Consiglio Nazionale delle Ricerche
Istituto di Matematica Applicata
e Tecnologie Informatiche
“Enrico Magenes”

REPORT SERIES

Diego Carrera, Fabio Manganini,
Giacomo Boracchi, Ettore Lanzarone

Defect Detection in Nanostructures

16-03

IMATI REPORT Series

Nr. 16-03

April 2016

Editorial Office
Istituto di Matematica Applicata e Tecnologie Informatiche "*E. Magenes*"
Consiglio Nazionale delle Ricerche
Via Ferrata, 5/a
27100 PAVIA (Italy)
Email: reports@imati.cnr.it
http://www.imati.cnr.it

IMATI Report Series Nr. 16-03 – 28th April 2016

Defect Detection in Nanostructures

Diego Carrera, Fabio Manganini, Giacomo Boracchi, Ettore Lanzarone

Abstract.

Nanoproducts represent a potential growing sector and nano brous materials are widely requested in industrial, medical and environmental applications. Unfortunately, the production processes at the nanoscale are difficult to control, and produced artifacts often exhibit local defects that prevent their functional properties. We present a fully-automated solution to detect defects in nano brous materials during their production, yielding smart-manufacturing systems that reduce quality-inspection times and wastes. We analyze SEM images of nano brous materials and learn, during an initial training phase, a model yielding sparse representations of the structures that characterize correctly produced nano bers. Defects are then detected by analyzing patches in test images and assessing the goodness-of- t of each patch to the learned model. The proposed solution has been successfully validated over 45 images acquired from samples produced by a prototype electrospinning machine. The low computational times indicate that the proposed solution can be adopted for real-time monitoring in an industrial-production scenario.

Keywords: Defect and Anomaly Detection, Nano brous materials, Quality control, Sparse Representation, Smart Manufacturing

DEFECT DETECTION IN NANOSTRUCTURES

DIEGO CARRERA[1] , FABIO MANGANINI[2]
, GIACOMO BORACCHI[1] , AND ETTORE LANZARONE[2]

[1]DEIB, Politecnico di Milano, Via Ponzio 34/5, I-20133 Milan, Italy
[2]CNR-IMATI "Enrico Magenes", Via Corti 12, I-20133 Milan, Italy

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 produced artifacts often exhibit local defects that prevent their functional properties. We present a fully-automated solution to detect defects in nanofibrous materials during their production, yielding smart-manufacturing systems that reduce quality-inspection times and wastes. We analyze SEM images of nanofibrous materials and learn, during an initial training phase, a model yielding sparse representations of the structures that characterize correctly produced nanofibers. Defects are then detected by analyzing patches in test images and assessing the goodness-of-fit of each patch to the learned model. The proposed solution has been successfully validated over 45 images acquired from samples produced by a prototype electrospinning machine. The low computational times indicate that the proposed solution can be adopted for real-time monitoring in an industrial-production scenario.

Key words. Defect and Anomaly Detection, Nanofibrous materials, Quality control, Sparse Representations, Smart Manufacturing

1. Introduction. Nanoproducts demand has steadily increased over the past few years [41]. In particular, nanofibrous materials are nowadays widely requested [17], e.g., in life sciences and medicine [11, 33], filtration and water treatment [8, 53, 10], surface coating [54, 55], and sensors [29]. However, despite the manifold production processes (see Section 2.1) and the flourishing of laboratory prototypes [43, 52], nanofibrous materials are not yet industrially produced at the large scale, and are recognized as one of the main challenges in high-tech manufacturing [45, 38], e.g., in the Horizon 2020 *Program Factory of the Future* (`ec.europa.eu/research/industrial_technologies/factories-of-the-future_en.html`).

Production processes at the nanoscale are still difficult to control and, as a result, the structure characterizing these materials (nanostructure) often exhibit local defects, like the *beads* and *films* shown in Figure 1.1, which might impair their properties (e.g., mechanical or filtering properties). These defects make the production long and costly, as they result in a large proportion of wasted materials, and at the same time they make it necessary thorough quality-inspection procedures.

Therefore, smart-manufacturing solutions and, in particular, highly-automated systems for monitoring nanostructures quality are of crucial importance for nanoproduct production. In fact, to guarantee a satisfactory production quality, it is necessary to raise prompt alerts as soon as defects exceed a tolerance level or, more in general, when the quality of the produced artifacts falls below a desired standard, as this would allow to take corrective actions – including halting the machinery.

Unfortunately, to the best of our knowledge, there are no automatic systems able to provide a comprehensive monitoring of nanofiber production and, in particular, there are no systems able to detect and quantitatively assess localized defects like

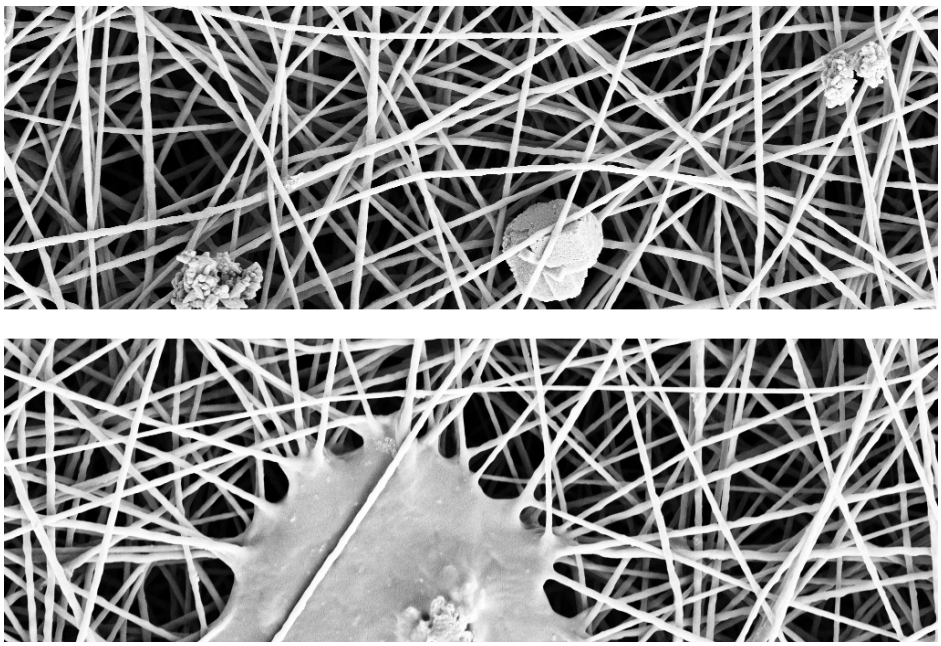


FIG. 1.1. *Two details from SEM images depicting nanofibrous material produced by electrospinning. The large area covered by filaments in both images refer to good quality fibers. The top image contains two sorts of localized defects: a small* speck of dust *at the center and two* beads*, namely fiber clots. The bottom image contains a* films*, which is a thin layer of material among the fibers.*

those in Figure 1.1. As discussed in Section 2.2, this sort of monitoring problems can be conveniently addressed by analyzing images acquired by Scanning Electron Microscopes (SEM), thanks to their resolution up to 1 nanometer. Few vision-based systems have been proposed for monitoring the nanofibrous materials, but they only measure the fiber diameter [57, 39] or orientation [1], while not the localized defects, which is indeed our goal. It is worth mentioning that defect-detection algorithms for different industrial applications (e.g., those in [51, 26, 7, 40]) are very *ad-hoc* and application-specific: as such, these cannot be applied to nanofribrous materials.

