this particular case, the modification parameter t represents the lens position which has been mapped into the interval $[0, 1]$. For illustration purposes, we examine a synthetic focus sequence of a real image that has been convolved with a Gaussian kernel of increasing width. The implicit curve that is generated onto the manifold is shown in Fig. 9 (a) together with the contour plot of the cost function from (9). The path traced by the curve on the cost function determines the one-dimensional SF $F(\sigma(t), k(t))$ in Fig. 9 (b). Observe how this SF behaves much in the same way as common SFs from literature. We stress once again that although we seek a minimum for $F(\sigma(t), k(t))$, we are in fact carrying out the optimisation in the 2-dimensional manifold curve in Fig. 9 (a) where the derivatives of F are well defined, unlike the unknown derivatives of F w.r.t t in Fig. 9 (b), that require costly numerical approximations.

Also notice in Fig. 9 (b) that for regular sampling in the lens parameter space t , we get irregular samples in the SF domain. This is more apparent near the narrow region around the minimum where the SF drops significantly, but otherwise is almost flat. The same behaviour occurs also in the 2d curve in Fig. 9 (a).

The basic interpretation of this is that when the images are blurred they all look very similar and one has to take very large lens steps to move away from the “blurred region” of the focus space. This is why the tail(s) of the SF typically look flat. The danger here is that for a fixed lens step λ the optimisation might terminate if no significant progress is made.

On the opposite side, when we start to obtain focused images, then each subsequent focused image will look considerably different from its predeces-

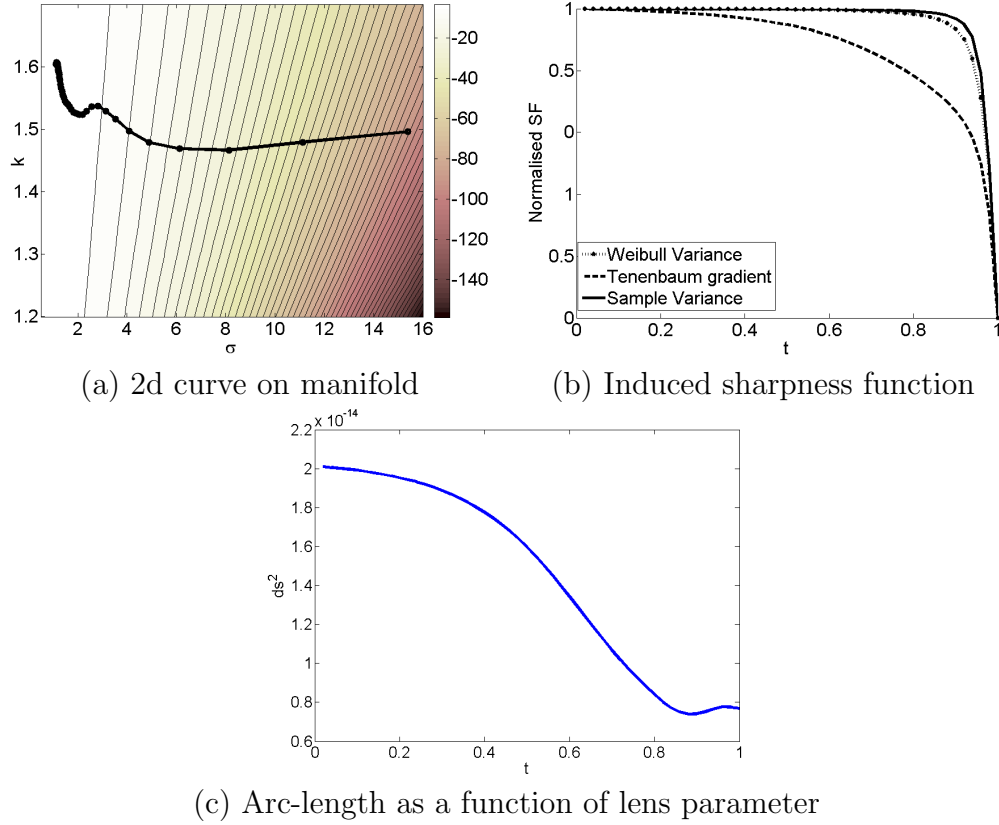


Figure 9: Illustration of the typical behaviour of a (synthetic) focus sequence.

sor. The SF will change dramatically, which explains the sharp drop. In the “focused region” (i.e. near the basin of the SF) a fixed lens step will lead to an ever-decreasing SF value until the minimum is obtained. The risk here of course is that we might overshoot a very narrow basin.

We may understand this behaviour further if we examine the squared arc length element in Fig. 9 (c), which is given by

$$ds^2 = \frac{k^2}{\sigma^2} d\sigma^2 + \frac{2(\xi - 1)}{\sigma} d\sigma dk + \frac{\xi^2 - 2\xi + \pi^2/6 + 1}{k^2} dk^2, \quad (22)$$

at an infinitesimally small region of the manifold around each position t . Where ds^2 is large, i.e. in the blurred region, neighbouring points on the manifold are far apart and for a fixed, small step-length we may not move very much on the manifold. The SF might thus appear flat and one would need to take larger steps λ in order to make progress. On the contrary, where

ds^2 is small, i.e. on the focused region, neighbouring points on the manifold are very close together and so the same λ might produce a very large manifold step. One would therefore need to take smaller steps to avoid missing the solution.

We emphasise here that regular t sampling versus irregular SF responses is not an effect of the manifold, since it occurs for all SFs (see for example the Tenenbaum gradient and sample variance in Fig. 9 (b)). All the pictured SFs have a similar monotonic decrease. However, we argue that ours is the only method that by using additional information from the geometry of the manifold, can provide a way of designing optimisation algorithms that can adaptively adjust λ depending on where in the focus space we are.

4. Experiments

In this section, we present the results from the experiments we have carried out in order to evaluate our framework in autofocusing and compare it against common baseline methods. The main purpose of these experiments is not to determine if our method outperforms every other method in literature, but to simply establish if the framework can be applied to and provide a reasonable solution in terms of speed and accuracy, for the problem of autofocusing. This is why we compare against more commonly used, baseline methods rather than against methods from consumer electronics literature [22], which are heavily tuned to the type of hardware used and the objects imaged.

