Autofocus on Moving Object in Scanning Electron Microscope

Andrey V. Kudryavtsev, Sounkalo Dembélé and Nadine Piat[1]

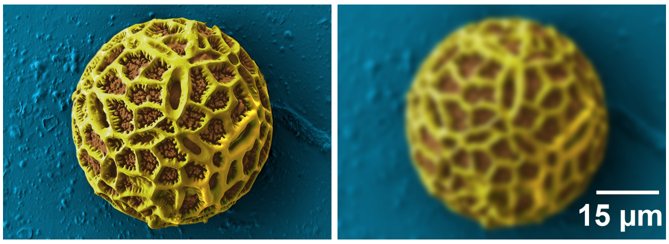
Abstract

The sharpness of the images coming from a Scanning Electron Microscope (SEM) is a very important property for many computer vision applications at micro- and nanoscale. It represents how much object details are distinctive in the images: the object may be perceived sharp or blurred. Image sharpness highly depends on the value of focal distance, or working distance in the case of the SEM. Autofocus is the technique allowing to automatically adjust the working distance to maximize the sharpness. Most of the existing algorithms allows working only with a static object which is enough for the tasks of visualization, manual microanalysis or microcharacterization. These applications work with a low frame rate, less than 1 Hz, that guarantees a low level of noise. However, static autofocus can not be used for samples performing continuous 3D motion, which is the case of robotic applications where it is required to carry out a continuous 3D position measurement, e.g., nano-assembly or nanomanipulation. Moreover, in addition to constantly keeping object in focus while it is moving, it is required to perform the operation at high frame rate. The approach offering both these possibilities is presented in this paper and is referred as dynamic autofocus. The presented solution is based on stochastic optimization techniques. It allows tracking the maximum of the sharpness of the images without sweep and without training. It works under uncertainty conditions: presence of noise in images, unknown maximal sharpness and unknown 3D motion of the specimen. The experiments, that were performed with noisy images at high scan rate (5 Hz), were conducted on a Carl Zeiss Auriga 60 FE-SEM. They prove the robustness of the algorithm with respect to the variation of optimization parameters, object speed and magnification. Moreover, it is invariant to the object structure and its variation in time.

Keywords: Autofocus, SEM, Stochastic Optimization, Gradient Descent

1. Introduction

Autofocus is a very useful feature for all types of visual sensors and, in particular, for Scanning Electron Microscopes (SEM): it makes possible to obtain sharp images with the least human intervention. Two SEM images with sharp and soft focus are represented in Fig. 1. Of the two usual types of autofocus, active with the use of a telemetric system, and passive with the use of images, only the last type is implemented in electron microscopy [1, 2, 3]. Autofocus can be also classified as static or dynamic depending on whether the target object is fixed or moving, respectively, because the same principles are not appropriate for both cases. In electron microscopy, static autofocus is the most widespread because in most applications objects are static. However, with the appearance of new applications in microscopy such as 3D reconstruction that requires smooth acquisition of multiple images with different object positions [4, 5, 6, 7, 8] or robotics in the microscope that requires 3D object tracking, i.e. including depth measurement, [9, 10, 11, 12, 13, 14, 15, 16, 17] static autofocus is no longer appropriate and dynamic autofocus needs to be developed. The object needs to stay in focus for the whole operation time.



(a) (b)

Figure 1: Images of a pollen grain acquired with a Carl Zeiss Auriga 60 FE-SEM: a) image in focus, b) out of focus. Images were coloured manually.

It is important to notice the relation between autofocus and depth estimation using defocus information. Actually these tasks are the same. Yet, the goal of autofocus is to adapt the value of focal distance in the range determined by the depth of field while depth estimation aims for precise value with smallest possible error. The algorithm presented below is initially developed for keeping object in focus, however, it allows to estimate the depth coordinate precisely enough for such applications as automatic manipulation, assembly, etc.

When working with autofocus algorithms, it is important to

Email address: firstname.familyname@femto-st.fr (Andrey V. Kudryavtsev, Sounkalo Dembélé and Nadine Piat)

[1] Authors are with FEMTO-ST Institute, AS2M department, Univ. Bourgogne Franche-Comté, Univ. de Franche-Comté/CNRS/ENSMM, 24 rue Savary, F-25000 Besançon, France.

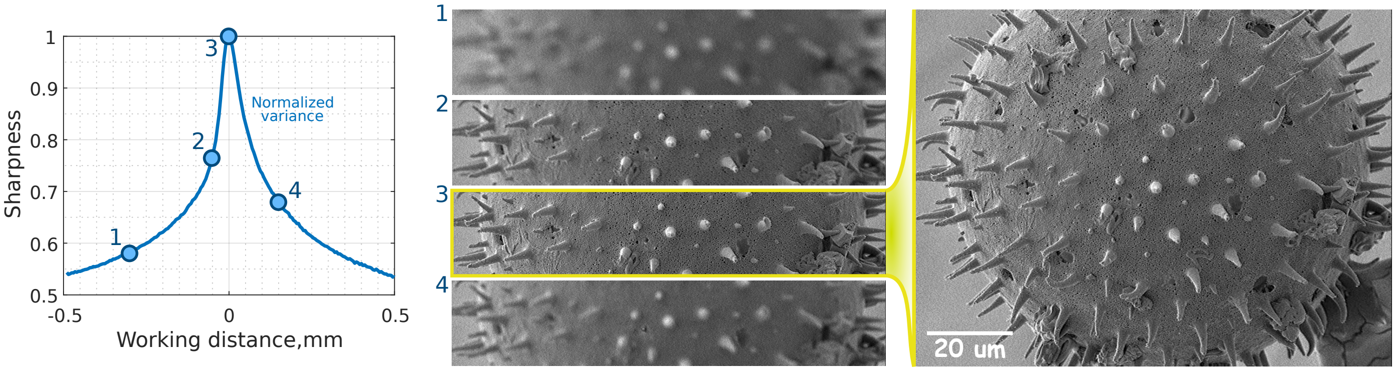


Figure 2: Left: sharpness function for a pollen grain of daisy flower (*Bellis Perrenis*). Middle: image ROIs for points in sharpness function. Right: in-focus image that corresponds to point 3 on sharpness function. Magnification: ×1000.

give a definition of an in-focus image. It can be defined as follows: any change of focal distance, or working distance in case of SEM, will not give a sharper image than the in-focus image. Many works are devoted to the modelling of defocus and, as a result, it is often represented as a convolution of in-focus image with a blur function [18]:

$$I = I' * h \tag{1}$$

where I' denotes the in-focus image, h is a blur function that depends on the focal distance, I is the observed defocused image. The goal of autofocus is to find the focal distance that minimizes the blur in the image and leads to $I = I'$.

At this point an important question arises: how can the level of blur be measured. There exist many different techniques based on statistical information, image gradient, Fourier or wavelet transforms. The dependence of image sharpness from the depth variation or equivalently from focal distance variation is called sharpness function. It has a special form characterized by the presence of maximum in the point where image is in focus. Depending on the object form, the function may have several maxima if object parts are located at different distances from the visual sensor. A very extensive comparative study of focus measure operators for general scenes was realized by Pertuz *et al.* in [19]. As for microscopy domain, one can refer to the study made by Rudnaya *et al.* [3]. After evaluating several sharpness functions, the normalized variance was selected:

$$S(I) = \frac{1}{MN}\frac{1}{\mu}\sum_{M}\sum_{N}(I(u,v)-\mu)^2 \tag{2}$$

where $S(I)$ is a sharpness of image I, μ is the mean of intensity values, M and N are image width and height, respectively. An example of sharpness function as well as some images at different defocus values are presented in Fig. 2. The object is a pollen grain of daisy flower (*Bellis Perennis*).

