The impact of metrics in mechanical imaging

Ahmed Chabib, Jean-Francois Witz, Pierre Gosselet, Vincent Magnier

▶ To cite this version:

Ahmed Chabib, Jean-Francois Witz, Pierre Gosselet, Vincent Magnier. The impact of metrics in mechanical imaging. Strain, 2025, 61, pp.e1249. 10.1111/str.12494 . hal-04251608

HAL Id: hal-04251608

https://hal.science/hal-04251608v1

Submitted on 20 Oct 2023

The impact of metrics in mechanical imaging

Ahmed Chabib, Jean-François Witz, Pierre Gosselet, Vincent Magnier

October 20, 2023

Abstract

Optical Flow (OF) is an alternative technique to the more classical digital image correlation methods (DIC), able to measure the motion between two images. It is a pixel-wise method, in the sense that the displacement is defined at each pixel and its computation does not require overpixel grids. This paper examines the impact on the identified strain field, the main quantity of interest for solid mechanics, of the metrics used to quantify the conservation of the optical flow and to impose the regularity of the displacement. The Charbonnier and Lorentzian loss functions are inspected in this work. A new regularization approach is used to locally vary the Tikhonov parameter using a mask in order to preserve the discontinuities and encourage the appearance of local phenomena while smoothing the computed fields elsewhere. Finally, an open-source GPU-accelerated python code has been implemented.

1 Introduction

Digital Image Correlation (DIC) and Optical Flow (OF) refer to techniques that analyze a sequence of images in order to estimate the displacement. In recent years, DIC has gained popularity as an efficient technique to analyze digital images of materials under testing, allowing for the high-resolution full-field measurement of strain. The basic principle of DIC is to track the displacement of a surface or a volume by comparing digital images taken before and after deformation. DIC methods involve partitioning images into smaller regions and then analyzing their intensity patterns in order to identify corresponding subsets between two images. The intensity patterns are encoded using a speckle or random dot pattern applied to the surface (or by the variation of the texture in the volume), and the resulting data can be presented as displacement or strain maps. These maps provide useful information about the mechanical behavior of the material or the structure being analyzed. Besides its full-field advantage compared to point sensors, DIC is a non-destructive technique that does not require direct contact with the surface, making it a suitable option to analyze fragile or sensitive materials.

Optical Flow is a popular method used in the field of computer vision and image processing to measure displacement and motion fields. This technique also provides full-field measurements and does not require a physical contact with the object being analyzed as DIC. After more than 40 years of research, optical flow models have steadily improved in accuracy over time. Unlike DIC, these models are capable of describing the movement at every pixel of the image.

Both Optical flow and DIC techniques are based on gray value constancy assumption (1) introduced by Horn and Schunk [12], assuming that the gray values of the pixels do not change during the motion:

$$I_1(x) = I_2(x + u(x)), \forall x \in \Omega, \tag{1}$$

where I_1 is the reference image, I_2 is the deformed image, x are the coordinates of a given pixel, Ω represents the domain of the image and u is the displacement field.

Moreover, most of DIC software [15, 8] uses least square as a metric to quantify the difference between the sequence of the images and they subsequently try to minimize this gap. This optimization problem is ill-posed due to the number of degrees of freedom exceeding the number of equations. To overcome the issue, DIC approaches may only focus on subsets (Ω_e) of the image and use a kinematic vector space V_h based on shape functions associated with a grid spaced by several pixels [3] (like in finite element methods):

$$\min_{u_h \in V_h} \sum_{\Omega_e \subset \Omega} \int_{\Omega_e} \Big(I_1(x) - I_2\big(x + u_h(x)\big)\Big)^2 \mathrm{dx}. \tag{2}$$

It is very common when decreasing the spacing of the grid to add a Tikhonov regularization [13] to the previous functional:

$$E_{DIC}(u_h) = \sum_{\Omega_e} \int_{\Omega_e} \left\{ \Big(I_1(x) - I_2\big(x + u_h(x)\big) \Big)^2 + \lambda (\nabla u_h)^2 \right\} \mathrm{dx}, \qquad \text{with } \lambda > 0. \tag{3}$$

Most of the current optical flow methods closely resemble the first Horn and Schunck formulation (4). As done with DIC, they typically combine a data term that assumes gray value constancy, along with a weighted smoothness term that models the variation of the flow across the image. The only difference is that the problem here is formulated at each pixel of the whole image with a kinematic defined at the pixel scale (without shape functions over grids):

$$E(u) = \int_{\Omega} \left\{ \Big(I_1(x) - I_2\big(x + u(x)\big) \Big)^2 + \lambda \big((\nabla u)^2 \big) \right\} \mathrm{dx}, \qquad \text{with } \lambda > 0. \tag{4}$$

It has been noticed that this classical formulation utilizing the quadratic function is not robust to outliers. Black and Anandan [4] solved this issue by introducing a new formulation (5) where the quadratic error function is substituted by an adequate error function robust to outliers, hence the term "robust functions". The paper of Sun [9] drew inspiration from these robust models to define new classical methods capable of enhancing the quality of the computed fields. It analyzed the impact of the penalty functions and the effect of other parameters on the displacement field from Middlebury optical flow benchmark [16] which contains real-life images. For those datasets, the texture that refers to the visual features that allow different regions of an image to be distinguished is not imposed as is the case of mechanical sequences where the speckle pattern is imposed by the user. It has been chosen in different works [18, 19] to utilize an anisotropic diffusion approach as a replacement for the homogeneous regularization in the Horn-Schunck model. This decision stems from the frequent occurrence of discontinuities in optical flow, which often coincide with high image gradients. Introducing a metric ρ, the functional to be minimized can be written as:

$$E(u) = \int_{\Omega} \left\{ \rho \left((I_1(x) - I_2(x + u(x)))^2 \right) + \lambda \left(\rho \left((\nabla_x u)^2 \right) + \rho \left((\nabla_y u)^2 \right) \right) \right\} \mathrm{dx}. \tag{5}$$

Our research contributes to a better understanding of the methods proposed in [9] and their implementation, in the context of the analysis of mechanical strain and the detection of mechanical cracks. In particular, we compare the effect of replacing the quadratic norm by the Charbonnier's or the Lorentzian for the image and smoothness terms. Note that we provide a simple matrix-free python code accelerated by GPU so that researchers from all the scientific disciplines who are not familiar with GPU programming can test their images with theses techniques easily in a reasonable time.

The rest of the paper is organized as follows. In Section 2, we provide the details for the robust and efficient minimization of the functional (5). In Section 3, we assess the method and compare it different settings on a numerically generated cracked samples before presenting a local regularization approach and moving on to Section 3.3 to present our conclusions and perspectives.

2 Methodology

The methods used in this work are similar to the classical brightness methods [9] introduced by Sun with some differences that will be mentioned later. The penalty functions used in [9] are taken into account: first, the classical quadratic norm $\rho(x^2) = x^2$ utilized by Horn and Schunk(HS) and widely used in DIC, then the Charbonnier [5] loss function defined as $\rho(x^2) = \sqrt{x^2 + \varepsilon^2}, \varepsilon = 0.001$, which is a differentiable and convex robust function and a variant of the L^1 norm, and finally the non-convex Lorentzian [4] function defined as $\rho(x^2) = \log(1 + \frac{x^2}{2\sigma^2}), \sigma > 0$. These functions are illustrated by their curves on Figure 1.

We denote by u and v the horizontal and the vertical displacement fields respectively. The equation (5) can be discretized in the form of:

$$E(u, v) = \sum_{i,j} \rho\big((I_2(i + u_{i,j}, j + v_{i,j}) - I_1(i, j))^2\big) + \lambda \Big(\rho\big((\nabla_x u)^2_{i,j}\big) + \rho\big((\nabla_y u)^2_{i,j}\big) + \rho\big((\nabla_x v)^2_{i,j}\big) + \rho\big((\nabla_y v)^2_{i,j}\big) \Big) \tag{6}$$

where (i, j) are the coordinates of pixels, ∇_x and ∇_y are respectively the horizontal and vertical derivation operators.