4.1. Setup, datasets and proposed experiments

First are the off-line experiments, where we used 8 focus sequences, 6 of them captured with an inexpensive consumer camera¹ and 2 captured with an optical microscope. All the images are single-channel grayscale. For the 6 camera sequences, we captured between 200-300 frames per sequence, at a maximum resolution of 3264×2448 pixels, by varying the position of the camera lens between its minimum and maximum settings, thereby sampling the full range of focal plane positions at regular intervals. The 2 microscopy sequences contain 34 frames each, at a resolution of 696×520 pixels, and have been selected from the dataset in [5]. All the images in the 8 sequences

¹Canon IXY DIGITAL 20 IS

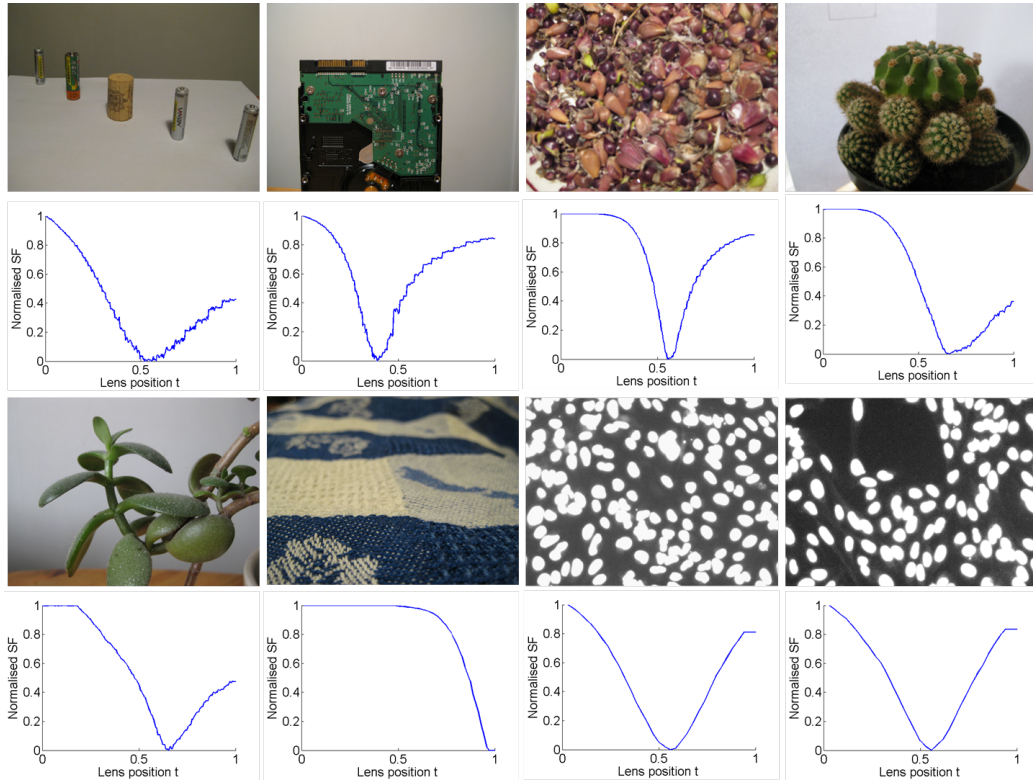


Figure 10: The in-focus frames from the 8 sequences used in our off-line experiments together with their normalised Weibull variance SFs.

were stored on disk, with the lens positions mapped between the range $[0,1]$ and made available to all optimisation algorithms in a look-up table fashion, thereby simulating the real camera lens control. The in-focus frames for all the 8 sequences together with their corresponding Weibull variance SFs are shown in Fig. 10.

The off-line experiments are partitioned into two sets. First, we wish to evaluate the performance of the proposed optimisation scheme on the 2-manifold against standard (baseline) 1-dimensional optimisation schemes; Second, we wish to compare the complete solution (Weibull sharpness function and manifold optimisation) against common autofocus schemes.

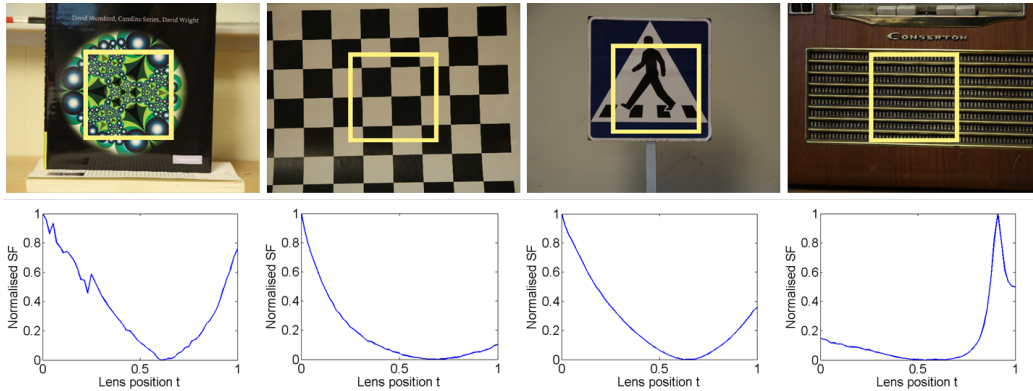


Figure 11: The in-focus frames from the 4 sequences used in our on-line experiments together with their normalised Weibull arctangent SFs. Note also the rectangle in the centre of each image, which indicates the region inside which the SF was calculated.

For the first set, it was necessary to have the same SF across all optimisation methods in order to evaluate the performance of the optimisation solution alone. Since it also had to be one SF that was comparable with our method, we opted for the Weibull variance. All optimisation methods modify the lens parameter t and for each value at t , calculate an SF value from (9). As such, we have performed 1000 tests for each sequence, where in order to keep the experiments as fair as possible, we used the same lens starting position t_0 and the initial step-length Δt_0 across all tested optimisation methods, at every test run. In addition, to evaluate the sensitivity on starting position and initial step-length, we sampled uniformly at random $t_0 \sim U(0.2, 0.5)$ and $\Delta t_0 \sim U(0.1, 0.5)$ for each test. For the second set, which is the evaluation of the complete autofocus schemes, we compared our method (using both cost functions from Section 2.5) against the methods described in the next sub-section. We have also carried out 1000 test runs for each sequence, while varying t_0 and Δt_0 at each test run.

