Manufacturing Quality Prediction Using Smooth Spatial Variable Selection Estimator with Applications in Aerosol Jet® Printed Electronics Manufacturing

Yifu Li[1], Hongyue Sun[2], Xinwei Deng[3], Chuck Zhang[4], Hsu-Pin (Ben) Wang[4], Ran Jin[1],

[1] Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA, USA, [2] Department of Industrial and Systems Engineering, University at Buffalo, Buffalo, NY, USA, [3] Department of Statistics, Virginia Tech, Blacksburg, VA, USA, [4] H. Milton Stewart School of Industrial and Systems Engineering, Georgia Tech, Atlanta, GA, USA.

Abstract

Additive manufacturing (AM) has advantages over production cycle time, flexibility, and precision compared with traditional manufacturing. Spatial data, collected from optical cameras or *in situ* sensors, are widely used in various AM processes to quantify the product quality and reduce variability. However, it is challenging to extract useful information and features from spatial data for modeling because of the increasing spatial resolutions and feature complexities due to the highly diversified nature of AM processes. Motivated by the aerosol jet® printing process in printed electronics, we propose a smooth spatial variable selection (SSVS) procedure to extract meaningful predictors from spatial contrast information in high-definition microscopic images to model the resistances of printed wires. The proposed method does not rely on extensive feature engineering, and has the generality to be applied to a variety of spatial data modeling problems. The performance of the proposed method in prediction and variable selection through simulations and a real case study has proven to be both accurate and easy to be interpreted.

Keywords: Additive manufacturing modeling; fused Lasso; printed electronics; spatial variable selection; spatial modeling.

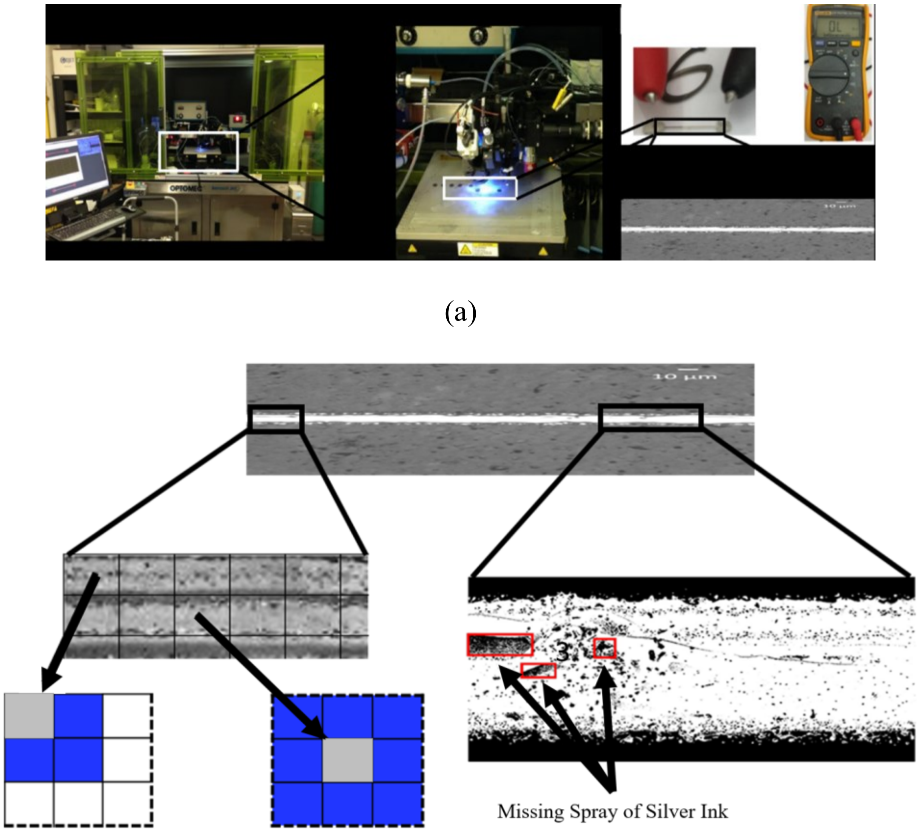
1. Introduction

The challenges of additive manufacturing (AM) processes related to product quality quantification, monitoring and control impede its broader application. Extensive efforts have been made in recent years on quality modeling of AM using *in situ* process information (Rao *et al.*, 2016). Among these efforts, spatial data are widely collected and analyzed in various AM processes (Huang *et al.*, 2015). However, due to the increase of spatial resolution and feature complexity, it is still challenging on efficiently using spatial data in AM quality engineering (Schabenberger *et al.*, 2017). More investigations thus are needed to improve the modeling and interpretations of spatial data in AM quality control. In this paper, we focus on quality modeling of additive-manufactured electronics with spatial data to reflect process conditions.

Along with this direction, our research is motivated by an aerosol jet® printing (AJP) process, which is a direct write type of AM for flexible electronics printing. The basic procedures of the AJP process are as follows. (1) Nanoparticle silver ink is atomized into droplets (mists) by an atomizer first. (2) Then, the mists are delivered by carrier gas to the nozzle. (3) The mists are pushed out of the nozzle at a high velocity onto the surface of printing. More details of the process can be found in Sun *et al.* (2017).

In the AJP process, high-resolution microscopic images are taken on the surfaces of the printed electronics, as shown in Figure 1(a), to measure their quality indirectly. Because the microscope takes the images in a non-contact way, it can prevent potential damages and human errors in measurements caused by traditional contact-based measuring tools, such as multi-meters (Mahajan *et al.*, 2013). These microscopic images, reflecting the distribution of the silver inks on the printed electronics, have been proven to strongly correlate with the printed electronics' resistance and other electronic properties (Sun *et al.*, 2017).

However, there is lack of efficient and systematical ways to identify interpretable features from the raw images to model electronic properties. Instead of performing the complex engineering-driven feature extractions requiring domain knowledge as shown on the right of Figure 1(b), we intend to use spatially correlated predictor generated from image pixel or mesh to directly predict the quality response, as shown on the left of Figure 1(b). Here, a mesh consists of multiple pixels from a certain area in an image with its intensity calculated by the mean intensity of all pixels in the mesh. Specifically, our objective focuses on generation and variable selection methods of spatially correlated predictors (i.e., images in the AJP process) to help identify interpretable features for predicting conducting wires' resistance. Note that the wires with missing segments and infinite resistances are treated as outliers and removed from the modeling.



(a)

(b)

Figure 1. (a). Optomec aerosol jet® system (left), the printing process (middle), and the resistance measurement on printed electronics using a multi-meter and a microscopic image of printed electronics (right). (b). Two distinct ways of generating variables for resistance prediction: spatial predictor extraction with an example of contrast between a pixel and its 8-Connected neighborhoods (defined in Adams *et al.* (1994)) (left) and engineering-driven feature extraction (Sun *et al.*, 2017) (right).

In the literature, there are various works that use spatially correlated predictors for variable selection and modeling. Note that these methodologies are different from the spatial statistics, which typically emphasizes the modeling and analysis of spatially correlated responses (Ripley, 2005). Here our focus is on "spatial-data-in-scalar-out" types of regression, which are also known as the "scalar-on-image" (Kang *et al.*, 2016) regressions for image data. Two of the most commonly used methodologies are feature extraction based modeling, and direct modeling (e.g., treat the intensity of each pixel as a predictor). For feature extraction based modeling, wavelet analysis (Bukkapatnam *et al.*, 1999), homologous features (Li *et al.*, 2015), and Fourier transforms (Li *et al.*, 2017a), etc., can be first used to extract features from raw spatial data. Then, regression models can either use the extracted features as input variables or employ feature reductions (Bai *et al.*, 2018) before performing variable selection and prediction. Kernel-based models are also widely adopted, such as kernel-ridge regression (Vovk, 2013), relevance vector regression (Zheng *et al.*, 2015), etc. The kernel-based model transforms the original datasets into dual forms of various kernel spaces in order to reduce the modeling complexity and change the model linearity for better performances. One potential drawback of methods mentioned above is that sufficient domain knowledge on the corresponding manufacturing processes and a large amount feature engineering are often needed to generate high-quality predictors. As a result, a strategy to directly select individual variables without complex feature extraction procedures or kernel function selection is more useful to directly identify important features, which is called direct modeling method in this paper.