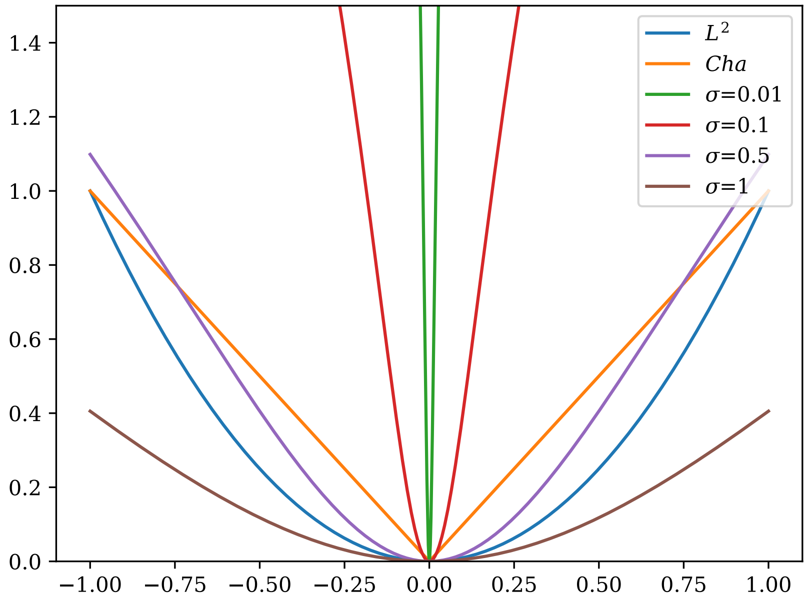


Figure 1: The curve of the tested penalty functions: the quadratic (L^2), Charbonnier (Cha.) using $\varepsilon = 0.001$ and the Lorentzian with different values of σ

2.1 Graduated Non-Convexity (GNC) and pyramidal approach

The minimization of a non-convex function is not always straightforward due to the existence of local optima that may appear during the minimization process. The Graduated Non-Convexity (GNC) scheme is used to tackle the complex optimization problems. GNC algorithm works by minimizing a sequence of energy functions $(E_\omega)_\omega$ (7) given as a linear combination of a fully convex quadratic energy function E_Q and the fully robust energy function E (5). E_Q is obtained by changing the robust penalties in (5) by a quadratic loss function. The parameter ω decreases gradually from 1 to 0 and varies the convexity of the compound objective, allowing the transition from the quadratic form to the proposed robust form.

$$E_\omega(u, v) = \omega E_Q(u, v) + (1 - \omega)E(u, v) \tag{7}$$

Throughout the process, the solution obtained from the previous stage is utilized as an initial guess for the current stage. It has been demonstrated [17] that using three stages of *convexification* typically yields satisfactory results. During each step, a straightforward local minimization of the energy function is executed.

For the purpose of predicting large motion fields, a coarse-to-fine [6] approach is often adopted as in DIC. The coarse-to-fine approach consists in creating a pyramid of images by repeatedly reducing the size of the original image. The pyramidal structure is created by stacking the smaller resized images on top of each other, ending with the highest resolution image at the bottom. Afterwards, at every level of the pyramid we perform a processing step, the output is then resized and used as a starting point for the next level. The process is repeated until reaching the original image. An illustration of this approach is shown in Fig. 2. We set the standard deviation of the Gaussian filter to be $\sqrt{\frac{1}{2d}}$, where d represents the downsampling factor. During the minimization of the quadratic formulation which appears during the first iteration of the GNC process, we dynamically set the number of pyramid levels to ensure that the highest level has a width or height of approximately 20 to 30 pixels with a downsamling factor of 0.5 since it has been explored [9] that using a convex penalty, a factor of 0.5 is sufficient. For the remaining stages of the GNC, we will utilize two levels with a factor of 0.8 to benefit from the solution obtained in the prior *convexification* stage. The GNC is also used for the Charbonnier function despite its convexity, since it leads to better minimization [9] due to the nonlinearity of its derivative. In this method ten flow increments are computed per direction at each pyramid level and a 5×5 median filter is applied in the end of each warping step to the current fields to remove the outliers as described in [9].

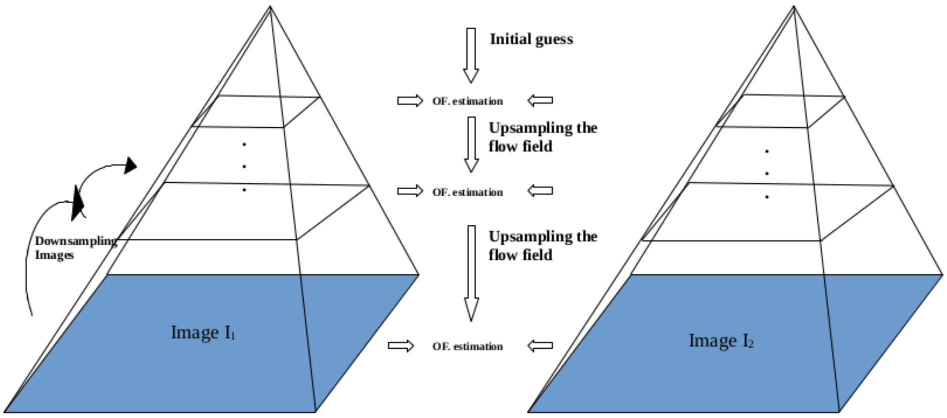


Figure 2: Representation of the pyramidal approach, OF=Optical Flow

2.2 Methodological details

The estimation of the optical flow fields requires computing a finite difference approximation of the spatial derivatives of images as well as the temporal derivative defined as the difference between the reference image and the corrected image $I_2(i + u^k_{i,j}, j + v^k_{i,j})$ where u^k and v^k represent the displacement fields computed at the iteration k obtained by a bi-cubic interpolation. Instead of the classical finite difference kernels, we approximate the spatial derivatives using a five-point kernel [18] $h = \frac{1}{12}\left[-1, 8, 0, -8, 1\right]$. We neglect pixels whose movement exceeds the border by setting their corresponding derivatives to zero. We interpolate both the second image and its derivatives using the current flow field. In agreement with [11], we use the average of the gradient of the reference image and the interpolated derivatives of the second image to define the tangent system to be solved.

Unlike the methods used in Sun's paper [9], where the regularization parameter λ and the parameter σ of the Lorentzian function are fixed to provide a displacement close to the real fields obtained from the image sequences in the Middlebury dataset, our approach consists of varying the value of these parameters to see their impact on the strain field and to study the effect of this change on discontinuities.

The choice of the parameter in the Lorentzian function is far from evident, and there is no universal parameter value that suits all situations. The selection of this parameter requires a thorough understanding of the data and objectives. For instance, if the expected motion fields are smooth and continuous, a higher value of this parameter can be a good choice to encourage regularization and reduce discontinuities. Conversely, if the field contains discontinuities, a lower value of σ allows increasing their effect. It may be necessary to conduct empirical tests by adjusting the parameter of the Lorentzian function and evaluating the results on real optical flow data.

It should be noted that the Rudin-Osher-Fatemi [18](ROF) texture-structure decomposition frequently used to eliminate changes in brightness is not applied here. Indeed, this decomposition consists in filtering input images by breaking each of them down into two parts, a texture part which preserves image details, and a part containing shadows called structure. In mechanical engineering, however, experimental setups are designed to ensure that there is no change in brightness during the test. All the more so as, in the example presented in this paper, the images are synthetic and do not pose any brightness-related problems.