In addition, we have also carried out a small set of on-line autofocusing experiments using a Canon EOS 50D camera. The camera was connected to a computer via a USB cable and the lens focus motor was controlled using the Digital EOS SDK (EDSDK 2.8 API) [8]. Images were captured in a real-time, on-line fashion at specified lens positions. The rest of the framework and interface remained unchanged and identical to the off-line experiments previously. The on-line setup allows for examining our method in more realistic and uncontrolled lighting and noise conditions. In other

words, a scene imaged multiple times at the same lens position, might result in slightly different sharpness values each time, due to the stochastic noise process in the camera and the minute variations in the global illumination properties. One advantage with having a computer-controlled camera setup, is that we can obtain a ground truth (g.t.) focused frame using the camera’s autofocus functionality. This allowed us to obtain more accurate estimates of the sharpness relative to the g.t., for all compared methods. In addition, the exact lens position could be read out directly from the camera, giving more precise measurements of frame distance from the g.t. As such, we imaged 4 different objects (see Fig. 11) with a 85mm lens, at fixed aperture and shutter speed. Each experiment was repeated 10 times and we computed the Weibull arctangent SF as described in (11), over a 200×200 pixel rectangular region, around the center of each image. As we have already mentioned in section 2.5, the Weibull arctangent SF, is in general smoother and wider than the Weibull variance SF we examined in the first set of experiments. This can be seen clearly in the plots in the second row of Fig. 11.

Performance was evaluated in all cases (on-line and off-line experiments), in terms of *speed* and *accuracy*. We used 4 different but complementary measures. These are:

i)(NFEs): the number of function evaluations, or equally the number of captured images until convergence. Note that the same termination tolerances have been used for all methods. The NFEs is the most important factor that determines optimisation speed, since the majority of the time used is spent in image acquisition, especially for microscopy. Obviously, the lower the NFEs is, the faster the method is.

ii)(F_{\min}): the minimum normalised SF value achieved at convergence. This is an accuracy measure and only makes sense when the same SF is used across all optimisation schemes, and so we only used it for the first set of off-line and the on-line experiments. The normalisation was done for illustration purposes and for comparing accuracies across sequences. Thus, although we minimise the negative SF, in the figures that follow, F_{\min} varies between $[0,1]$ and the higher it is, the more accurate the optimisation method is.

iii)(T_{val}): the distance (in image frames) from the ground truth, focused frame. This is an additional measure of accuracy, and it is especially useful in situations where different SFs are compared. In particular, it is often the case that a specific SF graph is very wide around the global minimum. This means that even though a recovered optimum may be far from the ground truth solution in terms of frames, in actual reality the SF difference between

the two is very small. It would be useful to have an additional accuracy measure that is able to differentiate between the two cases, especially in order to give some contextual meaning to different SF values. The distance measure we use here is defined as

$$T_{\text{val}} = |T_{\text{gt}} - T_{\text{opt}}| \cdot \text{frames}, \quad (23)$$

where T_{gt} is the ground truth lens position and T_{opt} is the recovered lens position.

iv) (N_r): the total number of mechanical lens movements (steps) required to converge to the optimum. This measure is used to differentiate between methods that might converge to a minimum within the same NFEs but have different strategies on how to get there. Obviously, the lower the number of motor steps, the faster the method is, and in order to have some way of comparing between different sequences, we evaluate instead

$$N_r = |N_{\text{ms}} - N_{\text{gt}}|. \quad (24)$$

N_{ms} is the total number of lens motor steps that the optimisation method requires, and N_{gt} is the theoretical ground truth lens steps necessary if we were to obtain a focused image with just a single function evaluation from the current starting point. N_r has units of minimum lens step.

4.2. Autofocusing methods: SFs and optimisation schemes

Here we present a selection of 1-dimensional optimisation methods and a number of commonly used SFs from autofocusing literature that together define the set of baseline methods we used for comparative evaluation. In terms of SFs, all of which obtain a global minimum at the focused image, we looked at the image *negative variance*

$$F_{\text{var}}(x, y, t) = -\frac{1}{N} \sum_x \sum_y (I(x, y, t) - \bar{I}(t))^2, \quad (25)$$

where N is the number of pixels in the image I , and $\bar{I}(t)$ signifies the mean image at lens parameter t . We also looked at the Shannon entropy

$$F_E(t) = -\sum_i p_i(t) \log_2 p_i(t), \quad (26)$$

where p_i is the grayscale image histogram and the sum is over the histogram bins. Finally we looked at another widely used SF, which is the Tenenbaum gradient [45] and is defined as follows

$$F_{\text{TB}} = \sum_x \sum_y S_x(x, y, t)^2 + S_y(x, y, t)^2, \quad (27)$$

where

$$S_x(t) = I(t) * \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \text{and} \quad S_y(t) = I(t) * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}. \quad (28)$$

For the optimisation methods, we tested 5 different approaches. These are: *Boddeke's method* [4], the *Golden Search* (GS) [36], *Nelder-Mead downhill simplex* [33], the *Global adaptive* (GA) method [26] and finally the *Hill climbing* (HC) method [42]. Due to economy of space we refer the interested reader to the respective publications for additional information on each method, but it suffices to say that all of them work on the SF directly, without calculating numerical derivatives. We also mention that we have not tested the 15 possible combinations of SFs and optimisation methods, as this would be rather uninformative and time consuming. Instead, we determined the best overall performing optimisation method from the first type of experiments, and used that method with the 3 SFs in the second type of experiments.