It is important to notice that there exist two different classes of autofocusing tasks. Firstly, many works deal with situation where $I' = I'_{const}$ which occurs only when the object is not moving. Autofocus in such case is referred as static. As the sharpness is a function of the working distance, autofocus may be considered as a problem of optimization: the search for the peak of sharpness along the optical axis. When the object is not moving, this function is constant and the desired image I' stays the same. Besides the simple method of sweeping all possible values of working distance and choosing the one with the best sharpness score, two main types of static autofocus can be distinguished. They differ on whether the model is used explicitly or not. In the first type, a fitting method is used to estimate the peak: Nicolls *et al.* [20] assumed a Gaussian model and used two algebraic equations from three measurements, Wu *et al.* [2] assumed a quasi-Gaussian model and used least squares fitting from five measurements, Rudnaya *et al.* [21] assumed a quadratic model and used ordinary least squares fitting from three measurement. These methods are not adapted for high level of image noise as they highly rely on the model estimated from training data. In paper [22], coarse-to-fine hill climbing method is used to obtain the initial guess of peak, then, robust least squares fitting, support vector machines, allows accurate estimation of the peak. Although these methods have a good level of precision, their main drawback is the execution speed. In the second case, where there is no training data, the peak is estimated directly: in several works, coarse-to-fine hill climbing method [23, 24, 25] or Fibonacci search ([26]) were used. These methods are fast enough but lack of accuracy.

Second, autofocus on the object that is in motion, dynamic autofocus, where I' varies in time, $I' = I'(t)$. Previously mentioned methods can not be applied in this case as the in-focus image is not constant. It also implies that dynamic autofocus represents the continuous search for maximum value of sharpness. This process is equivalent to estimate the depth coordinate using defocus information. In recent work [27], Marturi *et al.* proposed a method of depth estimation in SEM which is based on visual servoing of focus. The object is moving gradually: the motion is stopped when the object is out of depth of field which reactivates the autofocusing task. When the autofocus is done, the object is allowed to continue its motion. Thus, this scheme allows to keep the object in focus at some discrete positions but not during the whole operation procedure. It can also be seen as static autofocus at discrete intervals of time. Another solution to dynamic autofocus is presented by Le Cui *et al.* in [28].

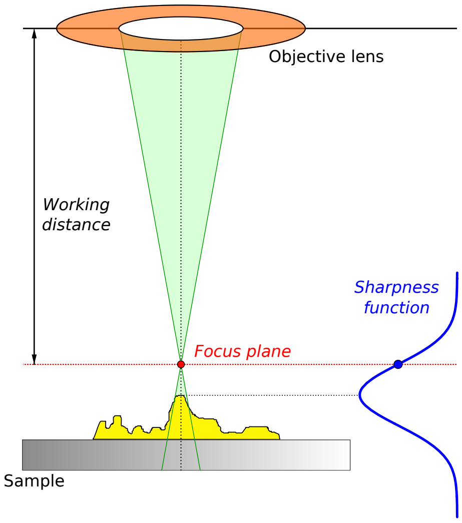


Figure 3: Principle of focusing in SEM.

Authors work on three-dimensional tracking inside SEM and the problem of keeping object in focus is also treated as the problem of depth estimation. The relation between the image gradient and depth variation is approximated with quadratic rational function. The coefficients of this function are determined by fitting the training data which implies preliminary scanning of the scene (taking images at different sample positions) before the movement can be started. The parameters are then updated using the particle filter during the operation which allows to recover the depth coordinate.

This paper presents a method of *keeping* object in focus online, during its movement, both in translation and rotation. It corresponds to the situation where neither in-focus image nor blur function are constant. The proposed method, being based on online stochastic optimization, has the following advantages compared to literature solutions:

- absence of calibration step, no model is used, i.e. there is no need for training data, no focus sweeping of the scene;
- the algorithm is invariant to the object structure which is directly derived from the previous point;
- no scanning procedure during operation, only two images are used to estimate best focus position (depth variation);
- being robust to noise, it allows to work with high frame rate (approximately 5 Hz) that is confirmed by experiments;
- adaptive to the variation of object speed.

The remainder of the paper is organized as follows: Section 2 presents the experimental setup as well as the analysis of SEM image and its dependence on the frame rate. Influence of object movement is also demonstrated. Section 3 describes two main parts of the algorithm: first-order derivative approximation and optimization for non-stationary sharpness function, i.e. that vary in time. Experimental results showing the algorithm performance in different conditions are presented and analyzed in Section 4.

2. Imaging conditions in SEM

The experimental setup of this paper consists of a Carl Zeiss Auriga 60 FE-SEM along with the computer and the control software written in C++. Auriga has Schottly field emission Gemini electron column that converges the beam towards the sample surface (Fig. 3). The distance between the focus plane i.e. the point of convergence and the lower pole of the objective lens defines the working distance that represents the equivalent of focal distance for SEM. When the sample is located at the focus plane, the image sharpness reaches its maximum (peak). The distance around the peak, where the image remains sharp, is called the depth of field. It should be emphasized that the process of image formation in SEM is different from classical cameras: the image is obtained by scanning the surface using an electron beam, thus, only one pixel is acquired at a time. As it will be shown further, the time of image acquisition (cycle time) influences greatly its quality. SEM Zeiss Auriga 60 has a special parameter (ScanSpeed) that defines the frame rate. It varies from 1 to 16: the lower the number the bigger the frame rate is. For an image with size of 1024×768 pixels, ScanSpeed 1 and 8 correspond to a frame rate of 17.54 Hz and 0.1 Hz, respectively.

2.1. Frame rate influence on image noise

As it was mentioned previously, the goal of this work is to be capable to maintain an object in focus inside SEM during operation, i.e. while object is moving. Thus, the frequency of image acquisition needs to be much higher than for classical visualization tasks. Yet, smaller cycle time (bigger frame rate) leads to decreasing image quality. In order to demonstrate the relation between frame rate and image noise the following experiment was conducted: several images were acquired at different frame rates while measuring the sharpness. On every image, there is a region corresponding to the background that is completely out of depth of field (upper left corner 50×50 pixels, Fig. 4). Therefore, the standard deviation of intensity values in this region should be close to zero and then reflects the amplitude of noise in the whole image. It is confirmed by experimental results (Fig. 4, Table 1): with increasing frame rate, the noise amplitude grows. For instance, with acquisition frequency of approximately 4.5 Hz (cycle time of 220 ms), the standard deviation of intensity values is 34.67, which is about 13% of the maximal value (255). Another way to show the influence of scanning speed on image is to make images of static scene with different frame rates. The resulting variations of sharpness are shown in Fig. 5: bigger cycle time gives smaller oscillations of sharpness value. These results show the complexity of sharpness estimation in the case of high frame rate which should be taken into account by the algorithm. For the remainder of this paper, we will work with ScanSpeed2 that corresponds to 4.5 Hz.

