Defect Detection in SEM Images of Nanofibrous Materials

Diego Carrera*, Fabio Manganini, Giacomo Boracchi, and Ettore Lanzarone

Abstract—**Nanoproducts represent a potential growing sector and nanofibrous materials are widely requested in industrial, medical and environmental applications. Unfortunately, the production processes at the nanoscale are difficult to control, and nanoproducts often exhibit localized defects that prevent their functional properties. Therefore, defect detection is a particularly important feature in smart-manufacturing systems, which aims at reducing quality-inspection times and production wastes. Here we present a novel solution to detect defects in nanofibrous materials by analyzing SEM images. We employ an algorithm that learns, during a training phase, a model yielding sparse representations of the structures that characterize correctly produced nanofiborus materials. Defects are then detected by analyzing each patch of an input image and extracting features that quantitatively assess whether the patch conforms or not to the learned model. The proposed solution has been successfully validated over 45 images acquired from samples produced by a prototype electrospinning machine, a promising technology for producing nanofibrous materials. The low computational times indicate that the proposed solution can be effectively adopted in a monitoring system for industrial production.**

Index Terms—**Defect and Anomaly Detection, Nanofibrous Materials, Quality Control, Sparse Representations, Smart Manufacturing, SEM Images**

I. Introduction

Nanoproducts demand has steadily increased over the past few years [1]. In particular, nanofibrous materials are nowadays widely requested [2], e.g., in life sciences and medicine [3], [4], filtration and water treatment [5], [6], [7], surface coating [8], [9], and sensors [10].

However, despite the large number of production processes and the flourishing of laboratory prototypes [11], [12], nanofibrous materials are not yet industrially produced at the large scale. They are indeed recognized as one of the main challenges in high-tech manufacturing [13], [14], e.g., in the Horizon 2020 Program *Factory of the Future*. In fact, production processes at the nanoscale are still difficult to control and, as a result, the nanostructure characterizing these materials often exhibits local defects, which may impair their mechanical or filtering properties. Defects also make the production long and costly, resulting in an increased proportion of wasted materials and long time spent in thorough quality-inspection procedures.

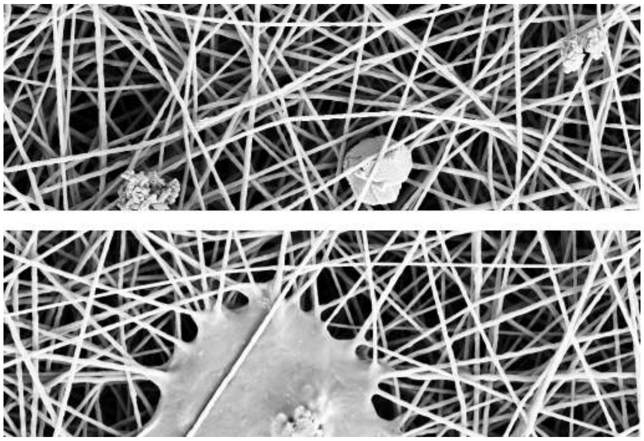


Fig. 1. Two details from SEM images depicting nanofibrous material produced by electrospinning. The large area covered by filaments in both images refer to good quality fibers. The top image contains two sorts of localized defects: a small *speck of dust* at the center and two *beads*, namely fiber clots. The bottom image contains a *films*, which is a thin layer of material among the fibers.

Hence, automated systems for monitoring the quality of nanostructures are of crucial importance for the development of nanoproducts industry. On the one hand, these systems allow to raise alerts as soon as the amount/size of defects exceeds a given tolerance level, in order to take corrective actions (including halting the machinery) and guarantee a satisfactory production quality. On the other hand, automatic defect-detection solutions provide a quantitative assessment which can be used to accurately design/tune the production process to both optimize physical properties and control the defectiveness of the produced materials [15].

As we will discuss in Section II-B, defects in nanofibrous materials can be conveniently detected by analyzing Scanning Electron Microscopes (SEM) images, since these instruments have a resolution up to 1 nanometer. However, despite the relevance of this industrial problem, to the best of our knowledge, there are no automatic systems able to provide a comprehensive monitoring of nanofiber production. In particular, there are no systems able to detect and quantitatively assess localized defects, like *beads* and *films* shown in Figure 1. While humans can easily identify these defects, thus distinguishing between normal and anomalous/defective regions, this is not a simple task for a machine. In fact, both normal and anomalous regions are far from being regular: fibers follow different orientations and randomly overlap, and

This work has been partially supported by Italian CNR and MIUR organizations through the Flagship Projects Factory of the Future "NanoTWICE" and "Autospin". *Asterisk indicates corresponding author.*

*D. Carrera and G. Boracchi are with the Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, Milan, Italy (e-mail: diego.carrera — giacomo.boracchi @polimi.it).

F. Manganini and E. Lanzarone are with the Istituto di Matematica Applicata e Tecnologie Informatiche (IMATI), Consiglio Nazionale delle Ricerche (CNR), Milan, Italy (e-mail: fabio.manganini@gmail.com; ettore.lanzarone@cnr.it).

defects can be very different in appearance and shape. Existing vision-based systems for monitoring nanofiber production only measure fiber diameter [16], [17] or orientation [18], while not localized defects like those in Figure 1, which is indeed our goal.

Our contribution is the first solution able to automatically detect and quantitatively assess localized defects in SEM images of nanofibrous materials. We successfully solve this problem by adopting an anomaly-detection algorithm which can be easily executed on a machinery that embeds a SEM microscope and all the accessories to prepare samples for spot checks. We test our solution on SEM images of nanofibrous materials produced by a prototype electrospinning machine and show that *i*) it achieves excellent detection performance, and *ii*) it requires a negligible processing time with respect to the sample-preparation time. The proposed solution can thus be employed in a comprehensive monitoring system that performs spot checks on the produced material and generates alarms and feedbacks to correct the production process.

We locate defects as regions covered by anomalous patches, i.e., small portions of the image that do not conform the structure of normal images. To this end, we use the algorithm in [19], which is able to handle patches that (given their large dimensionality and complex structure) cannot be straightforwardly analyzed by traditional multivariate statistical methods [20], e.g., maximum likelihood approaches. Patch dimensionality/complexity are in fact reduced by extracting features, namely meaningful indicators with a known response from normal data. However, manually designing effective features is difficult, since patches extracted from the considered SEM images are far from being regular. Hence, we adopt [19] to extract data-driven features that measure the conformance of test patches with respect to a learned dictionary, which provides sparse representations [21] of normal patches. As typical in the anomaly-detection context, anomalies are then detected as patches corresponding to outliers with respect to the features distribution [22]. Another reason for using [19] is that this algorithm can be extended to operate at different magnification levels [23].

The algorithm in [19] was introduced for detecting anomalies in image textures, and we here introduce specific pre- and post-processing stages to improve the defect-detection performance on SEM images of nanofibrous materials. Moreover, we perform an experimental campaign that is substantially larger than that in [19], as we also include comparisons against state-of-the-art anomaly-detection methods [24], [25], and quantitative performance assessment over a dataset of 45 SEM images. In contrast, quantitative tests in [19] were performed on a dataset of texture images [26] and only few SEM images were considered in a qualitative assessment.

We made our dataset (consisting of 45 SEM images) and the defect annotations publicly available for download[1]. These images were acquired from a prototype electrospinning machine, developed by the *Istituto per lo Studio delle Macromolecole* of the National Research Council of Italy within the Project NanoTWICE (see Acknowledgments), which aims at equipping the already existing and fully-functioning prototype with all the facilities necessary for industrialization.

The paper is structured as follows. Nanofibrous materials and electrospinning are presented in Section II, together with the methods for monitoring nanofiber production and an overview of anomaly-detection algorithms for images. The addressed anomaly-detection problem is formulated in Section III, and the proposed solution is illustrated in Section IV. Experiments are detailed in Section V, while conclusions and future research directions are presented in Section VI.