Our paper fills the gap towards the development of a fully-automated vision-based system that monitors the production of nanofibrous materials. In particular, we propose an algorithm that automatically detects defects affecting the nanofiber structure by analyzing SEM images. While humans can easily identify defects in images like Figure 1.1, thus distinguishing between normal and anomalous regions (i.e., the defects), 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 defects can be very different in their appearance and shape. Hence, it is not straightforward to design effective criteria that discriminate between normal and anomalous regions in these images.

We address this task as an anomaly-detection problem, which we effectively solve by learning a model that describes normal regions by sparse representations [16]. Our algorithm analyzes a SEM image in a patch-wise manner, and detects anomalies by assessing whether each patch fits or not the learned model. Sparse representations are nowadays one of the leading models in image and signal processing applications [16, 34], and sparsity has shown to be a very effective prior for learning adaptive representations. Differently from the majority of defect-detection solutions, our algorithm

is general and can be extended to other monitoring scenarios, like analyzing MEMS signals [6] or textures [12].

Our algorithm can be reasonably implemented on a machinery that embeds a SEM microscope and all of the accessories to prepare samples for spot checks. In fact, the time required for processing an image is negligible with respect to the sample preparation: as such, the proposed algorithm is an important step towards the development of a comprehensive monitoring system which will be able to generate timely alarms and feedbacks for correcting the production process.

We have successfully validated our algorithm on SEM images of nanofibrous materials produced through a prototype electrospinning machine (see Section 2.1). Such a machine was 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, including a testing module whose our algorithm is the core.

This paper extends our conference publication [12] which presents the bare anomaly-detection algorithm. Here, we describe the industrial production scenario and the conditions where this algorithm operates. Furthermore, we introduce specific algorithmic enhancements for handling this sort of images and we substantially extend the experimental section with respect to [12]. On the algorithmic front, we introduce pre- and post-processing stages to improve the defect-detection performance in nanostructures. Concerning the experiments, we present an extensive campaign to quantitatively assess the performance of our algorithm on a dataset of 45 SEM images, and we compare it against state-of-the-art methods for anomaly detection in images [3, 59]. The only quantitative tests in [12] were performed on a dataset of texture images [15] and few SEM images were considered only for qualitative analysis.

The paper is structured as follows. Nanofibrous materials and their production processes are presented in Section 2, together with an overview on existing solutions to monitor the production of nanofibrous materials and to detect anomalies in images. Our anomaly-detection problem is formulated in Section 3, while the proposed algorithm is presented in Section 4. Experiments, including a description of our dataset, are detailed in Section 5. Finally, the conclusions and future research directions are presented in Section 6.

2. Background and Related Works. Nanofibrous materials are porous materials made of polymer nanofibers, i.e., fibers whose diameter is smaller than 100 nanometres [2], which are embedded together in an amorphous structure. Thanks to the possibility of tuning the material properties working on the morphology at the nanoscale, nanofibrous materials are nowadays applied in several fields, and have generated a lot of innovative applications in the last few years [17, 47]. Most of applications concern medicine, e.g., anti-adhesion materials in surgery, biofunctionalized materials for biomedical applications, and scaffolds for tissue engineering [4, 11, 33]. Other important applications concern the construction of filtration membranes [8, 53, 10], where nanofibrous materials are inserted in a media to provide superior filtration capabilities. Finally, nanofibrous materials are also employed in surface coating [54, 55] and sensors [29].

2.1. Nanofibrous Materials Production and Electrospinning. Several techniques for producing nanofibers have been presented in the literature, e.g., self-assembly, templating, lithography and electrospinning [17, 31]. 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, nanofi-

brous materials produced through electrospinning typically feature higher surface-to-volume ratio and better controlled porosity than those produced by other techniques. Electrospinning also allows large flexibility in the size and shape of the produced nanofibers [17, 47]. Electrospinning was originally studied by Rayleigh in 1897, and patented in 1934 [25]; however, it has only recently gained momentum thanks to the explosive growth of nanostructured materials [17].

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 nanofiber collecting counter-electrode. 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 a random fashion, forming the non-woven structure. Further details about electrospinning process can be found in [17, 47].

Unfortunately, external environmental variables and process instabilities (e.g., in the solution jet [23, 9]) may seriously affect the production, introducing high variability in the fiber characteristics and localized defects like those in Figure 1.1. 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 among them. Finally, tiny speck of dust might also get trapped in the fibers, and this also prevent the functional properties of the nanofibrous materials. Another issue is the presence of big holes in the material, i.e., parts of the artifact where no fibers are present. Obviously, a *physiological* porosity is expected and desired; 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.

2.2. 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 artifacts through sample checks. Monitoring production parameters is simpler, because it just requires to process signals acquired from the machine; however, the quality of the produced materials can be affected by several other stochastic factors and analyzing these signals may be not sufficient to detect problems in the produced artifacts. 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., nanomechanical properties through atomic force microscopy [21] or filtering behaviour in case of filters [53, 10]). The second option does not allow to generate timely alerts to correct the production of nanofibrous material. 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, and large-scale industrial scenarios requires automatic solutions. So far, the only existing automatic solutions are meant to measure the fibers diameter and orientation [57, 39, 1], while they are not able to detect nor quantitatively assess defects like those in Figure 1.1.

2.3. Anomaly Detection in Images. Algorithms for detecting anomalies in images can be divided in reference-based and reference-free ones. In reference-based methods (e.g., [58]), 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 [58].

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 [19] 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* [42, 36, 37] (in the machine-learning community this problem is also known as one-class classification [44]). In this paper we present a novelty-detection algorithm, even though we refer to anomaly/defect detection, as they are more appropriate descriptions for the considered defect-detection 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.