For the direct modeling method, popular approaches include tensor regressions (Li *et al.*, 2013), matrix regression (Zhou *et al.*, 2014), deep learning (Chen *et al.*, 2014), Gaussian process (GP) models (Kang *et al.*, 2016), which were re-engineered to handle "scalar-on-image" regression based on traditional GP models (Colosimo *et al.*, 2015), etc. However, many tensor or matrix regressions are usually built upon the assumption of low-rank approximations (Zhou *et al.*, 2014), which might not always be adequate to various spatial datasets collected in the AM processes (Sun *et al.*, 2017). The models used in deep learning, such as CNN, are relatively complex and require large sample sizes of input data (Luo, 2017). But in a highly flexible AM process, the sample size can become very small (e.g., measured at most in tens or hundreds), since AM often produces "one-of-a-kind" design of products to satisfy personalized needs. Furthermore, complex model structures produced by deep learning can be difficult to generate interpretable features for further diagnosis on product quality (Lou *et al.*, 2012). The use of Gaussian process models also has high computational costs and large sample size requirements, making it difficult for data modeling with a large number of variables and limited sample sizes in AM processes (Tripathy *et al.*, 2016).

In this paper, we focus on using spatial data to study the resistance variation of printed wires due to the missing ink spray defects (black pinholes/regions in the image of Figure 2(b)). It has been proven that increment of such defects significantly increases the resistance of a printed wire (Zhao *et al.*, 2012). It is worth mentioning that: not only in AJP process, various other printing processes for flexible electronics, including flexographic printing (Krebs, 2009), inkjet printing (Haverinen *et al.*, 2010), and gravure printing (Kang *et al.*, 2012), are also suffering from quality issues due to missing ink spray defects, such as short circuit, high energy loss, etc. In this paper, we propose to generate and select spatial predictors directly for accurate quality prediction and good interpretability. The proposed approach is to encourage similar-but-non-identical effects among neighborhood predictors, pursuing the smoothness among neighborhood parameters during variable selection. The underlying motivation is that predictors in the neighborhood are spatially correlated, which would take similar roles (e.g., either be all significant or insignificant) on affecting the response. Thus, the parameter value changes among spatially adjacent predictors are expected to be smooth. For example, in microscopic images of printed wires, the parameter values would gradually decay as the corresponding predictors are far away from the defect regions, which are strongly associated with the response. We refer to such a phenomenon as the *neighborhood effect*. Such an effect is inherent to many spatially adjacent predictors in spatial data modeling (Kang *et al.*, 2016). In the AJP process, the spatial contrast information generated from the microscopic images can directly predict resistances of conducting wires through the proposed spatial modeling framework.

For predictor generation, three different types of spatial contrast information (Adams *et al.*, 1994) will be used to form spatial predictors: (a) contrast (difference) between each mesh's intensity and the overall image's mean intensity, (b) contrast (difference) between each mesh's intensity and its 8-Connected neighborhood meshes' mean intensity, and (c) contrast (difference) between each mesh's intensity and the largest row-wise mean intensity among the image. Comparing with the past AJP-related quality quantification work (Sun *et al.*, 2017), an advantage of using spatial predictors is that it requires neither heavy feature engineering nor manually labeled quality indices (e.g., silver ink over-spray) for product quality quantification. An example of spatial predictors from the microscopic images is shown in Figure 2, which is motivated by the work of Adams *et al.* (1994) to reflect local intensity variation.

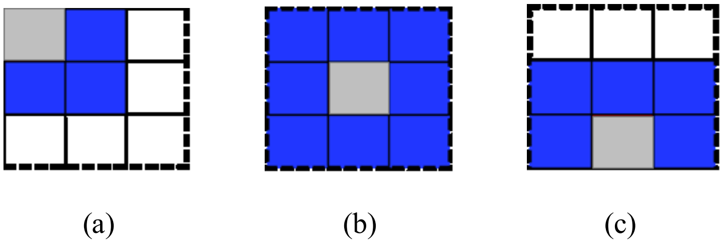


Figure 2. (a)-(b). The 8-Connected neighborhoods of the underlying mesh are its surrounding meshes.

In total, we would remark that the proposed model, smooth spatial variable selection (SSVS), is easy to interpret since the spatial predictors are directly generated from the spatial data, such as the contrast information among pixels. Both 1-D and 2-D simulations show that SSVS better identifies inherent structures among model parameters, by encouraging their smoothness. The real case study shows that given the highly personalized nature of AM processes, SSVS under small training sample sizes offers smaller prediction errors on testing data and yields meaningful variable selection results. This gives SSVS the potential to be widely adopted for quality modeling in various AM processes and spatial data modeling problems.

The remaining part of this paper is organized as follows. In Section 2, the proposed modeling framework, SSVS, and the estimation algorithm are introduced. In Section 3, the simulation studies for SSVS comparing with the benchmark models were introduced. The benchmark models in this research mainly include Lasso, fused Lasso, and matrix regression, where Lasso is known for its variable selection under sparse model parameter spaces, fused Lasso and matrix regression are known for their capability of recovering certain types of model parameter structures. Details of these models and their comparisons with the proposed method will be discussed later. Section 4 shows a real case study of SSVS using the AJP process. Finally, Section 5 summarizes the proposed methodology and discusses the future work.

2. Smooth Spatial Variable Selection for Models with Spatial Predictors

2.1. Model Formulation

Without loss of generality, we consider a linear regression between a single response variable y, such as the resistance of a wire, and 2-D spatial predictors $\mathcal{X} = \begin{pmatrix} \mathcal{X}_{1,1} & \cdots & \mathcal{X}_{1,m} \\ \vdots & \ddots & \vdots \\ \mathcal{X}_{l,1} & \cdots & \mathcal{X}_{l,m} \end{pmatrix}$, a $l \times m$ matrix, where each predictor can correspond to a pixel or a mesh location for microscopic images of conducting wires. Given that each predictor in the matrix $\mathcal{X}$, such as $\mathcal{X}_{i,j}$, is spatially located by its index (i,j) in a 2-D space, we can vectorizie $\mathcal{X}$ and the corresponding $l \times m$ model parameter matrix B without losing their spatial information. Specifically, $tr(\mathcal{X}^T B)$ defines the summation of the pair-wise multiplication of the corresponding elements in matrix $\mathcal{X}$ and B. Therefore, if one vectorizie $\mathcal{X}$ and B, both by appending the columns sequentially into $\boldsymbol{x} = \left(x_{1,1}, \cdots, x_{l,m}\right)^T$ and $\boldsymbol{\beta} = \left(\beta_{1,1}, \cdots, \beta_{l,m}\right)^T$, into $(lm) \times 1$ vectors, each pair of elements in $\mathcal{X}$ and B to be multiplied remain unchanged. As a result, the summation of pair-wise multiplication defined by $tr(\mathcal{X}^T B)$ remains equivalent to performing vector-wise multiplication $\boldsymbol{x}^T \boldsymbol{\beta}$. Thus, we can present the 2-D model formulation in 1-D format

$$y = tr(\mathcal{X}^T B) + \varepsilon = \boldsymbol{x}^T \boldsymbol{\beta} + \varepsilon, \tag{1}$$

where $\varepsilon \sim N(0, \sigma^2)$ is independently and identically distributed (i.i.d) error term. It is worth to pointing out that such an indexing of predictors can be easily generalized to data organized in a higher dimensional format, such as 3-D colored image data and even 4-D spatial-temporal video data (Yan *et al.*, 2015), where the neighborhood effect can be enforced among spatially adjacent model parameters.