2.3 Implementation

We provide a python source code accelerated by GPU using *CuPy* [1] which proposes a NumPy-compatible API for GPU computing with NVIDIA CUDA. As the matrix to be solved at each iteration is a positive definite symmetric matrix, the conjugate gradient method [10] is used to solve the problem. The choice of this type of solvers is not only motivated by its ability to solve large-scale systems, but also by its capacity to be implemented in a matrix-free version, which allow fewer resources consumption and consequently enables the processing of much larger datasets. The code [7] as well as the examples are uploaded to GitHub.

3 Results

We tested our algorithm on real images and demonstrated that it yields satisfactory results. The choice to use synthetic images is also influenced by the absence of a reference solution in the context of physical experiments. The use of synthetic images is of great importance in computer vision, as they allow for the generation of precise data for assessing the effectiveness of image processing algorithms. By using these images, it is possible not only to avoid the difficulties associated with collecting real experimental data, which can be costly in terms of time and resources, but also they allow for the creation of various test scenarios.

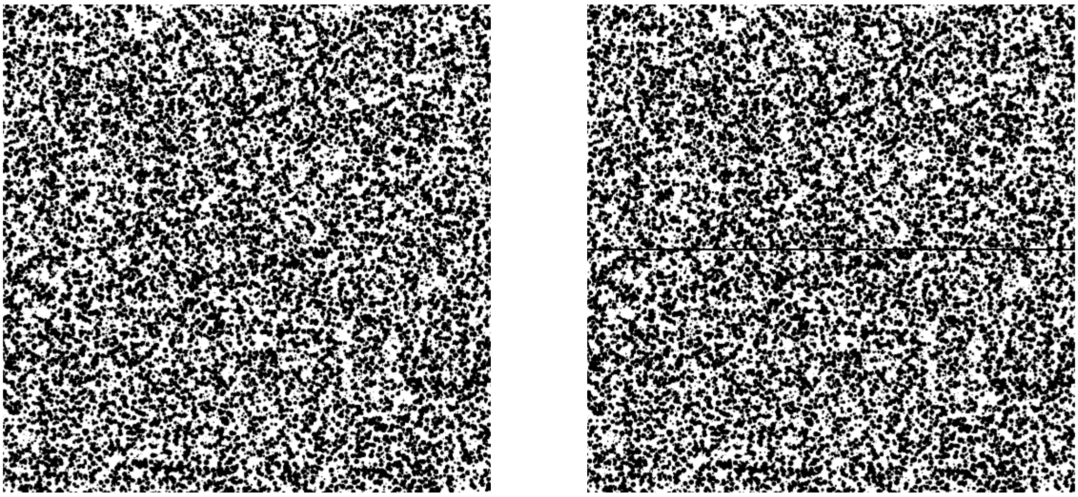
A natural measure of quality in this context is the Average Endpoint Error (AEE) which corresponds to the average (over N pixels) of the Euclidean distance between the calculated (u, v) and the ground-truth $(\hat{u}, \hat{v})$ motion fields:

$$AEE^2 = \frac{\|u - \hat{u}\|^2 + \|v - \hat{v}\|^2}{N^2}$$

In order to study the influence of the previous cited functions over the discontinuities, a synthetic 512×512 speckle patterns image is generated with grain sizes averaging between three and five pixels using the algorithm developed in [2].

3.1 Crossing crack

We create an artificial transverse horizontal crack of one pixel by imposing a one-pixel downward displacement on the lower part of the created discontinuity. The images are presented in the Fig. 3.

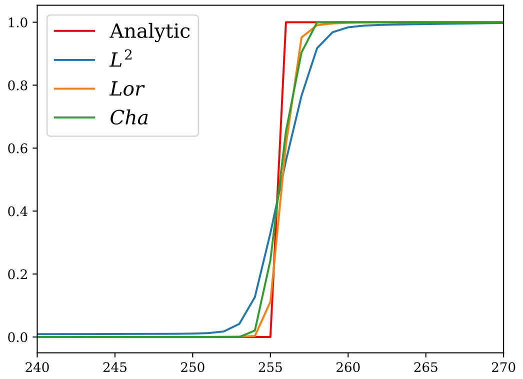


(a) Generated speckle pattern

(b) Transverse crack image

Figure 3: First synthetic test case

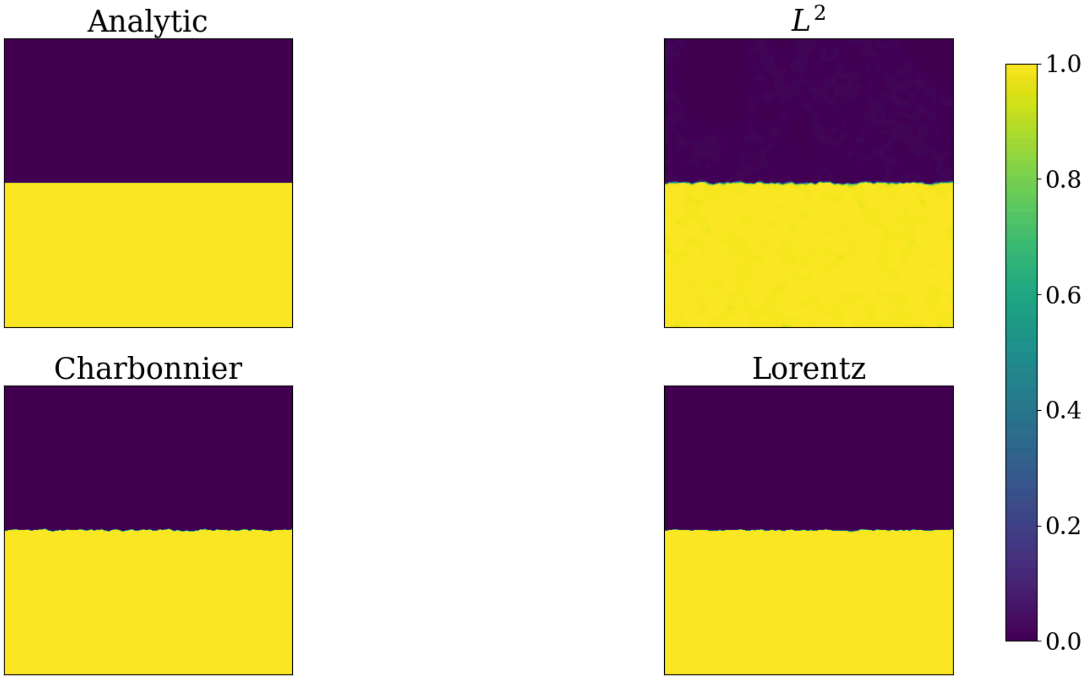
The horizontal displacement in this test case is zero across the entire image whereas the vertical displacement field is invariant for each part of the image, since it is zero for the entire upper part of the crack and imposed to be one pixel for the lower part as shown in the analytic motion field of Fig. 4c. The average of vertical displacements along the horizontal direction of the image must then be constant and the curve of these averages over the entire image should be represented by a Heaviside step function, discontinuous at the crack row. In this case, the maximum standard deviation reached on each section is an indicator of the noise of the calculated solution. The results of Fig. 4 demonstrate that the quadratic norm behaves as predicted by diffusing discontinuities and by requiring more pixels to capture the crack than the robust functions (Charbonnier and Lorentzian) that are able to converge towards a better solution in terms of AEE and standard deviation. Indeed, the L^2 norm captures the discontinuities with seven pixels, while other metrics that only require three pixels. This highlights the importance of studying the impact of these metrics over the estimation of strain on mechanical images.



(a) Average of vertical displacements along the horizontal direction in the vicinity of the crack.