II. Background and Related Works

Nanofibrous materials are porous materials made of polymer nanofibers, i.e., fibers whose diameter is smaller than 100 nanometers [27], which are embedded together in an amorphous structure. Thanks to the possibility of tuning the material properties by modifying nanostructures characteristics, nanofibrous materials are nowadays applied in several fields, and have generated a lot of innovative applications in the last few years [2], [28]. Most of applications concern medicine, e.g., anti-adhesion materials in surgery, biofunctionalized materials for biomedical applications, and scaffolds for tissue engineering [29], [3], [4]. Other important applications concern the construction of filtration membranes [5], [6], [7], where nanofibrous materials are inserted in a media to provide superior filtration capabilities. Finally, nanofibrous materials are also employed in surface coating [8], [9] and sensors [10].

A. Nanofibrous Materials Production and Electrospinning

Several techniques for producing nanofibers have been presented in the literature, e.g., self-assembly, templating, lithography and electrospinning [2], [30]. This latter is perhaps the most promising and versatile one, since it directly produces nanofibrous materials whose nanostructure is mainly in the form of non-woven nanofibers. Moreover, nanofibrous materials produced through electrospinning typically feature higher surface-to-volume ratio and a porosity that can be better controlled than in materials produced by other techniques. Electrospinning also allows large flexibility in the size and shape of the produced nanofibers [2], [28]. Electrospinning was originally studied by Rayleigh in 1897, and patented in 1934 [31]; however, it has only recently gained momentum thanks to the explosive growth of nanostructured materials [2].

Briefly, the electrospinning process takes place between a spinning head and a plate. The spinning head has a capillary opening where a high voltage is applied, and it is connected with a reservoir of polymer solution under pressure. On the other side, the static plate (usually grounded) acts as a counter-electrode that collects nanofibers. The high voltage difference between the spinning head and the ground results in a nano- to micron-sized electrically-driven polymer solution jet, which is drawn out from the apex of a cone (the so-called *Taylor cone*) formed at the capillary opening of the spinning head. The solvent rapidly evaporates from the jet during the run and, under optimal conditions, a continuous nano-sized filament is deposited to the collecting electrode in

[1] http://web.mi.imati.cnr.it/ettore/NanoTwice

a random fashion, forming the non-woven structure. Further details about electrospinning process can be found in [2], [28].

Unfortunately, external environmental variables and process instabilities (e.g., instabilities in the solution jet [32], [33]) may seriously affect the production, introducing high variability in the fiber characteristics and localized defects like those in Figure 1. In particular, beads are generated by drops coming off from the spinning head, which are deposited on the plate together with the fibers. Similarly, films are caused by larger drops that, once deposited on the plate, widen and thin; then, once the solvent evaporates, a film thinner than the fibers remains. Finally, tiny speck of dust might also get trapped in the fibers, and this also prevents the functional properties of the nanofibrous materials. Another issue is the presence of big holes in the material, i.e., parts of the sample where no fibers are present. Obviously, a *physiological* porosity is expected and desired; however, problems arise when pores are larger than a reference value. Similarly, problems may arise when fibers characteristics (e.g., the average fiber diameter and their distribution) depart from their reference value.

B. Monitoring Nanofibrous Materials Production

Several solutions to monitor nanofibrous material production, including electrospinning, have been presented in the literature. They can be roughly divided into solutions that continuously control production parameters during the process (e.g., the current or the pressure in the equipment) and solutions that control the produced materials through spot checks. Monitoring production parameters is simpler, as it consists of analyzing signals acquired from the machine; however, the quality of the produced materials can be affected by several other stochastic factors and analyzing these signals might not be sufficient to detect defects in the produced materials. Thus, it is preferable to inspect samples of the produced material, and two directions are possible: *i*) to directly analyze the structure of the nanofibrous material by SEM imaging, or *ii*) to assess some functional properties (e.g., nano-mechanical properties through atomic force microscopy [34] or filtering behavior in case of filters [6], [7]). The latter option does not allow to generate alerts in a short time, thus is less suited for regular spot checks. As such, the most effective approach consists in acquiring SEM images of few material samples and directly analyze their nanostructures.

Of course, visual inspection by human operators is not a viable option, as large-scale industrial scenarios require automatic and repeatable solutions that quantitatively assess defects. So far, the only existing automatic solutions are meant to measure the fibers diameter and orientation [16], [17], [18], while they are not able to detect defects like those shown in Figure 1.

C. Anomaly Detection in Images

Algorithms for detecting anomalies [35] in images can be divided in reference-based and reference-free ones. In reference-based methods (e.g., [36]), anomalies are detected by comparing the test image against a reference one that does not contain anomalies and can be used as a template. These methods apply in many industrial scenarios, e.g., in semiconductor production where wafers correspond, up to some small misalignment, to a reference template [36].

Reference-free algorithms do not use any template image to compare with, and they are the only viable option in scenarios like the one considered here, where normal images depict filaments that follow pseudo-random rather than geometrical patterns. Reference-free algorithms detect anomalies by either computing *i*) features that are able to discriminate between normal and anomalous regions, or *ii*) features that provide a known response to normal regions. In the latter case, any region yielding an unusual response is considered anomalous. Methods implementing this latter strategy are typically referred as *novelty detection* [22], [37], [38] (in the machine-learning community this problem is also known as one-class classification [39]). In this paper we present a novelty-detection algorithm, even though we refer to anomaly/defect since these terms are more appropriate descriptions for the considered application.

On the one hand, anomaly-detection methods are easy to use, since they simply require a training set of normal images (which is often easy to collect) and are in principle able to detect as anomalous any pattern that does not conform training images. On the other hand, this problem is more challenging because it does not rely on any information about the anomalies to be detected. Overviews of anomaly-detection algorithms for images can be found in [35], [22]; not surprisingly, these algorithms have been also proposed for industrial monitoring purposes [40], [41], [42].

Here, we adopt an anomaly-detection algorithm based on a dictionary yielding sparse representations, which are nowadays one of the leading models in image and signal processing applications [21], [43]. In particular, we pursue the approach in [19] to represent normal data, in which a dictionary is learned during an initial training phase. Then, test images are analyzed in a patch-wise manner, computing features which assess the conformance of each patch with the structures characterizing normal ones. Anomalies are then identified as outliers in the feature distribution. This approach proved to be particularly successful on image texture. A different anomaly-detection algorithm that uses sparse representations is [24], where the anomalous data are identified during the sparse-coding stage, by means of an *ad hoc* procedure. Convolutional-sparse models [44] were also shown to be effective in detecting anomalies, even though they are more computationally demanding than traditional patch-based models like those described in Section IV-A.

While several defect-detection algorithms used in industrial applications (e.g., those in [45], [46], [47], [48]) are very application-specific and cannot be straightforwardly applied to nanofibrous materials, anomaly-detection algorithms based on sparse representation are rather general. In fact, these have been customized for different monitoring scenarios, like analyzing MEMS [49] signals in environmental monitoring applications, or ECG tracings to detect arrhythmias in wearable devices [50].

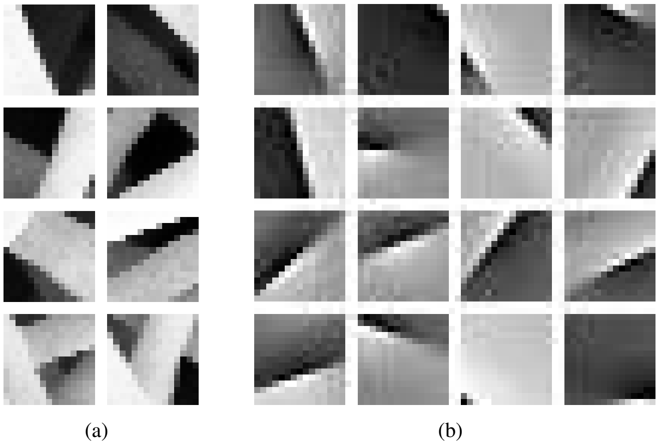


Fig. 2. (a) Examples of training patches. (b) Atoms of the dictionary learned by the ADMM algorithm [51]. Here we show only 16 atoms.

