Content-Aware Enhancement of Images with Filamentous Structures

Haris Jeelani, *Student Member, IEEE*, Haoyi Liang, *Student Member, IEEE*, Scott T. Acton, *Fellow, IEEE*, and Daniel S. Weller, *Member, IEEE*

***Abstract*—In this article we describe a novel enhancement method for images containing filamentous structures. Our method combines a gradient sparsity constraint with a filamentous structure constraint for effective removal of clutter and noise from the background. The method is applied and evaluated on three types of data: confocal microscopy images of neurons, calcium imaging data and images of road pavement. We found that images enhanced by our method preserve both the structure and the intensity details of the original object. In the case of neuron microscopy, we find that the neurons enhanced by our method are better correlated with the original structure intensities than the neurons enhanced by well-known vessel enhancement methods. Experiments on simulated calcium imaging data indicate that both the number of detected neurons and the accuracy of the derived calcium activity improved. Applying our method to real calcium data, more regions exhibiting calcium activity in the full field of view were found. In road pavement crack detection, smaller or milder cracks were detected after using our enhancement method.**

***Index Terms*—Image enhancement, Denoising, Neuron imaging, Calcium imaging, Crack detection.**

I. Introduction

IMAGE enhancement is an important pre-processing step required for extracting useful information from images containing noise, clutter and other undesirable objects. Natural images exhibit significant variation in content, so the enhancement usually takes the approach of modeling natural scene statistics [1], [2]. On the contrary, imaging in science and engineering typically targets a specific class of objects. So, the structure and intensity values remain roughly similar across different acquisitions. In neuron microscopy, the 3D structure of individual neurons is of central importance for understanding the extent of neuronal diversity [3] and their characteristics. For instance, dendrite morphology [4] is used by neuroanatomists to classify types of neurons. Several functional behaviors [5] are derived from this shape-based neuron morphology analysis. The branching patterns [6] of serotonergic neurons are analyzed for studying their behavior. Various labs have collaborated and shared large quantities of neurons online [3], [7]. Neuroanatomists are trained to manually trace neurites using features such as intensities, branch orientations, distances between branches, calibers, tortuosities, colors, and spines or boutons. While autotracing algorithms [8]–[10] capture the general layout of neurites, often it is not possible to distinguish reliably between the processes belonging to different neurons [11]. It is because the above-mentioned features are not correctly recovered by the autotracing algorithms. Furthermore these algorithms can be sensitive to noise and clutter due to signal from inessential sources such as lipids. Confocal microscopy images are degraded by low signal to noise ratio (SNR), clutter and non-uniform illumination. Removing these artifacts and enhancing the desired structure can be beneficial for automatic morphological analysis of the neurons. For example, automatic tracing algorithms [9], [12] use all pixels with high intensity values as possible components of neurons to develop a skeleton for neuron classification. Having clear backgrounds and high contrast, these tracing methods may provide more accurate classifications.

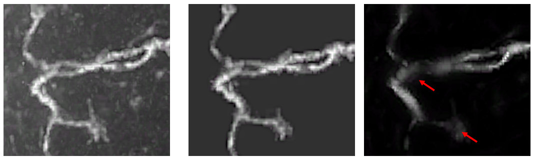

(a) Input (b) CAEFI (c) Vesselness

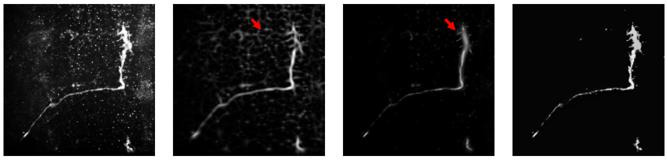
Fig. 1. The image in (a) is the cropped region of the CN1 neuron. (b) and (c) show the result of enhancement using (our) CAEFI method and the vesselness [13] method, respectively. Loss in the intensity and structure can be clearly seen in (c).

Most existing denoising methods employ prior probability distributions. For instance, non local denoising methods use Gaussian [14] or Poisson [15], [16] probability distributions as similarity measures to weight the patches around a given site and take the weighted average to remove noise at that site. MIDAL [17] assumes a Gamma distribution to remove multiplicative noise. A recent Bayesian method [18] applies denoising directly to MRI measurements in complex domain to remove additive Gaussian noise. These methods perform better if the noisy images follow a similar distribution. Furthermore, these methods require tuning various parameters that are sensitive to image contrast and geometrical structure. A non-

This research is supported by NIH R21 EB022309, NSF ABI 1759802, the Thomas F. and Kate Miller Jeffress Memorial Trust, Bank of America, Trustee, NVIDIA Corporation with the donation of the GPUs.

All authors are affiliated with the Charles L. Brown Electrical and Computer Engineering Department at the University of Virginia, Charlottesville, VA 22903 USA. H. Jeelani is the corresponding author (email: gj8an@virginia.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes a document with additional figures and discussions. Contact gj8an@virginia.edu or hl2uc@virginia.edu for further questions about this work.



(a) Input (b) LDE (c) Vesselness (d) CAEFI

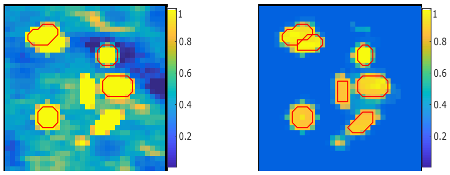
Fig. 2. A sample 2D neuron input image in (a) is enhanced using different methods. It can be seen that (d) has a clear background compared to the image in (b) and a better intensity profile of the original neuron than the LDE [31] image in (c).

parametric data-driven method [19] is based on empirical Bayes estimation, in which the prior is modeled using Markov random fields. A Rician noise model is additionally incorporated to denoise MRI images. Another data-driven method UINTA [20] does not rely on any prior statistical structure and estimates its parameters automatically, without imposing any shape constraints. BM3D [21] uses a shape-adaptive DCT but does not explicitly enforce filamentous shape constraints. Therefore these methods might be ineffective in removing clutter that does not have a filamentous shape. There exist image enhancement methods that exploit filamentous structure to enhance vessel-like objects in images. The Vesselness [13] method provides a shape-constraining framework for vessel enhancement. This method quantified the filamentous structure using eigenvalues derived from local gradients. Similar ideas have been exploited in [22], [23] to enhance sophisticated vascular structures. A data-adaptive method [24] introduces a multiscale network; it is basically similar to Vesselness, except that the network is trainable. The Jerman filter [25] method aims to provide robust vessel enhancement by regularizing the filter response obtained from eigenvalues. Other methods [26]–[28] employ specialized multi-direction and multi-scale filters to measure the presence of filamentous structure at each pixel location. Since a reasonable number of neurons have filamentous shape, these approaches have been applied to confocal microscopic neuron images as well. However, these methods modify the intensity levels during the enhancement process. Preserving the intensities of image structures restricts the enhancement achievable by existing algorithms. A new method is needed for these imaging problems. Such an intensity distribution is an important clue for classifying neuron phenotype and monitoring neuron activity [29]. Also, existing methods do not combine gradient sparsity and filamentous structure constraints together. The piece-wise smoothness of filamentous structure gives rise to sparse gradients that can be used along with structural measures to provide more effective enhancement. In our previous work [30] we show that by ignoring gradient sparsity, methods [13], [26] can enhance faint clutter. We replicate the observations in Fig. 2 for convenience.

In calcium imaging [32], the calcium ions in neurons are used for recording population dynamics such as neuronal signaling. Calcium imaging also can obtain functional maps of parts of the central nervous system [33]. The calcium activity usually occurs in cellular subcompartments, and the initial step in automatic calcium analysis is the detection of these regions of interest (ROIs). Typically a calcium indicator that chelates and illuminates the calcium ions is used. The intensity information of calcium indicators is captured by a microscope, and then the calcium activity is deconvolved from the much slower dynamics of the indicator in the ROI. The intensity information is important because most of the subsequent analyses mainly depend on the calcium activity obtained after deconvolution. The detection and deconvolution of calcium becomes harder due to nonspecificity of calcium ions, limited imaging rate, low SNR of the calcium signal [34], and the presence of unpredictable neural processes and/or neuropil activity [35]. Enhancing calcium imaging data could increase the number of detected neurons and provide accurate ROI detection for measuring the calcium activity. The enhanced images can also provide accurate whole-brain activity maps [36]–[38] for predicting network types based on animal behavior. Nonlinear matrix factorization techniques [39], [40] deconvolve calcium from noisy raw data. However, these methods do not effectively deal with demixing of overlapping neural sources. The constrained non-negative matrix factorization (CNMF) approach [35] avoids this problem by constraining calcium dynamics with autoregressive models. Furthermore, this method treats denoising, demixing and deconvolution simultaneously in one optimization framework.