Metric	AEE $\times 10^{-5}$	Std. dev. max	Crack width (pixels)
L^2	7.86	1.64×10^{-2}	7
Lorentz $\sigma = 0.05$	6.05	2.97×10^{-3}	3
Charbonnier	7.04	3.20×10^{-5}	3

(b) AEE, maximum standard deviation reached outside the crack, and number of pixels required to capture the discontinuity for each penalty function.



(c) Calculated and analytic motion fields.

Figure 4: Identification of the crossing crack for different metrics.

3.2 Crack with tip

The William's series [14] allows for the description of displacement and strain fields near a crack or notch in a solid material. This method assumes that the motion near a crack can be expressed as a series of functions dependent on the distance from the crack tip. From the pre-defined speckle pattern Fig. 3a, a deformed image is generated using Williams' fields resulting from a planar loading mode presented in Fig. 6j. The deformed image Fig. 5 of the generated sequence contains a 1 × 250 pixels discontinuity.

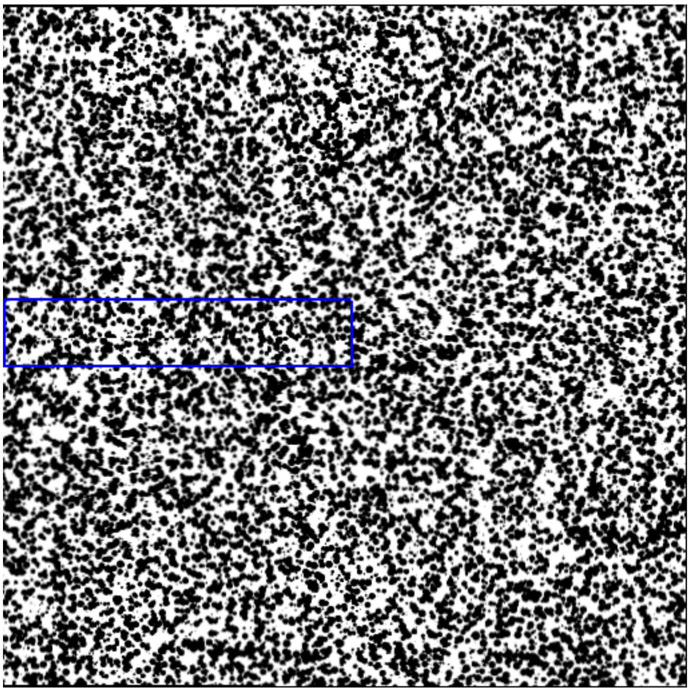
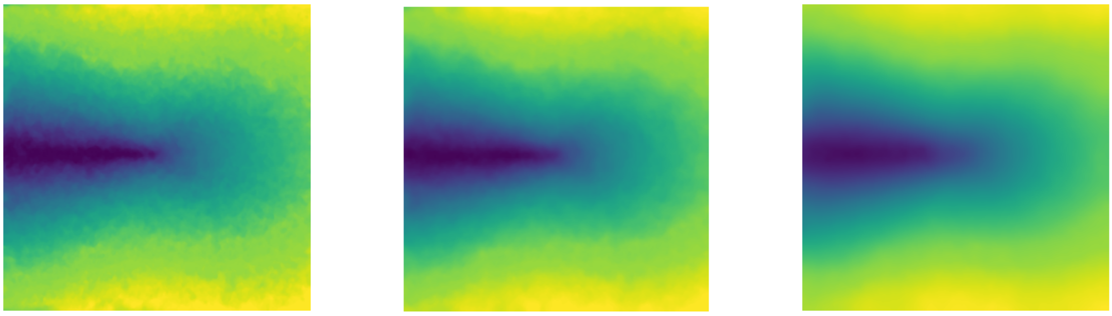


Figure 5: The deformed image for the second test case, the part framed in blue contains the generated crack

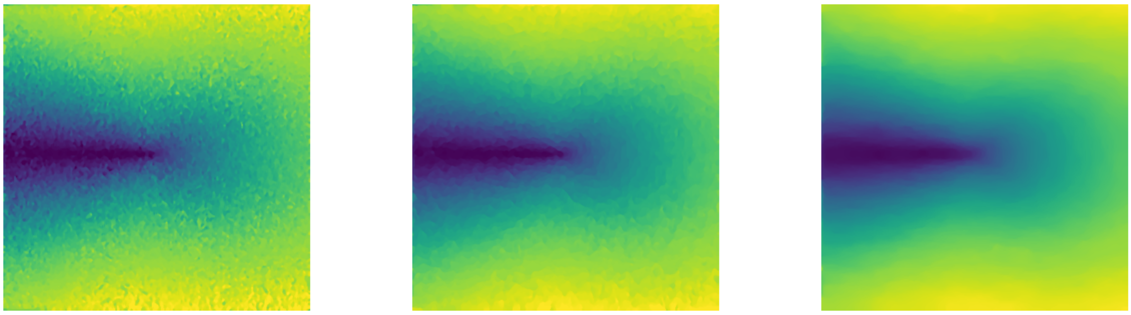
The regularization parameter in optical flow methods is used to balance the smoothness of the estimated motion field and the accuracy of the motion vectors. This parameter allows controlling the trade-off between fitting the data and maintaining the smoothness of the estimated flow field. The behavior of this parameter is similar to that of the grid size in DIC techniques [14]. A higher value of the regularization parameter leads to a smoother but less accurate motion estimation, while a lower value results in a more detailed but potentially noisy flow field, see Fig. 6.

The initial results show that the Charbonnier and the Lorentzian functions give more promising results than the quadratic norm by preserving discontinuities and providing better quality of motion fields. To highlight the difference between these two metrics, we present the horizontal and vertical strain fields obtained with different regularization in Fig 7 and Fig 8. Charbonnier produces spurious discontinuities (see Fig. 7c Fig 8c) that are not observed either in the analytical fields Fig. 7e and Fig. 8e or in the fields generated by the Lorentzian penalty function presented in the first rows of Fig 7 and Fig 8. Moreover, these unreal discontinuities become structured and difficult to distinguish from real cracks. This outcome can be explained by the fact that Charbonnier is an approximation of the non-smooth L^1 norm, which results in abrupt gradients in regions where the displacement values are slightly different, rendering it challenging to use on mechanical images.

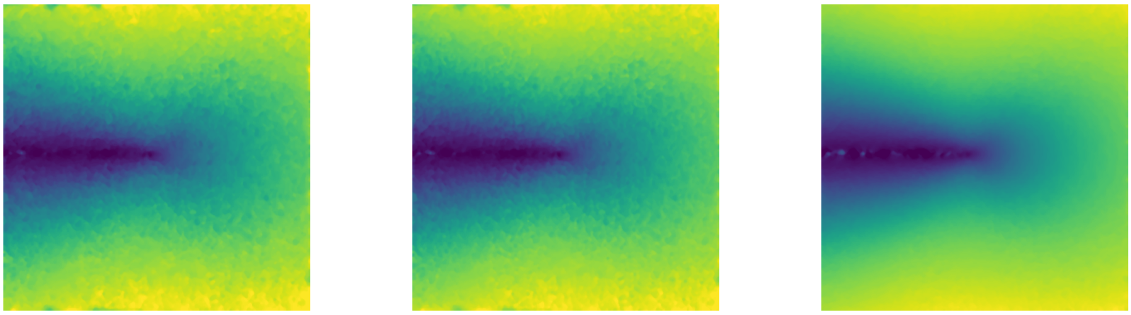
Discontinuities like cracks can be characterized as places where the strain field varies rapidly. This can be illustrated by Fig. 9e which shows the norm of the gradient of the strain field for the ground-truth solution, and where the crack clearly stands out. Thus, the norm of the difference between the value of the analytical second derivatives of the displacement and their calculated counterpart is an indicator that quantifies the ability of the methods to capture cracks. We denote this field by NSDE which stand for Frobenius Norm of the Strain Derivative Error, its relative mean value is denoted by MNSDE, see Eq. (8). We observe from the NSDE of the Fig. 9, that the fields obtained with the Lorentzian function shown in Fig. 9a and Fig. 9b are more homogeneous outside the crack, closer to reality, and contain less discontinuities than those estimated by the Charbonnier function presented in Fig. 9c and Fig. 9d. Furthermore, Fig. 9f shows that the lowest values of MNSDE are provided by the Lorentzian compared