4.3. Results

We begin with the results from the off-line experiments. The first set of experimental results, averaged over all the 8 datasets and over all the 1000 test runs, are shown in Fig. 12. From the F_{min} accuracy plot (Fig. 12(a)), we see that the Golden Search method is the most accurate and with a very low variance. Our manifold optimisation method performs relatively well, with a median of 0.9806, which is better than the 0.9389 of the commonly used Boddeke's method. Ours however has a larger variance, which might indicate an increased sensitivity to starting values, something which is expected from local gradient-descent methods. If we look at the average frame distance plot (Fig. 12(b)), we observe a similar behaviour. The Golden Search is the most accurate method with a median of 2.41, followed by the Simplex and Global

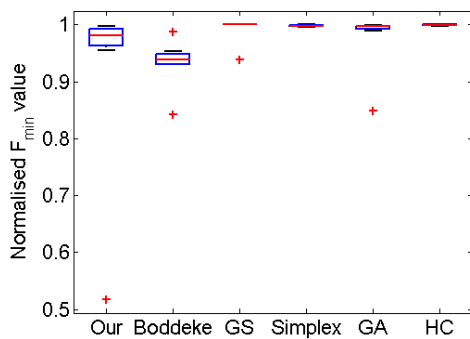
Adaptive methods. Our method has a median of 8.7 frames from the ground truth, which again is more accurate than Boddeke’s method.

In terms of speed, we observe a very different result. First, at the very important NFEs measure (Fig. 12(c)), the Golden Search is the most expensive method with a median of 33.85 frames required for convergence, and with the highest variance from all other optimisation methods. A similar behaviour is shown by the other highly-accurate methods such as the Simplex and the Global Adaptive. Our method together with Boddeke’s are amongst the fastest if we also consider the average number of motor steps (Fig. 12(d)). Notice that in terms of motor steps our method is quite efficient without too much of exploratory, overshooting or backtracking behaviour, unlike Boddeke’s. We may conclude by saying that the most accurate methods are also the most expensive, and for a small increase in accuracy they pay a very large penalty in captured image overhead. The Weibull optimisation method as well as Boddeke’s exhibit the best compromise between speed and accuracy, with ours being more efficient in terms of lens steps and much more accurate, while only requiring one more image frame to converge, on average.

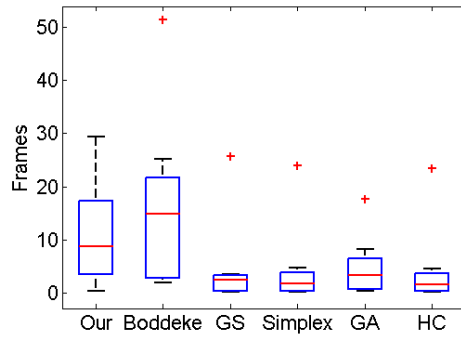
From the other competing methods, we note the Hill Climbing approach which also produces very good accuracy scores, at a moderate expense. For this reason, we have decided to use it as the optimisation component in the baseline methods in the second set of the experiments.

The averaged results from the second set are shown in Fig. 13. In terms of frame distance accuracy (Fig. 13(a)), the HC+V combination is the only one that outperforms our solutions. Surprisingly the Tenenbaum gradient is the least accurate solution. However, in terms of speed (Fig. 13(b,c)) all HC solutions are approximately twice as expensive as our approach with the HC+V combination being the most time consuming. From our two variants, the one using the Weibull arctangent in (11) is slightly more accurate and faster than its Weibull variance counterpart. This result supports in some respects our claim originally posed in Section 3 about the advantages of a wide region near the optimum.

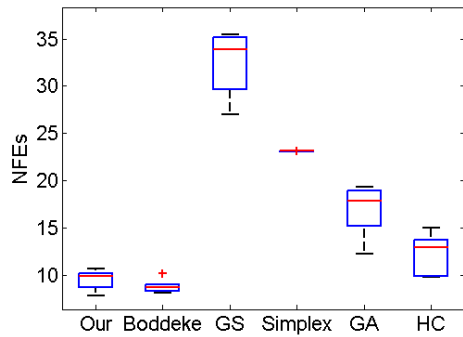
For the on-line experiments with the Canon EOS 50D, we can observe a similar pattern with the previous results. In Fig. 14(a) we can see the SF values for the 6 methods. Note here, that just like in Fig. 12(a) we are quoting normalised SF *accuracy*, with a larger value indicated a more accurate SF value recovered by the algorithm, relative to the ground truth. The g.t. value, is obtained by the camera’s internal autofocus functionality. However, unlike the off-line results, in this case due to the slight variations



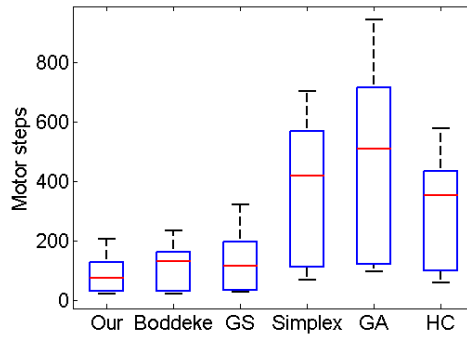
(a) Average SF value.



(b) Average frame distance.



(c) Average NFEs.



(d) Average motor steps.

Figure 12: Results from the first set of off-line experiments on optimisation methods with the same SF (Weibull variance). Each box in every plot contains averaged information from 8000 tests. The solid parts of the boxes show the data between the 25th and 75th percentile and the middle line is the median value. The crosses indicate statistical outliers. The labels on the abscissae are defined in Section 4.2.