For conducting variable selection for the model in (1), various methods are proposed in the literature such as Lasso (Tibshirani, 1996), which is a commonly used method, but not designated for spatial variable selections (Tibshirani *et al.*, 2005). Tibshirani *et al.* (2005) introduced fused Lasso to penalize the difference between model parameters, such encouraged smoothness of variables. The applications of fused Lasso for 2-D spatial datasets are discussed in the work of generalized Lasso (Tibshirani *et al.*, 2010), which can enforce model parameter structures using designated penalty matrices. Both fused Lasso and generalized Lasso mainly focus on piece-wise constant variable selection, that is, the similarity of two adjacent model parameters is encouraged by penalizing their differences in the model. Furthermore, the generalized Lasso does not enforce the overall sparsity of model parameters, which is not suitable for high-dimensional modeling under the sparsity assumption of model parameters. In the spirit of both fused Lasso and generalized Lasso, we propose smooth spatial variable selection (SSVS) to encourage the similarity and smoothness of spatially adjacent model parameters. For elaboration, we denote the parameter of a predictor $\mathcal{X}_{i,j}$ in $\mathcal{X}$ as $\beta_{i,j}$, and define the r -neighborhoods of $\beta_{i,j}$ as the set of parameters $\left\{\beta_{o,u}\colon \sqrt{(i-o)^2+(j-u)^2} \le r\right\}$, where the distance of any two parameters is measured by Euclidean distance. The SSVS is to encourage that if the value of $\beta_{i,j}$ is non-zero (i.e., the i,j-th predictor is important), the parameters $\beta_{o,u}$ in its r-neighborhoods also tend to have non-zero values. Furthermore, the magnitudes of $\beta_{o,u}$ might gradually decay (less significant) as the locations of the corresponding predictors are further away from $x_{i,j}$. The proposed SSVS does not only enforce the similarity among neighborhood model parameters, but also make the similarity enforced tunable through adjusting the value of r in model estimation (see Section 2.2). The larger the value of r considered for the variable selection, the smoother that the changes among neighborhood parameters will be enforced.

2.2. The Smooth Variable Selection Estimator

Suppose that $\boldsymbol{y} = (y_1, \dots, y_n)^T$ is an $n \times 1$ response vector observed for the y in (1), containing resistances of n printed wires, and $X = \begin{pmatrix} x_{1,(1,1)} & \cdots & x_{1,(l,m)} \\ \vdots & \ddots & \vdots \\ x_{n,(1,1)} & \cdots & x_{n,(l,m)} \end{pmatrix}$ is a $n \times lm$ predictor matrix observed for $\boldsymbol{x}$ in (1), containing the values of the spatial predictors from n printed wires' microscopic images. By incorporating SSVS, the model estimation in (1) can be solved by minimizing

$$min \, \frac{1}{2} \|\boldsymbol{y} - X\boldsymbol{\beta}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \lambda_2 \|S\boldsymbol{\beta}\|_1, \tag{2}$$

where $S = \begin{pmatrix} S_{(1,1),(1,1)} & \cdots & S_{(1,1),(l,m)} \\ \vdots & \ddots & \vdots \\ S_{(l,m),(1,1)} & \cdots & S_{(l,m),(l,m)} \end{pmatrix}$ is an $lm \times lm$ weight matrix with its component as

$$S_{(i,j),(o,u)} = \begin{cases} -\frac{exp\left(-\sqrt{(i-o)^2+(j-u)^2}\right)}{\sum_{z=1}^{r} exp(-z)} & \forall\, (o,u) \in \{\sqrt{(i-o)^2 + (j-u)^2} \leq r, i \neq o, j \neq u\} \\ 0 & \forall\, (o,u) \in \{\sqrt{(i-o)^2 + (j-u)^2} > r, i \neq o, j \neq u\} \\ -\sum_{o \neq i} \sum_{u \neq j} S_{(i,j),(o,u)} & o = i, u = j \end{cases}.$$

Here the smoothness is achieved by penalizing the difference between weighted parameter $S_{(i,j),(i,j)}\beta_{i,j}$, and the sum of its weighted neighborhood parameters $\sum_{o \neq i} \sum_{u \neq j} S_{(i,j),(o,u)}\beta_{o,u}$, which are no more than r distance away from $\beta_{i,j}$ measured by Euclidean distance. The weights are determined by the normalized exponential distance weighting function $\frac{exp\left(-\sqrt{(i-o)^2+(j-u)^2}\right)}{\sum_{z=1}^{r} exp(-z)}$, which ensures that the weight on each neighborhood parameter is in [0,1] with the denominator/normalizing factor $\sum_{z=1}^{r} exp(-z)$. The same r is considered for all predictors when estimating the model parameters. It is important to point out that when there is only one neighborhood considered ($r = 1$) for the estimator, $\|S\boldsymbol{\beta}\|_1$ only penalizes the differences of two consecutive model parameters, which are one distance away from each other. Hereafter, SSVS with only one neighborhood is the same as fused Lasso. Besides on spatial predictors from image data, the matrix S can further enforce the smoothness on predictors arranged in irregular grids. For example, we can use the adjacency information, such as a 0-1 binary indicator, instead of 2-D Euclidean distances among predictors to calculate $S_{(i,j),(o,u)}$.

2.3. Parameter Tuning and Model Estimation

In (2), we consider using extended Bayesian information criterion (EBIC) (Chen *et al.*, 2012) to select tuning parameters r, λ_1, and λ_2. Here, EBIC is used because traditional model selection criterions such as Bayesian information criterion (Schwarz, 1978) and Akaike information criterion (Akaike, 1998) do not have consistent and satisfying model selection when the number of variables is much larger than the sample size. EBIC leverages both the number of unknown parameters to estimate and the resulted model complexity for model selection. As a result, it has been shown that EBIC can yield more accurate and consistent modeling performance for high dimensional modeling under small sample sizes (Chen *et al.*, 2012).

The objective function in (2) is a quadratic function, which can be solved by standard quadratic programming algorithms, such as Newton method and interior point method. However, for a modeling problem with a high dimension of predictors, taking the second derivation required by some optimization methods is computationally intensive. To address this issue, we adopted the split Bregman algorithm, which is a type of alternating direction method of multipliers (ADMM) algorithm, which will quickly converge under l_1 norm (Ye *et al.*, 2011).

Ye *et al.* (2011) proposed a general framework for solving the constrained optimization problem as

$$min\, V(\boldsymbol{\beta}) + \lambda_1 \|\boldsymbol{a}\|_1 + \lambda_2 \|\boldsymbol{b}\|_1,$$

$$subject\ to\text{:}\quad \boldsymbol{a} = \boldsymbol{\beta}, \boldsymbol{b} = S\boldsymbol{\beta},$$

which has the augmented Lagrangian function as

$$L(\boldsymbol{\beta}, \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{u}, \boldsymbol{v}) = V(\boldsymbol{\beta}) + \lambda_1 \|\boldsymbol{a}\|_1 + \lambda_2 \|\boldsymbol{b}\|_1 + \langle \boldsymbol{u}, \boldsymbol{\beta} - \boldsymbol{a} \rangle + \langle \boldsymbol{v}, S\boldsymbol{\beta} - \boldsymbol{b} \rangle + \frac{\mu_1}{2} \|\boldsymbol{\beta} - \boldsymbol{a}\|_2^2 +$$

$$\frac{\mu_2}{2} \|S\boldsymbol{\beta} - \boldsymbol{b}\|_2^2,$$

where $V(\boldsymbol{\beta})$ is a convex function, $\langle \cdot, \cdot \rangle$ is the inner product of two vectors, $\boldsymbol{u}$ and $\boldsymbol{v}$ are dual variables corresponding to the two linear constraints, and μ_1 and μ_2 are positive augmented Lagrangian parameters controlling the speed of optimization. The convergence of the algorithm is guaranteed.

3. Simulation Studies

In the simulation studies, we consider two simulations with a 1-D format of predictors and a 2-D format of predictors to systematically evaluate the prediction and variable selection performance of the proposed SSVS estimator. The SSVS estimator will be compared with three benchmark models, including Lasso, fused Lasso, and nuclear-norm based matrix regression (for the 2-D dataset). Other models, such as CNN and Gaussian process models, were not selected as benchmark models due to their large sample size requirements in model estimation (Tripathy *et al.*, 2016).

3.1 2-D Simulation Study of SSVS

We created a simulation study with 2-D spatial predictors to illustrate the benefits of SSVS on 2-D datasets. To ensure the generality of the simulation results and showcase the importance of having a smoothness-tunable variable selection process, we assumed that model parameters, and predictors' mean values and covariance all had varying spatial correlations across different replications respectively. Furthermore, we intentionally designed the generation of the parameter smoothness to make the decay pattern of model parameters different from the weights assigned in the smoothing matrix S of Equation (2). As a result, a random decay pattern on model parameters will be chosen for each replication of the simulation.