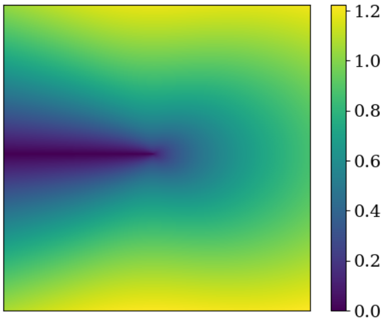
(a) Lorentzian, $\lambda = 1$ (b) Lorentzian, $\lambda = 10^1$ (c) Lorentzian, $\lambda = 10^2$



(d) Charbonnier, $\lambda = 1$ (e) Charbonnier, $\lambda = 10^1$ (f) Charbonnier, $\lambda = 10^2$



(g) L^2,$\lambda = 5 \times 10^2$ (h) L^2,$\lambda = 10^3$ (i) L^2,$\lambda = 3 \times 10^4$

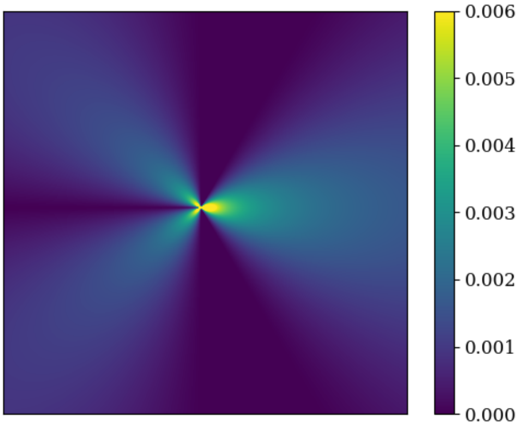


(j) Analytic horizontal displacement

Figure 6: Horizontal displacement field estimated with different values of λ, and different loss functions for the crack with tip experiment.

(a) Lorentzian, $\lambda = 10^1$ (b) Lorentzian, $\lambda = 10^2$

(c) Charbonnier, $\lambda = 10^1$ (d) Charbonnier, $\lambda = 10^2$



(e) Analytic

Figure 7: The horizontal strain fields ε_{xx} provided with different regularization amplitude of λ for the Lorentzian and Charbonnier functions

to Charbonnier's. This result makes the Charbonnier's function challenging to use in mechanics.

$$NSDE(\boldsymbol{\varepsilon}) = \sum_{i,j} || \frac{\partial(\varepsilon_{x_i x_j} - \hat{\varepsilon}_{x_i x_j})}{\partial x_j} ||_F^2, \qquad MNSDE(\boldsymbol{\varepsilon}) = \frac{\overline{NSDE(\varepsilon)}}{\overline{\sum_{i,j} || \frac{\partial \hat{\varepsilon}_{x_i x_j}}{\partial x_j} ||_F^2}} \tag{8}$$

3.3 Local regularization

Global regularization of the fields may result in excessive smoothing of the solution, leading to loss of details and structures in the image and decreasing the accuracy of the registration computation. Moreover, this strategy may not be suitable for situations involving regions with discontinuities. Hence, the interest in considering local regularization weights is to preserve the discontinuities as much as possible on the calculated fields. We have adopted the strategy to adjust the amplitude of the regularization in such a way that it decreases around the crack to ensure that there is no diffusion of the crack whereas the strain fields becomes homogeneous and smooth outside the area containing the discontinuity. This leads to the minimization of a novel energy whose regularization weight ($\lambda_{i,j}$) is defined locally and changes depending on the pixel coordinates:

$$E(u,v) = \sum_{i,j} \rho\big((I_2(i+u_{i,j}, j+v_{i,j}) - I_1(i,j))^2\big) + \lambda_{i,j}\Big(\rho\big((\nabla_x u)^2_{i,j}\big) + \rho\big((\nabla_y u)^2_{i,j}\big) + \rho\big((\nabla_x v)^2_{i,j}\big) + \rho\big((\nabla_y v)^2_{i,j}\big)\Big) \tag{9}$$

To test our approach, we create a local smoothing mask Fig. 10 with values varying between 7 (near the crack) and 600 (away). In order to avoid sudden changes in λ, a Gaussian filter is applied as shown in Fig. 10b. The analytical strain field ε_{yy} shown in Fig. 11k and 11l allows to visualize the whole length of the discontinuity.

The choice of a global large smoothing weight results in excessive smoothing in the area near the crack tip Fig. 11f which is not observed for the small values of λ Fig. 11c. This can be explained by the fact that when the regularization amplitude is high, there is a strong influence of the neighboring pixels on the smoothing process, which can lead to a loss of important details. But the global small λ results in structured noise away from the crack.

The local change of regularization shown in Fig. 11g, 11h, 11i allows us to take advantage of the high regularization outside the crack by homogenizing the strain fields and to preserve the totality of discontinuity on the ε_{yy} field as in the case of small regularization, as the impact of the neighboring pixels is forced to become weak at the crack's neighborhood.

One can observe in Fig. 12 that the local regularization permits to correctly capture the strain gradients. Also, the MNSDE computed with the local regularization matches the best approaches in their region of validity (near the crack for the low regularization, away from the crack for the large regularization).

4 Conclusion

Along this paper, we have tested the impact of different metrics on the strain fields estimated using optical flow methods. Robust penalty functions like Charbonnier and Lorentzian often give very good results on displacements and preserve cracks unlike the quadratic L^2 norm often used in DIC that tends to diffuse them. Despite its advantage of preserving discontinuities, the Charbonnier function creates some spurious cracks which could be hardly distinguished from the physical ones. The use of the Lorentzian function gives results comparable to those obtained by the Charbonnier regarding the preservation of the actual cracks, without creating artificial cracks, but it is a non-convex function which makes the minimization of the energies more delicate.

The Tikhonov parameter that balances the influence of the fidelity term and the smoothness term plays a major role in these methods. Increasing this term is similar to enlarging the size of the sub-images in DIC algorithms. The increase of this parameter can cause oversmoothing on some regions of the image, which could be unpleasant in the presence of local phenomena, while its decrease can encourage the presence of noise in the estimated strain fields. A strategy is adopted to change the value of this parameter locally using a mask. This approach enabled us to combine the advantage of preserving the discontinuities in the areas where this parameter is low and of smoothing the fields where its weight is greater.

This work has enabled us to test a few examples of robust metrics. In future work, we'd like to explore other metrics and their added value in mechanics. The creation of the regularization mask used in the illustrative example of this article is manual and with advance knowledge of the kinematic field which is not often the case. One of the perspectives is to automate the creation of masks. One initial idea to reach this objective could be to take the advantage of the pyramidal approach and to leverage the GNC structure used during the minimization process. This automation could also be achieved through the introduction of an artificial intelligence model able to distinguish between image areas requiring stronger smoothing and those requiring weaker smoothing.

Finally, in order to use these penalties on other image sequences, we implemented an open source, GPU-accelerated, and matrix-free code [7] written in python that can be easily modified and allows the use of a regularization mask.

References

[1] Akiba, Takuya, Fukuda, and Kota. CuPy: A numpy-compatible library for nvidia CUDA. *IPSJ Transactions on Programming*, 9(0):4–7, 2016.

[2] M. Baconnais. *Méthode intégrée de corrélation d'images et de corrélation d'images virtuelles.* PhD dissertation, Centrale Nantes, 2019.