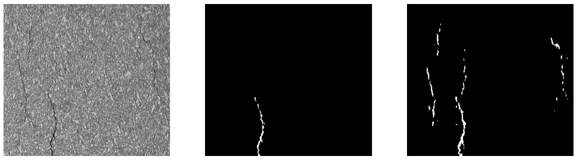
Besides the necessity of image enhancement in the two biomedical imaging techniques mentioned above, enhancement is also needed in civil engineering and materials imaging. One such application is the identification and analysis of cracks in images of various materials. Cracks are often filamentous in shape and can be seen in pavements, concrete buildings and various other types of materials. They can be surrounded by other materials such as sand and stones. Detected cracks in images can be used to perform sensitivity analyses of longitudinal cracking of pavements made from different materials [41]. Failure modes, maximum loads of notched concrete beams and various other assessments [42] can be performed on detected cracks as well. Therefore enhancement of images containing cracks can be a useful preprocessing step for accurate crack detection and its consequent analyses. The data acquired for crack detection can be too big for manual analysis, so automatic crack detection techniques are needed. Several automatic crack detection algorithms [43], [44] for road pavement surfaces have been developed. The overall performance of these algorithms can be improved by feeding enhanced images, as the enhanced images will mostly contain cracks. The number of features to be learned by a neural network-based crack detection algorithm [45], [46] also can be reduced since the enhancement removes unnecessary details from training images. In road pavement crack detection, a MATLAB toolbox named CrackIT [43] uses pre-processing techniques like intensity normalization, histogram equalization, and pixel saturation for improving crack detection; the images are not enhanced otherwise. The convolutional neural network approaches [45], [46] learn features from insignificant image patches in background as well as important patches containing cracks to detect the cracks in the whole image. Both of these methods can take advantage of image enhancement techniques.

Unlike our previous work, in this paper we apply and



(a) Input (b) CAEFI

Fig. 3. The raw unenhanced image of simulated calcium data in (a) above shows four detected ROIs by the CNMF detection algorithm [35] and the image in (b) is enhanced by CAEFI and shows seven detected ROIs by the same algorithm.



(a) Input (b) Unehanced (c) CAEFI

Fig. 4. The image in (a) above shows raw input of a road pavement. The image in (b) above shows the cracks detected by CrackIT [43] algorithm without any enhancement. The image in (c) shows cracks detected by the same algorithm after using CAEFI enhancement. It can be seen near the regions pointed by the red arrows that more cracks are found after enhancement.

evaluate our method on different types of data. We also assess our method by adding a high amount of clutter and noise to the images from the BigNeuron (BN) dataset. The effects of the algorithm parameters and the algorithm convergence are also described.

A. *Our Contribution and Outline*

We present a Content-Aware image Enhancement method for FIlamentous structures (CAEFI) that includes both gradient sparsity and structural priors. The enhancement framework is formulated as an optimization problem with data fidelity and the structure weighted gradient sparsity terms in Section II. We demonstrate that our framework can be used on various types of images, by applying it on three types of data that have filamentous structure: neuron microscopy images, calcium imaging data and road pavement images. We evaluate the efficacy of our method in the presence of different amounts of clutter and noise. We show that the CAEFI approach is effective at removing artifacts from these types of images. The appropriate experimental setups and the evaluation approaches for three types of data are described in Section III. Our approach is shown to preserve both the structure and the intensities of the objects. The results of our evaluation, the influence of the parameters and the convergence of our framework are discussed in detail in Section IV. We conclude this paper in Section V.

II. Theory

The CAEFI method exploits two main characteristics in a given image: gradient sparsity and filamentous structure. These characteristics are used as regularizers in the CAEFI framework and are described separately below.

A. *Gradient Sparsity*

We assume the underlying structure varies and the background remains constant. However, clutter and noise give rise to many non-zero gradient magnitudes. By enforcing gradient sparsity, we can eliminate these spurious gradient magnitudes. Therefore, gradient sparsity is an effective and valid assumption sufficient to obtain clean backgrounds free from noise and clutter.

If I is the original input image and S is the enhanced image, we formulate the enhancement task for minimizing the error between S and I, and the gradient sparsity in the background as:

$$S = \underset{S}{\operatorname{argmin}} \|S - I\|_i^j + \lambda \cdot \|\mathcal{D} \circ S\|_0 \tag{1}$$

Here, $\mathcal{D}$ is an operator that computes the differences between pixel values in vertical and horizontal directions. The value of i can be chosen as 0 or 1 with $j = 1$, or as 2 with $j = 2$. However, many image restoration tasks [47]–[49] use i as 2 because the $L2$-norm is differentiable. But using an $L1$ or $L2$ norm in the data fidelity tends to create layering artifacts or retain inhomogeneous illumination. The effects of using $L2$ and $L0$ norms for the data fidelity are shown pictorially in Fig. 5. Faint clutter and inconsistent illumination are common in neuron images. Therefore, we use the $L0$-norm for the data fidelity.

B. *Filamentous Structure*

Gradient sparsity alone is not sufficient to remove clutter while preserving the filamentous structure such as that of neurons. The high intensity noise cannot be differentiated from the structure of interest. There are existing methods [13], [25], [26], [31] for neuron enhancement that claim to measure filamentous structure using singular values. In a 2D case, the singular values $s1$ and $s2$ signify the length of the object in horizontal and vertical directions, respectively. For a 2D filament, the difference between s1 and s2 is small; therefore the reciprocal of Q results in a higher value for a filamentous structure. We use the structure index [50], Q, to quantify this structure. For a 2D image, Q is defined as,

$$Q = s_1 \cdot \frac{s_1 - s_2}{s_1 + s_2} \tag{2}$$

where $s_1 \geq s_2$ are obtained by taking an SVD on local gradient matrices [50]. For the 3D case, the optimal version of Q depends on the appropriate shape of filamentous structure. In our case, we find two large and one small singular values characterize a 3D filamentous structure for our datasets. Hence, Q is defined as:

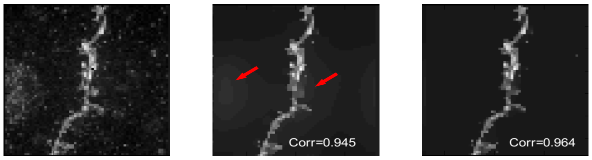
$$Q = s_1 \cdot s_2 \cdot \frac{s_1 - s_3}{s_1 + s_3}, s_1 \geq s_2 \geq s_3 \tag{3}$$

An illustration showing the advantage of using this structural enhancement prior is shown in Fig. 6.

C. *Content-Aware Framework*

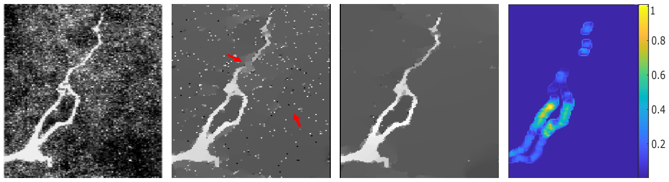
The gradient sparsity and filamentous structure priors are combined to form our content-aware image enhancement framework:

$$S = \underset{S}{\operatorname{argmin}} \|S - I\|_0 + \lambda \cdot \sum_p \frac{1}{Q(p) + \epsilon} \|\mathcal{D} \circ S(p)\|_0 \tag{4}$$



(a) Input (b) $L2$-norm (c) $L0$-norm

Fig. 5. The image in (a) shows a patch from the Condron dataset and the images enhanced using $L2$-norm and $L0$-norm are shown in (b) and (c) above, respectively. In (b), layering artifacts can be seen in the regions pointed by red arrows. A hand-segmented mask containing only the neuron region from the input image was used to compute the correlation coefficient shown in the bottom right corner.



(a) Input (b) Without Q (c) With Q (d) Q Map

Fig. 6. The image in (a) above is a cropped region of a neuron from the BigNeuron dataset with added clutter; this image was used as input to the CAEFI method. The images in (b) and (c) were generated without and with using structural weighting (Q) in the CAEFI framework, respectively. The artifacts due to not applying structural weighting can be seen in the regions pointed by red arrows in (b). In (d) pixel locations belonging to filamentous structure have higher Q-map weightings.