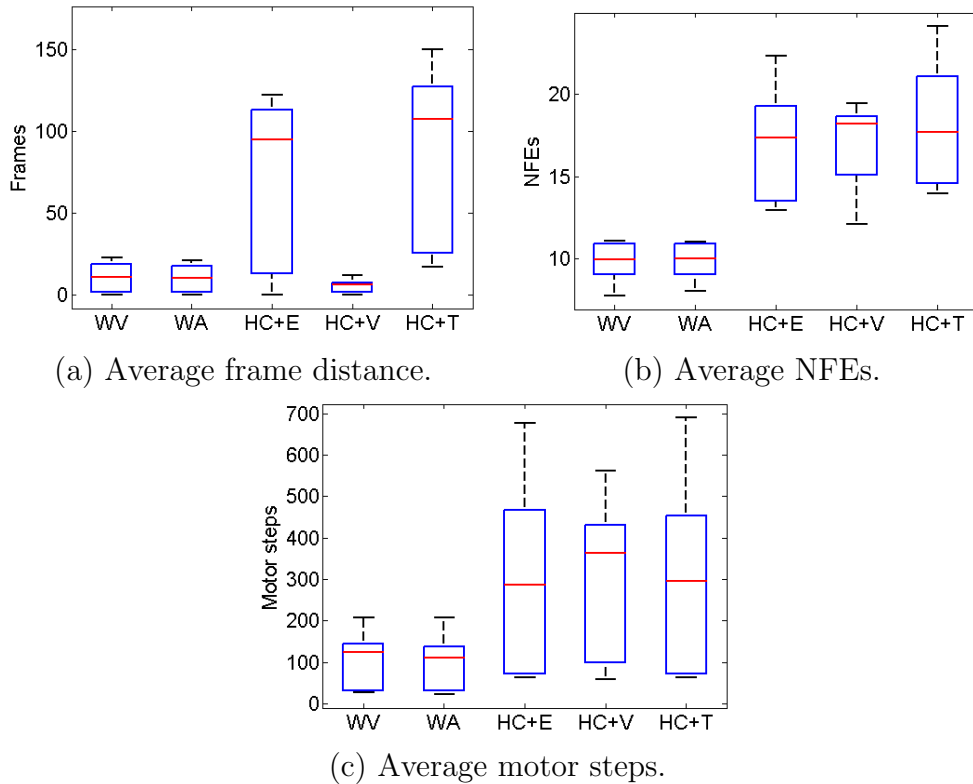


Figure 13: Results from the second set of off-line experiments on the complete approaches (optimisation method and cost function). The labels on the graphs are: (WV): Manifold optimisation with Weibull variance, (WA): Manifold optimisation with eigen-vector arc-tangent, (HC+E): Hill Climbing with image entropy, (HC+V): Hill Climbing with image variance and (HC+T): Hill Climbing with Tenenbaum gradient.

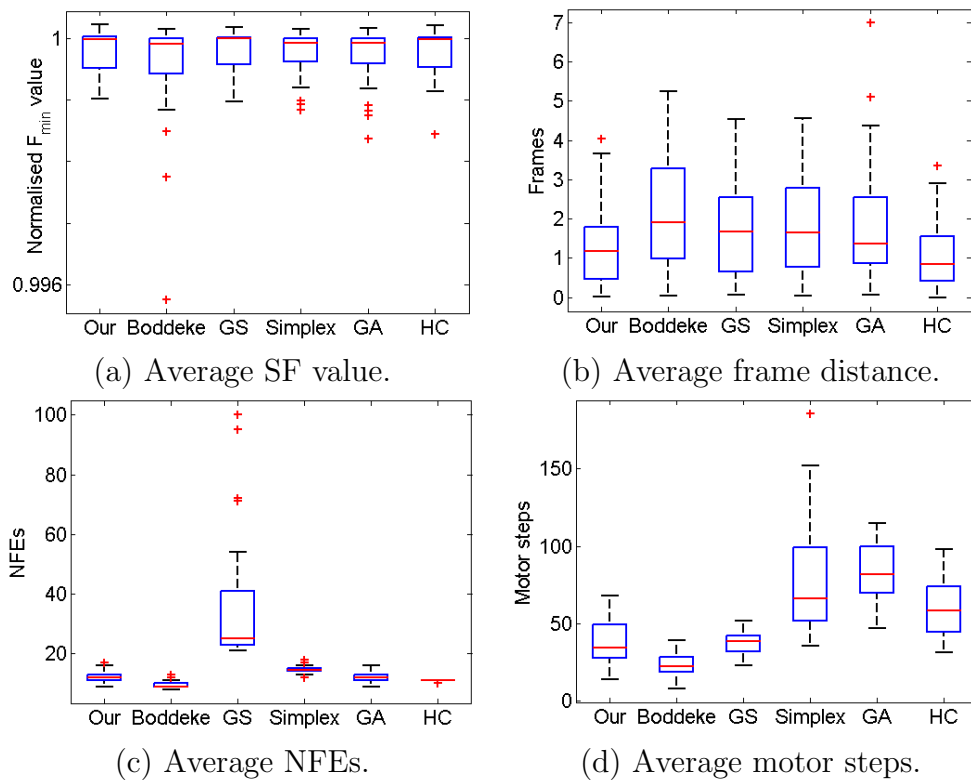


Figure 14: Results from the on-line experiments on optimisation methods with the same SF (Weibull arctangent). Each box in every plot contains averaged information from the different tests. The solid parts of the boxes show the data between the 25th and 75th percentile and the middle line is the median value. The crosses indicate statistical outliers. The labels on the abscissae are defined in Section 4.2.

in the imaging conditions between different test runs, there are small fluctuations in the SF values. Therefore, it is possible to obtain an SF value that is smaller than the ground truth and as a result normalised SF values that exceed 1 in Fig. 14(a). What we can see from this plot is that our method is very accurate, with no outliers and with low variance, a behaviour which is more of characteristic of global optimisation (and hence accurate) methods. For the average frame distance from the g.t. (Fig. 14(b)), which is an additional measure of accuracy, we see that our method is performing very well on average, just behind the global Hill Climbing method. Compare this with the performance in the off-line experiments (12(b)), where both our method and Boddeke’s were less accurate than all the global methods.

In terms of speed, the results in Fig. 12(c) and (d) are much in line with what we have already seen in Fig. 12(c) and (d). Our method together with Boddeke’s and the Hill Climbing methods, require the fewer number of acquired images (NFEs). For the average number of motor steps relative to the g.t., we see that in this case our method falls slightly behind Boddeke’s but still faster than the global optimisation approaches.

In summary we can say that our method offers a very good trade-off between speed and accuracy. We have seen some performance variations between the off-line and on-line realistic experiments, which are to be expected due to the different SFs examined with different characteristics (Weibull variance vs arctangent). What is important to note is that the good overall performance of our method generalises well across the different experimental setups and the different SFs used. These results, provide considerable support to our hypothesis that the Weibull framework can be applied to and provide a fast and accurate solution to the problem of autofocusing, in line with existing methods in literature.