[3] G. Besnard, F. Hild, and S. Roux. "finite-element" displacement fields analysis from digital images : application to portevin-le châtelier bands. *Experimental Mechanics*, 46:789–804, 2006.

[4] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *CVIU*, 104:63–75, 1996.

[5] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *ECCV*, page 25–36, 2004.

[6] A. Bruhn, J. Weickert, and C. Schnörr. combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005.

[7] A. Chabib, JF. Witz, P. Gosselet, and V. Magnier. Link to the code repository, 2023.

[8] V. Couty, J-F. Witz, P. Lecomte-Grosbras, J. Berthe, E. Deletombe, and M. Brieu. Gpucorrel: A gpu accelerated digital image correlation software written in python. *SoftwareX*, 16, 2021.

[9] D. Sun and S. Roth and M. J. Black. Secrets of optical flow estimation and their principles. In *Computer society conference on computer vision and pattern recognition.*, pages 2432–2439. IEEE, 2010.

[10] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49, 1952.

[11] B. Horn. *Robot Vision.* MIT Press, 1986.

[12] B.K.P. Horn and G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[13] JC. Passieux and JN. Périé. High resolution digital image correlation using proper generalized decomposition: Pgd-dic. *International Journal for Numerical Methods in Engineering*, 92:531–550, 2012.

[14] C. Roux-Langlois, A. Gravouil, M.-C. Baietto, J. Réthoré, F. Mathieu, F. Hild, and S. Roux. Dic identification and x-fem simulation of fatigue crack growth based on the williams' series. *International Journal of Solids and Structures*, 53:38–47, 2015.

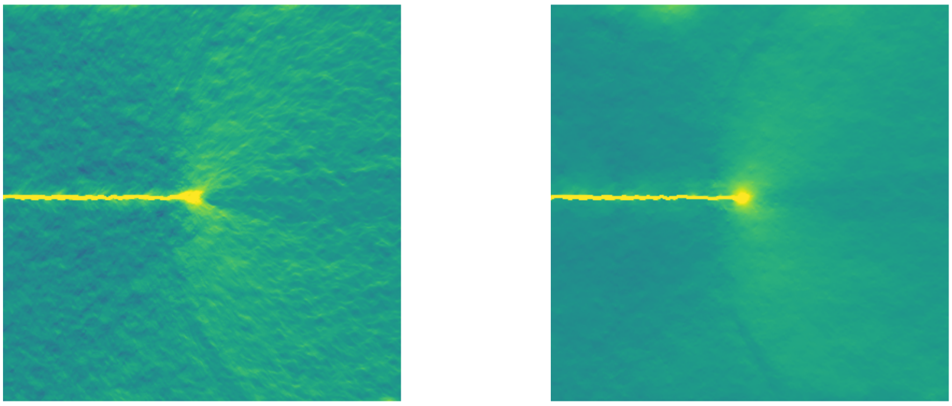
[15] J. Réthoré. UFreckles, 2018.

[16] S. Baker and D. Scharstein and J. Lewis and S. Roth and MJ. Black and R.Szeliski. A database and evaluation methodology for optical flow. *ICCV*, page , 2007.

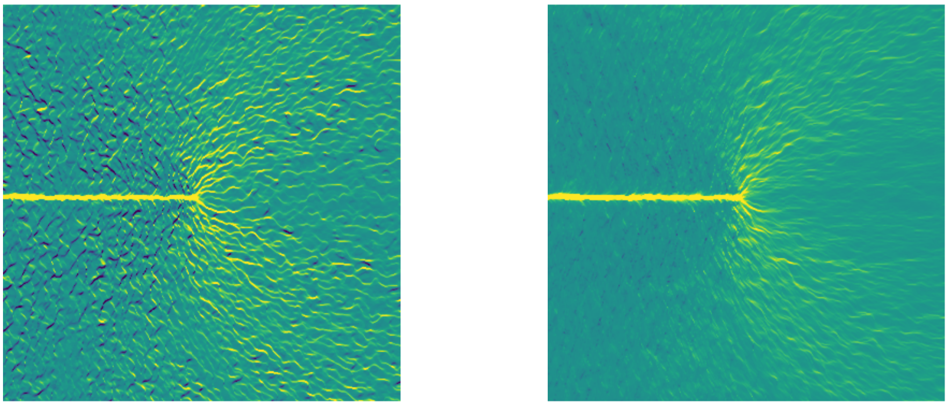
[17] D. Sun, S. Roth, J.P. Lewis, and M. J. Black. Learning optical flow. *ECCV*, 3:83–97, 2008.

[18] A. Wedel, T. Pock, C. Zach, D. Cremers, and H. Bischof. An improved algorithm for tv-l1 optical flow. *Dagstuhl Motion Workshop*, 2008.

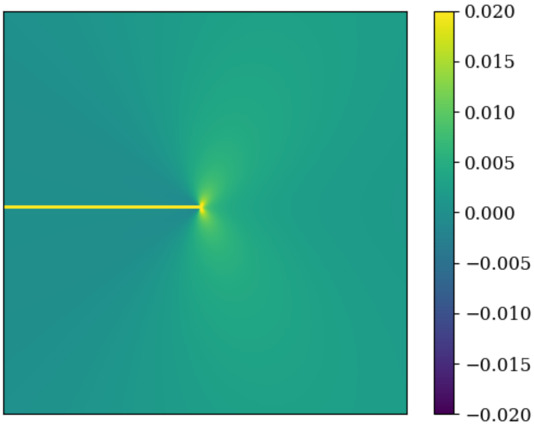
[19] J. Weickert and T. Brox. Diffusion and regularization of vector- and matrix-valued images. *Contemporary Mathematics*, 268:313–251, 2002.