The variable S represents the enhanced image, I is the input image, and p is the pixel location. $Q(p)$ is the structure index obtained from the local gradient approximated around the pixel location p. A 3D version of Q defined in (3) is used on calcium and neuron data, and since the road pavement data are 2D we use a 2D version of Q defined in (2). A negligibly small value ϵ is added to the denominator to avoid divide-by-zero errors. Notice the gradient sparsity constraint in (4) is weighted by a factor of $\frac{1}{Q}$. This weighting complies with our assumption that the structure is less sparse than the background. It will appropriately decrease the gradient sparsity near the structure and increase it in the background region. Our optimization problem in (4) has two terms with a nondifferentiable $L0$-norm. Therefore to simplify our iteration updates, we separate them using two auxiliary variables: $E = S - I$ and $U = \mathcal{D} \circ S$. We rewrite our problem as in (5). Furthermore, by splitting the optimization problem we can also apply it to larger datasets without compromising the speed of convergence [51].

$$\min_{S,E,U} \|E\|_0 + \sum_p \frac{\lambda}{Q(p)+\epsilon} \|U(p)\|_0 + \eta \|S - I - E\|_2^2 + \beta \|\mathcal{D} \circ S - U\|_2^2 \tag{5}$$

To solve for each optimization variable, we further divide (5) into three subproblems:

$$\min_S \|S - I - E\|_2^2 + \frac{\beta}{\eta} \|\mathcal{D} \circ S - U\|_2^2 , \tag{6}$$

$$\min_U \|\mathcal{D} \circ S - U\|_2^2 + \sum_p \frac{\lambda}{\beta \cdot Q(p) + \epsilon} \|U(p)\|_0 , \tag{7}$$

$$\min_E \|E\|_0 + \eta \|S - I - E\|_2^2 . \tag{8}$$

Algorithm 1 Algorithm for solving (4)

```
Initialize: $E = \mathbf{0}$, $S = I$, $tol = 10^{-3}$, $err = \infty$,
$MaxIter = 60$, $i = 0$
while $err > tol$ and $i < MaxIter$ do
calculate $Q$ using (3) or (2) and $S$
    $\hat{S} \leftarrow \frac{\eta(I+E)+\beta D_x^T H + \beta D_y^T V}{\eta + \beta D_x^T D_x + \beta D_y^T D_y}$    ▷ Sol. for (6)
    $H \leftarrow D_x \hat{S}, \quad V \leftarrow D_y \hat{S}$
    if $(H(p)^2 + V(p)^2) < \frac{\lambda}{\beta \cdot (Q(p)+\epsilon)}$ then    ▷ Sol. for (7)
        $H(p) \leftarrow 0, V(p) \leftarrow 0$
    end if
    $E \leftarrow \hat{S} - I$
    if $E(p)^2 < \frac{1}{\gamma}$ then $V(p) \leftarrow 0$    ▷ Sol. for (8)
    end if
    $\beta \leftarrow \kappa \cdot \beta, \quad \eta \leftarrow \kappa \cdot \eta$
    $err \leftarrow \|S - \hat{S}\|, \quad S \leftarrow \hat{S}, \quad i \leftarrow i + 1$
end while
```

The gradient U has two parts, the horizontal gradient H and the vertical gradient V. They are computed using the horizontal and vertical difference matrices D_x and D_y respectively. Our continuation strategy enforces equality constraints on two auxiliary variables and scales β and η by a factor of κ at every iteration. The problem in (6) is quadratic, and the closed form solution can be obtained by differentiating (6) with respect to S and setting it to zero:

$$\eta(S - I - E) + \beta[\mathcal{D}_x^T(\mathcal{D}_x S - H) + \mathcal{D}_y^T(\mathcal{D}_y S - V)] = 0 \tag{9}$$

Solving for S directly from (9) requires a computationally expensive large matrix inversion. We solve for S in the Fourier domain because the expensive inversion operation can be replaced by element-wise division. So, S in the Fourier domain is given as:

$$S = \mathcal{F}^{-1}\left(\frac{\eta\mathcal{F}(I+E) + \beta[\mathcal{F}(\mathcal{D}_x)^T\mathcal{F}(H) + \mathcal{F}(\mathcal{D}_y)^T\mathcal{F}(V)]}{\eta\mathcal{F}(\mathbf{1}) + \beta[\mathcal{F}(\mathcal{D}_x)^T\mathcal{F}(\mathcal{D}_x) + \mathcal{F}(\mathcal{D}_y)^T\mathcal{F}(\mathcal{D}_y)]}\right) \tag{10}$$

In (10), $\mathcal{F}$ is the Fourier operator and $\mathcal{F}(\mathbf{1})$ is the Fourier transform of the delta function. The constants η and β are multiplied element-wise. The subproblems in (7) and (8) are solved for each pixel separately using non-iterative hard thresholding [48]. The key steps for solving all three subproblems are described in Algorithm 1. The initial values of η, β, κ are assigned empirically, and we found them to be stable for all experiments (see Section IV-D for details). Our Algorithm 1 stops if the $L2$-norm of the difference between the image at the current iteration and the image at previous iteration does not increase beyond a tolerance level defined in Algorithm 1. Although we can also use the metric-Q [50] that measures amount of cleanliness without a reference image, we did not observe any noise after convergence by using this difference measure. The additional maximum iterations criterion ensures that the algorithm does not run forever. Note the non-convex $L0$-norm and the alternating minimization approach make our framework converge to a local minimum.

III. Methods

To demonstrate the benefits and wide applicability of our enhancement method, we evaluate CAEFI on three types of datasets: neuron microscopy, calcium imaging and road pavement datasets. The experimental setups and evaluation approaches for these datasets are described in the following subsections. Although the evaluation approach varies for each dataset, the CAEFI method will generally be assessed based on its ability to extract the desired underlying structure from a noisy environment. We also qualitatively and quantitatively compare the performance against existing methods.

A. Neuron Microscopy Data

Neurons acquired by confocal microscopy contain measurement noise and clutter due to lipids and other unaccounted sources, so they need to be enhanced for further analysis. Although neurons come in a variety of shapes, we restricted our analysis to neurons having filamentous shape due to the underlying assumptions of our method. We considered two versions of 3D confocal microscopy images: Drosophila larvae neurons [31] from Dr. Barry Condron's (CN) lab at the University of Virginia and human pluripotent stem cell neurons from Dr. Rick Livesey's lab at Cambridge University, which are part of the BigNeuron (BN) dataset [3], [52]. The CN dataset has natural clutter, whereas the BN dataset has some noise in the background but is mostly clean. We added randomly generated clutter to make the background in the BN data more challenging. A clutter dictionary is learned from 3D patches of size 7x7x5 extracted from whole images of CN dataset using the K-SVD algorithm [53]. The error threshold and iteration limit of K-SVD algorithm is set to 0.001 and 10 respectively. The dictionary has 500 atoms with $122,500$ dictionary coefficients. In order to add clutter to each clean neuron, one atom is randomly picked to generate one patch of clutter. This patch is added to the top-left region of the clean image. Another patch is generated similarly and added to the previous patch with an overlap size of 6x6x4; the overlapped patches are averaged within 7x7x5 for normalization. The process is repeated until we finish filling the background regions with the clutter. Finally, we apply the CAEFI method on 5 images from the BN dataset and 5 images from the CN dataset. To evaluate our method, we test how effectively CAEFI can suppress background while preserving the neuron structure. We compare our method against the Vesselness method [13], Jerman [25], PNL [15], and BM3D [21]. In the Vesselness algorithm, there are five main parameters to tune. Three of which are vesselness constants (α, β, C) that determine: if a structure is a line or a plane, the deviation from a blob-like structure, and threshold between eigenvalues of noise and vessel structure, respectively. The two other parameters are the vesselness filter scale range and the spacing between the range. A brute force strategy was used to find each of these parameters. For both datasets, the vesselness constants were the same with values: $\alpha = 0.5, \beta = 0.5, C = 500$. The scale range and its spacing was different for two datasets. The BN dataset gave the best results for a scale range between $[7, 9]$ with spacing of 0.5, and the CN dataset gave the best result for the scale range between $[1, 5]$ with the spacing of 2. Unlike the Vesselness filter, the Jerman method aims to amplify and provide uniform intensity profiles for filamentous structure. For both datasets, the vesselness constants were and the scale range were same as the Vesselness method. However, the parameter τ that controls the magnitude of the filter response is set in the range of $[6, 10]$ for CN data and in $[100, 150]$ for BN data. The video version of BM3D had only one parameter, i.e. the standard deviation of noise σ_{bm} to tune. The range of σ_{bm} for the BN and CN data was $[500, 1000]$ and $[25, 100]$, respectively. The PNL method normally uses PURE to find its optimal parameters; this did not work with our data, so we manually tuned the parameters. This method had six parameters, and the most sensitive among these were: α_{pnl}, which controlled the amount of Poisson priors to be enforced; β_{pnl}, which adjusted the effect of a pre-estimated image (refer [15] for more details); w_{pnl}, the search neighborhood; and q_{pnl}, which scaled input and rescaled the output. The ranges of $\alpha_{pnl}, \beta_{pnl}, w_{pnl}, q_{pnl}$ for BN data were $[0.17, 0.3], [0.2, 0.4], [12, 15], [0.95, 0.98]$, respectively. For CN data, these ranges were $[0.4, 0.5], [0.2, 0.3], [9, 12], [0.25, 0.45]$, respectively.