III. Problem formulation

Let us denote by $s : \mathcal{X} \to \mathbb{R}^+$ the SEM image depicting the nanostructures to analyze for anomaly-detection purposes, where $\mathcal{X} \subset \mathbb{Z}^2$ is the regular pixel grid corresponding to the image domain. The image intensity[2] at pixel $c \in \mathcal{X}$ is denoted by $s(c)$. Our goal is to locate anomalous regions in s; as such, the problem can be formulated as estimating the unknown anomaly mask

$$\Omega(c) = \begin{cases} 0 & \text{if } c \text{ falls inside a normal region} \\ 1 & \text{if } c \text{ falls inside an anomalous region} \end{cases} \quad (1)$$

which has to report as many anomalies as possible, independently of their dimension and shape. In particular, we are interested in estimating an anomaly mask $\widetilde{\Omega}$ that *i*) covers most of the anomalous regions in s and that *ii*) reports the largest number of anomalies, including the smallest ones.

Our only assumption is that, for training purposes, a set of normal (i.e., anomaly-free) images T is given, while no training images containing anomalous regions are provided. This is a reasonable assumption since anomalous regions might be very different in shape, dimension and appearance, and a training set might not encompass all possible anomalies that could occur during operations. In this sense, anomalies remain unknown and we detect as anomalous any region that do not conform the structure of normal data.

IV. Proposed Solution

To better locate anomalies, we process the input image s in a patch-wise manner. A patch is a small image region having a given shape and extracted at a specific pixel, denoted as patch center. In particular, the patch centered in $c \in \mathcal{X}$ is defined as

$$\mathbf{s}_c = \{s(c+u), u \in \mathcal{U}\}, \ \forall c \in \mathcal{X} \quad (2)$$

where $\mathcal{U}$ is a neighborhood of the origin which defines the patch shape. While in principle patches $\mathbf{s}_c$ can be defined over arbitrary shapes, in practice, we chose $\mathcal{U}$ as a square neighborhood of $\sqrt{p} \times \sqrt{p}$ pixels, where p is the cardinality of $\mathcal{U}$. Note that $\mathbf{s}_c$ will be usually considered as a column vector in $\mathbb{R}^p$, and thorough the paper we use bold letters to indicate vectors. Figure 2(a) shows examples of patches extracted from normal images.

A. Normal Patches Model

Our modeling assumption is that patches in normal images are drawn from a stationary, stochastic process $\mathcal{P}_N$ and can be ideally contained in an unknown union of low dimensional subspaces of $\mathbb{R}^p$ [52]. Such union of subspaces corresponds to our model which has to be learned from the training set T. This model can be expressed in term of a *dictionary* $D \in \mathbb{R}^{p \times n}$ [53] that provides sparse approximation of all normal patches $\mathbf{s}_c \in \mathbb{R}^p$, i.e.,

$$\mathbf{s}_c \approx D\mathbf{x}_c \,. \quad (3)$$

In (3), $\mathbf{x}_c \in \mathbb{R}^n$ denotes the coefficient vector which is *sparse*, namely only few coefficients are nonzero, thus $\|\mathbf{x}_c\|_0$, which is the number of nonzero components of $\mathbf{x}_c$, is small. Note that (3) corresponds to the following expression

$$\mathbf{s}_c \approx \sum_{i=1}^{n} x_{c,i} \mathbf{d}_i \,, \quad (4)$$

where $x_{c,i}$ denotes the i-th component of $\mathbf{x}_c$ and $\mathbf{d}_i$ the i-th column of D, which is commonly referred to as a *dictionary atom.* Sparsity implies that only few $x_{c,i}$ in (4) are nonzero and that each normal patch can be (well) approximated by a linear combination of few dictionary atoms. The dictionary D is typically overcomplete, i.e., the number of atoms exceeds space dimension ($n > p$): overcompleteness allows more flexibility in the definition of atoms than in basis expansions, and this flexibility enables sparsity in (4).

The coefficients of the sparse representation $\mathbf{x}_c$ are computed by solving the *sparse coding* problem which is formulated as

$$\mathbf{x}_c = \underset{\widetilde{\mathbf{x}} \in \mathbb{R}^n}{\arg\min} \frac{1}{2} \|D\widetilde{\mathbf{x}} - \mathbf{s}_c\|_2^2 + \lambda \|\widetilde{\mathbf{x}}\|_1, \quad (5)$$

where the ℓ^1 regularization term $\|\widetilde{\mathbf{x}}\|_1$ promotes sparsity in the solution [54]. According to (4) sparsity should be promoted by the ℓ^0 regularization term $\|\widetilde{\mathbf{x}}\|_0$ instead of $\|\widetilde{\mathbf{x}}\|_1$. However, while the ℓ^0 regularization makes the sparse coding problem NP-Hard, the formulation (5) is convex and can be solved by standard convex optimization algorithms. The problem (5) is theoretically grounded, since it can be shown that under proper assumptions concerning the dictionary D and the number of nonzero coefficients of $\mathbf{x}_c$, the sparse coding problems involving the ℓ^0 and ℓ^1 regularizations have the same solution [55]. The solution of (5) corresponds to the maximum a posteriori (MAP) estimate of $\mathbf{x}_c$ given the patch $\mathbf{s}_c$, when the nonzero coefficients follow a Laplace distribution. This result can be extended to other ℓ^p regularizations, which corresponds to assuming that the nonzero coefficients of $\mathbf{x}_c$ follow a Gibbs distribution [56].

The *dictionary learning* problem actually corresponds to learning both the dictionary $D \in \mathbb{R}^{p \times n}$ and the sparse

[2] SEM produces grayscale images, since each pixel reports the number of electrons received by the detector. However, the proposed algorithm can be easily extended to color images by jointly analyzing different color bands.

representations $X \in \mathbb{R}^{n\times m}$ for a given training set $S \in \mathbb{R}^{p\times m}$, containing m normal patches arranged as the columns of S. Dictionary learning is then formulated as solving

$$[D, X] = \underset{\widetilde{D}\in\mathbb{R}^{p\times n},\, \widetilde{X}\in\mathbb{R}^{n\times m}}{\arg\min} \frac{1}{2}\|\widetilde{D}\widetilde{X} - S\|_2^2 + \lambda\|\widetilde{X}\|_1, \quad (6)$$

where the ℓ^1 regularization is applied to each column of X. The problem (6) is typically solved by using the Alternating Direction Method of Multipliers (ADMM) [57] which alternates the calculation of the dictionary atoms and the sparse representations X of training patches S with respect to the current dictionary.

In our experiments we learn the dictionary D solving (6) by means of the algorithm in [51] based on the ADMM and solve the sparse coding (5) using the LARS algorithm [58]. Figure 2(b) reports a few atoms of a dictionary learned from a normal image, and shows that these atoms actually depict the peculiar structures of the filaments that characterize normal patches. It seems quite likely that a linear combination of few of these atoms can reliably approximate the normal patches reported in Figure 2(a).

B. Detection of Anomalous Patches

We assume that anomalous patches come from a process $\mathcal{P}_A$ yielding structures that are different from those generated by $\mathcal{P}_N$. Unfortunately, it is not often possible to learn a dictionary that approximates anomalous patches, since these are rarely provided in a sufficient amount for training. Thus, we detect anomalies by determining whether test patches fall inside/outside of the union of low-dimensional subspaces that well approximate normal patches.

In practice, each patch in a test image is independently analyzed to determine whether it admits a sparse representation with respect to D. To this purpose, we compute a feature consisting in a *bivariate indicator* which jointly accounts for the reconstruction error and the sparsity of the representation (3). In particular, given a patch $\mathbf{s}_c$, we compute $\mathbf{x}_c$ (sparse coding) by solving the BPDN problem (5), and we extract as feature in c the following indicator vector:

$$\mathbf{f}(c) = \begin{bmatrix} \|D\mathbf{x}_c - \mathbf{s}_c\|_2 \\ \|\mathbf{x}_c\|_1 \end{bmatrix}. \quad (7)$$

Indicators extracted from normal patches follow a stationary, albeit unknown, distribution. Anomalous patches are expected to substantially deviate from normal ones in either their sparsity or reconstruction error (or possibly both). Thus, the corresponding indicators would be outliers with respect to the distribution ϕ_0 of indicators extracted from normal patches. While in [19] outliers in the indicators are detected by means of a confidence region built according to the multivariate Chebyshev inequality, we here model the distribution ϕ_0 by Kernel Density Estimation (KDE), adopting a kernel based on linear diffusion with automatic bandwidth selection [59].