To generate the spatially correlated covariance of the spatial predictors, we defined the covariance between variable $x_{i,j}$ and $x_{o,u}$ on a 2-D space with an exponential decay function $\sigma_{i,j;o,u} = cov(x_{i,j}, x_{o,u}) = \frac{1}{\tau^{\sqrt{(i-o)^2+(j-u)^2}}}$, where the covariance shrinkage parameter (τ) had the value of 5. Using this way, the covariance of variables, which were spatially close to each other, had larger correlations. Similarly, to generate the means of variables, we first randomly selected five predictors as the centers of the defects, which were bounded by radius φ following a discrete uniform distribution of {3, 4, 5} to simulate defects with different sizes. Then the spatially correlated mean of each predictor within the defect area having the center at predictor $x_{i,j}$ was defined as $\mu_{o,u} = 255\left(1 - \frac{1}{\omega^{\sqrt{(i-o)^2+(j-u)^2}}}\right) \forall\ (o,u) \in \{\sqrt{(i-o)^2 + (j-u)^2} \leq \varphi, o \neq i, u \neq j\}$, where the shrinkage parameter (ω) followed a discrete uniform distribution of {1.1, 1.2, 1.3, 1.4, 1.5}. Here, we define defect areas, which have significantly lower intensities in microscopic images (Figure 1), as the regions that were not uniformly sprayed with silver ink in the printing process. Different values of ω could simulate defect areas with different intensity shrinkage rates from the centers of the defects. For predictors outside of the defect area, the means were simply defined as $\mu_{o,u} = 255\ \forall\ (o,u) \in \{\sqrt{(i-o)^2 + (j-u)^2} > \varphi, o \neq i, u \neq j\}$. Such a mean value generation approach ensured that the means of predictors within defect areas, which were darker than non-defect areas, were assigned with values smaller than 255. For the means of predictors not within the defect areas, they were assigned with 255 (the largest value/brightest color in a grey-scale image). As result, each predictor $x_{o,u}$ within each image sample followed i.i.d. $N(\mu_{o,u}, \sigma_{i,j;o,u})$, with one defect area existing. We repeatedly used such methods to generated image samples containing five defect areas selected previously.

To generate spatially correlated model parameter, we assumed that only predictors within the defect areas having non-zero values. Here, we used the same five predictors select previously as the centers of five defect areas and the following equation to generate the parameters of variables within the defect areas $\beta_{o,u} = \frac{\gamma}{\omega^{\sqrt{(i-o)^2+(j-u)^2}}}\ \forall\ (o,u) \in \{\sqrt{(i-o)^2 + (j-u)^2} \leq \varphi, o \neq i, u \neq j\}$, where the defect center's parameter value (γ) followed a discrete uniform distribution of {1, 2, 3, 4, 5}. As the variable within the defect area was further away from the defect center, it had a smaller parameter value. For parameters outside of the defect areas, they were assigned with zeros ($\beta_{o,u} = 0\ \forall\ (o,u) \in \{\sqrt{(i-o)^2 + (j-u)^2} > \varphi, o \neq i, u \neq j\}$). Eventually, the response was generated based on (1) with the values generated above and ε following i.i.d. $N(0,1)$.

Similar to the simulation settings in Zhou *et al.* (2014), we used the above methods to generate 500 64 × 64 image samples with an example shown in Figure 3(a) and a set of 64 × 64 2-D model parameters shown in Figure 3(b). To evaluate the consistency of the results, we simulated 100 replications for each simulation scenario. Within each replication, we randomly used 90% of simulated samples as the training dataset and the rest 10% as the testing dataset (Zeng *et al.*, 2016). The tuning parameters were selected based on EBIC for Lasso and SSVS, and BIC for matrix regression (Zhou *et al.*, 2014). The simulation results including the testing data prediction accuracy in root mean squared errors (RMSEs) and the corresponding standard errors (SEs) are presented in Table 1. The value shown in bold is the best result (smallest error) obtained from different models and SSVS achieved the lowest error with r = 2. The variable selection results are presented in Table 2 based on variable selection accuracies (ACCs=$\frac{True\ Negative+True\ Positve}{Total\ Amount\ of\ Predictors}$) and Figure 3. The modeling assumptions were validated by residual plots in supplementary materials.

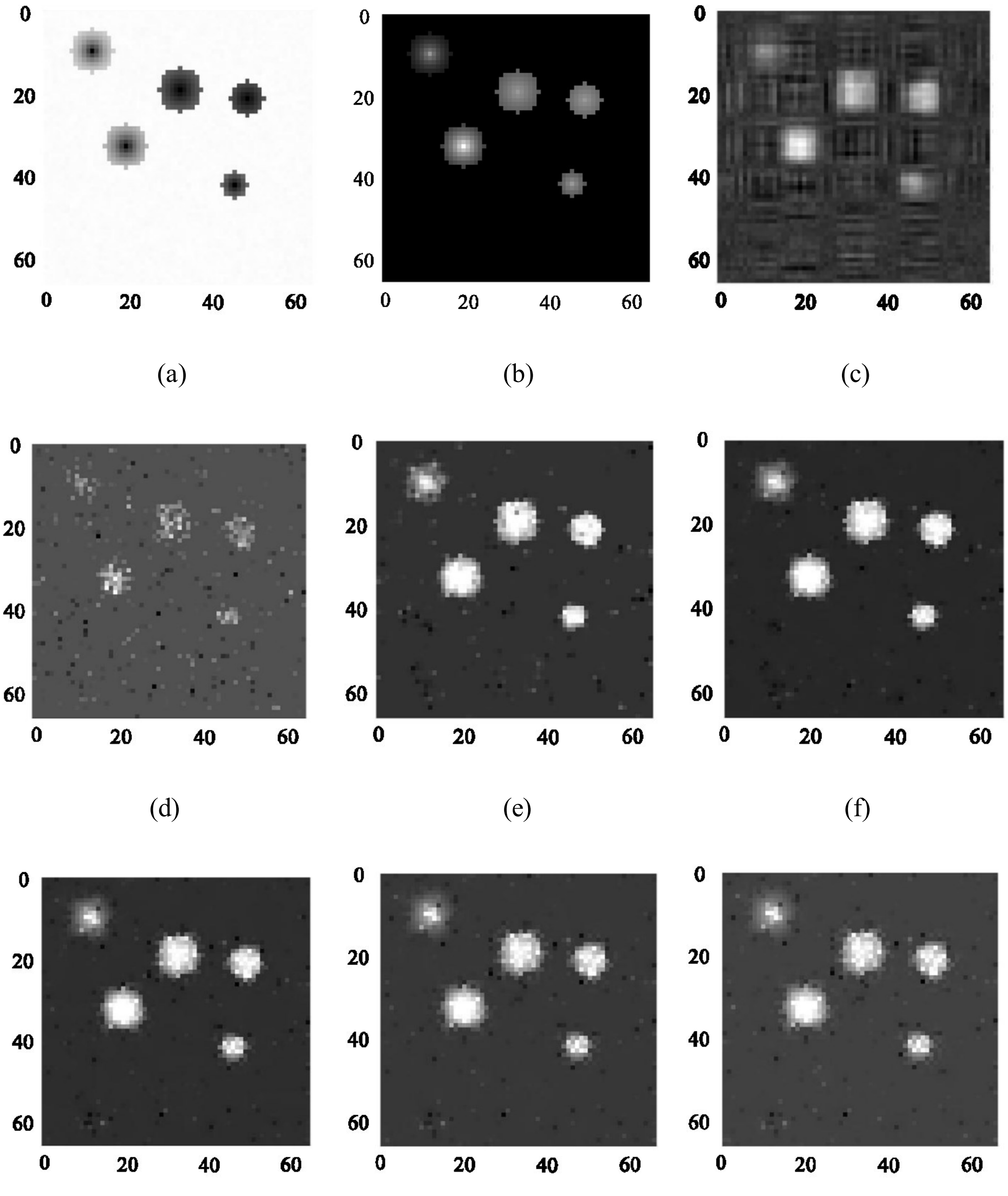
Table 1. Testing RMSEs over 100 replications for the 2-D simulation dataset (standard errors in parenthesis).