The parameters of our method are described in Section IV-D. The values of major parameters, i.e. window size and smoothness λ parameter, are set as 9 and 0.002, respectively, for all images in CN data. Since the BN data has high intensity clutter, the λ value was scaled by ratio of mean intensity of BN data to the mean intensity found in CN data. The window size was the same as that of CN data. The datasets include hand-segmented ground truths stored in the 'swc' format so direct comparison of enhanced results is not possible. Therefore for quantitative evaluation, we obtain a binary mask (see supplementary material Section 2) of ground truth from the 'swc_to_mask plugin' of the Vaa3D [54] software. The resulting mask had a different neuron thickness but encapsulated most of the actual neuronal region. A Pearson's correlation coefficient $\rho = \frac{cov(X,Y)}{\sigma_X \sigma_Y}$ is computed in masked regions between the ground truth image and the enhanced images to show that intensity values are not altered after using CAEFI. Since the data is 3D, the SSIM is computed for each frame, and an average score of all frames is used for evaluation and discussion of results. The SSIM, PSNR, correlation scores for both datasets are combined, and a matched-pairs t-test is performed to conclude if scores of CAEFI are statistically different from other methods. Arrows are used to show preservation of structure and the presence or absence of noise in the enhanced images.

B. Calcium Imaging Data

Calcium imaging is a useful technique for measuring neuronal firing activity by capturing illuminating calcium ions into a series of digital images recorded by a microscope. Image intensity represents calcium fluorescence and is the governing factor that determines the instance of neuron firing and other behaviors, justifying evaluation of the CAEFI method on such data. Various methods [55], [56] exist for detection of neurons but the CNMF method [35] recovers calcium activity by decomposing the observed fluorescence data (F) into spatial components (a_i) containing appropriate neuron ROIs and temporal components $c_i(t)$ having calcium information of the corresponding ROI. Mathematically, the observed

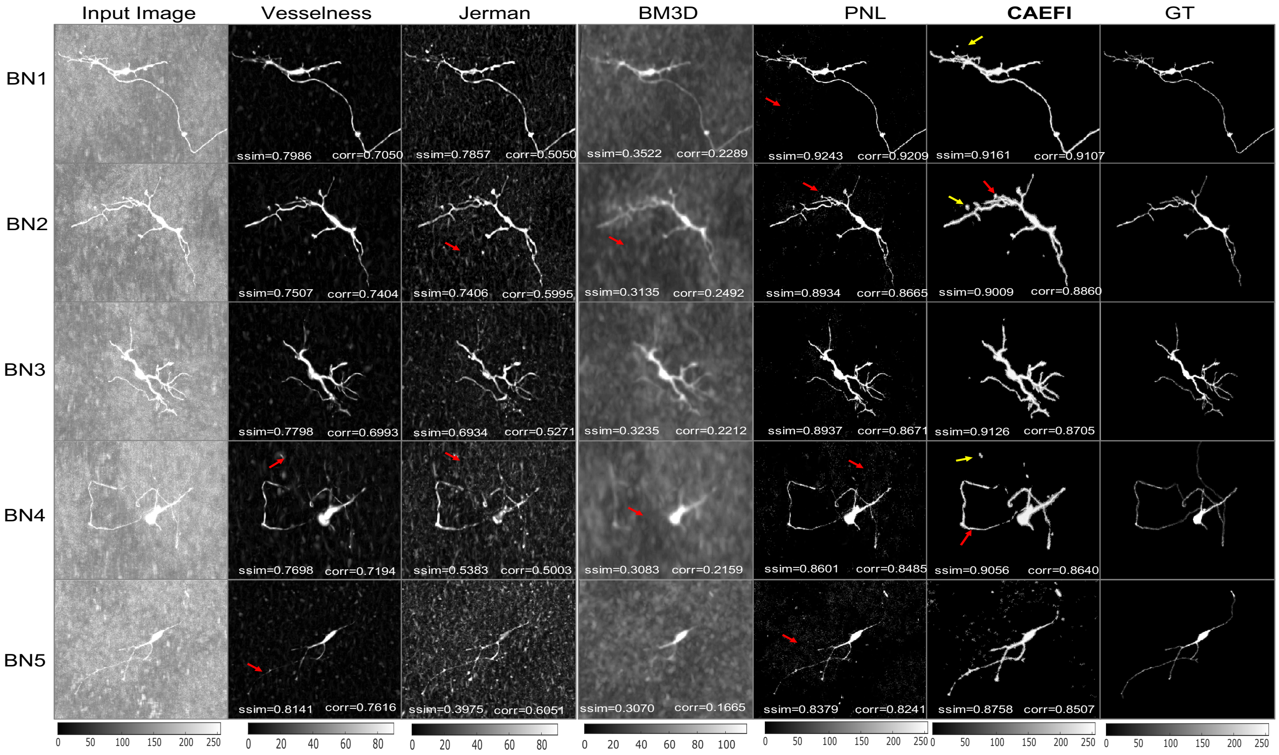


Fig. 7. Enhancement results of five neurons from BigNeuron (BN) data for different method. In the left most column are input images that were generated by adding synthetic clutter to images. The following columns show images after being enhancement by Vesselness, Jerman, BM3D, PNL, and CAEFI methods respectively. The SSIM and correlation scores are shown in the bottom right corner of appropriate image. The colorbars shown in the last row of each column is the same for all images in that column.

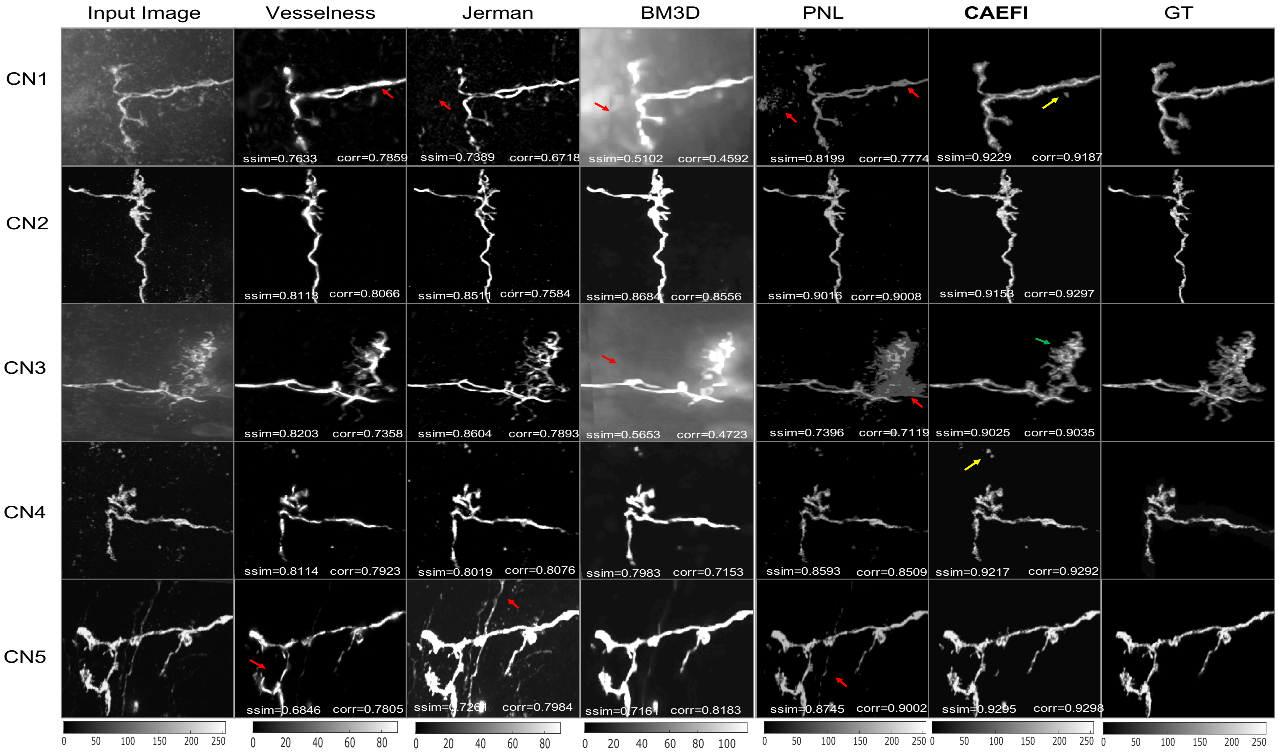


Fig. 8. Enhancement results of five neurons from Condron (CN) data for different methods. This figure is similar to Fig. 7, however, this data did not have synthetic clutter.