Then, a patch $\mathbf{s}_c$ is considered anomalous when $\mathbf{f}(c)$ falls in a low-density region of ϕ_0. Therefore, an initial estimate of the anomaly mask is given by

$$\widetilde{\Omega}(c) = \begin{cases} 0 & \text{if } \phi_0\left(\mathbf{f}\left(c\right)\right) < \gamma \\ 1 & \text{if } \phi_0\left(\mathbf{f}\left(c\right)\right) \geq \gamma \end{cases}, \quad (8)$$

being $\gamma > 0$ a parameter that tunes the responsiveness of the anomaly detector. In particular, the value of γ can be empirically chosen to provide an acceptable false positive rate.

C. Preprocessing

To effectively capture the structure that characterizes normal filaments, we consider quite small patches; thus, there might be patches that do not overlap with any filament and are completely dark. Patches that are entirely zero can be perfectly reconstructed by any linear model, and achieve a (very) sparse representation, having all coefficients in (3) equal to zero. Unfortunately, null indicator vectors can impair the estimation of ϕ_0, and it is safer to remove them from both the training and test patches. Thus, we consider for training only patches in the set S:

$$S = \{\mathbf{s}_c \,|\, \mathrm{median}(\mathbf{s}_c) > \varepsilon\}, \quad (9)$$

where $\varepsilon > 0$ is a manually tuned parameter. The same operation is applied on test images. The median in (9) was used to remove also dark patches that marginally overlap with a filament. It is worth mentioning that nanofibrous materials having too large holes might yield porosity values that are far from the reference ones. However, this sort of anomalies can be detected by straightforward morphological operations on the whole image and certainly do not require any learning method.

Another pre-processing operation to perform before dictionary learning and sparse coding is to subtract the average value from each patch $\mathbf{s}_c$, which is quite a customary operation in the sparse-representation literature.

D. Postprocessing

Even though the anomaly detector (8) takes as input the whole patch $\mathbf{s}_c$, its final decision concerns only the patch center c and no other pixels belonging to the same patch. However, since patches centered in neighboring pixels largely overlap, it would be better to aggregate the decisions of the anomaly detector in all those patches that overlap with c. We perform such aggregation by post-processing the anomaly mask (8) by majority voting:

$$\widehat{\Omega}(c) = \begin{cases} 0 & \text{if } \#\mathcal{A}_c < \#\mathcal{N}_c \\ 1 & \text{if } \#\mathcal{A}_c \geq \#\mathcal{N}_c \end{cases}, \quad (10)$$

where $\mathcal{A}_c = \{u \in \mathcal{U} \,|\, \widetilde{\Omega}(c+u) = 1\}$ denotes the set of pixels in $\mathbf{s}_c$ that are considered anomalous and $\mathcal{N}_c = \{u \in \mathcal{U} \,|\, \widetilde{\Omega}(c+u) = 0\}$ the set of pixels that are considered normal.

Finally, to smooth the borders of anomalous regions in $\widehat{\Omega}$ we perform an additional post-processing by customary morphological operators [60]. More precisely, we apply an erosion followed by a dilation, which are nonlinear filters based on order statistics: the minimum and the maximum over a given support, respectively. We experienced that adopting these binary operations over a neighborhood smaller than $\mathcal{U}$ can improve the coverage of anomalous regions.

Algorithm 1 The training phase of the proposed algorithm.

```
Require: a training set of normal images T.
 1: Prepare two training sets of normal patches $S$ and $V$.
 2: Remove dark patches from $S$ and $V$ as in (9).
 3: Subtract the average value from each patch in $S$ and $V$.
 4: Learn the dictionary $D$ solving (6) from $S$.
 5: Compute the sparse representation of $V$ solving (5).
 6: Compute $\{\mathbf{f}(c), \forall \mathbf{s}_c \in V\}$ as in (7), the indicators of normal patches.
 7: Fit $\phi_0$ through KDE, and define a suitable the threshold $\gamma > 0$ to be used in the detector (8).
```

E. Algorithm Summary

We here describe the training phase (Algorithm 1) and detail all of the steps of the proposed solution during operations (Algorithm 2). To prevent overfitting, it is convenient to split T in two sets of patches, S and V, which are both preprocessed as in Section IV-C (Algorithm 1, lines 1-3). We use patches in S to learn the dictionary D by solving (6) via the ADMM algorithm [51] (line 4), while we compute the sparse representations (line 5) and the indicators (line 6) for all the patches in V. Then, we fit the distribution ϕ_0 only to these latter (line 7), ignoring patches used to learn D. Finally, a suitable threshold $\gamma > 0$ is chosen.

During operations, each test image is processed in a patch-wise manner and each patch $\mathbf{s}_c$ undergoes the preprocessing steps described Section IV-C (Algorithm 2, line 2-3). The sparse representation $\mathbf{x}_c$ is computed by solving the sparse-coding problem (line 4) and the corresponding indicator vector $\mathbf{f}(c)$ is obtained (line 5). A simple thresholding on $\phi(\mathbf{f}(c))$ as in (8) provides a preliminary estimate of the anomaly mask, i.e., $\widetilde{\Omega}$ (line 6). The anomaly mask is then refined through the post-processing to obtain $\widehat{\Omega}$ (line 9-12).

V. Experiments

We analyze SEM images acquired from samples produced by the prototype electrospinning machine developed for the NanoTWICE Project. Electrospinning sessions were performed with experimental conditions (machine parameters and environmental variables) that typically produce good quality nanofibrous materials with no macroscopic defects. There is no point of considering worse production conditions, as these can be easily detected by other macroscopic approaches (e.g., they yield altered current patterns during the process) and do not require vision-based monitoring.

A. Dataset Description

Our SEM images were acquired with the FE-SEM (Carl Zeiss Sigma NTS, Gmbh Öberkochen, Germany). A sample of 4×4 cm from the produced material is placed on a metallic support, and a thin gold coating of 5 nm is applied on the sample surface to guarantee satisfactory electrical conduction. All images are acquired in the same conditions and using the same parameters, i.e., magnification of 8000x, extra high tension of 5 kV, working distance of 7 mm, brightness of 45%, and contrast of 52%.

Algorithm 2 The operational phase of the proposed algorithm.

```
Require: test image $s$; $D$, $\phi_0$, $\gamma > 0$ from Algorithm 1
 1: for all patch $\mathbf{s}_c$ in $s$ do
 2:     if $\mathbf{s}_c$ satisfies (9) then
 3:         Subtract form $\mathbf{s}_c$ its average value.
 4:         Solve (5) to compute $\mathbf{x}_c$.
 5:         Compute the indicator $\mathbf{f}(c)$ as in (7).
 6:         Define the anomaly mask value $\widetilde{\Omega}(c)$ as in (8).
 7:     end if
 8: end for
 9: for all pixel $c$ of the anomaly mask $\widetilde{\Omega}$ do
10:     Set the anomaly mask $\widehat{\Omega}$ via majority voting (10).
11: end for
12: Apply erosion and dilatation operators to $\widehat{\Omega}$.
```

Our dataset contains 45 SEM images (dimension 1024×696 pixels): 5 images are anomaly-free, while 40 images contain anomalies of different size. For each image, we manually select all defects, defining the anomaly mask Ω that is used as a ground truth in our tests. Overall defects in these images are very small: on average they cover 1.3% of the image, and only the 0.5% of the anomalies exceed the 2% of the image size.

B. Figures of Merit

To assess the performance of the proposed solution we compute the following figures of merit: *False Positive Rate* (FPR), namely the percentage of pixels which are erroneously identified as anomalous; *True Positive Rate* (TPR), namely the percentage of pixels which are correctly identified as anomalous.

Since both FPR and TPR depend on the threshold γ, which sets the responsiveness of Algorithm 2 and of the alternative solutions described below, we analyze the Receiver Operating Characteristic (ROC) curve, which is obtained by plotting the TPR against the FPR for different values of γ. Moreover, the area under the ROC curve (AUC) is used as a single quantitative performance indicator, which is equal to 1 in the case of the perfect detector (i.e., TPR = 100% and FPR = 0%).