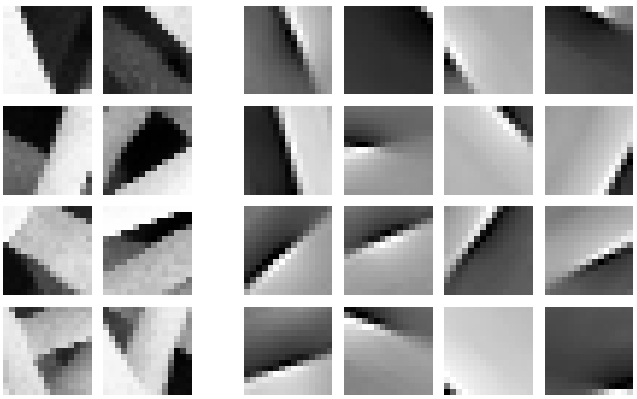
A general overview of anomaly-detection algorithms for images is provided in [19, 42]; not surprisingly, they have been also proposed for industrial monitoring purposes [28, 50, 49]. We here focus on anomaly-detection algorithms based on sparse models, which stands at the core of the proposed algorithm and of our previous work [12]. In [12], a model yielding sparse representation of normal patches was learned and used to compute indicators that assess the conformance of each patch with respect to the learned model. Then, anomalies are identified as outliers in the distribution of such indicators. A different approach was proposed in [3], where anomalous data are identified while computing the sparse representation by means of a specific sparse-coding procedure. Convolutional-sparse models [18] were shown to be particularly effective in detecting anomalies, even though they are more computational demanding than traditional patch-based models like that described in Section 4.1. It is worth also mentioning that sparse models have been used to detect changes in streams of signals [6] and unusual events in video sequences [56, 22].

3. 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 at pixel $c \in \mathcal{X}$ is denoted by $s(c)$ [1]. Our goal is to locate anomalous regions in s; as such, the problem can be formulated as estimating the unknown anomaly mask

$$\Omega(c) = \begin{cases} 0 & \text{if } c \text{ falls inside a normal region} \\ 1 & \text{if } c \text{ falls inside an anomalous region} \end{cases} \tag{3.1}$$

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*)

[1] 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.



(a) (b)

FIG. 4.1. *(a) Examples of training patches. (b) Atoms of the dictionary learned by the ADMM algorithm [30]. Here we show only 16 atoms.*

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.

4. 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 extracted at a specific pixel, denoted as patch center. In particular, the patch centered in $c \in \mathcal{X}$ is defined as

$$\mathbf{s}_c = \{s(c+u), u \in \mathcal{U}\},\ \forall c \in \mathcal{X} \tag{4.1}$$

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 4.1(a) shows examples of patches extracted from normal images.

4.1. Normal Patches Modeling. Our modeling assumption is that patches in normal images are drawn from a stationary, stochastic process $\mathcal{P}_N$, and we use the training set T to learn a *dictionary* $D \in \mathbb{R}^{p \times n}$ [24] that provides sparse approximation of all normal patches $\mathbf{s}_c \in \mathbb{R}^p$, i.e.,

$$\mathbf{s}_c \approx D\mathbf{x}_c\,. \tag{4.2}$$

In (4.2), $\mathbf{x}_c \in \mathbb{R}^n$ denotes the coefficient vector which has few nonzero components, namely, $\mathbf{x}_c$ is *sparse*. Note that (4.2) corresponds to the following expression (where c has been omitted for notation sake)

$$\mathbf{s} \approx \sum_{i=1}^{N} x_i \mathbf{d}_i\,, \tag{4.3}$$

where x_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_i in (4.3) are nonzero and that each normal patch can be (well) approximated by a linear combination of few dictionary atoms. Thus, our modeling assumption (4.2) leads to learning a union of low-dimensional subspaces that contains normal patches [32]. 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 (4.3).

The coefficients of the sparse representation $\mathbf{x}_c$ in (4.2) are obtained by projecting a patch $\mathbf{s}_c$ on the closest subspace spanned by the atoms of D. This problem is referred to as *sparse coding* and formulated as

$$\mathbf{x}_c = \underset{\widetilde{\mathbf{x}} \in \mathbb{R}^n}{\arg\min} \|D\widetilde{\mathbf{x}} - \mathbf{s}_c\|_2 \;\text{ such that }\; \|\mathbf{x}_c\|_0 \leq \kappa. \tag{4.4}$$

where $\|\mathbf{x}\|_0$ denotes the ℓ^0 norm of $\mathbf{x}$, i.e., its number of non-zero coefficients, and $\kappa > 0$ is the maximum number of nonzero coefficients allowed in the representations. The minimization (4.4) is typically solved by greedy algorithms such as the Orthogonal Matching Pursuit [48].

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

$$\begin{aligned}[D, X] = \underset{\widetilde{D} \in \mathbb{R}^{p \times n},\, \widetilde{X} \in \mathbb{R}^{n \times m}}{\arg\min} & \|\widetilde{D}\widetilde{X} - S\|_2, \\ \text{such that } \|\widetilde{\mathbf{x}}_i\|_0 < k, \;\; & i = 1, \ldots, n\end{aligned} \tag{4.5}$$

where the sparsity constraint applies to each column of X. The problem (4.6) is typically solved [5] by alternating between the calculation of the dictionary atoms and of the sparse representations X of training patches S with respect to the current dictionary.

The sparsity of a representation can be also measured by the ℓ^1 norm $\|\mathbf{x}\|_1 = \sum_{i=1}^{p} |x_i|$, thus the sparse coding (4.4) and dictionary-learning (4.6) problems are replaced by their convex relaxation. Problem (4.4) becomes

$$\mathbf{x}_c = \underset{\widetilde{\mathbf{x}} \in \mathbb{R}^N}{\arg\min} \frac{1}{2}\|D\widetilde{\mathbf{x}} - \mathbf{s}_c\|_2^2 + \lambda\|\widetilde{\mathbf{x}}\|_1, \tag{4.6}$$

which is referred to as basis pursuit denoising (BPDN) problem [20], and can be solved by convex optimization algorithms. The dictionary learning (4.6) becomes

$$[D, X] = \underset{\widetilde{D} \in \mathbb{R}^{p \times n},\, \widetilde{X} \in \mathbb{R}^{n \times m}}{\arg\min} \frac{1}{2}\|\widetilde{D}\widetilde{X} - S\|_2^2 + \lambda\|\widetilde{X}\|_1, \tag{4.7}$$

which can still be solved by alternating the computation of D and X. In the following, we adopt the ℓ^1 norm penalized expressions (4.6) and (4.7), which are solved by Alternating Direction Methods of Multipliers (ADMM) [14].

Normal patches are thus modeled by learning a dictionary D that solves (4.7), by means of the algorithm in [30] based on the ADMM. Figure 4.1(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 4.1(a).