fluorescence is written as $F(t) = \sum_{i=1}^{K} a_i c_i(t) + B(t) + \epsilon(t)$. The variable $B(t)$ represents background and $\epsilon(t)$ is the zero mean Gaussian noise at the frame t. Additionally, an autoregressive model is enforced on the temporal component to recover calcium activity. These components are solved using an alternating minimization framework. We show that this method can be improved by pre-processing the data with our enhancement method. Since the CNMF method is non-convex, their algorithm is sensitive to the initial center location of the neurons. We choose greedy initialization [35] for our experiments. In this approach, the user roughly defines the number of neurons to be found in the data, and the algorithm will correlate each position with a predefined 2D Gaussian kernel. The process will stop after passing through all positions in a given 2D image, and then it will greedily consider the top correlated locations. The values of parameters set for this approach have been described in appropriate sections below. For evaluation on real data, we do not have the ground truth calcium activity, so we first evaluate benefits of using our method on simulated data with known ground-truth. We then apply it on real data obtained from the neurofinder [57] website to measure the increase in the number of neurons detected. Further details are given below:

1) Simulated data: Seven Gaussian shaped neurons of various orientations are generated as shown in Fig. 9. Their firing activity is modeled using an iid Poisson process with firing threshold of 0.3. The calcium activity is modeled as an autoregressive process with two parameters $(0.98, 1)$. Gaussian noise is added to the generated calcium activity. Finally, clutter cropped from real data along with time varying 2D spatial Gaussian noise is added to the simulated neurons. The BM3D parameter was set as $\sigma_{bm} = 25$. The PNL parameters were: $\alpha_{pnl} = 0.1$, $\beta_{pnl} = 0.3$, $w_{pnl} = 7$, $q_{pnl} = 0.75$. The CAEFI parameters are set as: $\lambda = 0.007$, window size $w = 5$. The parameters of the CNMF were fixed while comparing all methods and are assigned as follows: an initial guess (K) is set as 8, the standard deviation of the Gaussian kernel (used for initialization) is set as 4, autoregressive dynamics is set to 2, merge threshold is set to 0.7, and finally the number of iterations is set as 3. The enhancement method is evaluated based on the number of neurons detected. The $L2$-norm of the difference between recovered calcium activity with the ground truth is also computed.

2) Real data: The calcium imaging data set labelled as "00.00" [57] is used. This 2D data has large dimensions of $512 \times 512 \times 3024$. Due to its large size, the dataset is divided into four $128 \times 128 \times 3024$ regions to fit into memory. For BM3D, $\sigma_{bm} = 35$ was used. The PNL parameters were: $\alpha_{pnl} = 0.1$, $\beta_{pnl} = 0.3$, $w_{pnl} = 9$, $q_{pnl} = 0.27$. All analysis is performed within these cropped regions. The CAEFI parameters are set as follows: $\lambda = 0.001$, window size of 9. Due to high spatial resolution and small size of neurons only a cropped region is shown in paper for better visibility. For evaluation, we compare the number of extra cells detected by the CNMF method before and after enhancement of images from each region.

C. Road Pavement Data

The data supplied with the CrackIT toolbox [43] is used. Unlike neuron microscopy images, these images are obtained from a conventional digital camera. They were captured during sunlight hours with camera's optical axis perpendicular to the road surface and its lateral edges parallel to the road axis. The 2D images have an image size of 2048×1536 pixels and each pixel corresponds to a road area of 1mm^2. We use an existing crack detection algorithm [43] to show improvements in crack detection with our method. Parameter settings for BM3D were $\sigma_{bm} = 90$ and for the PNL were: $\alpha_{pnl} = 0.17$, $\beta_{pnl} = 0.27$, $w_{pnl} = 9$, $q_{pnl} = 0.4$. The CAEFI parameters are same as the real calcium imaging data. All the parameters of the CrackIT method had default values. This algorithm uses anisotropic diffusion along with other pre-processing techniques such as adjusting saturation levels to enhance crack detection. It also has the ability to classify and characterize the detected crack. The ground truth contains rough hand-segmented rectangular blocks that include cracks and are smaller in the size compared to the size of input images. The ground truths are of size 27×20, and the input images are of size 2048×1536 pixels. As a result, we did not compute quantitative metrics. Therefore we visually evaluated the results and show improvements with the aid of arrows. Note that results were obtained after including CAEFI in the CrackIT pre-processing pipeline. We also include the other pre-processing steps of the CrackIT algorithm.

IV. Results

A. Neuron Microscopy

The top row of Fig. 7 shows neurons from the BN data with heavy added clutter and noise in the background. The intensities of Vesselness and Jerman method have been adjusted for better visibility (See colorbars in Fig. 7 and 8). It can be seen that all methods remove noise-like patterns, however the residual clutter can be seen near red arrows in Vesselness and Jerman methods. The PNL method effectively removes clutter for most cases; however, noise can be seen in regions pointed by red arrows. Although the CAEFI method did not have noise and clutter for most cases, boundaries of the neuron (pointed by the red arrows) show hairy structures. These structures cause the neurons to appear wider than their ground truth. This effect can be reduced by tuning the λ parameter for each image. The SSIM and correlation values of PNL and CAEFI methods were close to each other for the BN data. The images of BM3D method show significant clutter and loss of structure in all neurons, especially in case of BN4 and BN5. This method has the lowest scores among all other methods. There are cases where all methods fail to remove isolated structure pointed by yellow arrows. In case of real data, i.e. the CN data in Fig. 8, higher performance of CAEFI can be clearly seen, especially in the cases CN1 and CN3 neurons where the background clutter is higher. The green arrow shows CAEFI better preserves structure compared to all other methods. However, some aliasing can also be seen near this region. For the PNL method, structure and intensity loss can be clearly seen near the red arrows. The performance of Jerman method was lower on average compared to Vesselness as seen in the SSIM and correlation values. The Vesselness and the Jerman methods remove clutter. However, we observed intensity loss; therefore, we scaled the images for better visibility for CN data as well. Some structural loss can be seen near the red arrows. The SSIM and correlation values shown in Fig. 7 and Fig. 8 for the CAEFI enhanced neurons are higher than for other competing methods. The average scores of both datasets are summarized in tables I and II. Multiple comparison pairwise t-tests were performed using SSIM, PSNR and correlation values of each method against CAEFI. The t-tests indicated that values of CAEFI were significantly different from other methods. The p-values comparing CAEFI against PNL were higher than for the other existing methods (see supplementary material Section 3).

TABLE I
AVERAGE SCORES OF BIG NEURON DATA

Method	SSIM	PSNR(dB)	Corr.
Vesselness	0.7826	29.9	0.7272
Jerman	0.6311	27.34	0.5474
BM3D	0.3211	19.61	0.229
PNL	0.87	35.37	0.8654
CAEFI	0.9022	36.19	0.8764

TABLE II
AVERAGE SCORES OF CONDRON DATA

Method	SSIM	PSNR(dB)	Corr.
Vesselness	0.7785	32.55	0.703
Jemran	0.7897	31.39	0.7587
BM3D	0.6917	18.81	0.664
PNL	0.839	38.14	0.822
CAEFI	0.9185	45.51	0.922

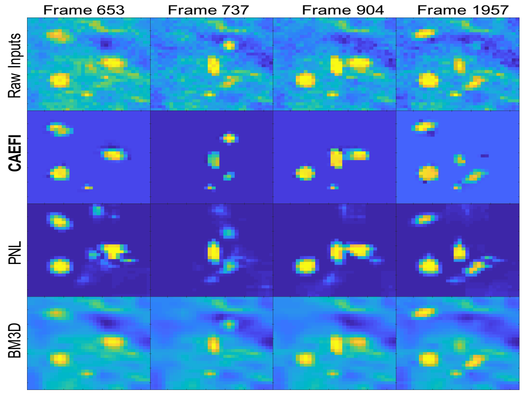
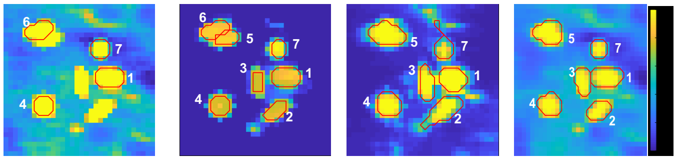


Fig. 9. The top row shows sample frames from 2000 frames of raw neuron data and their results after using various enhancement methods referred by their labels in the following rows. It can be seen that CAEFI provides cleaner background than other methods. It also does not cause blurring that is seen in other methods.