However, although the AUC is commonly adopted in detection problems, in the considered scenario it is mainly influenced by large defects, while our goal is to detect all of them disregarding their size. Therefore, to quantitatively assess the coverage of all defects, we extract the connected components [60] of the ground truth Ω, thus assigning a blob to each defect. Then, we measure the *Defect Coverage* as the percentage of pixels covered by the output $\widehat{\Omega}$ of a detector yielding FPR = 5%. Of course, each defect yields one *Defect Coverage* value, and different solutions have to be contrasted by comparing the distribution of *Defect Coverage* values.

C. Alternative Solutions

We compare our algorithm against five anomaly-detection solutions that, like ours, operate patch-wise without any anomalous patch for training. All solutions have been tested in

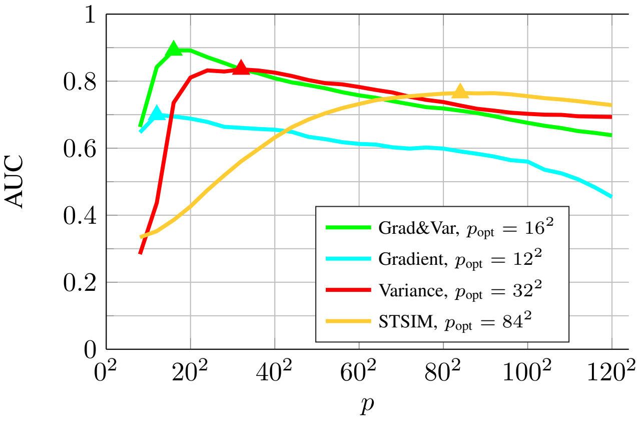


Fig. 3. Values of the AUC for the considered methods obtained by varying the patch size. The solid triangles indicate the points where the AUC is maximized and in the legend the optimal patch size p_{opt} is reported.

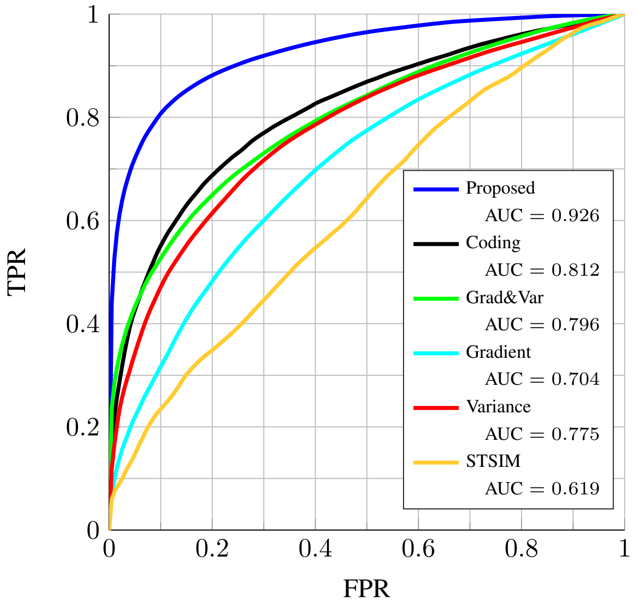


Fig. 4. ROC curves for all solutions considered in Section V, with the corresponding AUC values reported in the legend. *Proposed* and *Coding* solutions use a patch size $p = 15^2$, while the others the optimal patch size p_{opt} selected in Figure 3. The *Proposed* solution outperforms by far all of the others.

the same setting: they are trained from a set T of 5 anomaly-free images, they preliminary remove dark patches as in (9), and they perform the same postprocessing described in Section IV-D.

The first three (*Variance*, *Gradient*, *Grad&Var*) are baseline solutions that implement manually designed features to distinguish between normal and anomalous patches. In particular, the indicator vectors associated to these baseline solutions are suggested by the fact that defects are often flat, whereas normal regions are characterized by prominent edges (see Figures 1 and 2(a)). The fourth solution (*STSIM*) is based on the structural texture similarity measure proposed in [25], which achieves state-of-the-art performance in texture classification. Finally, the fifth solution (*Coding*) was proposed in [24] and also learns a model based on sparse representation, thus it extracts a data-driven feature like the proposed one.

Baseline solutions follow the same framework of the proposed algorithm: more precisely, during the training phase, we compute an indicator vector $\mathbf{f}(c)$ for all of the patches extracted from the images in T. Then, we fit the distribution $\phi_{\mathbf{0}}$ on the computed indicators by KDE [59] and set a suitable threshold γ. During operations, the anomaly mask $\widetilde{\Omega}$ is computed as in (8). The only difference between the baseline solutions is the indicator vector $\mathbf{f}$ used:

- *Variance*: the indicator vector $\mathbf{f}(c)$ corresponds to the sample variance $v(c)$ computed over the patch $\mathbf{s}_c$.
- *Gradient*: the indicator vector $\mathbf{f}(c)$ corresponds to $g(c)$, the average magnitude of the gradients in the patch $\mathbf{s}_c$. More precisely, we compute at first the image of gradient magnitude d as

$$d = \sqrt{(s \circledast d_x)^2 + (s \circledast d_y)^2}, \tag{11}$$

  where $d_x = [-1, 1]$ and $d_y = [-1; 1]$ are the horizontal and vertical derivative filters [60], respectively, and $\circledast$ denotes the 2-dimensional convolution. If we denote by $\mathbf{d}_c$ the patch centered at c extracted from d, then $g(c)$ is the average value of the patch $\mathbf{d}_c$.
- *Grad&Var*: this solution stacks the indicators $v(c)$ and $g(c)$ in a two-dimensional indicator vector

$$\mathbf{f}(c) = \begin{bmatrix} g(c) \\ v(c) \end{bmatrix}. \tag{12}$$

The *STSIM* solution is based on structural texture similarity metric [25], which assesses the similarity between different textures. More precisely, a texture image s is decomposed into steerable-filter subbands [61], and a feature vector $\mathbf{h}(c)$ is obtained by computing subband statistics over $\mathbf{s}_c$. In [25] this is used for texture classification: each feature vector is assigned to the closest class in terms of Mahalanobis distance. In our scenario there is only one texture corresponding to normal images, and we perform anomaly detection using feature vectors as follows: during the training phase we compute the feature vectors from the training images, their mean $\overline{\mathbf{h}}$ and their covariance. Then, during operations, we compute $\mathbf{h}(c)$ for each patch and consider $\mathbf{s}_c$ anomalous when the Mahalanobis distance between $\mathbf{h}(c)$ and $\overline{\mathbf{h}}$ exceeds a fixed threshold γ. This is equivalent to consider as anomalous any patch having an indicator falling outside a confidence region around $\overline{\mathbf{h}}$, defined by the Chebyshev inequality.

The *Coding* solution was presented in [24] and also assumes that normal data admit sparse representation with respect to a dictionary D as in (3). Differently from our algorithm, the anomaly-detection phase in [24] is embedded in a specific sparse coding procedure. More precisely, for each patch $\mathbf{s}_c$ the sparse representation $\mathbf{x}_c$ is computed together with a term $\mathbf{a}_c$, which exhibits a large magnitude when the approximation (3) is not good enough. Thus, anomalies are detected when the magnitude of $\mathbf{a}_c$ exceeds a fixed threshold γ. To enable a fair comparison, we use the same dictionary D in the *Coding* and *Proposed* solutions.

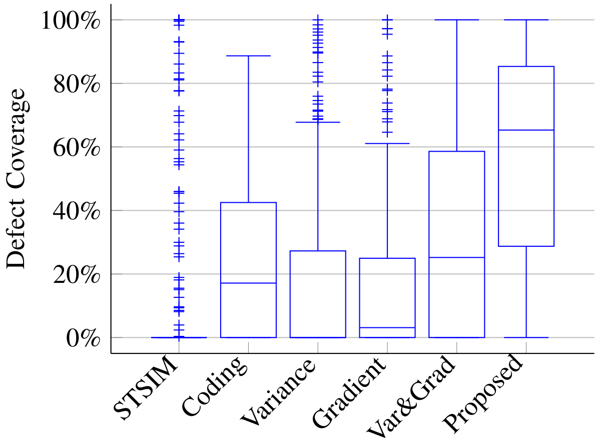


Fig. 5. Box-plots reporting the distribution of the *Defect Coverage*. All the considered solutions have been configured to yield at a fixed FPR = 5%. The proposed algorithm achieves the best performance, as it covers at least 50% of the anomalies for more than 60% of their area.