4.2. 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 *bivariate indicator* which jointly accounts for the reconstruction error and the sparsity of the representation (4.2). In particular, given a patch $\mathbf{s}_c$, we compute $\mathbf{x}_c$ (sparse coding) by solving the BPDN problem (4.6), and we define the indicator in c:

$$\mathbf{f}(c) = \begin{bmatrix} \|D\mathbf{x}_c - \mathbf{s}_c\|_2 \\ \|\mathbf{x}_c\|_1 \end{bmatrix}. \tag{4.8}$$

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), and the corresponding indicators would be outliers with respect to the distribution ϕ_0 of indicators extracted from normal patches. While in [12] we detect outliers in the indicators by means of a confidence region built according to the multivariate Chebyshev's inequality, we here model the distribution ϕ_0 by Kernel Density Estimation (KDE), adopting a kernel based on linear diffusion with automatic bandwidth selection [13].

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

$$\widetilde{\Omega}(c) = \begin{cases} 0 & \text{if } \phi_0\left(\mathbf{f}\left(c\right)\right) < \gamma \\ 1 & \text{if } \phi_0\left(\mathbf{f}\left(c\right)\right) \geq \gamma \end{cases}, \tag{4.9}$$

being $\gamma > 0$ a parameters 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.

4.3. 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 (4.2) equal to zero. Unfortunately, null indicators 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 the patches

$$S = \{\mathbf{s}_c \,|\, \text{median}(\mathbf{s}_c) > \varepsilon\}, \tag{4.10}$$

where $\varepsilon > 0$ is a manually tuned parameter. The same operation is applied on test images. The median in (4.10) was used to remove also dark patches that marginally overlap with a filament. It is worth mentioning that nanofibers 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 not at patch level.

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.

1. **Input:** a training set of normal images T.
2. Prepare two training sets of normal patches S and V.
3. Remove dark patches from S and V as in (4.10).
4. Subtract the average value from each patch in S and V.
5. Learn the dictornary D solving (4.6) from S.
6. Compute the sparse representation of V solving (4.6).
7. Compute $\{\mathbf{f}(c), \forall \mathbf{s}_c \in V\}$ as in (4.8), the indicators of normal patches.
8. Fit ϕ_0 through KDE, and define a suitable the threshold $\gamma > 0$ to be used in the detector (4.9).

Algorithm 1: The training phase of the proposed algorithm.

4.4. Postprocessing. The anomaly detector (4.9) takes as input the whole patch $\mathbf{s}_c$ even though its final decision concerns only the patch center c and not 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 (4.9) by majority voting:

$$\widehat{\Omega}(c) = \begin{cases} 0 & \text{if } \#\mathcal{A}_c < \#\mathcal{N}_c \\ 1 & \text{if } \#\mathcal{A}_c \geq \#\mathcal{N}_c \end{cases}, \tag{4.11}$$

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 those 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 [27]. More precisely, we apply an erosion followed by a dilatation, which are nonlinear filters based on order statistics: the minimum and the maximum over a given support, respectively. We experienced that these binary operations applied over a neighborhood smaller than $\mathcal{U}$ slightly improve the coverage of anomalous regions.

4.5. Algorithm Summary. We here describe the training phase (Algorithm 1) and detail all of the steps of the proposed anomaly-detection algorithm during operations (Algorithm 2). To prevent overfitting, it is convenient to extract from T two sets of patches, S and V, which are both pre-processed as in Section 4.3 (Algorithm 1, lines 1-3). We use patches in S to learn the dictionary D by solving (4.7) via the ADMM algorithm [30] (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 4.3 (Algorithm 2, line 2-3). Then, 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 (4.9) 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 10-13).

5. 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 parame-

Input: test image s; D, ϕ_0, $\gamma > 0$ from Algorithm 1
foreach *patch* $\mathbf{s}_c$ *in* s **do**
    `// Pre-processing`
    if $\mathbf{s}_c$ *satisfies* (4.10) **then**
        Subtract form $\mathbf{s}_c$ its average value.
        `// Anomaly Detection`
        Solve (4.6) to compute the sparse representation $\mathbf{x}_c$.
        Compute the indicator $\mathbf{f}(c)$ as in (4.8).
        Define the anomaly mask value $\widetilde{\Omega}(c)$ as in (4.9).
    else
    end
end
`// Post-processing`
foreach *pixel* c *of the anomaly mask* $\widetilde{\Omega}$ **do**
    Set the anomaly mask $\widehat{\Omega}$ via majority voting (4.11).
end
Apply erosion and dilatation operators to $\widehat{\Omega}$.

Algorithm 2: The operational phase of the proposed algorithm.

ters and environmental variables) that typically produce good quality nanofibers with no macroscopic defects. There is no point of considering worse production conditions, as these can be easily macroscopically detected by other approaches (e.g., altered current patterns during the process) and do not require vision-based monitoring.

5.1. Dataset Description. SEM images are acquired with the FE-SEM (Carl Zeiss Sigma NTS, Gmbh Öberkochen, Germany). A sample of 4×4 cm from the artifact 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%.

The dataset used in our experiments 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 the defects, defining the anomaly mask Ω that is used as a ground truth. 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. SEM images together with their ground truth are available for download at `http://web.mi.imati.cnr.it/ettore/NanoTwice`.

5.2. Figures of Merit. To assess the performance of the proposed solution we compute the following figure of merits: *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 quantitative performance index, which is equal to 1 in the case of the perfect detector (i.e., TPR = 100% and FPR = 0%).

However, the AUC is mainly influenced by large defects, while our goal is to

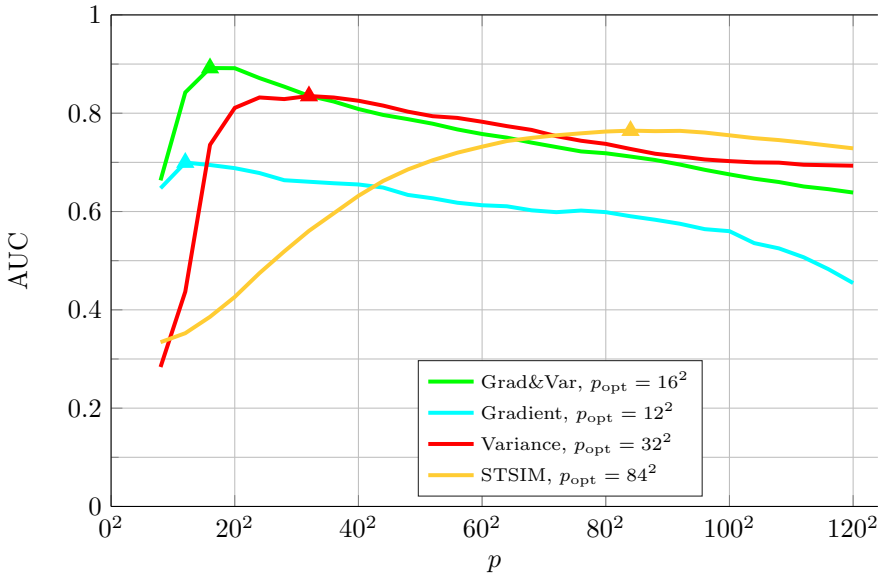


FIG. 5.1. *Values of the AUC for the considered methods obtained by varying the patch size. The solid triangles indicate the points where the AUC is maximized and in the legend the optimal patch size* p_{opt} *is reported.*

detect all of them, disregarding their size. Therefore, to quantitatively assess the coverage of all defects, we extract the connected components [27] 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.