(a) Without Enh. (b) CAEFI (c) PNL (d) BM3D

Fig. 10. A maximum intensity projection of simulated data without enhancement in (a) is followed by the projection images after using different enhancement techniques. It can be seen that neuron detections increase after using enhancement. Unlike other methods, CAEFI detects overlapped neurons 6 and 5. The ROI detected for neuron 7 and 2 by the PNL method includes some of the background regions.

B. Calcium Imaging

1) Simulated data: The image in Fig. 10a is the maximum intensity projection showing contours of neurons detected by the CNMF method without using CAEFI enhancement; each of the four detected neurons are labelled. Comparing this result to the image in Fig. 10b, we can see the number of detected neurons increases from 4 to 7. We observe that the contours of detected neurons do not perfectly match those of the simulated ground truth, but the number of neurons detected after enhancement is exact. The clutter below neuron 2 that appears like a neuron is present before and after enhancement; however, in Fig. 10c the PNL method considers it as a part of neuron 2. This phenomenon can also be seen near neuron 7. In Fig. 9, CAEFI produced some structure and intensity loss in frames 737 and 1957, although without the blur visible with other methods. Consequently, the intensity loss also is visible in the projection plot (Fig. 10b). Aliasing effects near neuron boundaries can also be seen in CAEFI as compared to PNL in Fig. 9 and 10. Tuning the parameters of CAEFI trades off between background clutter and structure/intensity preservation. The calcium activity deconvolved from the region within the contours is plotted in Fig. 10 for all methods. The peak firing locations match the ground truth (shown in black dotted line). Comparing the calcium activity of neuron 6 of the unenhanced case (magenta) with the activity of CAEFI enhanced neuron 6 (green), the presence of noise can be seen in the CAEFI case since the detected ROI is not accurate as in the unenhanced case. The error in recovery can be compared from table III. However, all other methods fail to detect neuron 6, and merge it with neuron 5, except CAEFI. We argue that this separation is achieved by enforcing structural constraints. For neuron 7, only PNL and CAEFI have activities close to the ground truth, with CAEFI being more accurate. Furthermore, the activity of neuron 2 in the PNL case is noisy due to inaccurate detection of the ROI. It can be observed from the table III that CAEFI reduces calcium activity recovery errors for a majority of the neurons more than the other methods.

2) Real data: Similar observations were found in the case of real data. The Fig. 12 shows an increase in the number of neuron detections from 46 in Fig. 12a to 57 in Fig. 12b after using CAEFI. The exact evaluation of extracted activity of these neurons could not be evaluated, as we lack the ground truth. The PNL in Fig. 10c did not detect some neurons near regions pointed by the white arrows and ROIs appeared to be smaller than other methods. On the other hand, the BM3D method in Fig. 12d had larger ROIs. CAEFI had similar ROIs as that of unenhanced case. The total number of neurons detected in the cropped regions for each method is reported in table IV. As there are 16 regions (4×4), we sum the number of detected cells in each row, which corresponds to a region of size $128 \times 512 \times 3024$, and report the value for each row of regions in table IV to reduce the table length.

C. Crack Detection

The top row of Fig. 13 shows a high resolution input image with significant noise-like background along with the cracks in the road. It can be seen near regions pointed by

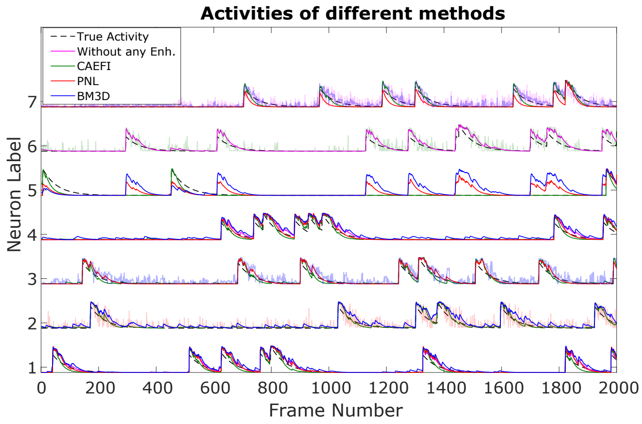
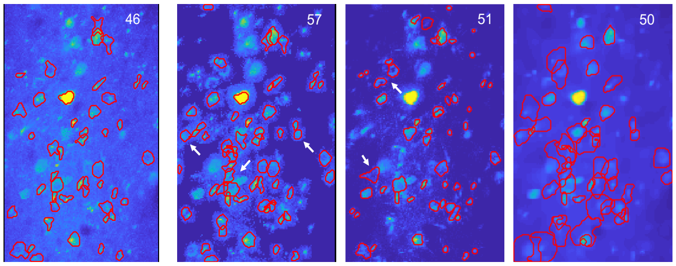


Fig. 11. Calcium activity of the detected neurons shown in Fig. 10

TABLE III
CALCIUM RECOVERY ERRORS ($L2$-NORM) OF ACTIVITIES SHOWN IN THE FIG. 11 FOR DIFFERENT METHODS

Neuron	No Enh.	BM3D	PNL	CAEFI
1	4.08	5.76	4.48	3.38
2	n/a	5.81	6.33	3.36
3	n/a	7.33	4.036	3.58
4	4.32	6.21	4.18	3.39
5	n/a	14.47	8.81	2.74
6	5.13	n/a	n/a	6.42
7	6.23	4.32	5.02	3.33



(a) Without Enh. (b) With CAEFI (c) With PNL (d) With BM3D

Fig. 12. Maximum intensity projections of a 200×128 cropped region of real calcium data with red contours showing neuron parts detected for different scenarios. The number of neuron parts detected for each scenario is shown on the top right corner. The white arrows shown in CAEFI case shows parts that are not detected by CAEFI but are detected by PNL. The converse is shown by the white arrows in the PNL case.

green arrows that more cracks are detected after incorporating CAEFI enhancement in the CrackIT algorithm. Comparing the images in the second row of column 1, 3 and 4 in Fig. 13 with the images in the third row of the corresponding columns, at least two smaller cracks were undetected before enhancement. This is shown using the green arrows in the images of the third row. The red arrows show structures that did not resemble cracks. The green arrows in BM3D method show some extra filamentous structure being detected by BM3D which was not seen in CAEFI. Conversely, green arrows in the first column of Fig. 13 show CAEFI detecting smaller cracks better than BM3D. The validity of these cracks could not be evaluated as we lack the ground truth.

TABLE IV
NUMBER OF DETECTED NEURONS IN ROWS OF CROPPED REGIONS OF REAL DATASET BEFORE AND AFTER USING DIFFERENT ENHANCEMENT METHODS.