Of course, choosing the right patch size is very important, since small patches might not exhibit the typical structure of normal data, while large patches might prevent the detection of small anomalies. To fairly compare different methods, each one has to be tested using its optimal patch size. Therefore, we choose the best value of p for baseline and STSIM solutions by testing $p \in \{4^2, 8^2, 12^2, \dots, 120^2\}$ over a validation set of 5 images containing anomalies. Figure 3 shows the average AUC values obtained for each solution, and reports the optimal patch sizes p_{opt} that are used in our experiments. As far as the *Coding* and the *Proposed* solution are concerned, we manually set the patch size $p = 15^2$, since using larger patches would require too many training data to avoid overfitting in model (3) and would substantially increase the computational costs. The same 5 validation images are used to set the other parameters for all of the considered solutions, using cross-validation to maximize the AUC: λ in (6) and (5), ε in (9), as well as the parameters in the *Coding* solution. The *Defect Coverage* is computed by configuring the parameter γ in each method to yield FPR = 5% in these 5 validation images. Finally, these 5 images used for validation are not considered for performance assessment, thus our experiments involve the remaining 35 images.

D. Results

The proposed solution is compared in two experiments against the five solutions described in Section V-C.

At first, we test each solution over the entire dataset, and we assess the overall anomaly-detection performance by the ROC curves averaged over 35 images. These curves are reported in Figure 4, together with the corresponding AUC values in the figure legend. ROC curves clearly indicate that the *Proposed* solution outperforms all the others, achieving AUC values that are at least superior of 0.2. In particular, the proposed solution outperforms the *Coding*, which uses the same dictionary D. Thus, we can conclude that (at least in this specific application) it is not convenient to embed the anomaly detection into the sparse-coding stage, while it is better to separately compute the indicators and then identify anomalies as outliers. The STSIM solution achieves the worse performance, probably because the anomalies in these images are very small and cannot be detected when using large patch sizes. However, as observed in [25] and in Figure 3, the performance of STSIM solution degrades when considering smaller patches, since the local subband statistics cannot capture the texture structure.

TABLE I
COMPUTATIONAL COMPLEXITY OF THE PROPOSED SOLUTION

Operation	Flops per patch	Time per image (s)
Median filter	$O(\sqrt{p}\log p)$	0.0087 ± 0.0011
Mean Subtraction	$O(p)$	0.1522 ± 0.0022
Sparse Coding	$O(pnk + nk^2 + k^3)$	51.190 ± 1.7717
Indicator Vector	$O(pn)$	1.6716 ± 0.0411
Density Evaluation	$O(1)$	0.0273 ± 0.0012
Majority Voting	$O(\sqrt{p})$	0.0159 ± 0.0020
Morphological filtering	$O(\sqrt{p}\log p)$	0.0101 ± 0.0005

In the second experiment we compare the *Defect Coverage* values of all these solutions, to make sure that the superior performance achieved by the proposed solution is not due to a superior coverage of few large defects (like the film in Figure 1). The box-plots in Figure 5 confirm that the proposed solution guarantees a *Defect Coverage* that is often better than others, having most of the defects covered more than 60%. Thus, considering that small anomalies far outnumber the large ones (as described in Section V-A) we can safely conclude that the proposed solution provides superior detection performance also of small defects. We also provide a visual comparison of the anomaly-detection performance. Figure 6 reports the masks $\widehat{\Omega}$ over three meaningful images for the three most effective solutions (according to Figures 4 and 5), generated by setting the same values γ used to compute the *Defect Coverage* values. These masks confirm that the proposed solution provides a superior coverage of very small anomalies, as it clearly emerges in the second image. The large film in the first image is successfully detected by all methods (and in particular by the *Coding* solution). However, the tiny anomalies in the second image are much better detected by the *Proposed* solution. Also, the *Coding* solution completely misses a large bead in the third image. Finally, most of the false alarms in the *Proposed* solution appears at junctions and pairs of filaments that are very close to each other (see third image), which however correspond to very few patches.

E. Computational Complexity

We analyze the computational complexity of the Algorithm 2 to show that our solution can be effectively adopted in a system monitoring the industrial production of nanofibrous materials through spot checks. Table I reports the order of floating point operations (*flops*) that each step of the Algorithm 2 performs to analyze a single patch $\mathbf{s}$, together with the time (mean value ± standard deviation) required to process an entire image using our MATLAB implementation of the algorithm on a PC mounting an Intel Core i7 3.40GHz CPU and 16GB RAM.

The preprocessing consists of (9), where the computation of the median is performed in a sliding manner and requires

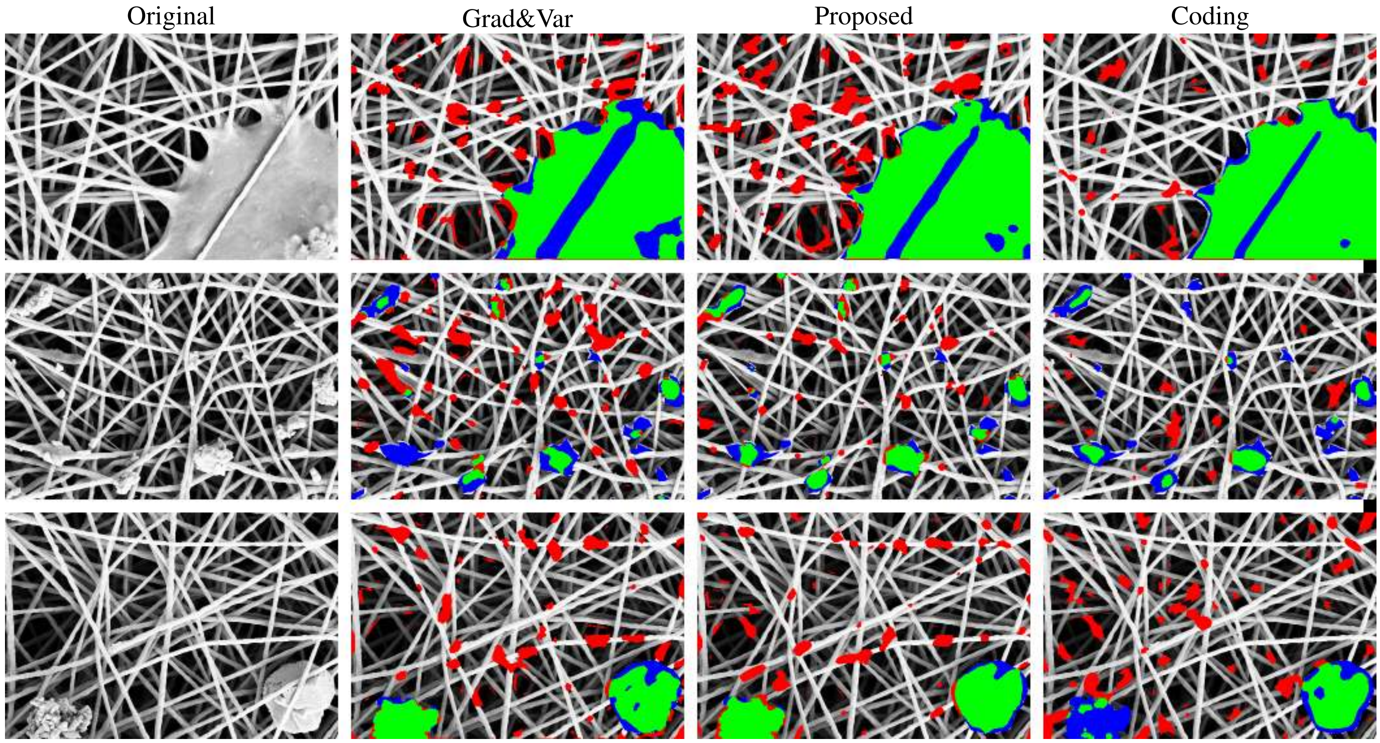


Fig. 6. Examples of anomaly detection on three meaningful SEM images. The first column reports the three original images, while the following ones present the detections obtained from *Grad&Var*, *Proposed* and *Coding* solutions, respectively. These solutions have been chosen as the best performing ones according to Figures 4 and 5. The threshold γ has been set as in Figure 5, to yield FPR = 5%. Pixels correctly identified as anomalous are marked in green, false positives in red, while false-negatives in blue.