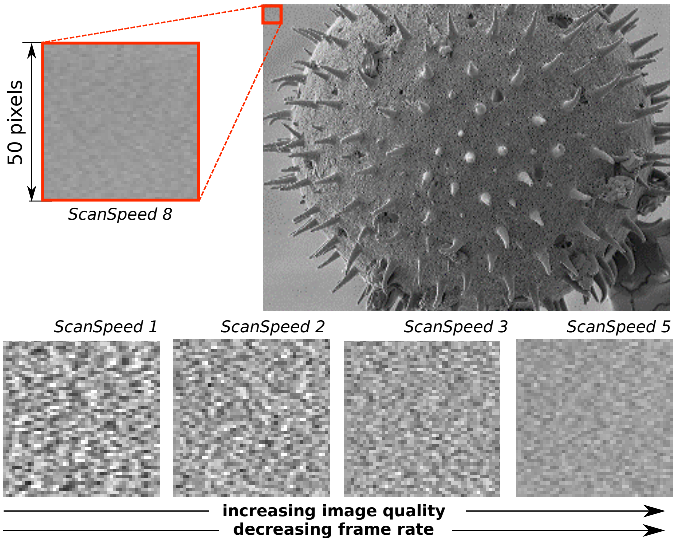


Figure 4: Influence of frame rate on image quality. Images represent the region of interest of 50 × 50 pixels from original frames acquired at different frame rates.

Table 1: Characteristics of image acquisition in a Carl Zeiss Auriga 60 FE-SEM. Values are given for the images size 1024× 768 pixels. Size of region of interest (ROI) is 50 × 50 pixels.

Scan speed	Cycle time, ms	Frame rate, Hz	Standard deviation of intensity values in ROIs
1	57	17.54	41.90
2	**220**	**4.55**	**34.67**
3	380	2.63	26.78
5	1400	0.71	13.81
8	10600	0.09	5.08

2.2. *Influence of movement on sharpness*

Another matter that influences the sharpness function is the object movement. For instance, in case of pure Z displacement, the form of the sharpness function should not have been changed. Hence, due to several factors, it is not true. The form and the value of maximum sharpness score change due to, first of all, the dependence of the depth of field on the working distance. Another factor is the change of brightness and contrast because of the lower electron energy when the object moves away from the column. This effect is amplified when the object performs a movement that is more complex, such as out of plane rotations, which is illustrated in Fig. 6. This figure represents four sharpness functions acquired at different orientations of the pollen grain. The observation is that the maximum value may be multiplied twice even in case of 5 degrees rotation. The importance of this remark is crucial because it shows, that, for dynamic autofocus, one can not use the techniques based on the error between current and desired sharpness. The desired sharpness, i.e. the maximum of the curve, is never the same if object is moving. Due to the fact that the rotation axis was not aligned with the center of the pollen grain, the maximums of sharpness correspond to different values of working distance.

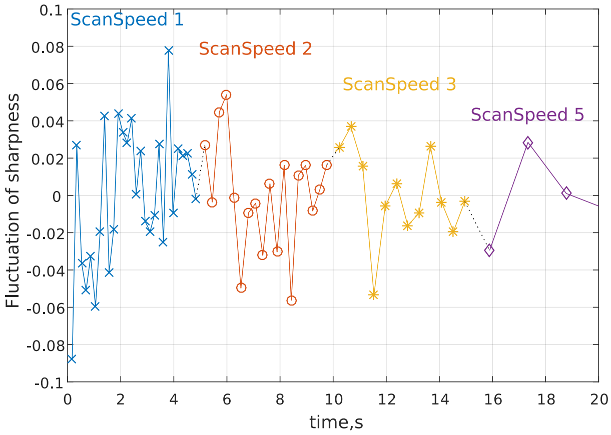


Figure 5: Sharpness variation for a static scene at different frame rates.

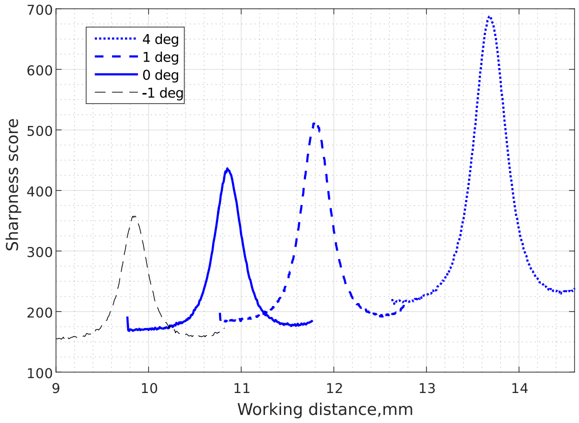


Figure 6: Sharpness functions for the same *Bellis Perrinis* scene in different orientations. Rotation was performed around horizontal axis of the image. The depth variation is due to misalignment of rotation axis with the object center.

3. Methods

As it was stated previously, the presented approach of dynamic autofocus is based on mathematical optimization. Optimization is the process of finding the minimum or maximum of a function $f(\theta)$ by varying its input parameter θ:

$$\min_{\theta \in \mathbb{R}} f(\theta) = f(\theta^*) \tag{3}$$

where θ^* is the value of θ that minimizes $f(\theta)$ subject to θ^* satisfying a set of constraints. The function $f(\theta)$ is often referred as objective or loss function; θ is the varying or input parameter.

In the context of the present work, the objective function is the sharpness function. In our case, the function is changing in time, i.e. is non-stationary. The input parameter is the working distance. The goal of the work is to keep the sharpness in maximum, thus, continuously update the value of θ, so that:

$$\theta_n = \arg\max_{\theta\in\mathbb{R}} f_n(\theta) = \arg\min_{\theta\in\mathbb{R}}(-f_n(\theta)) \tag{4}$$

where θ_n is the current value of the working distance that maximizes the current image sharpness ($f_n(\theta)$). As it was mentioned above, the objective function may contain one or several maxima. However, considering that the starting point of the dynamic autofocus is a well focused image (not necessary the best focused one), the function may be considered convex in the neighborhood of the maximum point. The size of this neighborhood is equal to the current depth of field of the microscope.

The remainder of this section presents the reflections that allowed to choose and adapt the optimization algorithm. There are two main types of optimization algorithms: deterministic and stochastic. According to the classification given by Spall in [29], the deterministic algorithms are characterized by the fact that the objective function is known (as well as its derivatives) and that this information is used to determine the search direction at every iteration. In contrast, stochastic optimization deals with two types of problems: either the algorithm of optimization itself makes a random choice of direction or the function is not known but only noisy measurements are available, which is the case of sharpness function.

Another criterion that allows to differentiate optimization algorithms is whether they use only function evaluations, first order derivative (gradients) or second order derivative (Hessians). For the algorithms that belong to the first group, most of them, such as Golden-section search [30], are based on the reduction of the interval that contains the maximum. They are not suitable for our application because the objective function is not stationary. The second group contains the approaches based on derivative. In autofocus problem, neither the sharpness function nor its derivatives are available. The only possible solution is to use the approximations which is not readily apparent for the unknown function in the case where only noisy measurements can be obtained. The approach allowing to approximate the first order derivative is presented in Section 3.1. The presence of noise makes irrelevant the idea to use second-order approximation: apart from the fact that it requires more images for one Hessian estimation, from the experiments presented above, its value would likely be unusable due to the high noise level at high frame rate. All these factors confine the choice of the optimization algorithms to the first-order methods. The most used of them is the gradient descent or ascent in our case (the difference consists only in the movement direction). Moreover, there exist several solutions allowing to improve its performance. Among them are Momentum [31], AdaGrad [32], RMSProp [33], Adam [34]. Having different properties and parameters, they are analyzed in Section 3.2. The last one is then adapted for the task of dynamic autofocus.