Region Row Name	Wihout Enh.	With BM3D	With PNL	With CAEFI
R1	80	140	145	150
R2	123	150	159	199
R3	114	146	149	151
R4	86	138	140	185

D. Parameter selection and Convergence

There are five parameters in the CAEFI framework (see algorithm 1). The parameters β, η, κ that control numerical stability and the convergence rate of the solution were the same for all datasets. We observed only minor changes in enhancement for all CN data for β values in the range $[0.1\lambda, 10\lambda]$, for η values in the range $[10^3\lambda, 10^6\lambda]$, and for κ in the range $[1, 4]$ (see supplementary material Section 4 for more details). However, our method has two major parameters, λ and window size, that need to be tuned. These two parameters influence the removal of clutter and preservation of the original structure in the images. For all three datasets, we found λ to lie within the range of 0.001 to 0.007. The sensitivity of λ was assessed on all samples from 3D CN dataset. We calculate the foreground error (fg_{err}) and background error (bg_{err}) for each image at different λ values using the equations below.

$$fg_{err} = \frac{\sqrt{\sum_{i \in \mathcal{F}_g} |I(i) - I_{enh}(i)|^2}}{\sqrt{\sum_{i \in \mathcal{F}_g} I(i)^2}} \quad bg_{err} = \frac{\sqrt{\sum_{j \in \mathcal{B}_g} |I(j) - \overline{I}_{enh}|^2}}{\sqrt{|\mathcal{B}_g| \overline{I}_{enh}^2}}$$

The unenhanced input image is I, and I_{enh} is the enhanced image. The i and j are the pixels locations belonging to the foreground F_g and the background B_g respectively. The average error values of all samples are shown in Fig. 14b. It can be observed that a trade-off between fg_{err} and bg_{err} needs to be achieved. While conducting our experiments, we found that once the λ is selected (often between 0.001 and 0.007), it doesn't vary much within samples of same dataset. For the Fig. 14b the average variance across samples was 0.0092 and 0.0039 for foreground and background error respectively. The window size for all datasets except the simulated calcium imaging data was set as 9. Increasing or decreasing the windows size by 2 did not significantly alter the results. For simulated data, we set the window size as 5 because the maximum size of the ROI of the simulated neurons was 4; this size is smaller than the ROI size typically seen in the real calcium imaging data. We found that our algorithm converges before 35 iterations for images from all datasets. To observe the convergence rate we took all sample images from the 3D CN dataset. We ran the algorithm for 60 iterations to obtain the I_{cnvg} as our final converged image for each sample (I). After saving the converged image we set the tolerance level to 10^{-3} and ran the algorithm again. At each iteration we computed the L_2-norm of difference between the I_{cnvg} and the image generated at that iteration and normalize it by L_2-norm to obtain the NRMSE. We average the L_2 norm-based difference for each sample and plot the values to show the convergence in Fig. 14a. For 35 iterations, our algorithm took an average time of 15.4 seconds per sample on a computer running Windows 7 with a 32 GB

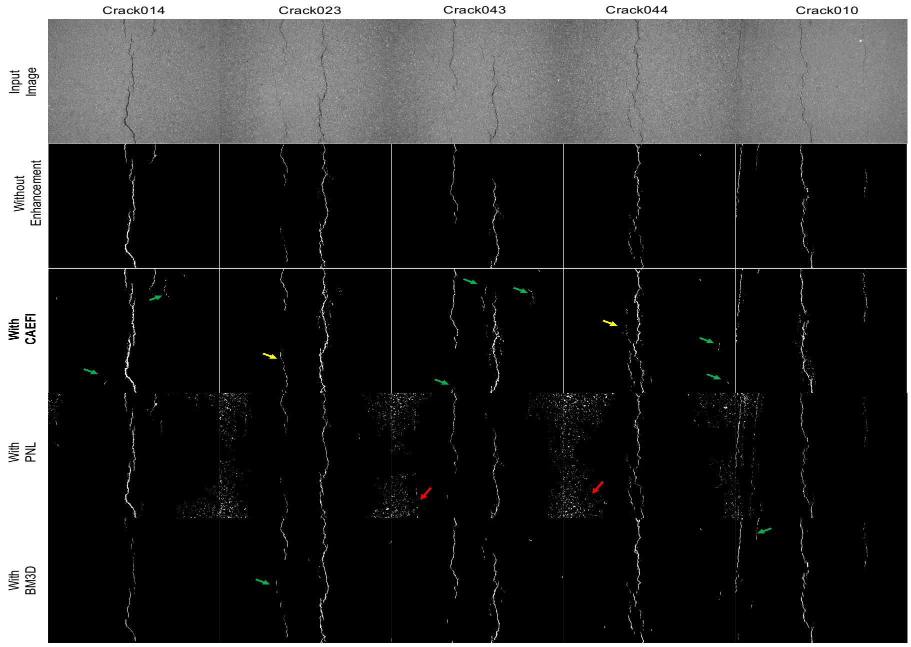
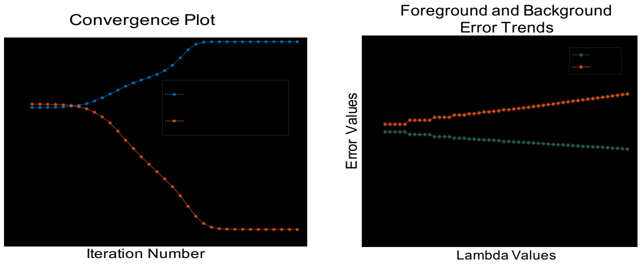


Fig. 13. Top row shows input images and the second row shows segmented cracks obtained by the CrackIT algorithm [43] without using any enhancement. The green arrows in the third row show milder cracks being detected by CAEFI. The yellow arrows show extra pixels belonging to the cracks (see first row) being detected by the CrackIT algorithm after using CAEFI.

of RAM and an i7 Intel processor; the average size of the sample images was 205×215. However, we observed the background of images obtained after 60 iterations was scaled by an arbitrary constant value. This problem occurs because our optimization framework involves the $L0$-norm that does not distinguish between values very close to zero and values far from zero. Also, the gradients are insensitive towards a constant value being added to all pixel values. We did not notice a significant loss in neuron structure in images after 60 iterations or by decreasing the tolerance level.



(a) (b)

Fig. 14. The plot in 14a shows the average rate of convergence of the CAEFI framework for the NRMSE and SSIM values w.r.t converged image. The plot in 14b shows the sensitivity of the parameter λ towards foreground and background errors for the CN dataset.

V. Conclusion

We introduced a novel image enhancement method that exploits the gradient sparsity and the structure priors for enhancing images containing filamentous structures. These two important priors are combined into one cost function and solved using a variable splitting method. The method was shown to preserve object structure and its intensity by applying it on images from neuron confocal microscopy, calcium imaging, and road pavement surface imaging. The background of enhanced confocal microscopy images had significantly less clutter, and the foreground intensities of enhanced images were highly correlated with the foreground intensities of the input images. The number of neurons detected in calcium imaging data increased, and additional cracks were detected in road pavement images after using the CAEFI enhancement.

References

[1] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-laplacian priors," in *Advances in Neural Information Processing Systems*, 2009, pp. 1033–1041.

[2] P. Chatterjee and P. Milanfar, "Is denoising dead?" *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 895–911, 2010.

[3] H. Peng, M. Hawrylycz, J. Roskams, S. Hill, N. Spruston, E. Meijering, and G. Ascoli, "Bigneuron: Large-scale 3d neuron reconstruction from optical microscopy images," *Neuron*, vol. 87, no. 2, pp. 252 – 256, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0896627315005991

[4] G. Stuart, N. Spruston, and M. Häusser, *Dendrites*. Oxford University Press, 2016.

[5] R. R. Llinás, "The contribution of santiago ramon y cajal to functional neuroscience," *Nature Reviews Neuroscience*, vol. 4, no. 1, p. 77, 2003.

[6] J. Chen and B. G. Condron, "Branch architecture of the fly larval abdominal serotonergic neurons," *Developmental biology*, vol. 320, no. 1, pp. 30–38, 2008.

[7] G. A. Ascoli, D. E. Donohue, and M. Halavi, "Neuromorpho. org: a central resource for neuronal morphologies," *Journal of Neuroscience*, vol. 27, no. 35, pp. 9247–9251, 2007.

[8] Y. Al-Kofahi, N. Dowell-Mesfin, C. Pace, W. Shain, J. N. Turner, and B. Roysam, "Improved detection of branching points in algorithms for automated neuron tracing from 3d confocal images," *Cytometry Part A*, vol. 73, no. 1, pp. 36–43, 2008.

[9] H. Peng, F. Long, and G. Myers, "Automatic 3d neuron tracing using all-path pruning," *Bioinformatics*, vol. 27, no. 13, pp. i239–i247, 2011.

[10] R. Parekh and G. A. Ascoli, "Neuronal morphology goes digital: a research hub for cellular and system neuroscience," *Neuron*, vol. 77, no. 6, pp. 1017–1038, 2013.

[11] R. Gala, J. Chapeton, J. Jitesh, C. Bhavsar, and A. Stepanyants, "Active learning of neuron morphology for accurate automated tracing of neurites," *Frontiers in neuroanatomy*, vol. 8, p. 37, 2014.

[12] H. Xiao and H. Peng, "App2: automatic tracing of 3d neuron morphology based on hierarchical pruning of a gray-weighted image distance-tree," *Bioinformatics*, vol. 29, no. 11, pp. 1448–1454, 2013.

[13] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," in *Medical Image Computing and Computer-Assisted Intervention-MICCAI'98*, W. M. Wells, A. Colchester, and S. Delp, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 130–137.

[14] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 60–65.