$O(\sqrt{p}\log p)$ flops for each patch. Subtracting the mean requires $O(p)$ operations. The sparse coding is performed using the implementation of the LARS algorithm provided in the SPAMS [62] library and requires $O(pnk + nk^2 + k^3)$ flops, where k is the maximum number of iterations of the LARS and is set to 200 in our experiments. Then, the indicators in (7) are computed at the cost of $O(pn)$ operations. Since the values of the density ϕ_0 are sampled in a regular grid, the evaluation of $\phi_0(\mathbf{f})$ can be performed retrieving a value in a lookup-table, at a constant cost $O(1)$. Finally, the mask $\widehat{\Omega}$ is computed by performing the majority voting and applying the morphological operators, which require $O(\sqrt{p})$ and $O(\sqrt{p}\log p)$ flops, respectively.

The computational times reflect the complexity of each step, thus are dominated by the sparse coding. However, the time required to analyze an image is well below the time needed to prepare the next sample material, thus multiple SEM images could be analyzed from the same sample. Moreover, the sparse coding can be easily sped up by considering fewer patches in the analysis, e.g., skipping every second patch. We experienced that such a running process does not substantially affect the resolution of the map $\widehat{\Omega}$, while it decreases the time required by a factor of 4. Finally, it is worth noting that the sparse coding is embarrassingly parallel, as the solution (5) can be executed independently over multiple patches. Hence, a GPU implementation of the proposed solution would dramatically reduce the computational times.

VI. Conclusions

We present an anomaly-detection algorithm that can successfully detect defects in nanofibrous materials. Experiments conducted on a large dataset of SEM images show that the proposed algorithm can effectively detect also tiny defects, and that it processes images in a reasonable time. Thus, this algorithm can be implemented in smart manufacturing systems for nanofibrous material production, to control the quality of the produced material by spot checks. These checks allow to adjust the production process parameters and, when regularly performed, to raise alerts when the production quality falls below a desired standard, yielding both economical and environmental advantages. Ongoing work concerns new dictionary-learning methods to exploit examples of defects during the training phase.

References

[1] L. Persano, A. Camposeo, C. Tekmen, and D. Pisignano, "Industrial upscaling of electrospinning and applications of polymer nanofibers: a review," *Macromolecular Materials and Engineering*, vol. 298, no. 5, pp. 504–520, 2013.

[2] C. Burger, B. S. Hsiao, and B. Chu, "Nanofibrous materials and their applications," *Annu. Rev. Mater. Res.*, vol. 36, pp. 333–368, 2006.

[3] N. Bölgen and A. Vaseashta, "Nanofibers for tissue engineering and regenerative medicine," in *3rd International Conference on Nanotechnologies and Biomedical Engineering*. Springer, 2016, pp. 319–322.

[4] C. Mahoney, D. Conklin, J. Waterman, J. Sankar, and N. Bhattarai, "Electrospun nanofibers of poly (ε-caprolactone)/depolymerized chitosan for respiratory tissue engineering applications," *Journal of Biomaterials Science, Polymer Edition*, no. just-accepted, pp. 1–21, 2016.

[5] R. Barhate and S. Ramakrishna, "Nanofibrous filtering media: filtration problems and solutions from tiny materials," *Journal of membrane science*, vol. 296, no. 1, pp. 1–8, 2007.

[6] K. M. Yun, C. J. Hogan, Y. Matsubayashi, M. Kawabe, F. Iskandar, and K. Okuyama, "Nanoparticle filtration by electrospun polymer fibers," *Chemical Engineering Science*, vol. 62, no. 17, pp. 4751–4759, 2007.

[7] D. Bjorge, N. Daels, S. De Vrieze, P. Dejans, T. Van Camp, W. Audenaert, J. Hogie, P. Westbroek, K. De Clerck, and S. W. Van Hulle, "Performance assessment of electrospun nanofibers for filter applications," *Desalination*, vol. 249, no. 3, pp. 942–948, 2009.

[8] S. Zali, F. Jalali, A. Es-Haghi, and M. Shamsipur, "Electrospun nanostructured polystyrene as a new coating material for solid-phase microextraction: Application to separation of multipesticides from honey samples," *Journal of Chromatography B*, vol. 1002, pp. 387–393, 2015.

[9] F. Zeighami, "Developing optically efficient nanofiber coatings inspired by cyphochilus white beetle," *Journal of Industrial Textiles*, p. 1528083715589751, 2015.

[10] X. Liang, T.-H. Kim, J.-W. Yoon, C.-H. Kwak, and J.-H. Lee, "Ultrasensitive and ultraselective detection of h_2s using electrospun cuo-loaded in_2o_3 nanofiber sensors assisted by pulse heating," *Sensors and Actuators B: Chemical*, vol. 209, pp. 934–942, 2015.

[11] M. Pokorny, J. Rebicek, J. Novak, A. Kotzianova, J. Klemes, J. Ruzickova, and V. Velebny, "Produced nanofibers and technological possibilities of laboratory apparatus 4spin®," *ACC Journal*, 2015.

[12] A. Varesano, C. Vineis, C. Tonetti, D. Omar Sanchez Ramirez, G. Mazzuchetti, S. Ortelli, M. Blosi, and A. Luisa Costa, "Multifunctional hybrid nanocomposite nanofibers produced by colloid electrospinning from water solutions," *Current Nanoscience*, vol. 11, no. 1, pp. 41–48, 2015.

[13] H. Şengül, T. L. Theis, and S. Ghosh, "Toward sustainable nanoproducts," *Journal of Industrial Ecology*, vol. 12, no. 3, pp. 329–359, 2008.

[14] D. E. Meyer, M. A. Curran, and M. A. Gonzalez, "An examination of existing data for the industrial manufacture and use of nanocomponents and their role in the life cycle impact of nanoproducts," *Environmental Science & Technology*, vol. 43, no. 5, pp. 1256–1263, 2009.

[15] M. Borrotti, E. Lanzarone, F. Manganini, S. Ortelli, A. Pievatolo, and C. Tonetti, "Defect minimization and feature control in electrospinning through design of experiments," *Journal of Applied Polymer Science,* in press.

[16] M. Ziabari, V. Mottaghitalab, S. McGovern, and A. Haghi, "Measuring electrospun nanofibre diameter: A novel approach," *Chinese Physics Letters*, vol. 25, no. 8, p. 3071, 2008.

[17] R. Milašius and J. Malašauskienė, "Evaluation of structure quality of web from electrospun nanofibres," *Autex Research Journal*, vol. 14, no. 4, pp. 233–238, 2014.

[18] "Electrospinz sem analyser," electrospinz Ltd, Blenheim, New Zealand; www.electrospinz.co.nz.

[19] G. Boracchi, D. Carrera, and B. Wohlberg, "Novelty detection in images by sparse representations," in *Proceedings of IEEE Symposium on Intelligent Embedded Systems (IES)*, 2014, pp. 47–54.

[20] R. Johnson and D. Wichern, *Applied multivariate statistical analysis*. Prentice Hall, 2002.

[21] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM review*, vol. 51, no. 1, pp. 34–81, 2009.

[22] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Processing*, vol. 99, pp. 215–249, 2014.

[23] D. Carrera, G. Boracchi, A. Foi, and B. Wohlberg, "Scale-invariant anomaly detection with multiscale group-sparse models," in *IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3892–3896.

[24] A. Adler, M. Elad, Y. Hel-Or, and E. Rivlin, "Sparse coding with anomaly detection," in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2013, pp. 1–6.

[25] J. Zujovic, T. N. Pappas, and D. L. Neuhoff, "Structural texture similarity metrics for image analysis and retrieval," *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2545–2558, 2013.