3.1. Derivative approximation for non-stationary function

This section shows how the first-order derivative of the sharpness function is approximated. When the derivative of the function is not known, it is possible to approximate it using directly its values in different points. The approximations that are usually used for stationary functions are Forward Euler, Backward Euler and Centered difference. These approximations, while being equivalent in continuous case, lead to different results in the discrete one. Hence, using decomposition in Taylor series, it can be demonstrated that the centered difference gives the best approximation in terms of truncation error. It represents the error between the actual derivative value and its approximation. It is also important to notice that in all cases only two measurements are needed, i.e. two images taken at different values of working distance. Thus, the centered difference approximation is beneficial. It has the following form:

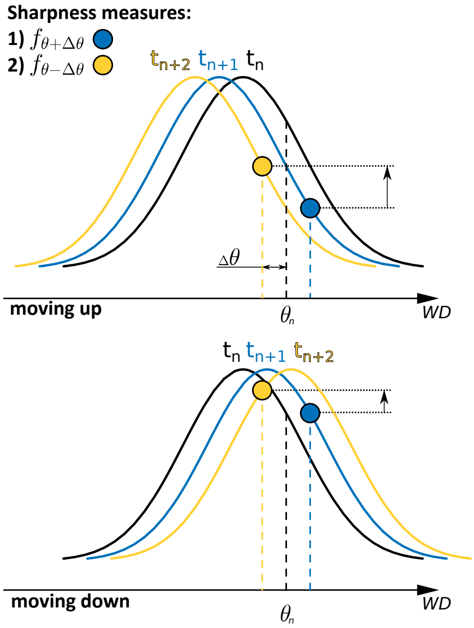


Figure 7: Derivative approximation. When using the same order (first $+\Delta\theta$, then $-\Delta\theta$), the derivative sign is false when the object is moving down.

$$f'(\theta_n) = \frac{f(\theta_n + \Delta\theta) - f(\theta_n - \Delta\theta)}{2\Delta\theta} \tag{5}$$

However, at this step, an important feature of dynamic autofocus needs to be taken into account: the sharpness function is non-stationary. For instance, using the formula of centered difference (Eq. 5), one has to make two estimations of the objective function. It means that the delay between two measurements is equal at least to the acquisition time of one image. Thus, depending on the moving direction, the order of measurements, $f(\theta + \Delta\theta)$ and then $f(\theta - \Delta\theta)$ or vice versa, can greatly influence the value of the approximated derivative. This aspect is illustrated in Fig. 7. The three sharpness curves correspond to different positions of an object in time. At time moment t_n the value of working distance was equal to θ_n. In order to approximate the derivative at this point, two measures are needed: at points $\theta_n + \Delta\theta$ and $\theta_n - \Delta\theta$. Assuming that they are performed at time moments t_{n+1} and t_{n+2} respectively, one can get the estimate of the derivative. It should be noted that the most important part in derivative approximation is its sign and not the amplitude because the error in the amplitude can be compensated during optimization. However, the estimate of the derivative would correspond to the point where the object was two images ago while the decision about the update of working distance should be taken at instant t_{n+2}. Getting back to Fig. 7, one can remark that in the upper case, when the object is moving up, the sign of approximated gradient is negative ($f_{\theta+\Delta\theta} < f_{\theta-\Delta\theta}$) that corresponds to the actual slope of sharpness function at t_{n+2}. However, when object is moving down (Fig. 7, lower panel), it is no longer true, thus, the update of working distance would be done in false direction. To tackle this issue, we propose to use a random variable, $d = \pm 1$, that will determine the order of function evaluations, or simply the sign of $\Delta\theta$. The value of this variable is generated with Bernoulli probability distribution. This approach was also used in SPSA algorithm (Simultaneous Perturbation for Stochastic Approximation) introduced for the first time in [35]. However, authors work with multivariable stationary functions and use the random variable vector to disturb all gradient directions at the same time: it allows to reduce the number of function evaluations which makes the optimization more efficient. In this paper, being applied to a non-stationary function, it allows to be independent of the direction change of the function variation. It should also be noted that for small displacement speeds, when the speed tends to zero, the order of function evaluations will not change the sign of the gradient. To summarize, the final expression for the first order derivative of the sharpness function has the following form:

$$f'(\theta_n) = \frac{f(\theta_n + d\Delta\theta) - f(\theta_n - d\Delta\theta)}{2d\Delta\theta} \tag{6}$$

where the evaluation of $f(\theta_i + d\Delta\theta)$ is performed first and then $f(\theta_i - d\Delta\theta)$.

Finally, some remarks need to be added about the value of $\Delta\theta$. The goal of the present work is to keep an arbitrary object in focus during its movement. It means that, while estimating the derivative, the object must also stay in focus. Thus, the values of the working distance $\theta \pm \Delta\theta$ should lie in the depth of field, i.e. the value of $\Delta\theta$ must be at least twice less than the depth of field.

3.2. Proposed algorithm

The most common first order algorithm allowing to find the minimum of a function is the gradient descent. In the present work, the objective function has only one input parameter, thus, it is more correct to speak about derivative. However, all the algorithms presented below are also true for a multivariable case. Therefore, to simplify the notations, the approximation of the first order derivative taken with the negative sign (as we look for the maximum) will be denoted as $\hat{g}$ and referred as gradient:

$$\hat{g} = -f'(\theta_n) \tag{7}$$

The update rule of the gradient descent algorithm is the following:

$$\theta_{n+2} = \theta_n - \alpha\hat{g} \tag{8}$$

where α denotes the gain or learning rate. Its value determines how important is the update in one iteration. In the context of the present work, θ_{n+2} represents the estimate of the working distance that would give the best value of image sharpness. As the evaluation of $\hat{g}$ takes two images, the time elapsed between θ_{n+2} and θ_n is twice the time of one image acquisition, that is why at odd time moments ($t_{n+1}, t_{n+3}, t_{n+5}$) the update is not performed. In general, gradient descent achieves good results when the objective function in not corrupted by noise, which is not true for SEM images sharpness. When there is an important change in the gradient value, which is chaotic due to noise, the algorithm will change dramatically the value of working distance and lose the focus. Another drawback of gradient descent consists in high dependence on the value of α. If the gain is too low, the convergence speed will also be low. In the case of autofocus it would greatly limit the maximum displacement speed. In contrast, if the gain is too high, the algorithm may suffer from oscillations about the maximum value. Therefore, several techniques were proposed in the literature to improve the performance of gradient descent.

Momentum. The first method is based on the following idea: if the sign of gradient does not change for a certain amount of time, i.e. the update direction stays the same, the update in this direction can be accelerated. It gives the following update rule:

$$\begin{aligned} m_{n+2} &\leftarrow \mu m_n - \alpha\hat{g} \\ \theta_{n+2} &\leftarrow \theta_n + m_{n+2} \end{aligned} \tag{9}$$

where m is a first moment variable. Thus, instead of integrating the gradient, the velocity is integrated. The acceleration depends on the factor $\mu \in (0, 1)$. Its typical value is 0.9. This algorithm allows to improve the convergence speed and prevents the value of θ from chaotic jumps. However, it is not suitable for non-stationary functions. Assuming that the object is moving in one direction for some time and then changes it, the algorithm would not be able to respond quickly, thus, the focus would be lost.

AdaGrad. In this case the learning rate is adaptive. It scales the current value of gradient according to the history of squared gradient values for previous iterations:

$$\begin{aligned} v_{n+2} &\leftarrow v_n + \hat{g}^2 \\ \theta_{n+2} &\leftarrow \theta_n - \alpha\frac{\hat{g}}{\sqrt{v_{n+2}}+\varepsilon} \end{aligned} \tag{10}$$

where ε is a small constant (typical 10^{-8}) allowing to avoid division by zero at first iterations, v is a second moment variable. Despite the robustness of this algorithm, it is also not adapted for non-stationary functions: the history of gradient is stored for the whole time of optimization. As a result, it has the same drawback as Momentum, impossibility to quickly respond at the change of movement direction.