5. Other applications

The autofocus application described so far is only one example where we can manipulate a single parameter (in this case the focus settings of the camera) in order to minimise a cost function defined on a statistical manifold, and thus solve a specific image processing problem. Another application where our Weibull optimisation framework can be used, is in the combination of multiple channels in an image, producing a scalar-valued image. Specifically, we have looked into the task of weighted linear combinations of the RGB channels in a colour image into a single channel intensity image, in order

to enhance certain properties such as edge information or contrast in the resulting image. This type of processing is particularly useful in subsequent tasks such as object segmentation. In that respect, we have published two papers [29, 30] using the same Weibull framework and optimisation method. In more detail, we used the trace of the Fisher information matrix as the objective function

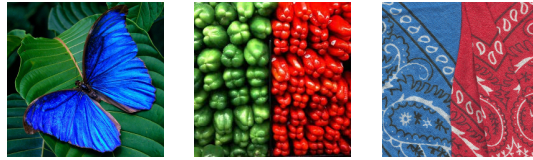
$$F_t(\sigma, k) = \frac{6 + 6(\xi - 2)\xi + \pi^2 + 6k^4/\sigma^2}{6k^2}, \quad (29)$$

and since instead of one we had to optimise over three parameters (RGB weights), we carried out a single search in each parameter independently (coordinate search scheme). Some results that show the efficacy of this method over standard approaches, in terms of a grayscale mapping for image enhancement, are illustrated in Fig. 15.

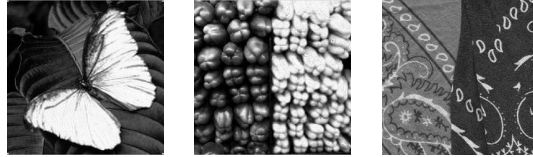
An additional application of the Weibull framework in image processing, is our ongoing work on tonemapping optimisation for dynamic range reduction in High Dynamic Range (HDR) images, to the 8-bit pixel values used in most processing applications today. An appropriate Weibull-based cost function, related to the gamma value of the image, is optimised using our framework over regularly sampled, overlapping, local patches from the HDR image and the mapped results are combined into the final output. We can see some initial results in Fig. 16. Compared with standard techniques, the Weibull optimised method gives visually more pleasing results, with a better contrast response, while avoiding over-exposure.

One further potential application of the Weibull framework, is in the area of saliency enhancement. The connection between the Weibull and image saliency has been identified and used by [48]. A straightforward extension to their work would be to cast saliency enhancement as optimisation, by tuning the appropriate saliency response of the image using our Weibull framework.

The aforementioned examples, are just a few of the possible additional applications of the Weibull manifold and optimisation framework, and illustrate that our proposed approach is not limited to the autofocus scenario presented here, but may be used with little or no modification to a variety of different image processing problems (discrete, continuous and multi-parameter).



Original RGB images.



Weibull grayscale mapping.



Fixed grayscale mapping.

Figure 15: Example from [30], of RGB to grayscale mapping using the Weibull framework. In the first row we have three examples of RGB images dominated by different combinations of primary colours (GB, RG and RB). In the second row we can see the results from the Weibull framework. Note how the grayscale mapping (RGB weights) adapt to the image content, in order to improve the overall contrast. Such a mapping is ideal for obtaining scalar images suitable for segmentation. Compare this with a standard fixed mapping ($0.298R+0.587G+0.114B$) that is unable to separate the different colours adequately.



8-bit quantization Histogram tonemapping Weibull tonemapping

Figure 16: Tonemapping examples on HDR images. In the first column we see a basic, fixed 8-bit quantization of the image. Note the overall poor contrast response. In the second column we show a common method, which adapts to the image content using histogram equalisation. Even though the contrast response has improved, the overall image is over-exposed. In the third column we show the adaptive Weibull method that uses our framework. We observe a much better contrast response that does not contain any over-exposed regions.

6. Conclusion

In this work we have exploited the connection between pixel-difference type of filter functions and the Weibull distribution, and devised a basic framework for low-level processing and subsequent analysis of images. Any image, once filtered can be represented as a point on the 2D Weibull manifold. Since we know many of the basic properties of that manifold, such as curvature, geodesic distances and its Riemannian metric, we can easily and efficiently carry out tasks like optimisation, clustering, curve fitting and extrapolation on the manifold. This undertaking would undoubtedly be very difficult in the original feature space.

In order to demonstrate our proposed framework, we have looked at its application to the image processing task of autofocusing. In that problem, we modify the lens position while capturing new images, until the most focused image is recovered. Such a modification scheme suggests that one may use an optimisation approach to solve the problem. In fact, most of the existing methods in literature are 1-dimensional minimisation schemes working on an appropriate sharpness function. We have followed the same route, but instead perform the optimisation on the 2D manifold, constrained to lie on

an implicit curve. There we are free to define cost functions with known, analytic derivatives and which can be used in a very efficient gradient-descent minimisation algorithm.

We have compared our method against common baseline approaches from literature, on different image sequences and across a number of different evaluation measures. What we found is that our manifold method is one of the best overall performers in terms of combined speed and accuracy. Both of these measures are important for practical applications where we require a well focused image but without capturing too many additional images. As a result, we have established that the additional cost of fitting the Weibull distribution to the filtered data, is more than compensated by the reduction in the number of required images for convergence.

Further to the autofocusing scenario we have also presented some additional application areas where our framework can be used (e.g. RGB-to-grayscale mapping, HDR tonemapping and saliency boosting), that demonstrate the flexibility and usefulness of our proposed approach in a variety of different image processing tasks.

Acknowledgements

This work was supported with funding from the ECs 7th Framework Programme (FP7/2007-2013), grant agreement 247947 (GARNICS); by the Swedish Foundation for Strategic Research through grant IIS11-0081; and from the Swedish Research Council through a grant for the project Extended Target Tracking (within the Linnaeus environment CADICS).