(a) Lorentzian, $\lambda = 10^1$ (b) Lorentzian, $\lambda = 10^2$

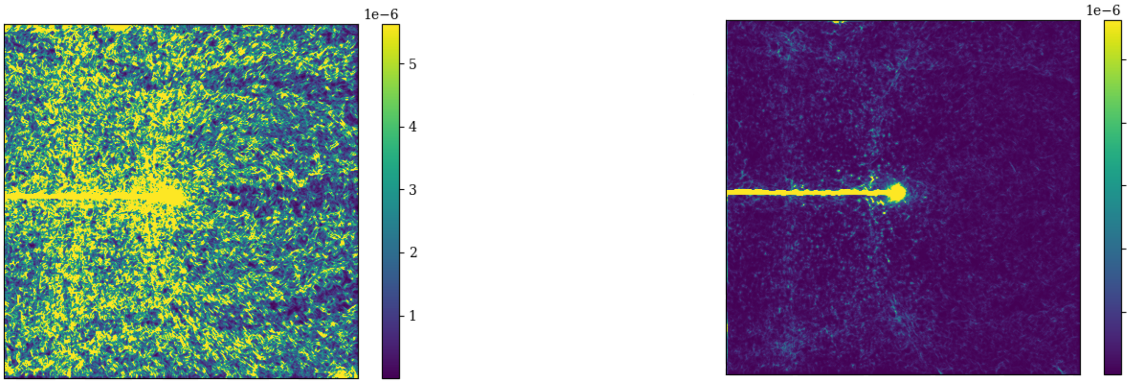


(c) Charbonnier, $\lambda = 10^1$ (d) Charbonnier, $\lambda = 10^2$



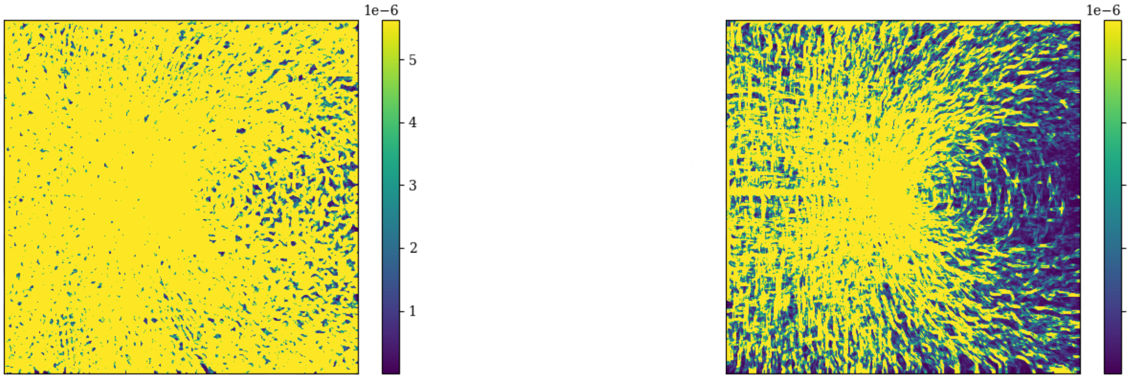
(e) Analytic

Figure 8: The vertical strain fields ε_{yy} provided with different regularization amplitude of λ for the Lorentzian and Charbonnier functions.



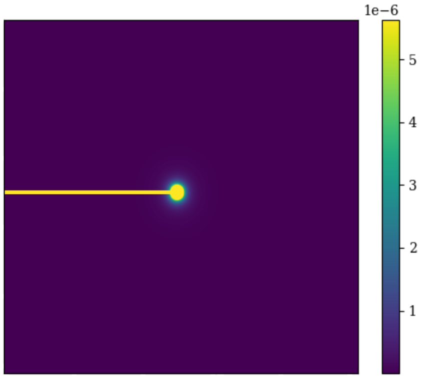
(a) Lorentzian, $\lambda = 10^1$

(b) Lorentzian, $\lambda = 10^2$



(c) Charbonnier, $\lambda = 10^1$

(d) Charbonnier, $\lambda = 10^2$

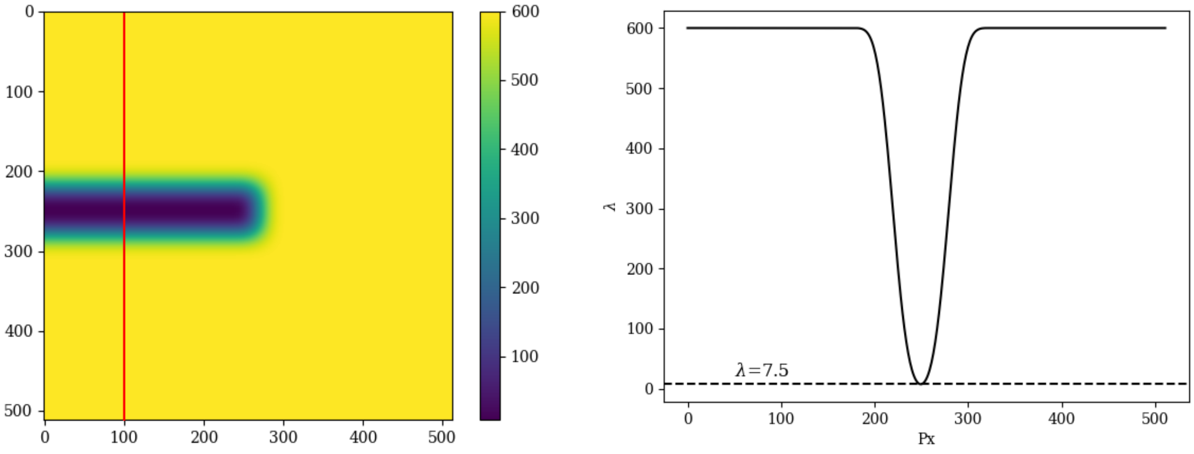


(e) Analytic

Function	λ	MNSDE
Lorentz	10	3.037×10^{-1}
Lorentz	10^2	8.226×10^{-2}
Charbonnier	10	2.029×10^{0}
Charbonnier	10^2	5.681×10^{-1}

(f) The MNSDE in terms of λ

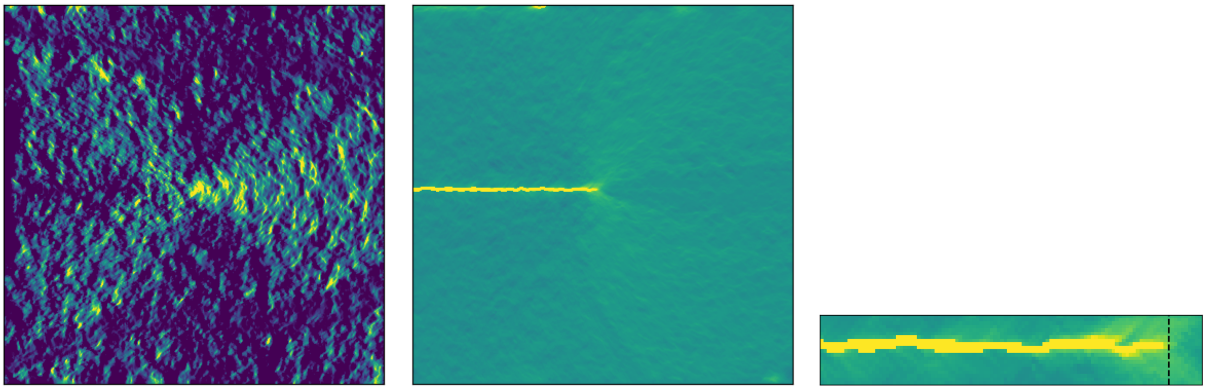
Figure 9: NSDE for both functions Lorentz in the first row and Charbonnier in the second one and with different regularization amplitudes and their corresponding MNSDE. The image (e) represents the analytical norm of the second-order gradient.



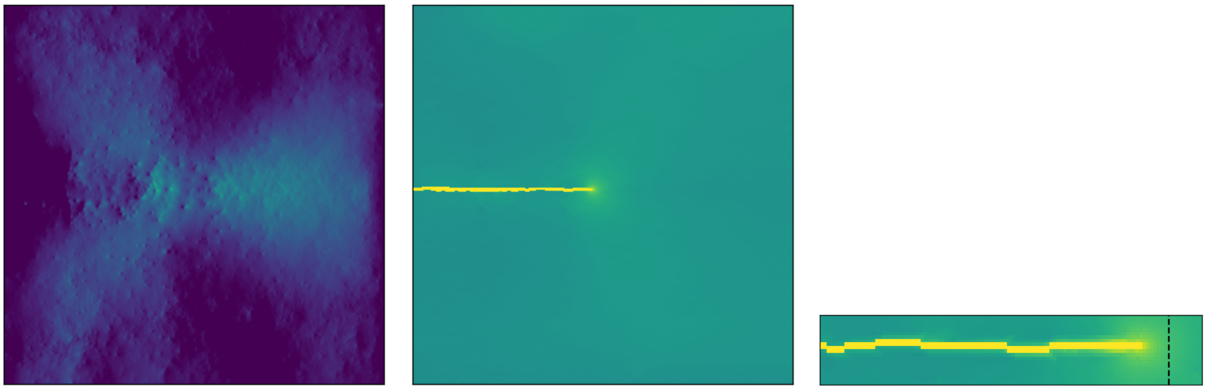
(a)

(b)