	Benchmark Models			Proposed Model
	Matrix Regression	SSVS (r = 0) / Lasso	SSVS (r = 1) / Fused Lasso	**SSVS (r = 2)**
RMSEs	23.56 (<0.00)	33.32 (0.34)	11.07 (0.18)	**10.03** **(0.12)**

Table 2. Average variable selection accuracies (ACCs) over 100 replications for the 2-D simulation dataset (standard errors in parenthesis).

	Benchmark Models			Proposed Model
	Matrix Regression	SSVS (r = 0) / Lasso	SSVS (r = 1) / Fused Lasso	**SSVS (r = 2)**
RMSEs	0.48 (<0.01)	0.89 (<0.01)	0.96 (<0.01)	**0.97** **(<0.01)**

From Table 1, Table 2, and Figure 3, it is clear that Lasso, which did not consider the neighborhood effect, performed the worst in prediction. It also had a large number of misdetections on significant model parameters (insignificant parameters misidentified as significant) during variable selection (Figure 3(d)). Furthermore, we can see that the matrix regression method, which relied on low-rank approximation, did not recover the parameter structures as good as SSVS did in this simulation (Figure 3(c)), and had a significantly lower prediction accuracy. For results of using SSVS, using one neighborhood (fused Lasso) had a larger prediction errors and more misdetections on significant model parameters in variable selection (Figure 3(e)). However, increasing the size of neighborhoods enforced in SSVS may significantly improve the modeling results. When there were more than two neighborhoods considered in variable selection, the prediction error and misdetections in variable selection gradually increased. This simulation did not only show the values of SSVS in spatial variable selection problems, but also emphasize the importance of varying the size of neighborhoods considered for SSVS in order to achieve the best modeling performance.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 3. (a). An example of 2-D predictors simulated. (b). The 2-D model parameters simulated (underlying true parameters). (c). Variable selection result using matrix regression. (d). Variable selection result using Lasso. (e)-(i). Variable selection result using SSVS considering 1-5 neighborhoods (2 being the optimal).

3.2 1-D Simulation Study of SSVS

Additionally, we created a 1-D data simulation to test the performance of SSVS by varying the sample size of input data (n), the number of model parameters (p), and the covariance of predictors controlled by covariance shrinkage parameter (τ) defined later in this simulation. The sample size had the values of $n = 40$ and $n = 400$, the number of predictors (p) had values of $p = 4000$ and $p = 8000$, and the covariance shrinkage parameter (τ) had values of 1.01 and 1.05. Based on τ, the covariance between two variables was generated as $\sigma_{i+d,i} = cov(x_{i+d}, x_i) = cov(x_{i-d}, x_i) = \frac{1}{\tau^d}$, where x_{i-d} and x_{i+d} were the variables d distances before or after x_i. Then, covariance among variables would reflect neighborhood shrinkage effects, i.e., the larger the d, the smaller the covariance between x_{i+d} and x_i or x_{i-d} and x_i was. Such a neighborhood shrinkage effect was indeed observed in our case study dataset when we sort the spatial predictors based on their contrast values from small to large. In total, we had 8 simulation scenarios. The model parameters were generated from a uniformly distributed random combination of piece-wise constant functions and exponential decay functions (see Figure 4(a)). If a piece-wise constant function was used, then 100, 150 or 200 parameters in series with equal and non-zero parameter values were generated, and the values were selected from a discrete uniform distribution of {-3, -2, -1, 0, 1, 2, 3}. Similarly, if an exponential decay function was used, 100, 150 or 200 parameters in series were generated using $\beta_{i-d} = \beta_{i+d} = \frac{\beta_i}{\theta^d}$, so that parameters of each series formed a symmetrical curve mimicking the shrinkage effect on parameters' values. Here, β_i was the parameter in the center of each series of parameters and followed a discrete uniform distribution of {-3, -2, -1, 0, 1, 2, 3}. β_{i-d} and β_{i+d} were the parameters d distances before or after β_i. θ was the shrinkage parameter following a discrete uniform distribution of {1.01, 1.02, 1.03}, which generated a symmetrically decaying parameter curve. Finally, the response was generated based on (1) with values generated above and ε following i.i.d. $N(0, 1)$. The motivation to generate the model parameters containing two different structures was to test if SSVS could better recover the inherent model structure when both piece-wise constant and smooth model parameter structures exist.

Similar to the previous simulation, we used the 90%-10% random data split for training and testing within each replication, with 100 replications performed in total. The tuning parameters were selected based on EBIC. Due to the 1-D nature of the underlying dataset, matrix regression was not used for this simulation study. We varied the neighborhood value r from 1 to 5 in SSVS (1 being fused Lasso) and reported only the best result for SSVS with r > 1. Table 3 reports average RMSEs of prediction based on the testing data and the corresponding SEs (shown in parenthesis) over 100 replications for each scenario. We refer to Lasso, where the neighborhood effect was not considered as $r = 0$. The variable selection results are presented in Table 4 based on variable selection accuracies (ACCs=$\frac{True\ Negative+True\ Positve}{Total\ Amount\ of\ Predictors}$). The values shown in bold are the best results (smallest errors) obtained under each data generation scenario across different models. The modeling assumptions were validated by residual plots in the supplementary materials. The first observation is that SSVS performed significantly better than Lasso did in all scenarios. Within SSVS models, when the sample size was small (n = 40), a larger r offered better prediction accuracy, and when the sample size (n = 400) was large, a smaller r offered better prediction accuracy (SSVS with r = 1). The result indicates that emphasizing the neighborhood effect of variables can result in a better performance when the sample size is very limited. However, we also realize when the sample size reduced to only 40, the prediction accuracy of SSVS became much worse, which had approximately 10 times larger RMSE comparing with the "n = 400" case. This means that we should always increase the sample size as much as possible for an ideal prediction performance when the model dimension grows very large. Lastly, we can see that SSVS was very robust to the variation on the covariance of adjacent model predictors, comparing with Lasso. This suggest that SSVS can effectively handle the data collinearity by considering model parameter smoothness.

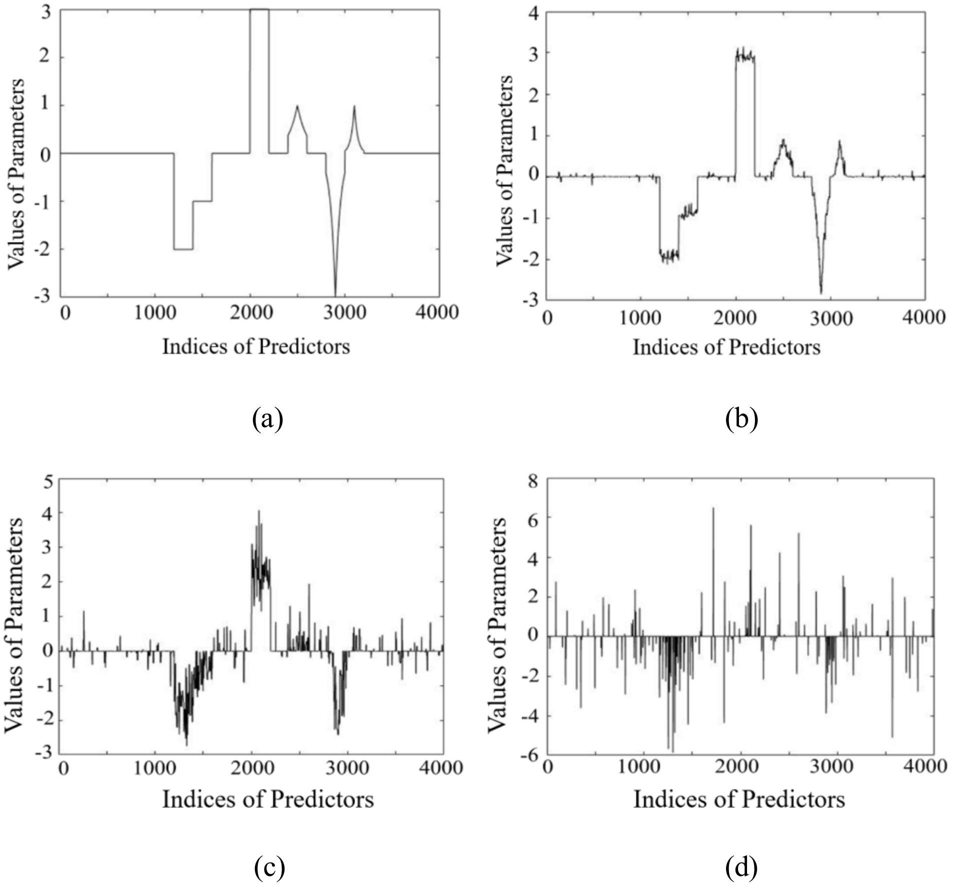
Table 3. Average RMSEs over 100 replications for testing data in 1-D simulation studies.