5.3. Alternative Solutions. We compare our algorithm against five anomaly-detection solutions that, like ours, operate patch-wise without any assumption about anomalies. All solutions have been tested in comparable settings: they are trained from a set T of 5 anomaly-free images, they preliminary remove dark patches (4.10), and they perform the same postprocessing described in Section 4.4.

The first three are baseline solutions (*Variance*, *Gradient*, *Grad&Var*); the fourth one (*Coding*) was proposed in [3] and is also based on sparse representation; the last one (*STSIM*) is based on the distance proposed in [59] which achieves state-of-the-art performance in texture classification.

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 [13] and set a suitable threshold γ. During operations, the anomaly mask $\widetilde{\Omega}$ is computed as in (4.9). 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

$$d = \sqrt{(s \circledast d_x)^2 + (s \circledast d_y)^2}, \tag{5.1}$$

where $d_x = [-1, 1]$ and $d_y = [-1; 1]$ are the horizontal and vertical derivative filters [27], 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

$$\mathbf{f}(c) = \begin{bmatrix} g(c) \\ v(c) \end{bmatrix}. \tag{5.2}$$

The *Coding* solution was presented in [3] and also assumes that normal data admit sparse representation with respect to a dictionary D as in (4.2). Differently from our algorithm, the anomaly-detection phase in [3] 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 $\mathbf{a}_c$, whose magnitude becomes large when approximation (4.2) is not good enough. Thus, anomalies are detected 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.

The *STSIM* solution is based on structural texture similarity metric [59], which assesses the similarity between different textures. More precisely, a texture image s is decomposed into steerable-filter subbands [46], and a feature vector $\mathbf{h}(c)$ is obtained by computing subband statistics over $\mathbf{s}_c$. In [59] this is used for texture classification: each feature vector is assigned to the class that is closest in term 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 $\bar{\mathbf{h}}$ and their covariance. Then, during operations, we compute $\mathbf{h}(c)$ for each patch and consider $\mathbf{s}_c$ as anomalous when the Mahalanobis distance between $\mathbf{h}(c)$ and $\bar{\mathbf{h}}$ exceeds a fixed threshold γ. This is equivalent to consider anomalous any patch having an indicator falling outside a confidence region around $\bar{\mathbf{h}}$, defined by the Chebyshev inequality.

Of course, choosing the right patch size is very important, since small patches might not report 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 $p \in \{4^2, 8^2, 12^2, \ldots, 120^2\}$ over a validation set of 5 images containing anomalies. Figure 5.1 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 (4.2) and would substantially increase the computational costs. The same 5 validation images are used to set other parameters for all the considered solution by cross-validation: λ in (4.7) and (4.6), ε in (4.10), 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.

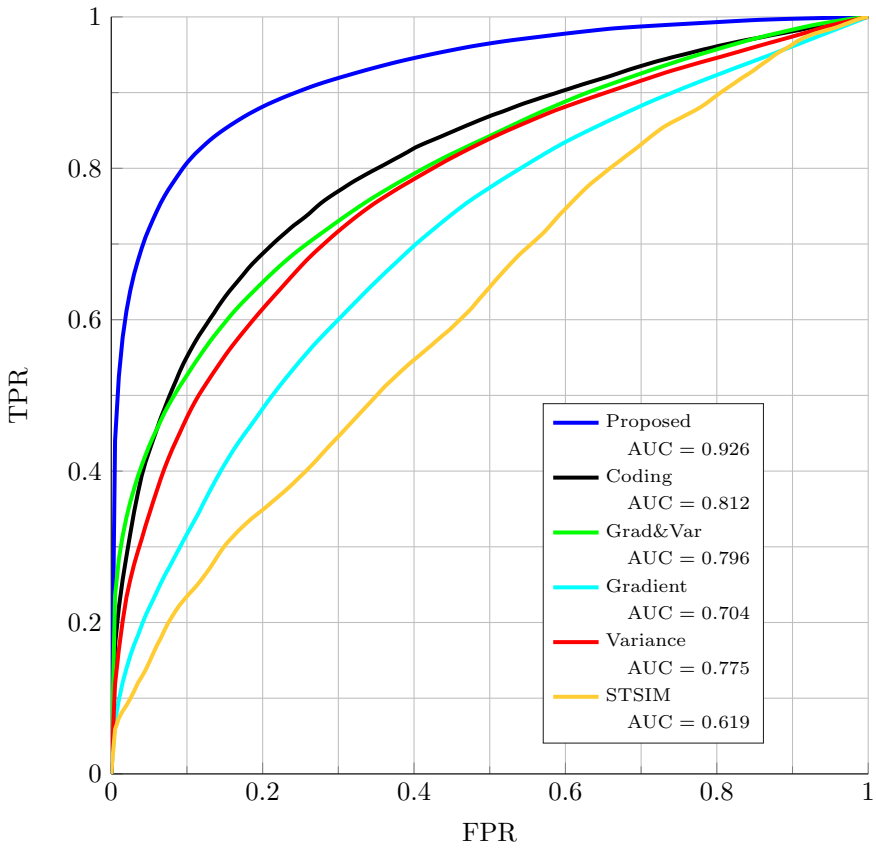


FIG. 5.2. *ROC curves for all the solutions considered in Section 5, with the corresponding AUC values reported in the legend.* Proposed *and* Coding *solutions use a patch size* $p = 15^2$*, while the others the optimal patch size* p_{opt} *selected in Figure 5.1. The* Proposed *solution outperforms by far all the others.*

5.4. Results. The proposed solution is compared against those five described in Section 5.3 in two experiments.

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 5.2, 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 [59] and in Figure 5.1, the performance of STSIM metric degrades when considering smaller patches, since the local subband statistics cannot capture the texture structure.

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 one is not due to a better coverage over few large defects (like the film in Figure 1.1).

