Fault Diagnosis in Microelectronics Attachment via Deep Learning Analysis of 3D Laser Scans

Nikolaos Dimitriou, Lampros Leontaris, Thanasis Vafeiadis, Dimosthenis Ioannidis, Tracy Wotherspoon, Gregory Tinker, and Dimitrios Tzovaras

Abstract—**A common source of defects in manufacturing miniature Printed Circuits Boards (PCB) is the attachment of silicon die or other wire bondable components on a Liquid Crystal Polymer (LCP) substrate. Typically, a conductive glue is dispensed prior to attachment with defects caused either by insufficient or excessive glue. The current practice in electronics industry is to examine the deposited glue by a human operator a process that is both time consuming and inefficient especially in preproduction runs where the error rate is high. In this paper we propose a system that automates fault diagnosis by accurately estimating the volume of glue deposits before and even after die attachment. To this end a modular scanning system is deployed that produces high resolution point clouds whereas the actual estimation of glue volume is performed by (R)egression-Net (RNet), a 3D Convolutional Neural Network (3DCNN). RNet outperforms other deep architectures and is able to estimate the volume either directly from the point cloud of a glue deposit or more interestingly after die attachment when only a small part of glue is visible around each die. The entire methodology is evaluated under operational conditions where the proposed system achieves accurate results without delaying the manufacturing process.**

Index Terms—**3DCNN, deep learning, PCB, smart manufacturing.**

I. INTRODUCTION

SMART manufacturing has greatly benefited from the Internet of Things (IoT) revolution and advances in Deep Learning (DL) techniques as well. In this regard, several frameworks have been proposed that enabled the collections and analysis of diverse data from factory equipment and external sensors in order to make better decisions and optimize production towards zero-defect manufacturing strategies.

This paper focuses on a characteristic use case falling into this category that often arise in the electronics manufacturing industry. In particular, a system is proposed for the scanning of PCBs and the inspection of die attachment on an LCP substrate. The key quantity that needs to be monitored is the

Nikolaos Dimitriou, Lampros Leontaris, Thanasis Vafeiadis, Dimosthenis Ioannidis and Dimitrios Tzovaras are with the Information Technologies Institute, Centre for Research and Technology Hellas (CERTH-ITI), 57001 Thessaloniki, Greece. (e-mail: nikdim@iti.gr, lleontar@iti.gr, thanvaf@iti.gr, djoannid@iti.gr, dimitrios.tzovaras@iti.gr).

Tracy Wotherspoon and Gregory Tinker are with Microsemi Corporation, a wholly owned subsidiary of Microchip Technology. (e-mail: Tracy.Wotherspoon@microchip.com, Greg.Tinker@microchip.com).
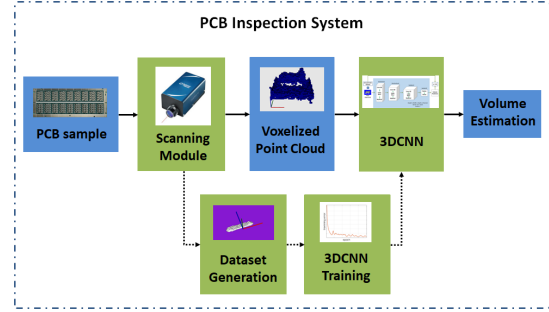

Fig. 1. The pipeline on the proposed system. Initially PCB's regions of interest are scanned in order to capture 3D information on the dispensed glue. The captured points clouds are fed to a 3DCNN that estimates the glue volume. Part of the proposed system is a procedure for generating a proper dataset for training the 3DCNN in a way that allows accurate volume estimation even when dies have been attached, occluding most of the dispensed glue.

volume of glue that is deposited on the LCP since insufficient or excessive glue is the primary source of malfunctions. The current industry practice for fault diagnosis is to have human operators inspect the manufactured PCBs and identify potential defects, a process that is error prone, cumbersome and time consuming causing delays in the production process.

The proposed system uses a custom scanning device consisting of a high-resolution laser scanning sensor and a modular motion control framework in order to scan the areas of interest on a PCB and to extract geometric information in terms of 3D point clouds. Using these point clouds, glue volumes are estimated using a 3DCNN where the captured point clouds are quantized into 3D voxels. A graphical overview of our system is shown in Figure 1. In summary, the main technical contributions of our work can be pinpointed to,

- deploying a modular scanning system for inspecting PCBs
- introducing a process for developing a volumetric dataset of glue deposits
- proposing a 3DCNN architecture for estimating the volume of glue
- correlating glue volume measurements before and after die attachment, enabling quality inspection after die attachment where only a trace of glue is visible.

The proposed approach counters difficulties commonly occurring in manual inspection tools stemming from subjective aspects and provides fast and accurate assessments. It is worth noting that while deep learning has certainly improved state of the art in fault diagnosis for industrial applications, to the

best of our knowledge there has not been any deep learning architectures used for quality inspection on 3D data. In this paper, the use of (R)egression-Net is yet another step closer to the exploitation of deep learning techniques in 3D industrial applications.