RMSProp. The idea proposed here is not to store all values of the gradient but use the exponentially weighted moving average:

$$\begin{aligned} v_{n+2} &\leftarrow \beta v_n + (1-\beta)\hat{g}^2 \\ \theta_{n+2} &\leftarrow \theta_n - \alpha\frac{\hat{g}}{\sqrt{v_{n+2}}+\varepsilon} \end{aligned} \tag{11}$$

where $0 \leq \beta < 1$ is the parameter that determines how many previous gradients would be taken into account and with which weight factor. For instance, if $\beta = 0$ only the current estimate of the gradient will be used, and the algorithm will perform the update in its direction by the value of α. It is worth to note that RMSProp is invariant to the scale of the gradient, as in previous example when $\beta = 0$. In practice, the value of β is taken equal to 0.9 or 0.99. This is the first algorithm that has the necessary properties for non-stationary functions: filtering of chaotic jumps in the gradient values, robustness and quick response on function variations (if the value of β is chosen correctly).

Adam. This recently introduced method stands for adaptive moment estimation. In addition to store the exponentially moving average of squared gradients like RMSProp, it also stores the exponentially moving average of the gradient itself:

$$\begin{gathered} m_{n+2} \leftarrow \beta_1 m_n + (1-\beta_1)\hat{g} \\ v_{n+2} \leftarrow \beta_1 v_n + (1-\beta_2)\hat{g}^2 \\ \theta_{n+2} \leftarrow \theta_n - \alpha \frac{m_{n+2}}{\sqrt{v_{n+2}}+\varepsilon} \end{gathered} \tag{12}$$

The update rule is very similar to RMSProp, however, not the noisy gradient estimate $\hat{g}$ is used, but its averaged value m. It allows better filtering of the gradient while keeping the functionality needed for non-stationary function optimization. Typical values of parameters are: $\beta_1 = 0.9, \beta_2 = 0.99, \varepsilon = 10^{-8}$. It is this algorithm that was used for the experiments.

The final algorithm for dynamic autofocus is represented in Algorithm 1. It should be noted that during operation, actual working distance is never equal to θ. Instead, the algorithm sets the working distance to $\theta \pm \Delta\theta$ to continuously estimate the derivative. In other words the actual value of working distance oscillates around the best focus position, which is updated every two images.

Algorithm 1 Dynamic autofocus in SEM

```
step ← 1
$\alpha \leftarrow 0.004$                              ▷ optimization parameters
$\beta_1 \leftarrow 0.6$
$\beta_2 \leftarrow 0.6$
$\Delta\theta \leftarrow 10^{-6}$
$\varepsilon \leftarrow 10^{-8}$
$m \leftarrow 0$
$v \leftarrow 0$
$\theta \leftarrow \theta_0$                            ▷ initial working distance
$d \leftarrow 1$                               ▷ evaluation direction
set working distance $\theta = \theta + d\Delta\theta$
while autofocus activated do
    acquire image $I_n$
    get sharpness score $s \leftarrow S(I_n)$ (Eq. 2)
    if step = 1 then
        evaluate $f(\theta + d\Delta\theta) = s$
        set working distance $\theta = \theta - d\Delta\theta$
        step ← 2
    else
        evaluate $f(\theta - d\Delta\theta) = s$
        estimate gradient $\hat{g}$ (Eq. 6, 7)
        $m \leftarrow \beta_1 m + (1-\beta_1)\hat{g}$
        $v \leftarrow \beta_2 v + (1-\beta_2)\hat{g}^2$
        $\theta \leftarrow \theta - \alpha \frac{m}{\sqrt{v}+\varepsilon}$
        $d \leftarrow rand(-1, 1)$
        set working distance $\theta = \theta + d\Delta\theta$
        step ← 1
    end if
end while
```

4. Results and discussion

The validation of the presented theoretical aspects was confirmed by three groups experiments: translation along Z axis (optical axis of the camera), translation along Z axis with varying parameters (optimization, speed, magnification), and rotational movement. The object used is a pollen grain of daisy flower that will be further referred as *Bellis Perennis*. The object is mounted on the robot (6 degrees of freedom) installed inside the SEM. For the rotational movement, additionally to the *Bellis Perennis*, the algorithm was tested on a scene with different objects including pollen grains: *Pollen Grains*. Both objects were coated with a layer of gold. The equipment used is a SEM Carl Zeiss AURIGA 60. The following parameters were constant for all of the experiments: acceleration voltage 3 kV, aperture size 30 μm. The frame rate was also constant, with the value of 4.5 Hz, that corresponds to ScanSpeed2.

Before starting the description of the experiments, it is important to give some information about the initialization of the algorithm. Three values are to be defined: m_0, v_0 and θ_0. The values of m_0 and v_0 are equal to zero. As a drawback, it leads to the fact that the values of m_n and v_n are biased towards zero at the initial steps as mentioned in [34]. It can be seen from the experiments that there are some oscillations at the first 5-15 frames in the values of estimated depth that disappear afterward (Fig. 8). Thus, we consider that it would be a good idea to activate the dynamic autofocus several images before the movement starts as it was done is one of the experiments. Another important variable is θ_0. The dynamic autofocus represents the tracking of the best sharpness position. A good choice would be a value of θ_0 that is close to the peak, i.e. in the depth of field. If the value is chosen far from maximum, the algorithm *may* catch up with the best focus value, but the convergence speed needs to be bigger than the movement speed. In addition, a different set of parameters should be used and nothing can guarantee that this set will still be optimal when the maximum will have been reached.

Translation along Z axis. The first experiment consisted in the object performing a translation along Z axis (along the optical axis of the camera), Fig. 8. The speed was defined as a sine function with its maximum of 10μm/s. It allowed to compare the results of dynamic autofocus with actual displacement which was known from the proprioceptive detectors of the robot. Results demonstrate that not only the object was correctly positioned inside the depth of field but it was performed with high accuracy (for the given example the depth of field was about 40 μm). The standard deviation of error is about 5 μm while the object dimensions are about 100 μm, magnifica-