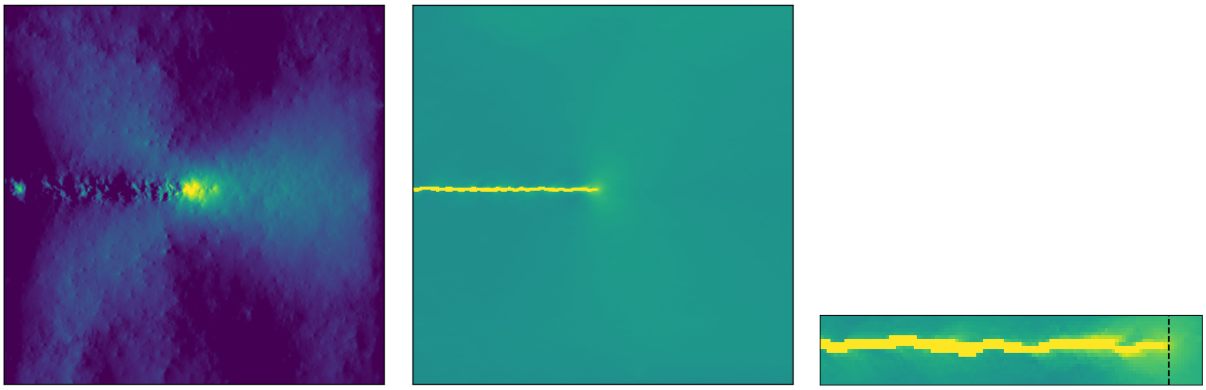
Figure 10: (a) Local regularization mask. The cut along the red line is shown in the figure (b).



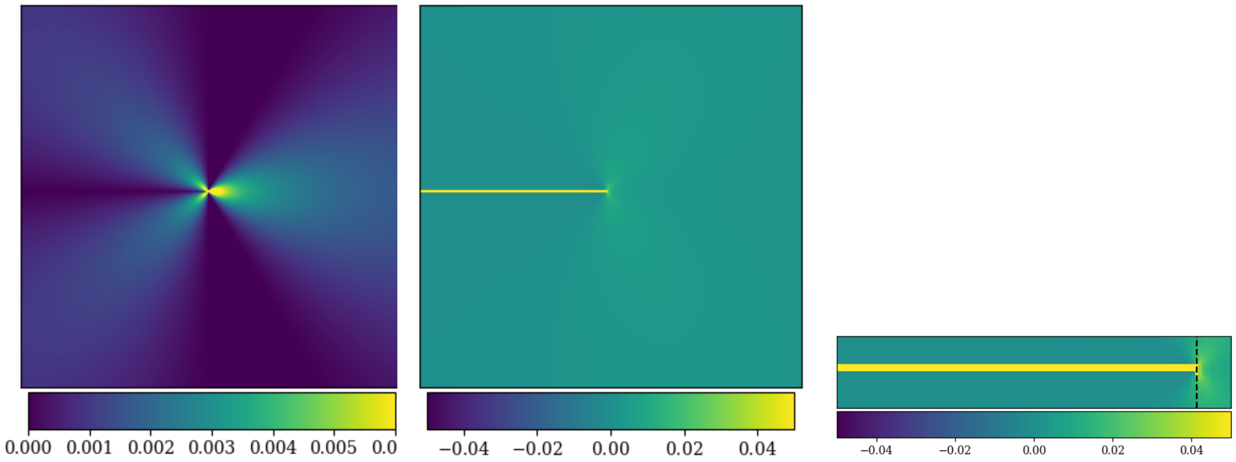
(a) $\varepsilon_{xx}, \lambda = 7$ (b) $\varepsilon_{yy}, \lambda = 7$ (c) $\varepsilon_{yy}, \lambda = 7$



(d) $\varepsilon_{xx}, \lambda = 6 \times 10^2$ (e) $\varepsilon_{yy}, \lambda = 6 \times 10^2$ (f) $\varepsilon_{yy}, \lambda = 6 \times 10^2$



(g) $\varepsilon_{xx}, \lambda$ local (h) $\varepsilon_{yy}, \lambda$ local (i) $\varepsilon_{yy}, \lambda$ local

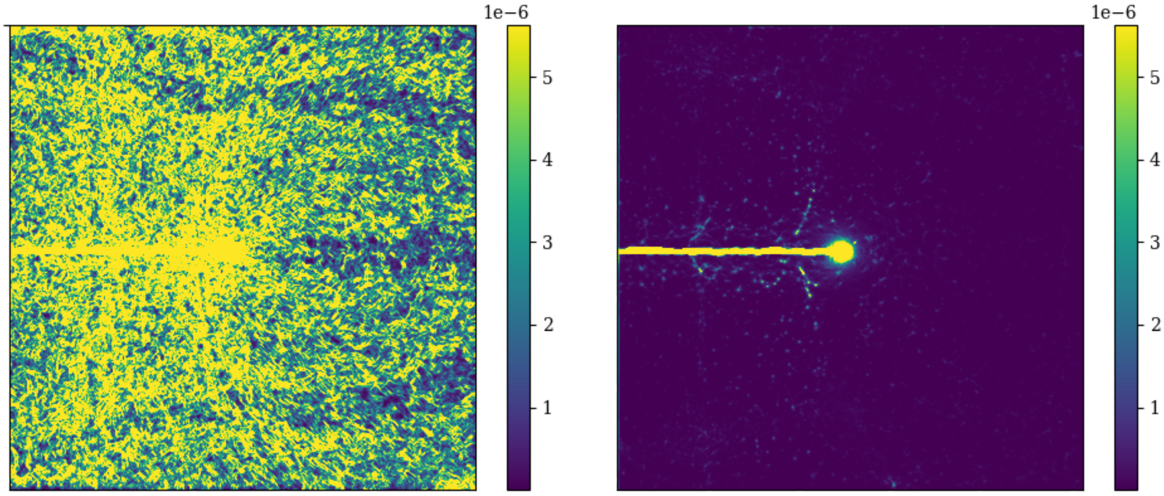


(j) ε_{xx} Analytic

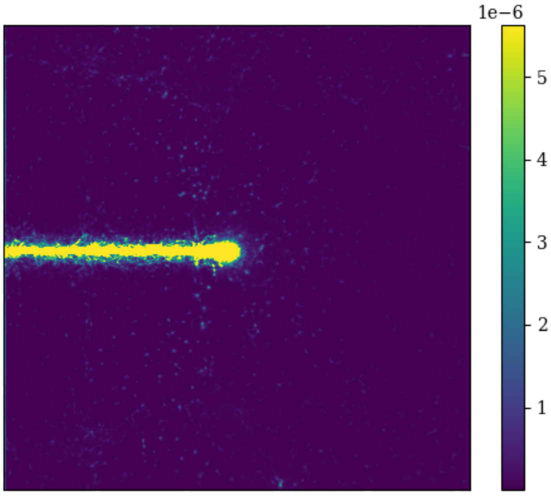
(k) ε_{yy} Analytic

(l) ε_{yy} Analytic

Figure 11: The strain fields in terms of λ. The third column shows a zoom of the second column over the crack. The first and second rows show fields obtained with values close to the minimum and maximum regularization parameter ($\lambda = 7$ and $\lambda = 600$) used in the mask, while the third row shows the results obtained with the local parameter change. Finally, the analytical fields are presented in the last row. The dotted line corresponds to the analytical abscissa of the crack tip.



(a) $\lambda = 7$ (b) $\lambda = 6 \times 10^2$



(c) λ local

λ	Crack Pixels	AEE$\times 10^{-4}$	MNSDE		
			Global	Near Crack	Out Crack
7	3	1.46	39.32×10^{-1}	17.44×10^{-1}	45.62×10^{0}
600	6	1.68	19.98×10^{-1}	24.35×10^{-1}	3.25×10^{0}
Local	3	1.53	15.03×10^{-1}	19.18×10^{-1}	3.12×10^{0}

(d)

Figure 12: NSDE obtained with different regularization amplitudes and using the local regularization mask. The table indicates the number of pixel required to capture the crack as well as the AEE, the MNSDE of the global field as well as in the vicinity and outside the crack.