[26] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. Peter Smith Publisher, Incorporated, 1981. [Online]. Available: http://sipi.usc.edu/database/database.php?volume=textures

[27] "Nanotechnologies - terminology and definitions for nano-objects - nanoparticle, nanofibre and nanoplate," iSO/TS 27687:2008.

[28] W. Teo and S. Ramakrishna, "A review on electrospinning design and nanofibre assemblies," *Nanotechnology*, vol. 17, no. 14, pp. R89–R106, 2006.

[29] S. Agarwal, J. H. Wendorff, and A. Greiner, "Use of electrospinning technique for biomedical applications," *Polymer*, vol. 49, no. 26, pp. 5603–5621, 2008.

[30] T. Liu, C. Burger, and B. Chu, "Nanofabrication in polymer matrices," *Progress in Polymer Science*, vol. 28, no. 1, pp. 5–26, 2003.

[31] A. Formhals, "Process and apparatus for preparing artificial threads," 1934, uS Patent 1,975,504.

[32] J. Deitzel, J. Kleinmeyer, D. Harris, and N. B. Tan, "The effect of processing variables on the morphology of electrospun nanofibers and textiles," *Polymer*, vol. 42, no. 1, pp. 261–272, 2001.

[33] B. Barua and M. C. Saha, "Investigation on jet stability, fiber diameter, and tensile properties of electrospun polyacrylonitrile nanofibrous yarns," *Journal of Applied Polymer Science*, vol. 132, no. 18, p. 41918, 2015.

[34] A. Chlanda, J. Rebis, E. Kijeńska, M. J. Wozniak, K. Rozniatowski, W. Swieszkowski, and K. J. Kurzydlowski, "Quantitative imaging of electrospun fibers by peakforce quantitative nanomechanics atomic force microscopy using etched scanning probes," *Micron*, vol. 72, pp. 1–7, 2015.

[35] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[36] M. Zontak and I. Cohen, "Defect detection in patterned wafers using anisotropic kernels," *Machine Vision and Applications*, vol. 21, no. 2, pp. 129–141, 2008. [Online]. Available: http://dx.doi.org/10.1007/s00138-008-0146-y

[37] M. Markou and S. Singh, "Novelty detection: a review - part 1: statistical approaches," *Signal processing*, vol. 83, no. 12, pp. 2481–2497, 2003.

[38] ——, "Novelty detection: a review - part 2: neural network based approaches," *Signal processing*, vol. 83, no. 12, pp. 2499–2521, 2003.

[39] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Advances in Neural Information Processing Systems*, 1999, pp. 582–588.

[40] W.-C. Li and D.-M. Tsai, "Defect inspection in low-contrast lcd images using hough transform-based nonstationary line detection," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 1, pp. 136–147, 2011.

[41] D.-M. Tsai and J.-Y. Luo, "Mean shift-based defect detection in multicrystalline solar wafer surfaces," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 1, pp. 125–135, 2011.

[42] D.-M. Tsai, I. Chiang, Y.-H. Tsai *et al.*, "A shift-tolerant dissimilarity measure for surface defect detection," *Industrial Informatics, IEEE Transactions on*, vol. 8, no. 1, pp. 128–137, 2012.

[43] J. Mairal, F. Bach, and J. Ponce, "Sparse modeling for image and vision processing," *Foundations and Trends in Computer Graphics and Vision*, vol. 8, no. 2-3, pp. 85–283, 2012. [Online]. Available: http://dx.doi.org/10.1561/0600000058

[44] D. Carrera, G. Boracchi, A. Foi, and B. Wohlberg, "Detecting anomalous structures by convolutional sparse models," in *Neural Networks (IJCNN), 2015 International Joint Conference on*, July 2015, pp. 1–8.

[45] D.-M. Tsai, S.-C. Wu, and W.-Y. Chiu, "Defect detection in solar modules using ica basis images," *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 1, pp. 122–131, 2013.

[46] H. Gao, C. Ding, C. Song, and J. Mei, "Automated inspection of e-shaped magnetic core elements using k-tsl-center clustering and active shape models," *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 3, pp. 1782–1789, 2013.

[47] X. Bai, Y. Fang, W. Lin, L. Wang, and B.-F. Ju, "Saliency-based defect detection in industrial images by using phase spectrum," *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 4, pp. 2135–2145, 2014.

[48] S. Mortazavian, M. M. Shabestary, Y. A.-R. I. Mohamed, and G. B. Gharehpetian, "Experimental studies on monitoring and metering of radial deformations on transformer hv winding using image processing and uwb transceivers," *Industrial Informatics, IEEE Transactions on*, vol. 11, no. 6, pp. 1334–1345, 2015.

[49] C. Alippi, G. Boracchi, and B. Wohlberg, "Change detection in streams of signals with sparse representations," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 5252–5256.

[50] D. Carrera, B. Rossi, D. Zambon, P. Fragneto, and G. Boracchi, "Ecg monitoring in wearable devices by sparse models," in *Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2016, pp. 145–160.

[51] Q. Liu, S. Wang, J. Luo, Y. Zhu, and M. Ye, "An augmented lagrangian approach to general dictionary learning for image denoising," *Journal of Visual Communication and Image Representation*, vol. 23, no. 5, pp. 753–766, 2012.

[52] Y. M. Lu and M. N. Do, "A theory for sampling signals from a union of subspaces," *Signal Processing, IEEE Transactions on*, vol. 56, no. 6, pp. 2334–2345, 2008.

[53] M. Elad, *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Science & Business Media, 2010.

[54] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, pp. 267–288, 1996.

[55] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[56] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse problems*, vol. 23, no. 3, p. 947, 2007.

[57] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

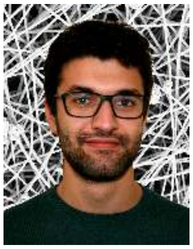
[58] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani *et al.*, "Least angle regression," *The Annals of statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[59] Z. I. Botev, J. F. Grotowski, D. P. Kroese *et al.*, "Kernel density estimation via diffusion," *The Annals of Statistics*, vol. 38, no. 5, pp. 2916–2957, 2010.

[60] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2008.

[61] E. P. Simoncelli and W. T. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *Proceedings of IEEE Internation Conference on Image Processing (ICIP)*, 1995, p. 3444.

[62] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *The Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.



Diego Carrera received the M.S. degree in Mathematics at Università degli Studi di Milano, Italy, in 2013. In 2014 he was research fellow at the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy, and in 2015 he was project researcher at the Department of Signal Processing, Tampere University of Technology, Finland. Currently, he is Ph.D. student at Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano. His research interests include: change and anomaly detection methods, learning techniques for statistical signal processing and algorithms for intelligent embedded systems.

Fabio Manganini received the M.S. degree in Mathematical Engineering from the Politecnico di Milano, Italy, in 2013. During 2014 - 2016 he worked as a research fellow at IMATI-CNR, Italy, dealing with mathematical and statistical methods to optimize the electrospinning production process for nanofibrous materials. Currently, he is an engineer in the computer vision R&D laboratory at the Sprinx Technologies S.r.l., Italy.

Giacomo Boracchi received the M.S. degree in Mathematics from the Università Statale degli Studi di Milano, Italy, and the Ph.D. degree in Information Technology at Politecnico di Milano, Italy, in 2004 and 2008, respectively. He was researcher at Tampere International Center for Signal Processing, Finland, during 2004-2005. Currently, he is an assistant professor at the Dipartimento di Elettronica, Informazione e Bioingegneria of the Politecnico di Milano. His main research interests include learning methods for nonstationary environments, as well as mathematical and statistical methods for image processing and analysis. In 2015 he received the IBM Faculty Award, and in 2016 the IEEE Transactions on Neural Networks and Learning Systems Outstanding Paper Award.

Ettore Lanzarone received the M.S. degree in Biomedical Engineering and the Ph.D. in Bioengineering from the Politecnico di Milano, Italy, in 2004 and 2008, respectively. Since 2011 he is Researcher at IMATI-CNR, Italy. He is Adjunct Professor at the Dipartimento di Matematica, Politecnico di Milano, Italy. His research interests include optimization and operations research, stochastic models, and bioengineering. He was involved in several research projects and he was the coordinator of the IMATI-CNR research unit for the projects "NanoTWICE" and "Autospin".