[15] C.-A. Deledalle, F. Tupin, and L. Denis, "Poisson nl means: Unsupervised non local means for poisson noise," in *Image processing (ICIP), 2010 17th IEEE international conference on*. IEEE, 2010, pp. 801–804.

[16] A. A. Bindilatti, M. A. Vieira, and N. D. Mascarenhas, "Poisson wiener filtering with non-local weighted parameter estimation using stochastic distances," *Signal Processing*, vol. 144, pp. 68–76, 2018.

[17] J. M. Bioucas-Dias and M. A. Figueiredo, "Multiplicative noise removal using variable splitting and constrained optimization," *IEEE Transactions on Image Processing*, vol. 19, no. 7, pp. 1720–1730, 2010.

[18] F. Baselice, G. Ferraioli, V. Pascazio, and A. Sorriso, "Bayesian MRI denoising in complex domain," *Magnetic resonance imaging*, vol. 38, pp. 112–122, 2017.

[19] S. P. Awate and R. T. Whitaker, "Feature-preserving MRI denoising: a nonparametric empirical bayes approach," *IEEE Transactions on Medical Imaging*, vol. 26, no. 9, pp. 1242–1255, 2007.

[20] ——, "Unsupervised, information-theoretic, adaptive image filtering for image restoration," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 364–376, 2006.

[21] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on image processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[22] D. Marin, Y. Zhong, M. Drangova, and Y. Boykov, "Thin structure estimation with curvature regularization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 397–405.

[23] J. Hannink, R. Duits, and E. Bekkers, "Crossing-preserving multi-scale vesselness," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2014, pp. 603–610.

[24] W. Fu, K. Breininger, R. Schaffert, N. Ravikumar, T. Würfl, J. Fujimoto, E. Moult, and A. Maier, "Frangi-net," in *Bildverarbeitung für die Medizin 2018*. Springer, 2018, pp. 341–346.

[25] T. Jerman, F. Pernuš, B. Likar, and Ž. Špiclin, "Beyond frangi: an improved multiscale vesselness filter," in *Medical Imaging 2015: Image Processing*, vol. 9413. International Society for Optics and Photonics, 2015, p. 94132A.

[26] S. Mukherjee and S. T. Acton, "Oriented filters for vessel contrast enhancement with local directional evidence," in *Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*. IEEE, 2015, pp. 503–506.

[27] K. Krissian, G. Malandain, N. Ayache, R. Vaillant, and Y. Trousset, "Model-based detection of tubular structures in 3d images," *Computer vision and image understanding*, vol. 80, no. 2, pp. 130–171, 2000.

[28] M. Sofka and C. V. Stewart, "Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures," *IEEE transactions on medical imaging*, vol. 25, no. 12, pp. 1531–1546, 2006.

[29] W. Ross, B. Salzberg, L. Cohen, A. Grinvald, H. Davila, A. Waggoner, and C. H. Wang, "Changes in absorption, fluorescence, dichroism, and birefringence in stained giant axons: optical measurement of membrane potential," *The Journal of membrane biology*, vol. 33, no. 1, pp. 141–183, 1977.

[30] H. Liang, S. T. Acton, and D. S. Weller, "Content-aware neuron image enhancement," in *2017 IEEE International Conference on Image Processing (ICIP)*, Sept 2017, pp. 3510–3514.

[31] S. Mukherjee, B. Condron, and S. T. Acton, "Tubularity flow field—a technique for automatic neuron segmentation," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 374–389, 2015.

[32] C. Grienberger and A. Konnerth, "Imaging calcium in neurons," *Neuron*, vol. 73, no. 5, pp. 862–885, 2012.

[33] W. C. Lemon, S. R. Pulver, B. Höckendorf, K. McDole, K. Branson, J. Freeman, and P. J. Keller, "Whole-central nervous system functional imaging in larval drosophila," *Nature communications*, vol. 6, p. 7924, 2015.

[34] M. Tada, A. Takeuchi, M. Hashizume, K. Kitamura, and M. Kano, "A highly sensitive fluorescent indicator dye for calcium imaging of neural activity in vitro and in vivo," *European Journal of Neuroscience*, vol. 39, no. 11, pp. 1720–1728, 2014.

[35] E. Pnevmatikakis, D. Soudry, Y. Gao, T. A. Machado, J. Merel, D. Pfau, T. Reardon, Y. Mu, C. Lacefield, W. Yang, M. Ahrens, R. Bruno, T. M. Jessell, D. Peterka, R. Yuste, and L. Paninski, "Simultaneous denoising, deconvolution, and demixing of calcium imaging data," *Neuron*, vol. 89, no. 2, pp. 285 – 299, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0896627315010843

[36] K. Ohki, S. Chung, Y. H. Ch'ng, P. Kara, and R. C. Reid, "Functional imaging with cellular resolution reveals precise micro-architecture in visual cortex," *Nature*, vol. 433, no. 7026, p. 597, 2005.

[37] R. Portugues, C. E. Feierstein, F. Engert, and M. B. Orger, "Whole-brain activity maps reveal stereotyped, distributed networks for visuomotor behavior," *Neuron*, vol. 81, no. 6, pp. 1328–1343, 2014.

[38] S. A. Romano, T. Pietri, V. Pérez-Schuster, A. Jouary, M. Haudrechy, and G. Sumbre, "Spontaneous neuronal network dynamics reveal circuit's functional adaptations for behavior," *Neuron*, vol. 85, no. 5, pp. 1070–1085, 2015.

[39] R. Maruyama, K. Maeda, H. Moroda, I. Kato, M. Inoue, H. Miyakawa, and T. Aonishi, "Detecting cells using non-negative matrix factorization on calcium imaging data," *Neural Networks*, vol. 55, pp. 11–19, 2014.

[40] F. D. Andilla and F. A. Hamprecht, "Sparse space-time deconvolution for calcium image analysis," in *Advances in Neural Information Processing Systems*, 2014, pp. 64–72.

[41] C. Zhang, H. Wang, Z. You, and B. Ma, "Sensitivity analysis of longitudinal cracking on asphalt pavement using mepdg in permafrost region," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 2, no. 1, pp. 40–47, 2015.

[42] Z. Shi, *Crack analysis in structural concrete: theory and applications*. Butterworth-Heinemann, 2009.

[43] H. Oliveira and P. L. Correia, "Crackit—an image processing toolbox for crack detection and characterization," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 798–802.

[44] A. Mohan and S. Poobal, "Crack detection using image processing: A critical review and analysis," *Alexandria Engineering Journal*, 2017.

[45] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361–378, 2017.

[46] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3708–3712.

[47] L. Condat, "A direct algorithm for 1-d total variation denoising," *IEEE Signal Processing Letters*, vol. 20, no. 11, pp. 1054–1057, 2013.

[48] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via $l0$ gradient minimization," in *ACM Transactions on Graphics (TOG)*, vol. 30, no. 6. ACM, 2011, p. 174.

[49] T. Goldstein and S. Osher, "The split bregman method for $l1$-regularized problems," *SIAM journal on imaging sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[50] X. Zhu and P. Milanfar, "Automatic parameter selection for denoising algorithms using a no-reference measure of image content," *IEEE Transactions on image processing*, vol. 19, no. 12, pp. 3116–3132, 2010.

[51] M. V. Afonso, J. M. Bioucas-Dias, and M. A. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE transactions on image processing*, vol. 19, no. 9, pp. 2345–2356, 2010.

[52] BigNeuron-Project. (Nov 2, 2016) Gold166 benchtesting neuron reconstructions. [Online]. Available: http://bigneuron.org

[53] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[54] H. Peng, J. Tang, H. Xiao, A. Bria, J. Zhou, V. Butler, Z. Zhou, P. T. Gonzalez-Bellido, S. W. Oh, J. Chen *et al.*, "Virtual finger boosts three-dimensional imaging and microsurgery as well as terabyte volume image visualization and analysis," *Nature communications*, vol. 5, p. 4342, 2014.

[55] Q. Spaen, D. S. Hochbaum, and R. Asín-Achá, "Hnccorr: A novel combinatorial approach for cell identification in calcium-imaging movies," *arXiv:1703.01999*, 2017.

[56] M. Pachitariu, C. Stringer, S. Schröder, M. Dipoppa, L. F. Rossi, M. Carandini, and K. D. Harris, "Suite2p: beyond 10,000 neurons with standard two-photon microscopy," *bioRxiv*, 2016. [Online]. Available: https://www.biorxiv.org/content/early/2016/06/30/061507

[57] Code-Neuro-Finder. 00.00. [Online]. Available: http://neurofinder.codeneuro.org/