The remainder of the paper is organized as follows. Section II focuses on related work in deep learning and industrial applications, while Section IV provides insight on the examined microelectronics use case. In Section V the proposed system is described in detail including the deployed scanning system, the generated volumetric dataset and the RNet architecture. Next, Section VI is devoted to the experimental evaluation of our solution and finally Section VII draws the conclusions of our work.

II. RELATED WORK

Deep Learning (DL) refers to a set of technologies, primarily Artificial Neural Networks (ANN), that has significantly improved state of the art in several tasks such as computer [1] and robotic [2] vision applications, image analysis [3], [4], audio signals [5], [6] and other complex signals [7]. In industrial systems, DL applications focus on inspection and fault diagnosis in industrial shopfloors and manufacturing processes. CNNs are the most representative category of deep learning models and their architectures are able to deal with big multimodal datasets. For instance, in [8] a CNN framework is presented to convert sensor signals from manufacturing processes to 2-D images which are used to solve image classification tasks for fault diagnosis whereas in [9] a method is reported for analyzing tomography sensory data using a convolutional neural network pipeline. Moreover, in [10], a deep convolutional computation model is proposed to learn hierarchical features of big data in IoT, by using the tensor-based representation model to extend the CNN from the vector space to the tensor space. Also, a convolutional discriminative feature learning method for induction motor fault diagnosis is introduced in [11], an approach that utilizes a combination of back-propagation neural network and a feed-forward convolutional pooling architecture.

In another interesting application of DL in industry, the authors of [12] propose a semisupervised deep learning model for data-driven soft sensor in order to monitor quality variables that cannot be measured directly. In a related line of research, [13] presents a soft sensor modeling method that combines denoising autoencoders with a neural network. Furthermore, a deep learning framework is introduced in [14] for adaptively extracting features from raw data that are used for fault classification while [15] presents a framework that utilizes multiple sparse autoencoders for tool condition monitoring. In a relevant work, the authors of [16] propose a deep learning model for early fault detection of machine tools under time-varying conditions. The authors in [17], have proposed an optimized deep architecture, namely a large memory storage retrieval neural network, to diagnose bearing faults. In a related application, [18] propose a convolutional deep belief network for bearing fault diagnosis applied in monitoring electric locomotive bearings while in the work of [19] a

deep residual network is used with dynamically weighted wavelet coefficients for planetary gearbox fault diagnosis. An unsupervised feature engineering framework is proposed in [20] that utilizes vibration imaging and deep learning for fault diagnosis in rotor systems. In another domain, [21] reports a crack detection method for nuclear reactors monitoring using a CNN and a Naive Bayes data fusion approach. Also a deep neural network architecture for chemical fault diagnosis on imbalanced data streams is presented in [22].

Deep learning has been also deployed in the field of semiconductor manufacturing which is the application of our work. In [23], authors introduce a method for wafer map defect pattern classification and image retrieval using CNN, so as to identify root causes of die failures during the manufacturing process. A deep structure machine learning approach is also proposed in [24] to identify and classify both single-defect and mixed-defect patterns by incorporating an information gain-based splitter as well as deep-structured machine learning. A fault detection and classification CNN model with automatic feature extraction and fault diagnosis functions has been proposed in [25] to model the structure of the multivariate sensor signals from a semiconductor manufacturing process.

While deep learning has certainly fostered progress in fault diagnosis of 1D and 2D signals it has not found wide application in processing 3D sensory from industrial shopfloors. Nonetheless, outside industry, 3DCNN models have recently gained traction for traditional computer vision tasks. For instance, VoxNet [26] integrates a volumetric occupancy grid representation with a supervised 3DCNN for object recognition. Similarly focussing on object recognition, a 3D volumetric object representation with a combination of multiple CNN models based on LeNet model is introduced in [27] while in [28], the authors proposed an algorithm for labeling complex 3D point cloud data and provide a solution for training and testing 3DCNN using voxels. A CNN has been also proposed in [29] that directly consumes 3D point clouds and has shown impressive results in 3D object recognition. In another interesting work [30] octree-based CNN is built upon the octree representation of 3D shapes and performs 3DCNN operations on the octants occupied by the 3D shape surface. Moreover, the authors of [31] use a subset of ModelNet [32] and propose a CNN combining a spatial transformation in an attempt to deal with the rotation invariance problem in 3D object classification. In the following section we will see that 3DCNN based deep models have potential for fault diagnosis on 3D data as is validated through a concrete microelectronics use case, namely die attachment on a LCP substrate.

III. INDUSTRIAL BACKGROUND

The proposed method is driven by a common problem faced in the electronics industry for the manufacturing of miniature electronic modules. The standard electronic equipment used in the industry for wire bonding, lacks the capability of auto-mated optical inspection of the process. During manufacturing, silicon dies are attached to an LCP substrate using conductive glue that is placed by the glue dispenser machine.