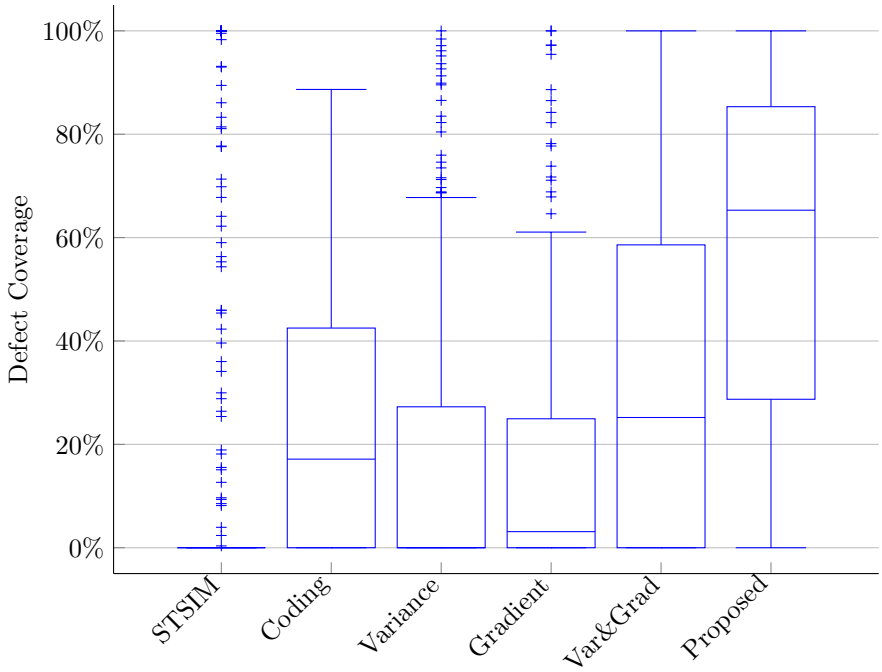


FIG. 5.3. *Box-plots reporting the distribution of the* Defect Coverage. *All the considered solutions have been configured to yield at a fixed* FPR $= 5\%$. *The proposed algorithm achieves the best performance, as it covers at least* 50% *of the anomalies for more than* 60% *of their area.*

The box-plots in Figure 5.3 confirm that the proposed solution guarantees a *Defect Coverage* that is often superior to the others, having most of the defects covered more than 60%. Thus, considering that small anomalies far outnumber the large ones (as described in Section 5.1) 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 6.1 reports the masks $\widehat{\Omega}$ over three meaningful images for the three most effective solutions (according to Figures 5.2 and 5.3), 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.

5.5. Computational Times. We measure the time necessary to run Algorithm 2 on a PC mounting an Intel Core i7 3.40GHz CPU. Our Matlab implementation adopts the SPAMS [35] library to perform sparse coding. On this machine, analyzing each image takes about $59\,s$ on average, with a standard deviation of $1.5\,s$. These computational times are well below the time needed to prepare the next sample before SEM acquisition, thus multiple SEM images could be analyzed from the same sample. If needed, the computational times can be further reduced by leveraging a GPU. In fact, the sparse coding of an entire image is highly parallel, as the solution (4.6) can

be executed in parallel over multiple patches. Hence, a GPU implementation of the proposed solution would dramatically reduce the computational times, allowing more exhaustive inspection of each sample.

6. Conclusions. We present an anomaly-detection algorithm that can successfully detect defects in nanostructures. 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, which can promptly raise alerts as soon as the production quality falls below a desired standard, yielding both economical and environmental advantages.

Ongoing work concerns improving the algorithm to handle also changes in the zooming level of these images, as well as new dictionary-learning methods to exploit examples of defects during training.

Acknowledgements. This work has been partially supported by Italian CNR and MIUR organizations through the Flagship Projects Factory of the Future "NanoTWICE" and "Autospin".

REFERENCES

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

[2] *Nanotechnologies - terminology and definitions for nano-objects - nanoparticle, nanofibre and nanoplate*. ISO/TS 27687:2008.

[3] Amir Adler, Michael Elad, Yacov Hel-Or, and Ehud Rivlin, *Sparse coding with anomaly detection*, in Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP), 2013, pp. 1–6.

[4] Seema Agarwal, Joachim H Wendorff, and Andreas Greiner, *Use of electrospinning technique for biomedical applications*, Polymer, 49 (2008), pp. 5603–5621.

[5] Michal Aharon, Michael Elad, and Alfred Bruckstein, *K-svd: An algorithm for designing overcomplete dictionaries for sparse representation*, IEEE Transactions on Signal Processing, 54 (2006), pp. 4311–4322.

[6] Cesare Alippi, Giacomo Boracchi, and Brendt 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.

[7] Xiaolong Bai, Yuming Fang, Weisi Lin, Lipo Wang, and Bing-Feng Ju, *Saliency-based defect detection in industrial images by using phase spectrum*, Industrial Informatics, IEEE Transactions on, 10 (2014), pp. 2135–2145.

[8] Rajendrakumar Suresh Barhate and Seeram Ramakrishna, *Nanofibrous filtering media: filtration problems and solutions from tiny materials*, Journal of membrane science, 296 (2007), pp. 1–8.

[9] Bipul Barua and Mrinal C Saha, *Investigation on jet stability, fiber diameter, and tensile properties of electrospun polyacrylonitrile nanofibrous yarns*, Journal of Applied Polymer Science, 132 (2015), p. 41918.

[10] Decostere Bjorge, Nele Daels, Sander De Vrieze, Pascal Dejans, Tamara Van Camp, Wim Audenaert, Joël Hogie, Philippe Westbroek, Karen De Clerck, and Stijn WH Van Hulle, *Performance assessment of electrospun nanofibers for filter applications*, Desalination, 249 (2009), pp. 942–948.

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

[12] Giacomo Boracchi, Diego Carrera, and Brendt Wohlberg, *Novelty detection in images by sparse representations*, in Proceedings of IEEE Symposium on Intelligent Embedded Systems (IES), 2014, pp. 47–54.

[13] Zdravko I Botev, Joseph F Grotowski, Dirk P Kroese, et al., *Kernel density estimation via diffusion*, The Annals of Statistics, 38 (2010), pp. 2916–2957.

[14] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends® in Machine Learning, 3 (2011), pp. 1–122.

[15] Phil Brodatz, *Textures: A Photographic Album for Artists and Designers*, Peter Smith Publisher, Incorporated, 1981.

[16] Alfred M Bruckstein, David L Donoho, and Michael Elad, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM review, 51 (2009), pp. 34–81.

[17] Christian Burger, Benjamin S Hsiao, and Benjamin Chu, *Nanofibrous materials and their applications*, Annu. Rev. Mater. Res., 36 (2006), pp. 333–368.

[18] Diego Carrera, Giacomo Boracchi, Alessandro Foi, and Brendt Wohlberg, *Detecting anomalous structures by convolutional sparse models*, in Neural Networks (IJCNN), 2015 International Joint Conference on, July 2015, pp. 1–8.