p	Size of Neighborhoods for SSVS	τ = 1.01		τ = 1.05	
		n = 40	n = 400	n = 40	n = 400
4000	SSVS (r = 0) / Lasso	145.29 (8.48)	14.32 (0.67)	199.55 (9.07)	26.15 (1.12)
	SSVS (r = 1) / Fused Lasso	31.57 (2.38)	**1.42** **(0.02)**	61.41 (4.04)	**1.48** **(0.03)**
	SSVS (r > 1)	**30.49 (r = 3)** **(2.16)**	1.52 (r = 3) (0.03)	**55.23 (r = 4)** **(3.58)**	1.55 (r = 2) (0.03)
8000	SSVS (r = 0) / Lasso	342.21 (15.59)	33.15 (1.42)	327.54 (13.41)	64.95 (2.72)
	SSVS (r = 1) / Fused Lasso	185.84 (11.68)	**2.01** **(0.04)**	229.56 (9.83)	**2.38** **(0.05)**
	SSVS (r > 1)	**176.28 (r = 4)** **(10.57)**	2.11 (r = 2) (0.04)	**209.87 (r = 5)** **(9.61)**	2.54 (r = 2) (0.06)

Table 4. Average variable selection accuracies (ACCs) over 100 replications for the 1-D simulation dataset (standard errors in parenthesis).

p	Size of Neighborhoods for SSVS	τ = 1.01		τ = 1.05	
		n = 40	n = 400	n = 40	n = 400
4000	SSVS (r = 0) / Lasso	0.83 (0.01)	0.89 (<0.01)	0.83 (9.07)	0.89 (0.01)
	SSVS (r = 1) / Fused Lasso	0.96 (<0.01)	**1.00** **(<0.01)**	0.92 (0.01)	**1.00** **(<0.01)**
	SSVS (r > 1)	**0.97 (r = 3)** **(<0.01)**	**1.00 ($All\ r$)** **(<0.01)**	**0.93 ($All\ r$)** **(0.01)**	**1.00 ($All\ r$)** **(<0.01)**
8000	SSVS (r = 0) / Lasso	0.84 (<0.01)	0.88 (<0.01)	0.84 (<0.01)	0.88 (<0.01)
	SSVS (r = 1) / Fused Lasso	**0.87** **(0.01)**	**1.00** **(<0.01)**	0.82 (0.01)	**1.00** **(<0.01)**
	SSVS (r > 1)	**0.87 (r = 5)** **(0.01)**	**1.00 ($All\ r$)** **(<0.01)**	**0.83 (r = 5)** **(0.01)**	**1.00 ($All\ r$)** **(<0.01)**

As a visual illustration of variable selection, we present the results from the same dataset generated with 4000 predictors. Figure 4(a) shows the true parameters generated and Figure 4(b) shows the parameters estimated by SSVS when there were more than one neighborhoods considered. By comparing the Figures, it is clear that when we considered more neighborhoods, SSVS recovered the original parameters significantly better comparing with SSVS only considering one neighborhood (Figure 4(c)), and Lasso, which did not consider the neighborhood effect (Figure 4(d)).



(a) (b)

(c) (d)

Figure 4. (a). True values of the simulated parameters. (b). The parameters learned considering more than one neighborhoods using SSVS. (c). The parameters learned considering one neighborhood using SSVS (fused Lasso). (d). The parameters learned considering no neighborhood (Lasso).

4. Case Study of the AJP for Printed Electronics

In this case study, we model the resistance for conducting wires printed by the AJP process. Single-layer silver nanoparticle wires were printed by varying several process setting parameters, including atomizer power voltage, process speed, gas flow rate, sheath gas flow rate and ink volume. In the end, 35 wires with microscopic images taken on their surfaces were produced and the corresponding resistances were measured. The primary defect type we are focusing on in this study is the missing spray of the silver inks.

Such defects can be seen as black pinholes/regions in Figure 5, which significantly increase the resistance of printed wires. A known cause of the missing ink spray is the inappropriate compositions of silver nanoparticle inks (Zhao *et al.*, 2012). Some recent research focuses on adding novel carbon nanotubes (CNTs), which can bridge the pinholes/defects, into silver nanoparticle ink to improve the printing quality (Oh *et al.*, 2008; Zhao *et al.*, 2012). Additionally, other process settings, including the sheath gas flow rate, the aerosol gas flow rate, and the stage speed, have been proven to affect the wire resistance and potentially contribute to missing ink spray issue (Sun *et al.*, 2017).

4.1. Contrast Generation on Microscopic Images

A high-resolution microscopic image (741×19491 pixels), combined from 10 consecutive microscopic images, was taken on the conducting wire having a small area (0.5 mm×10 mm). x. It can be inefficient and computationally expensive to use one predictor to represent such a small region for product quality characterization. For the ease of future modeling scale-up, we divided the raw image (Figure 5(a)) into rectangular meshes and averaged the intensities of all the pixels in each mesh (Figure 5(b)). This approach was also called average pooling operation with non-overlapping windows (Boureau *et al.*, 2010), and we referred images produced from the pooling as pooled images. When determining the optimal resolution for images after pooling, we first trained different Lasso models using different sets pooled images in resolutions of 10×100, 20×200, 30×300, 40×400, and 50×500 based on the underlying dataset as a pilot study. The model trained using the resolution setting of 20×200 provided us the best prediction results on the testing data. As a result, each pooled images has a resolution of 20×200, with each mesh representing a 50 μm×25 μm regions.

(a) (b)

Figure 5. (a). An original grey-scale image. (b). A newly formed image with lower resolution.

For the case study, we considered three types of contrasts, which could be easily generated from the images without engineering domain knowledge. The first set of 20×200 predictors was generated with the contrasts between each mesh's intensity and the overall image's mean intensity. The second set of 20×200 predictors was generated with the contrasts between each mesh's intensity and the mean intensity of the meshes in its 8-Connected neighborhood. Lastly, the third set of 20×200 predictors was generated with the contrasts between each mesh's intensity and the largest row-wise mean intensity. In an AJP process, this "brightest" region in the microscope image was the outcome of the most silver ink spray and could be considered as the most conductive path of the wire. However, one issue is that defect locations can vary significantly from one wire to another (dark regions in Figures 6), so that it was meaningless to fix each predictor to one corresponding location of images. Therefore, we sorted the values within each set of predictors from low to high and re-organized the predictors in quantiles. As a result, we expected that predictors at certain quantiles (e.g., high quantiles representing high contrast regions) were more strongly associated with wire resistance. Furthermore, the neighborhood effect among the model parameters of neighborhood quantiles preserves, which means that the changes of parameter values among neighborhood quantiles should be smooth. As a result, we can apply the 1-D version of SSVS model based on spatial predictors sorted in quantiles.

We adopted high dimensional ordinary least squares projection (HOLP) by Wang *et al.* (2015) to further filter the 12000 (3×20×200) predictors down to 6000 predictors. Here, the HOLP method is a screening method, which is built upon the ordinary least squares (OLS) estimator. Comparing with regularized regressions, HOLP can quickly screen the number variables in the original variable space down to a relatively low level so that the computational efficiency of SSVS can be further enhanced. However, we consider that determining the appropriate percentage of variables to preserve through HOLP can become a valuable future work. One thing worth emphasizing on dimension reduction in this work is that there are other popular methods, such as principal component analysis (PCA) (Jin *et al.*, 2000), to reduce the dimension of the data. However, different from HOLP, methods like PCA are widely known as "black box" approaches, which do not preserve the original variables of data, hence are significantly detrimental to the interpretability of the modeling and variable selection efforts.