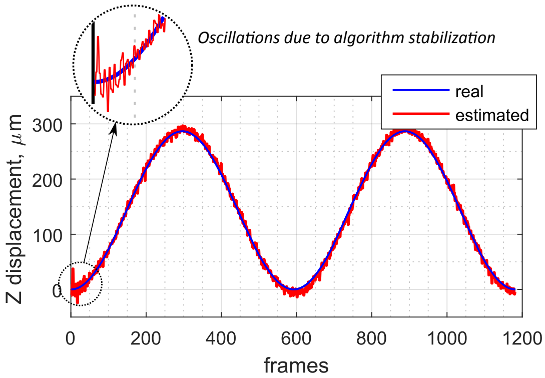


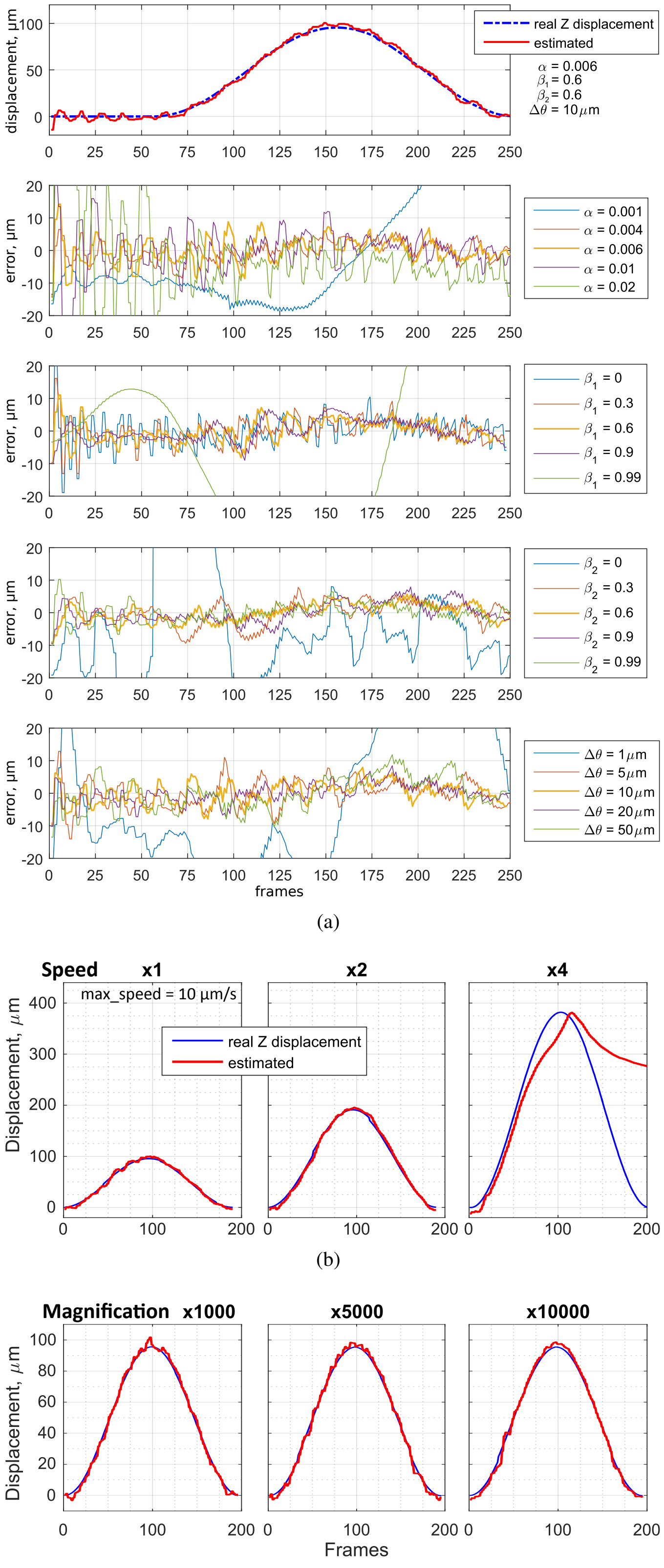
Figure 8: Results of dynamic autofocus in SEM. Object is performing a translation along Z axis, the software adapts the working distance to keep it in focus. Object: pollen grain *Bellis Perennis*. Magnification: ×1000. Maximal speed: 10 μm/s.

tion ×1000. It allowed to confirm the viability of the proposed approach. The next step was the test that allowed evaluation of the robustness of the proposed solution. For that, several experiments were conducted by varying the parameters of the algorithm: with the best set of parameters, the standard deviation of error is 3.4 μm. In the next paragraphs, the individual influence of each parameter as well as the choice of its value are presented.

Varying optimization parameters: $\alpha, \beta_1, \beta_2, \Delta\theta$ (Fig. 9(a)). While all optimization parameters were subject to change, the maximal speed and magnification were held constant: 10 μm/s and ×1000, respectively. First, the gain, or the learning rate, α have been varied. If the chosen value is too small, the algorithm fails to follow the best sharpness position very fast and finally loses it. In contrast, if the value is too big, one can notice a high level of oscillations around peak position. Even if it may be acceptable in some cases, e.g. while oscillations stay in the depth of field, this behavior is undesirable. Finally, even with variations of about 50% of α, the error stays in a ± 1 μm range, which is a good result. The intuition behind the choice of the value of α is the following: with high probability, after two iterations (derivative estimation) the object should stay in the region $\pm\alpha$ μm. Thus, this parameter relies on the displacement speed that will be analyzed below.

Second experiment was devoted to variation of parameter β_1 that defines how much the current update will depend on the previous values of gradient. When β_1 is equal to zero, the algorithm becomes equivalent to RMSProp and one can observe oscillations in the value of the error. Bigger values of β_1 allow to damp these oscillations while keeping high dynamic of the system, i.e. it correctly reacts to the change of speed direction. However, if the parameter is close to one, the system loses this indispensable feature. Thus, the retained value of $\beta_1 = 0.6$ allows to keep the advantages of momentum method without losing the possibility to work with non-stationary functions.

Third, as the algorithm depends on the exponentially decaying average, the parameter β_2 was analyzed. Its value indicates how many previous values will be taken into account. The value



(a)

(b)

(c)

Figure 9: Performance of dynamic autofocus algorithm with varying: a) optimization parameters, b) speed, c) magnification.

of zero corresponds to the situation where only the last derivative approximation is used. As it was mention previously, the results in this case are unacceptable due to high level of noise in images. This is also confirmed in Fig. 9(a). In contrast, the value close to one may give the same consequences as in case of β_1 is close to one, loss of system dynamics. However, if we are sure that the object has a constant speed, this situation may be advantageous. In the present case, the speed profile is a sine function and the values of β_2 in the interval of $[0.3, 0.9]$ give very similar results.

Finally, the impact of $\Delta\theta$ was identified. In order for object to stay in focus, the value of $\Delta\theta$ must be at least twice less than the depth of field, as it was mentioned in Section 3.1. In SEM, the depth of field can be estimated using the following expression:

$$DOF = \frac{2\delta_{screen} WD}{AM} \tag{13}$$

with working distance WD, aperture A and δ_{screen} as the image pixel size. Thus, the value of $\Delta\theta$ must respect the following inequality:

$$\Delta\theta < \frac{DOF}{2} = \frac{2\delta_{screen} WD}{AM} \tag{14}$$

Yet, we observed that, at least for the SEM used in this work (Carl Zeiss AURIGA 60), this formula should be used as an indication and not to obtain a precise value. By experiment, the better choice of $\Delta\theta$ would be the value at least four times smaller than the depth of field.

Varying object speed, Fig. 9(b). In this example, once again, the object had the sine function as speed profile. The maximal speed for each test was 10, 20 and 40 μm/s, respectively. All optimization parameters as well as magnification (×1000) were held constant, only the object speed was subject to change. The results show that the algorithm performs well when the speed is multiplied by a factor of two. However, it fails when the speed has a multiple of four. To overcome this, it is necessary to adapt the value of the parameter α. In general, the maximum object speed, for which dynamic autofocus is still viable, is principally limited by the value of frame rate that was equal to 4.5 Hz in this experiment.

Varying magnification, Fig. 9(c). In the next experiment, the algorithm was subject to changing magnification while the speed was the same. Three different values were tested: ×1000, ×5000, ×10000. The approach presents a high level of robustness. The value of standard deviations for all three cases is about 2 μm. For further increasing of magnification, the value of $\Delta\theta$ should be adapted as the depth of field becomes smaller with growing magnification (Eq. 13).