[19] Varun Chandola, Arindam Banerjee, and Vipin Kumar, *Anomaly detection: A survey*, ACM Computing Surveys (CSUR), 41 (2009), p. 15.

[20] Scott Shaobing Chen, David L Donoho, and Michael A Saunders, *Atomic decomposition by basis pursuit*, SIAM journal on scientific computing, 20 (1998), pp. 33–61.

[21] Adrian Chlanda, Janusz Rebis, Ewa Kijeńska, Michal J Wozniak, Krzysztof Roznia-towski, Wojciech Swieszkowski, and Krzysztof J Kurzydlowski, *Quantitative imaging of electrospun fibers by peakforce quantitative nanomechanics atomic force microscopy using etched scanning probes*, Micron, 72 (2015), pp. 1–7.

[22] Yang Cong, Junsong Yuan, and Ji Liu, *Sparse reconstruction cost for abnormal event detection*, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 3449–3456.

[23] Joseph M Deitzel, James Kleinmeyer, Donovan Harris, and Nora C Beck Tan, *The effect of processing variables on the morphology of electrospun nanofibers and textiles*, Polymer, 42 (2001), pp. 261–272.

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

[25] Anton Formhals, *Process and apparatus for preparing artificial threads*, 1934. US Patent 1,975,504.

[26] Huijun Gao, Changxing Ding, Chunwei Song, and Jiangyuan Mei, *Automated inspection of e-shaped magnetic core elements using k-tsl-center clustering and active shape models*, Industrial Informatics, IEEE Transactions on, 9 (2013), pp. 1782–1789.

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

[28] Wei-Chen Li and Du-Ming Tsai, *Defect inspection in low-contrast lcd images using hough transform-based nonstationary line detection*, Industrial Informatics, IEEE Transactions on, 7 (2011), pp. 136–147.

[29] Xishuang Liang, Tae-Hyung Kim, Ji-Wook Yoon, Chang-Hoon Kwak, and Jong-Heun 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, 209 (2015), pp. 934–942.

[30] Qiegen Liu, Shanshan Wang, Jianhua Luo, Yuemin Zhu, and Meng Ye, *An augmented lagrangian approach to general dictionary learning for image denoising*, Journal of Visual Communication and Image Representation, 23 (2012), pp. 753–766.

[31] Tianbo Liu, Christian Burger, and Benjamin Chu, *Nanofabrication in polymer matrices*, Progress in Polymer Science, 28 (2003), pp. 5–26.

[32] Yue M Lu and Minh N Do, *A theory for sampling signals from a union of subspaces*, Signal Processing, IEEE Transactions on, 56 (2008), pp. 2334–2345.

[33] Christopher Mahoney, Dawn Conklin, Jenora Waterman, Jagannathan Sankar, and Narayan Bhattarai, *Electrospun nanofibers of poly (ε-caprolactone)/depolymerized chitosan for respiratory tissue engineering applications*, Journal of Biomaterials Science, Polymer Edition, (2016), pp. 1–21.

[34] Julien Mairal, Francis Bach, and Jean Ponce, *Sparse modeling for image and vision processing*, Foundations and Trends in Computer Graphics and Vision, 8 (2012), pp. 85–283.

[35] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro, *Online learning for matrix factorization and sparse coding*, The Journal of Machine Learning Research, 11 (2010), pp. 19–60.

[36] Markos Markou and Sameer Singh, *Novelty detection: a review - part 1: statistical approaches*, Signal processing, 83 (2003), pp. 2481–2497.

[37] ———, *Novelty detection: a review - part 2: neural network based approaches*, Signal processing, 83 (2003), pp. 2499–2521.

[38] David E Meyer, Mary Ann Curran, and Michael 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, 43 (2009), pp. 1256–

1263.

[39] Rimvydas Milašius and Jolanta Malašauskienė, *Evaluation of structure quality of web from electrospun nanofibres*, Autex Research Journal, 14 (2014), pp. 233–238.

[40] Shahed Mortazavian, Masoud M Shabestary, Yasser Abdel-Rady I Mohamed, and Gevork 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, 11 (2015), pp. 1334–1345.

[41] Luana Persano, Andrea Camposeo, Cagri Tekmen, and Dario Pisignano, *Industrial upscaling of electrospinning and applications of polymer nanofibers: a review*, Macromolecular Materials and Engineering, 298 (2013), pp. 504–520.

[42] Marco A.F. Pimentel, David A. Clifton, Lei Clifton, and Lionel Tarassenko, *A review of novelty detection*, Signal Processing, 99 (2014), pp. 215–249.

[43] Marek Pokorny, Jiri Rebicek, Jindrich Novak, Adela Kotzianova, Jan Klemes, Jana Ruzickova, and Vladimir Velebny, *Produced nanofibers and technological possibilities of laboratory apparatus 4spin®*, ACC Journal, (2015).

[44] Bernhard Schölkopf, Robert C. Williamson, Alex J. Smola, John Shawe-Taylor, and John C. Platt, *Support vector method for novelty detection*, in Advances in Neural Information Processing Systems, 1999, pp. 582–588.

[45] Hatice Şengül, Thomas L Theis, and Siddhartha Ghosh, *Toward sustainable nanoproducts*, Journal of Industrial Ecology, 12 (2008), pp. 329–359.

[46] Eero P Simoncelli and William 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.

[47] Wee Eong Teo and Seeram Ramakrishna, *A review on electrospinning design and nanofibre assemblies*, Nanotechnology, 17 (2006), pp. R89–R106.

[48] Joel A. Tropp, *Greed is good: algorithmic results for sparse approximation*, IEEE Transactions on Information Theory, 50 (2004), pp. 2231–2242.

[49] Du-Ming Tsai, I Chiang, Ya-Hui Tsai, et al., *A shift-tolerant dissimilarity measure for surface defect detection*, Industrial Informatics, IEEE Transactions on, 8 (2012), pp. 128–137.

[50] Du-Ming Tsai and Jie-Yu Luo, *Mean shift-based defect detection in multicrystalline solar wafer surfaces*, Industrial Informatics, IEEE Transactions on, 7 (2011), pp. 125–135.

[51] Du-Ming Tsai, Shih-Chieh Wu, and Wei-Yao Chiu, *Defect detection in solar modules using ica basis images*, Industrial Informatics, IEEE Transactions on, 9 (2013), pp. 122–131.

[52] Alessio Varesano, Claudia Vineis, Cinzia Tonetti, Diego Omar Sanchez Ramirez, Giorgio Mazzuchetti, Simona Ortelli, Magda Blosi, and Anna Luisa Costa, *Multifunctional hybrid nanocomposite nanofibers produced by colloid electrospinning from water solutions*, Current Nanoscience, 11 (2015), pp. 41–48.