4.2. Numerical Results and Explanations

We performed 100 replications for both benchmarks (Lasso and nuclear-norm based matrix regression) and SSVS estimator under different neighborhood settings. For fair comparison, the matrix regression, which already incorporated low-rank regularization in model estimation, was trained using the original 2-D microscopic image data (Zhou *et al.*, 2014), while the SSVS and Lasso were trained using the proposed spatial predictors in 1-D filtered quantiles.

For each replication, we used 90%-10% data partition for model training and testing. The average testing RMSEs for resistance over 100 replications are presented in Table 5 with SEs in the parenthesis. The regularized matrix regression had a significantly poorer prediction performance comparing with the proposed approach. As a drawback mentioned previously, one major limitation on regularized matrix regression is that it fixes each predictor to the same location across different samples, but the defect areas which are strongly associated with the response (resistance) are constantly changing their locations. Furthermore, we can see that Lasso performed poorly in resistance prediction. On the contrary, SSVS with different neighborhood settings had significantly smaller RMSEs, and the smallest error was achieved at r = 3. This indicated that the neighborhood defined by r = 3 was the most appropriate to describe the predictors' smoothness. The modeling assumptions were validated by residual plots in supplementary materials.

Table 5. Testing RMSEs over 100 replications for the AJP dataset

	Benchmark Models			Proposed Model
	Matrix Regression	SSVS (r = 0) / Lasso	SSVS (r = 1) / Fused Lasso	**SSVS (r = 3)**
RMSEs	8.44 (0.87)	4.68 (0.20)	3.27 (0.14)	**3.21** **(0.13)**

One of the advantages of transforming original images directly into spatial predictors lies in good interpretability, because complicated feature extractions can lead to the loss of information depending on how well the underlying physical manufacturing process is understood. We illustrate the most selected features in Figures 6, where the colors of the mask indicate the number of times a predictor was selected over 100 replications (a darker color indicates the mesh being more frequently selected). Specifically, the top two wires of Figure 6 illustrate the variable selection among the first set predictors: the contrasts between each mesh's intensity and the overall image's mean intensity. The middle two wires of Figure 6 illustrate the variable selection among the second set predictors: the contrasts between each mesh's intensity and the mean intensity of the mesh's 8-Connected neighborhood. The bottom two wires of Figure 6 illustrate the variable selection among the third set predictors: contrasts between each mesh's intensity and the largest row-wise mean intensity of the image. The row with the largest intensity, due to a dense layer of ink sprayed, is the most conductive path of the wire.

For the first set in Figure 6, the selected meshes identified the missing ink defects. However, the intensities of selected meshes in the 40 Ω-wire were only 78 units (on a 0-255 scale) darker than the wire mean intensity on average, while such a difference increases to 120 units for the 15 Ω-wire. A relatively lower difference is due to the mean intensity of the 40 Ω-wire is low, which is the result of excessive missing prints shown in the image. For the second set in Figure 6, the selected meshes identified the areas of non-uniform ink spray (seen as black pinholes). However, the contrasts between each mesh and its 8-Connected neighborhood were significantly smaller for the 15 Ω-wire, which is on average 0 versus -4 for the 40 Ω-wire. The less contrast reflects that the 15 Ω-wire was printed more uniformly. For the third set in Figure 6, the selected meshes identified uniformly printed regions. The meshes in the 15 Ω-wire have similar intensity to the image's largest row-wise mean intensity (an average difference of 0). On the contrary, the meshes for 40 Ω-wire are on average 9 units darker than its largest row-wise mean intensity due to significantly more missing sprays. The variable selection results show that variation modeling based on simple and direct spatial features have inherently good interpretability for quality diagnosis and can result in an accurate resistance prediction when combined with SSVS estimator.

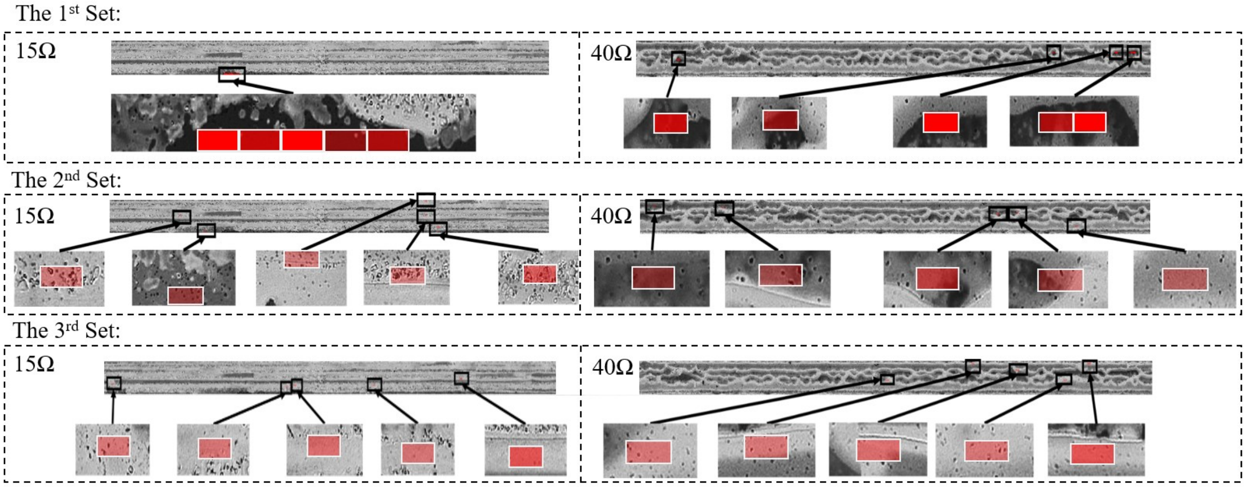


Figure 6. For the first set: the top five selected meshes representing the contrast between each mesh's intensity and the overall image's mean intensity for the wire with 15 Ω and 40 Ω. For the second set: the top five selected meshes reflecting the contrast between each mesh's intensity and the mean intensity of the each mesh's 8-Connected neighborhood for the wire with 15 Ω and 40 Ω. For the third set: the top five selected meshes reflecting the contrast between each mesh's intensity and the largest row-wise mean intensity for the wire with 15 Ω and 40 Ω.

5. Summary

Although spatial predictors are widely encountered in the AM process modeling, the modeling and feature extractions of the spatial predictors are typically challenging and case-specific. In most cases, such an effort requires engineering knowledge of the spatial dataset (e.g., feature extractions from images), or the manufacturing process. In this paper, the proposed SSVS aims to identify the significant predictors in pursuit of smooth parameters to reflect the neighborhood effect. The size of neighborhoods can be adaptively determined by the dataset, which is flexible for various AM processes, with different spatial correlations of predictors. The proposed SSVS relaxed the strong assumption of fused Lasso, which is SSVS's special case considering 1-neighborhood, and was able to estimate the underlying model parameters with higher accuracy. Both the simulation and case studies showed that the proposed methodology could not only accurately predict the quality response, but also automatically locate areas in quality modeling, which can be used in root cause diagnosis.

There are several directions for further investigations. First, we can validate the proposed methodology on other types of 2D or 3D spatial datasets, which have coherent spatial information across different samples, so that re-organizing the spatial data into quantiles for smoothness enforcement is not necessary anymore. Second, we may also relax the coherency requirement by projecting the raw image data into other domains, e.g., 2-D Fourier domains, so that the same set of projected spatial predictors may correspond to different regions with missing spray defects across different samples. As a result, ordering the predictors in quantiles will no longer be necessary before adopting SSVS for the AJP case study dataset. Third, we may adopt other spatial datasets to enhance the modeling performance. For example, we can use the 3-D surface scanning to capture the thickness variation, which can also contribute to the resistance variation of printed wires. However, the limitations, such as the slow data collection speed, of the current sensing technologies suggest that additional spatial data types might not be obtained efficiently for quick product quality quantification at this moment (Wang *et al.*, 2013). Finally, the proposed framework can be further used for spatial predictor monitoring and control for quality improvements and variation reduction, with some examples in Li *et al.* (2017b). The code package of SSVS is available upon request.