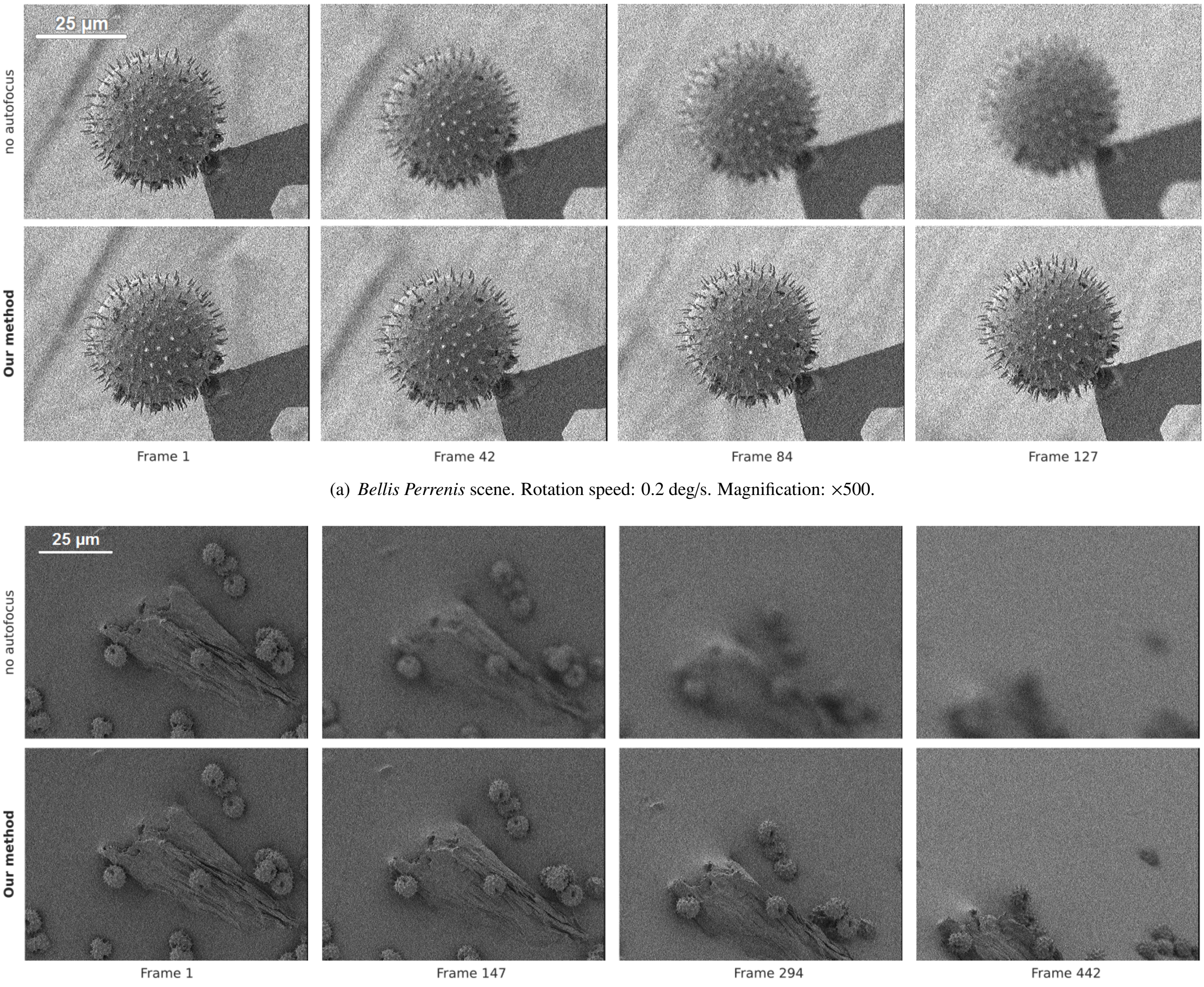
Rotating objects (Fig. 10). The last experiment was conducted to test the performance of the algorithm on rotating objects. Two scenes were used: *Bellis Perennis* and *Pollen Grains*. Rotation speed was constant and to 0.2 degrees per second. Magnification was ×500 and ×400, respectively. For the first object, *Bellis Perennis*, the total rotation was 15 degrees. The scene *Pollen Grains* rotated to approximately 60 degrees. It should be noted that the center of the scene was not aligned with the rotational axis of the robot. Thus, when the robot performs the rotational movement, the object rotates but not precisely about its center. It means that the rotational movement of one robotic axis results in a more complex movement of the object, i.e. rotation combined with uncontrollable translations. That is why, without autofocus, the object goes out of the depth of field. In contrast, when the dynamic autofocus is activated, the image stays sharp during the whole movement even when the scene highly differs from the beginning operation to the end like in the case of *Pollen Grains*: at the final frames, after the rotation of 60 degrees the scene was very different from the initial one, and only 10% of the image actually contained some visual information. It demonstrates that the algorithm is not only invariant to the scene itself but also to its change during operation.

5. Conclusion

Dynamic autofocus is a very useful feature when working with SEM. It helps to avoid readjusting of working distance after every position or orientation change. This feature also finds its application in micro/nanopositionning as it allows getting the information about the depth that is hard to retrieve using classical image processing. This paper presents an autofocus algorithm for scanning electron microscope that allows dealing with sample performing a continuous 3D motion at high frame rate. The complexity of the task was demonstrated through the tests made on Carl Zeiss Auriga 60 FE-SEM: with growing frame rate, the level of noise in images becomes bigger that makes difficult the measure of sharpness. Moreover, sample displacement results in unknown variations of maximal sharpness, i.e. the sharpness function is non-stationary with respect to the change of object position or orientation. For instance, in such conditions, static autofocus techniques are not applicable.

The paper describes an approach based on stochastic optimization of non-stationary sharpness function. An improved version of gradient descent (ascent) method was used along with the derivative approximation adapted for a non-stationary function. The introduction of averaging parameters allowed to filter the random variations of the gradient because of noise without loss of dynamic properties. The experiments on the Carl Zeiss Auriga 60 FE-SEM validated the algorithm. It is robust to variation of optimization parameters, to those of the scanning electron microscope, in particular the magnification and the frame rate, and to those of the sample. The computational cost of the algorithm is very low: calculation of normalized variance plus several algebraic expressions. The only step of initialization consists in getting the object in soft focus that was done manually in all presented experiments. No camera calibration, intrinsic or extrinsic, is needed. Moreover, the algorithm works under the conditions of high frame rate (5 Hz) and very noisy images. With optimal parameters, it was possible to track the object with displacement speed up to 20 μm/s (at ×1000 magnification) and 0.2 deg/s with a frame rate of 5 Hz and a format of 1024 x 768 pixels.

Further work will consist in combining of the presented approach with 2D visual servoing for realization of automatic nanomanipulation and image acquisition inside SEM.



(a) *Bellis Perrenis* scene. Rotation speed: 0.2 deg/s. Magnification: ×500.

(b) *Pollen grain* scene. Rotation speed: 0.2 deg/s. Magnification: ×400.

Figure 10: Dynamic autofocus algorithm on rotating objects in SEM.

Acknowledgment

This work has been supported by the Labex ACTION project (contract ”ANR-11-LABX-0001-01”), Equipex ROBOTEX project (contract ”ANR-10-EQPX-44-01”), the cross border project CITHADEL of Interreg France-Switzerland program through the European regional development fund. Authors would like to thank ChronoEnvironment Lab (Besançon, France) for providing samples of pollen grains, Patrick Rougeot[1] and Olivier Lehmann[1] for their help with sample preparation and image acquisition.