[53] Ki Myoung Yun, Christopher J Hogan, Yasuko Matsubayashi, Masaaki Kawabe, Ferry Iskandar, and Kikuo Okuyama, *Nanoparticle filtration by electrospun polymer fibers*, Chemical Engineering Science, 62 (2007), pp. 4751–4759.

[54] Sara Zali, Fahimeh Jalali, Ali Es-Haghi, and Mojtaba 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, 1002 (2015), pp. 387–393.

[55] Fateme Zeighami, *Developing optically efficient nanofiber coatings inspired by cyphochilus white beetle*, Journal of Industrial Textiles, (2015), p. 1528083715589751.

[56] Bin Zhao, Li Fei-Fei, and Eric P. Xing, *Online detection of unusual events in videos via dynamic sparse coding*, in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 3313–3320.

[57] Mohammad Ziabari, Vahid Mottaghitalab, Scott T. McGovern, and Akbar Khodaparast Haghi, *Measuring electrospun nanofibre diameter: A novel approach*, Chinese Physics Letters, 25 (2008), p. 3071.

[58] Maria Zontak and Israel Cohen, *Defect detection in patterned wafers using anisotropic kernels*, Machine Vision and Applications, 21 (2008), pp. 129–141.

[59] Jana Zujovic, Thrasyvoulos N Pappas, and David L Neuhoff, *Structural texture similarity metrics for image analysis and retrieval*, IEEE Transactions on Image Processing, 22 (2013), pp. 2545–2558.

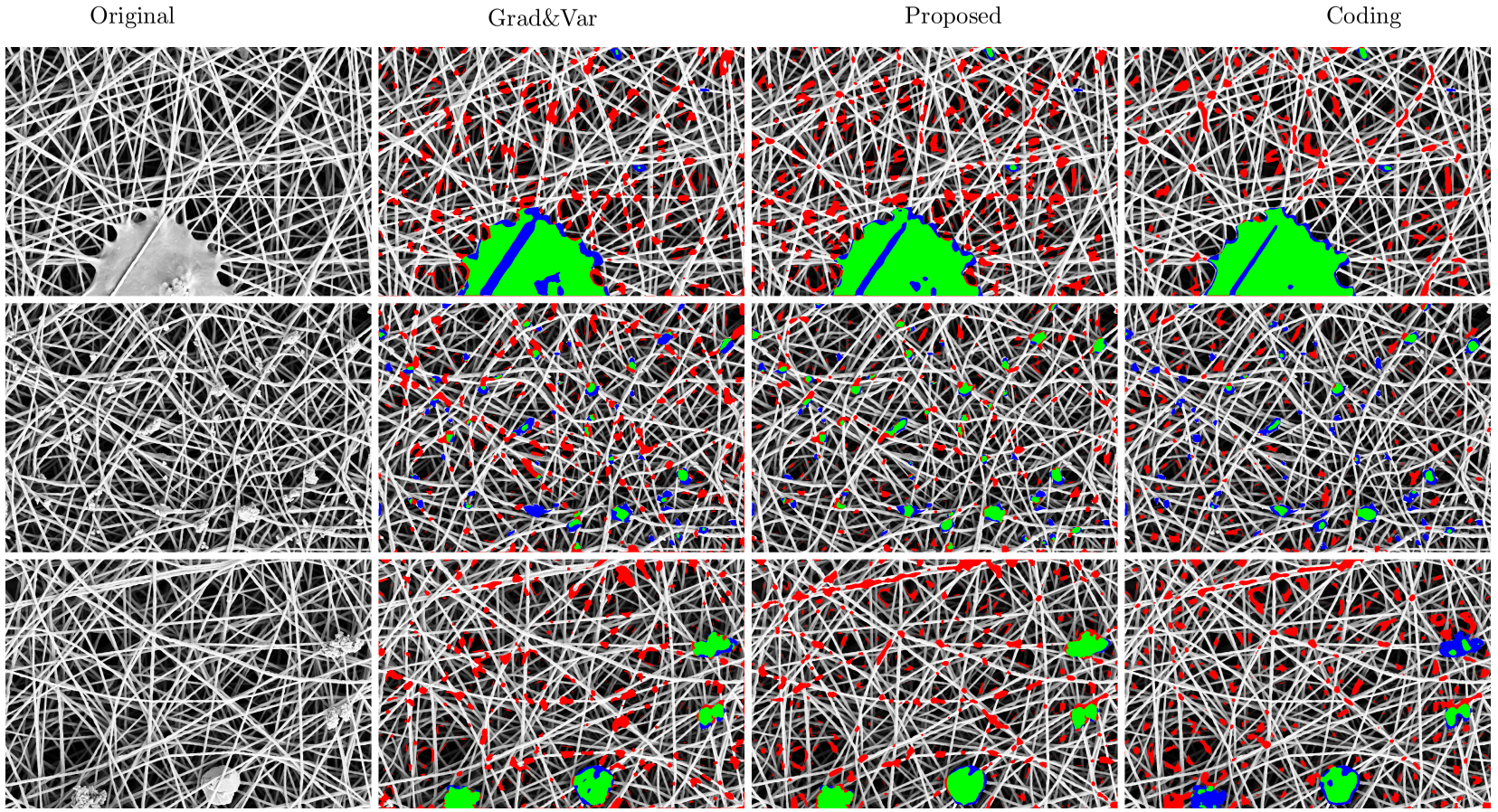


FIG. 6.1. *Examples of anomaly detection on three meaningful SEM images. The first column reports the three original images, while the following ones present the detections obtained from* Grad&Var, Proposed *and* Coding *solutions, respectively. These solutions have been chosen as the best performing ones according to Figures 5.2 and 5.3. The threshold γ has been set as in Figure 5.3, to yield $FPR = 5\%$. Pixels correctly identified as anomalous are marked in green, false positives in red, while false-negatives in blue.*

Recent titles from the IMATI-REPORT Series:

16-01: *Optimal strategies for a time-dependent harvesting problem*, G.M. Coclite, M. Garavello, L.V. Spinolo, February 2016.

16-02: *A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0,* R. Vázquez, April 2016.

16-03: *Defect detection in nanostructures*, D. Carrera, F. Manganini, G. Boracchi, E. Lanzarone, April 2016

Istituto di Matematica Applicata e Tecnologie Informatiche ``*Enrico Magenes",* CNR
Via Ferrata 5/a, 27100, Pavia, Italy
Genova Section: Via dei Marini, 6, 16149 Genova, Italy • **Milano Section:** Via E. Bassini, 15, 20133 Milano, Italy

http://www.imati.cnr.it/