Acknowledgments

The authors acknowledge the editor, associate editor, and reviewers for their constructive comments for us to revise this article, and the support from National Science Foundation CMMI-1634867 to this research.

References

Adams, R., and Bischof, L. (1994) Seeded region growing. *IEEE Transactions on pattern analysis and machine intelligence*, **16**(6), 641-647.

Akaike, H. (1998) Information theory and an extension of the maximum likelihood principle. in *Selected papers of Hirotugu Akaike*, Springer. New York, NY, pp. 199-213.

Bai, Y., Sun, Z., Zeng, B., Long, J., Li, L., de Oliveira, J. V., and Li, C. (2018) A comparison of dimension reduction techniques for support vector machine modeling of multi-parameter manufacturing quality prediction. *Journal of Intelligent Manufacturing*, 1-12.

Boureau, L., Ponce, J., and LeCun, Y. (2010) A theoretical analysis of feature pooling in visual recognition, in *Proceedings of the 27th international conference on machine learning*, ICML, pp. 111-118.

Bukkapatnam, S. T., Kumara, S., and Lakhtakia, A. (1999) Analysis of acoustic emission signals in machining. *Journal of manufacturing science and engineering*, **121**(4), 568-576.

Chen, J., and Chen, Z. (2012) Extened BIC for small-n-large-p sparse GLM. *Statistica Sinica*, **22**(2), 555-574.

Chen, Y., Lin, Z., Zhao, X., Wang, G., and Gu, Y. (2014) Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, **7**(6), 2094-2107.

Colosimo, B. M., Pacella, M., and Senin, N. (2015) Multisensor data fusion via Gaussian process models for dimensional and geometric verification. *Precision Engineering*, **40**, 199-213.

Haverinen, H. M., Myllyla, R. A., and Jabbour, G. E. (2010) Inkjet printed RGB quantum dot-hybrid LED. *Journal of display technology*, **6**(3), 87-89.

Huang, Q., Zhang, J., Sabbaghi, A., and Dasgupta, T. (2015) Optimal offline compensation of shape shrinkage for three-dimensional printing processes. *IIE Transactions*, **47**(5), 431-441.

Jin, J., and Shi, J. (2000) Diagnostic feature extraction from stamping tonnage signals based on design of experiments. *Journal of manufacturing science and engineering*, **122**(2), 360-369.

Kang, H., Kitsomboonloha, R., Jang, J., and Subramanian, V. (2012) High-Performance Printed Transistors Realized Using Femtoliter Gravure-Printed Sub-10 μm Metallic Nanoparticle Patterns and Highly Uniform Polymer Dielectric and Semiconductor Layers. *Advanced Materials*, **24**(22), 3065-3069.

Kang, J., Reich, B. J., and Staicu, A.-M. (2016) Scalar-on-Image Regression via the Soft-Thresholded Gaussian Process. *arXiv preprint arXiv:1604.03192*.

Krebs, F. C. (2009) Fabrication and processing of polymer solar cells: a review of printing and coating techniques. *Solar energy materials and solar cells*, **93**(4), 394-412.

Li, C., Ledo, L., Delgado, M., Cerrada, M., Pacheco, F., Cabrera, D., Sánchez, R.-V., and de Oliveira, J. V. (2017a) A Bayesian approach to consequent parameter estimation in probabilistic fuzzy systems and its application to bearing fault classification. *Knowledge-Based Systems*, **129**, 39-60.

Li, C., Sanchez, R.-V., Zurita, G., Cerrada, M., Cabrera, D., and Vásquez, R. E. (2015) Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis. *Neurocomputing*, **168**, 119-127.

Li, X., Zhou, H., and Li, L. (2013) Tucker tensor regression and neuroimaging analysis. *arXiv preprint arXiv:1304.5637*.

Li, Y., Mohan, K., Sun, H., and Jin, R. (2017b) Ensemble Modeling of In Situ Features for Printed Electronics Manufacturing With In Situ Process Control Potential. *IEEE Robotics and Automation Letters*, **2**(4), 1864-1870.

Lou, Y., Caruana, R., and Gehrke, J. (2012) Intelligible models for classification and regression, in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 150-158.

Luo, Y. (2017) Recurrent neural networks for classifying relations in clinical notes. *Journal of Biomedical Informatics*, **72**, 85-95.

Mahajan, A., Frisbie, C. D., and Francis, L. F. (2013) Optimization of aerosol jet printing for high-resolution, high-aspect ratio silver lines. *ACS Applied Materials & Interfaces*, **5**(11), 4856-4864.

Oh, Y., Suh, D., Kim, Y., Lee, E., Mok, J. S., Choi, J., and Baik, S. (2008) Silver-plated carbon nanotubes for silver/conducting polymer composites. *Nanotechnology*, **19**(49), 495602.

Rao, P. K., Kong, Z., Duty, C. E., Smith, R. J., Kunc, V., and Love, L. J. (2016) Assessment of dimensional integrity and spatial defect localization in additive manufacturing using spectral graph theory. *Journal of Manufacturing Science and Engineering*, **138**(5), 1-12.

Ripley, B. D. (2005). *Spatial Statistics*. Vol. 575, John Wiley & Sons, Hoboken, NJ.

Schabenberger, O., and Gotway, C. A. (2017). *Statistical methods for spatial data analysis*, CRC press, Boca Raton, FL.

Schwarz, G. (1978) Estimating the dimension of a model. *The Annals of Statistics*, **6**(2), 461-464.

Sun, H., Wang, K., Li, Y., Zhang, C., and Jin, R. (2017) Quality modeling of printed electronics in aerosol jet printing based on microscopic images. *Journal of Manufacturing Science and Engineering*, **139**(7), 1-10.

Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, **58**(1), 267-288.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005) Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **67**(1), 91-108.

Tibshirani, R. J., and Taylor, J. (2010) The solution path of the generalized lasso. *arXiv preprint arXiv:1005.1971*.

Tripathy, R., Bilionis, I., and Gonzalez, M. (2016) Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation. *Journal of Computational Physics*, **321**, 191-223.

Vovk, V. (2013) Kernel ridge regression. in *Empirical inference*, Springer. Berlin, Heidelberg, pp. 105-116.

Wang, K., Chang, Y.-H., Zhang, C., and Wang, B. (2013) Evaluation of Quality of Printed Strain Sensors for Composite Structural Health Monitoring Applications, in *Proceedings of 2013 SAMPE Fall Technical Conference*, SAMPE, pp. 21-24.

Wang, X., and Leng, C. (2015) High dimensional ordinary least squares projection for screening variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **78**(3), 589-611.

Yan, H., Paynabar, K., and Shi, J. (2015) Image-based process monitoring using low-rank tensor decomposition. *IEEE Transactions on Automation Science and Engineering*, **12**(1), 216-227.

Ye, G.-B., and Xie, X. (2011) Split Bregman method for large scale fused Lasso. *Computational Statistics & Data Analysis*, **55**(4), 1552-1569.

Zeng, Y., Young, T. M., Edwards, D. J., Guess, F. M., and Chen, C.-H. (2016) A study of missing data imputation in predictive modeling of a wood-composite manufacturing process. *Journal of Quality Technology*, **48**(3), 284-296.

Zhao, D., Liu, T., Park, J. G., Zhang, M., Chen, J.-M., and Wang, B. (2012) Conductivity enhancement of aerosol-jet printed electronics by using silver nanoparticles ink with carbon nanotubes. *Microelectronic Engineering*, **96**, 71-75.

Zheng, X., and Fang, H. (2015) An integrated unscented kalman filter and relevance vector regression approach for lithium-ion battery remaining useful life and short-term capacity prediction. *Reliability Engineering & System Safety*, **144**, 74-82.

Zhou, H., and Li, L. (2014) Regularized matrix regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **76**(2), 463-483.