[1] R. Nishi, Y. Moriyama, K. Yoshida, N. Kajimura, H. Mogaki, M. Ozawa, and S. Isakozawa, “An autofocus method using quasi-gaussian fitting of image sharpness in ultra-high-voltage electron microscopy,” *Journal of Microscopy*, vol. 62(5), pp. 515–519, 2013.

[2] Z. Wu, D. Wang, and F. Zhou, “Bilateral prediction and intersection calculation autofocus method for automated microscopy,” *Journal of Microscopy*, vol. 2048, no. 3, pp. 271–280, 2012.

[3] M. Rudnaya, R. Mattheij, and J. Maubach, “Evaluating sharpness functions for automated scanning electron microscopy,” *Journal of microscopy*, vol. 240, no. 1, pp. 38–49, 2010.

[4] L. Quan, “Self–calibration of an affine camera from multiple views,” *International Journal of Computer Vision*, vol. 19, no. 1, pp. 93–105, 1996.

[5] Z. Huang, D. Dikin, W. Ding, Y. Qiao, X. Chen, Y. Fridman, and R. Ruoff, “Three–dimensional representation of curved nanowires,” *Journal of microscopy*, vol. 216, no. 3, pp. 206–214, 2004.

[6] B. Kratochvil, L. Dong, L. Zhang, and B. Nelson, “Image–based 3D reconstruction using helical nanobelts for localized rotations,” *Journal of microscopy*, vol. 237, no. 2, pp. 122–135, 2010.

[7] R. Danzl, H. Schroettner, F. Helmli, and S. Scherer, “Coordinate measurement with nano–metric resolution from multiple SEM images,” in *European microscopy congress*, 2012.

[8] A. P. Tafti, A. B. Kirkpatrick, Z. Alavi, H. A. Owen, and Z. Yu, “Recent advances in 3D SEM surface reconstruction,” *Micron*, vol. 78, pp. 54–66, 2015.

[9] M.-F. Yu, M. J. Dyer, G. D. Skidmore, H. W. Rohrs, X. Lu, K. D. Ausman, J. R. Von Ehr, and R. S. Ruoff, “Three-dimensional manipulation of carbon nanotubes under a scanning electron microscope,” *Nanotechnology*, vol. 10, p. 244–252, 1999.

[10] M.-F. Yu, O. Lourie, M. J. Dyer, K. Moloni, F. T. Kelly, and R. S. Ruoff, “Trength and breaking mechanism of multiwalled carbon nanotubes under tensile load,” *Science*, vol. 287, pp. 637–640, 2000.

[11] T. Fukuda, F. Arai, and L. X. Dong, “Assembly of nanodevices with carbon nanotubes through nanorobotic manipulations,” in *Proceedings of the IEEE*, vol. 91(11), 2003, p. 1803–1818.

[12] B. J. Bell, L. Dong, B. J. Nelson, M. Golling, L. Zhang, and D. Grützmacher, “Fabrication and characterization of three-dimensional ingaas/gaas nanosprings,” *Nanoletters*, vol. 6, no. 4, pp. 725–729, 2006.

[13] C. Ru and S. To, “Contact detection for nanomanipulation in a scanning electron microscope,” *Ultramicroscopy*, vol. 118, pp. 61–66, 2012.

[14] S. Zimmermann, T. Tiemerding, S. Fatikow, T. Wang, T. Li, and Y. Wang, “Automated mechanical characterization of 2d materials using sem based visual servoing,” in *2013 International Conference on Manipulation, Manufacturing and Measurement on the Nanoscale (3M-NANO), 26-30 August 2013, Suzhou, China*, 2013.

[15] M. R. Mikczinski, G. Josefsson, G. Chinga-Carrasco, E. K. Gamstedt, and S. Fatikow, “Fabrication and characterization of three-dimensional ingaas/gaas nanosprings,” *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 115–119, 2014.

[16] J.-O. Abrahamians, B. Sauvet, J. Polesel-Maris, R. Braive, and S. Régnier, “A nanorobotic system for in situ stiffness measurements on membranes,” *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 119–124, 2014.

[17] C. Shi, D. K. Luu, Q. Yang, J. Liu, J. Chen, C. Ru, S. Xie, J. Luo, J. Ge, and Y. Sun, “Recent advances in nanorobotic manipulation inside scanning electron microscopes,” *IEEE Transactions on Robotics*, vol. 2, no. 16024, 2016.

[18] V. P. Namboodiri and S. Chaudhuri, “On defocus, diffusion and depth estimation,” *Pattern Recognition Letters*, vol. 28, no. 3, pp. 311–319, 2007.

[19] S. Pertuz, D. Puig, and M. A. Garcia, “Analysis of focus measure operators for shape-from-focus,” *Pattern Recognition*, vol. 46, no. 5, pp. 1415–1432, 2013.

[20] F. Nicolls, G. de Jager, and B. Sewell, “Use of a general imaging model to achieve predictive autofocus in the scanning electron microscope,” *Ultramicroscopy*, vol. 69, no. 1, pp. 25–37, 1997.

[21] M. Rudnaya, H. ter Morsche, J. Maubach, and R. Mattheij, “A derivative-based fast autofocus method in electron microscopy,” *Journal of Mathematical Imaging and Vision*, vol. 44, pp. 38–51, 2012.

[22] S. Dembélé, O. Lehmann, K. Medjaher, N. Marturi, and N. Piat, “Combining gradient ascent search and support vector machines for effective autofocus of a field emission–scanning electron microscope,” *Journal of microscopy*, 2016.

[23] K. Ong, J. Phang, and J. Thong, “A robust focusing and astigmatism correction method for the scanning electron microscope. part ii: Autocorrelation-based coarse focusing method,” *Scanning*, vol. 20, pp. 324–334, 1998.

[24] J. He, R. Zhou, and Z. Hong, “Modified fast climbing search autofocus algorithm with adaptive step size searching technique for digital camera,” *IEEE Transactions on Consumer Electronics*, vol. 12, pp. 244–252, 2003.

[25] N. Marturi, B. Tamadazte, S. Dembélé, and N. Piat, “Visual servoing-based approach for efficient autofocusing in scanning electron microscope,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo*, 2013, pp. 2677–2682.

[26] C. F. Batten, “Autofocusing and astigmatism correction in the scanning electron microscope,” Ph.D. dissertation, Citeseer, 2000.

[27] N. Marturi, B. Tamadazte, S. Dembélé, and N. Piat, “Visual servoing-based depth estimation technique for manipulation inside SEM.” *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 8, pp. 1847–1855, 2016.

[28] L. Cui, E. Marchand, S. Haliyo, and S. Régnier, “Three-dimensional visual tracking and pose estimation in scanning electron microscopes,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS’16*, Daejeon, Korea, October 2016, pp. 5210–5215.

[29] J. C. Spall, “Stochastic optimization,” in *Handbook of computational statistics*. Springer, 2012, pp. 173–201.

[30] J. Kiefer, “Sequential minimax search for a maximum,” *Proceedings of the American mathematical society*, vol. 4, no. 3, pp. 502–506, 1953.

[31] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural networks*, vol. 12, no. 1, pp. 145–151, 1999.

[32] J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization,” *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[33] T. Tieleman and G. Hinton, “RMSProp, Coursera: Neural networks for machine learning,” *Technical report*, 2012.

[34] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[35] J. C. Spall, “A stochastic approximation technique for generating maximum likelihood parameter estimates,” *IEEE American Control Conference*, pp. 1161–1167, 1987.