Appendix A. Cost function gradients

The gradient of the Weibull variance cost function in (9) is given by:

$$\nabla F_v = \begin{pmatrix} -2\sigma \left[-\Gamma \left(\frac{1+k}{k} \right)^2 + \Gamma \left(\frac{2+k}{k} \right) \right] \\ \frac{2\sigma^2}{k^4} \left[-\Gamma \left(\frac{1}{k} \right)^2 \psi \left(1 + \frac{1}{k} \right) + 2k\Gamma \left(\frac{2}{k} \right) \psi \left(\frac{2+k}{k} \right) \right] \end{pmatrix}, \quad (\text{A.1})$$

where $\psi(m, z) = \frac{d^{m+1}}{dz^{m+1}} \ln \Gamma(z)$ is the polygamma function. The gradient of the arctangent cost function in (11) is given by:

$$\nabla F_a = \begin{pmatrix} \frac{6(\xi-1)k^2(6k^4+\sigma^2\chi)}{\zeta} \\ \frac{-12(\xi-1)\sigma k(6k^4+\sigma^2\chi)}{\zeta} \end{pmatrix}, \quad (\text{A.2})$$

where $\zeta = 36k^8 + 12\sigma^2k^4(6(\xi - 1)^2 - \pi^2) + \sigma^4\chi^2$ and $\chi = 6(\xi - 1)^2 + \pi^2$.

Appendix B. Optimisation algorithms

Algorithm 1 The main minimisation algorithm

Input: $\lambda_1, \Delta t_1, t_1$, **Output:** $t_i, x_i, F(x_i)$

$\{F(x_1), x_1\} = \text{getSample}(t_1)$
 $i = 2$
while !termination **do**
 $t_i = t_{i-1} + \Delta t_{i-1}$
 $\{F(x_i), x_i\} = \text{getSample}(t_i)$
 if $F(x_i) > F(x_{i-1})$ **then**
 $\{F(x_t), x_t\} = \text{getSample}(t_{i-1} - \epsilon)$
 if $F(x_t) < F(x_{i-1})$ **then**
 $\{x_i, \lambda_i, F(x_i), t_i\} = \text{getWolfe}(x_t, x_{i-2}, \lambda_i, t_{i-1} - \epsilon)$
 else
 $\{x_i, \lambda_i, F(x_i), t_i\} = \text{getWolfe}(x_{i-1}, x_i, \lambda_i, t_{i-1})$
 end if
 else
 Calculate $T_i, T_i^\perp, \nabla F(x_i), F_c$ and F_t
 Calculate λ_i using (16)
 Calculate Δt_i using (18)
 end if
 $i = i + 1$
 Check termination criteria
end while

Algorithm 2 *getWolfe()* function. Wolfe point correction algorithm

Input: x_1, x_2, λ_1, t_1 , **Output:** $x_i, \lambda_i, F(x_i), t_i$

$\lambda_1 = \lambda/4$
 $i=2$
while $\lambda_i \geq$ some threshold τ **do**
 Calculate $T_i = [x_2 - x_1], T_i^\perp, \nabla F(x_2), F_c$ and F_t
 Calculate λ_i using (16)
 Calculate Δt_i using (18)
 $t_{i+1} = t_i + \Delta t_i$
 $\{F(x_{i+1}), x_{i+1}\} = \text{getSample}(t_{i+1})$
 if $F(x_{i+1}) < F(x_i)$ **then**
 Return $\lambda_i, t_{i+1}, x_{i+1}, F(x_{i+1})$
 else
 $\lambda_i = \lambda_i/4$
 end if
 $i = i + 1$
end while
Return $\lambda_i, t_i, x_i, F(x_i)$

References

- [1] Amari, S., Nagaoka, H., 2000. Methods of information geometry. volume 191 of *Translations of mathematical monographs*. American Mathematical Society.
- [2] Bertin, E., Clusel, M., 2006. Generalised extreme value statistics and sum of correlated variables. *Journal of Physics A: Mathematical and General* 39, 7607–7619.
- [3] Bhagavathy, S., Tesic, J., Manjunath, B.S., 2003. On the Rayleigh nature of Gabor filter outputs, in: *International Conference on Image Processing*.
- [4] Boddeke, F.R., van Vliet, L.J., Netten, H., Young, I.T., 1994. Autofocusing in microscopy based on the OTF and sampling. *Bioimaging* 2, 193–203.
- [5] Bray, M.A., Fraser, A.N., Hasaka, T.P., Carpenter, A.E., 2011. Workflow and metrics for image quality control in large-scale high-content screens. *Journal of Biomolecular Screening* .
- [6] Brenner, J., Dew, B., Horton, J., King, T., Neurath, P., Selles, W., 1976. An automated microscope for cytologic research a preliminary evaluation. *J Histochem Cytochem* 24, 100–111.
- [7] Brown, W.K., 1988. A theory of sequential fragmentation and its astronomical applications. *Journal of Astrophysics and Astronomy* 10, 89–112.
- [8] Canon Inc., 2010. Digital EOS SDK, Digital Imaging Developer Programme. <http://www.didp.canon-europa.com> .
- [9] Cao, L., Sun, H., Wang, X., 2008. The geometric structures of the Weibull distribution manifold and the generalized exponential distribution manifold. *Tamkang Journal of Mathematics* 39, 45–52.
- [10] Chen, C., Hong, C., Chuang, H., 2006. Efficient auto-focus algorithm utilizing discrete difference equation prediction model for digital still cameras. *IEEE Trans. Consum. Electron.* 52, 1135–1143.

- [11] Chen, C.Y., Hwang, R.C., Chen, Y.J., 2010. A passive auto-focus camera control system. *Appl. Soft Comput.* 10, 296–303.
- [12] Choi, K.S., Lee, J.S., Ko, S.J., 1999. New autofocusing technique using the frequency selective weighted median filter for video cameras. *IEEE Trans. Consum. Electron.* 45, 820–827.
- [13] Erasmus, S.J., Smith, K.C.A., 1982. An automatic focusing and astigmatism correction system for the SEM and CTEM. *Journal of Microscopy* 127, 185–199.
- [14] Evans, M., Hastings, N., Peacock, B., 2001. *Statistical Distributions*. Wiley series in probability and statistics. 3rd edition.
- [15] Fisher, R.A., 1925. Theory of statistical estimation. *Proceedings of the Cambridge Philosophical Society* , 700–725.
- [16] Geusebroek, J.M., 2005. The stochastic structure of images, in: *Scale Space*, pp. 327–338.
- [17] Geusebroek, J.M., Cornelissen, F., Smeulders, A.W.M., Geerts, H., 2000. Robust autofocusing in microscopy. *Cytometry* 39, 1–9.
- [18] Geusebroek, J.M., Smeulders, A.W.M., 2003. Fragmentation in the vision of scenes., in: *International Conference on Computer Vision*, pp. 130–135.
- [19] Goodman, J.W., 1996. *Introduction to Fourier optics*. McGraw-Hill, New York.
- [20] Groen, F.C.A., Young, I.T., Ligthart, G., 1985. A comparison of different focus functions for use in autofocus algorithms. *Cytometry* 6, 81–91.
- [21] Gumbel, E.J., 1958. *Statistics of Extremes*. Columbia University Press, New York.
- [22] Han, J.W., Kim, J.H., Lee, H.T., Ko, S.J., 2011. A novel training based auto-focus for mobile-phone cameras. *Consumer Electronics, IEEE Transactions on* 57, 232 – 238.

- [23] He, J., Zhou, R., Hong, Z., 2003. Modified fast climbing search auto-focus algorithm with adaptive step size searching technique for digital camera. *Consumer Electronics IEEE Transactions on* 49, 257–262.
- [24] Jutamulia, S., Asakura, T., Bahuguna, R.D., Guzman, P.C.D., 1994. Autofocusing based on power-spectra analysis. *Appl. Opt.* 33, 6210–6212.
- [25] Kehtarnavaz, N., Oh, H.J., 2003. Development and real-time implementation of a rule-based auto-focus algorithm. *Real-Time Imaging* 9, 197–203.
- [26] Lee, S.Y., Kumar, Y., Cho, J.M., Lee, S.W., Kim, S.W., 2008. Enhanced autofocus algorithm using robust focus measure and fuzzy reasoning. *IEEE Trans. Circuits Syst. Video Techn.* 18, 1237–1246.
- [27] Lenz, R., 1995. Investigation of receptive fields using representations of dihedral groups. *Journal of Visual Communication and Image Representation* 6, 209–227.
- [28] Lenz, R., Bui, T.H., Takase, K., 2005. A group theoretical toolbox for color image operators, in: *Proc. ICIP 05, IEEE*. pp. III–557–III–560.
- [29] Lenz, R., Zografos, V., 2012. RGB filter design using the properties of the Weibull manifold, in: *Proc. 6th European Conference on Colour in Graphics, Imaging, and Vision, Society for Imaging Science and Technology, 7003 Kilworth Lane Springfield, VA 22151 USA*. pp. 200–205.
- [30] Lenz, R., Zografos, V., 2013. Fisher Information and the Combination of RGB Channels, in: S. Tominaga, R.S., Trémeau, A. (Eds.), *Proc. CCIW*, pp. 250–264.
- [31] Lenz, R., Zografos, V., Solli, M., 2012. *Advanced Color Image Processing and Analysis*. Springer. chapter Dihedral Color Filtering.
- [32] Nayar, S.K., Nakagawa, Y., 1994. Shape from focus. *IEEE Trans. Pattern Anal. Mach. Intell.* 16, 824–831.
- [33] Nelder, J.A., Mead, R., 1965. A simplex method for function minimization. *The Computer Journal* 7, 308–313.

- [34] Oller, J.M., 1987. Information metric for extreme value and logistic probability distributions. *Sankhya: The Indian Journal of Statistics, Series A (1961-2002)* 49, 17–23.
- [35] Ong, K.H., Phang, J.C.H., Thong, J.T.L., 1997. A robust focusing and astigmatism correction method for the scanning electron microscope. *Scanning* 19, 553–563.
- [36] Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P., 2007. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press. chapter Golden Section Search in One Dimension. 3rd edition.
- [37] Rao, C.R., 1945. Information and the accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math. Soc.* 37, 81–91.
- [38] Rinne, H., 2008. *The Weibull Distribution: A Handbook*. CRC Press.
- [39] Rudnaya, M., den Broek, W.V., Doornbos, R., Mattheij, R., Maubach, J., 2011a. Autofocus and twofold astigmatism correction in HAADF-STEM. *Ultramicroscopy*, 111, 1043–1054.
- [40] Rudnaya, M., Mattheij, R., Maubach, J., ter Morsche, H., 2011b. Gradient-based sharpness function, in: *International Conference of Applied and Engineering Mathematics*, pp. 301–306.
- [41] Rudnaya, M.E., Morsche, H.T., Maubach, J., Mattheij, R., . A derivative-based fast autofocus method in electron microscopy. *Journal of Mathematical Imaging and Vision* , 1–14.
- [42] Russell, S., Norvig, P., 2003. *Artificial Intelligence: A Modern Approach*. Prentice Hall. 2nd edition.
- [43] Subbarao, M., Tyan, J.K., 1998. Selecting the optimal focus measure for autofocusing and depth-from-focus. *IEEE Trans. Pattern Anal. Mach. Intell.* 20, 864–870.
- [44] Tanaka, N., Hu, J., Baba, N., 1999. An on-line correction method of defocus and astigmatism in haadf-stem. *Ultramicroscopy* 78, 103 –110.

- [45] Tenenbaum, J.M., 1971. Accommodation in computer vision. Ph.D. thesis. Stanford University, Department of Computer Science. Stanford, CA, USA.
- [46] Vollath, D., 1987. Automatic focusing by correlative methods. *Journal of Microscopy* 147, 279–288.
- [47] Yang, G., Nelson, B.J., 2003. Wavelet-based autofocusing and unsupervised segmentation of microscopic images, in: *Intelligent Robots and Systems*, pp. 2143–2148.
- [48] Yanulevskaya, V., Geusebroek, J.M., 2008. Salient region detection from natural image statistics, in: *ASCI*.
- [49] Zografos, V., Lenz, R., 2011. Spatio-chromatic image content descriptors and their analysis using extreme value theory, in: Heyden, A., Kahl, F. (Eds.), *Image Analysis*, Springer. pp. 579–591.