The manufacturing process consists of two main stages performed sequentially. The first stage is the glue dispensing
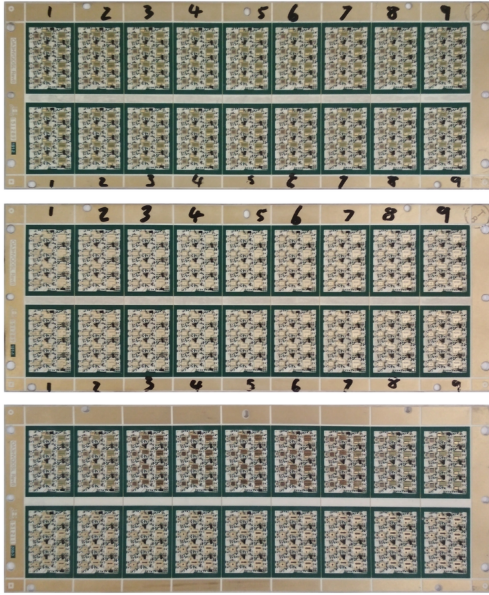
Fig. 2. Each of the PCBs used to validate the proposed system consists of two rows and nine columns for a total of 18 circuit modules depicted in Figure 3. The PCBs are specifically manufactured to simulate the potential fault conditions during production. For the same column of circuit modules of the PCBs, the same glue quantity is dispensed. Moreover, from the left to the right columns, a larger quantity of glue has been progressively dispensed. The top and middle PCB consists of regions where the dies are attached and unattached respectively, while the bottom PCB consists of both attached (first row of nine circuits) and unattached (second row of nine circuits).

Fig. 3. On the left and right image, we have a closer look at one circuit module of a PCB with dimensions $13mm \times 19mm$, before and after die attachment. Notice the different types of glue annotated as *A*, *B*, *C*, *D* and *E*. On each circuit there are four glue deposits on each type where approximately the same quantity of glue has been placed. As explained in Section VI the top three deposits are used for training a 3DCNN and the bottom one for testing. On top, we see a magnified view of glue regions before and after the attachment of dies.

stage and the second is the die (IC) attachment stage. The most critical variable for the attachment stage is the volume of the dispensed glue that is controlled indirectly by parameterizing the pressure on the glue dispenser machine. The two fault conditions that can be faced in the production are the excessive and insufficient glue volume. The excess in the amount of the glue leads to internal short circuits and on the other hand, insufficiency in the glue volume leads to weak die bonding. The current practice for the detection of these two fault conditions is the manual inspection of the glue volume at the first stage and the glue fillet, defined as the visible overflown glue around the die border, at the second stage.

The manual inspection by a human operator who decides on the faulty conditions of the process, introduces an error that derives from the limitations of the human factor. This error corresponds to overhead costs on newly introduced parts in the production line that require adjustment period for the stabilization of the manual inspection process, resulting in an increase of the failure rate. The automation of the inspection process is considered an important step towards the automated quality control in the microelectronics industry and constitutes the motivation of the proposed system.

IV. DESCRIPTION OF USE CASE

For the development and testing of the proposed system we have used three PCBs depicted in Figure 2. As is illustrated in Figure 3, there are 5 different types of dies annotated as

A,B,C,D and E. The type of die defines the shape of the glue that needs to be placed on the LCP substrate for its attachment. This correspondence between the glue and die types is depicted in Figure 3. Each PCB comprises 18 circuits over two rows and nine columns. Each of the 18 circuits hosts four placeholders for each of the five glue types A,B,C,D and E, thus totally 20 regions for the placement of the dies as illustrated in Figure 3.

To simulate the conditions that can be met during the production, the PCBs, in Figure 2, have been specifically manufactured to have a wide range of glue quantities, from insufficient glue to excessive glue. Concretely, the glue quantity progressively increases starting from the far left column of each of the three PCBs and moving to the right. To ensure the same quantity per column, aside appropriately parameterizing the pressure on the glue dispenser, glue quantity has been visually verified during manufacturing. Moreover, to simulate both the glue dispensing stage and the die attachment stage, the top PCB comprises regions at the die attachment stage, the middle PCB comprises regions at the glue dispensing stage and the bottom PCB comprises regions at both stages; top 9 circuits comprises regions after die attachment while the bottom 9 circuits comprises regions before die attachment.

V. PCB INSPECTION SYSTEM

In this section, the components of the proposed system are described. Initially we present the custom scanning system that was developed for acquiring 3D measurements from a PCB. Next the processing of the collected data is described in order to generate an annotated data set of point cloud measurements and finally the proposed 3DCNN architecture is presented for glue volume estimation.

A. Scanning system

For PCB inspection, a modular laser scanning system has been built. As depicted in Figure 4a, it includes an Optimet Conopoint-10 sensor [33], a Newport XPS-RL2 motion controller [34], two linear stages and the support breadboards. In
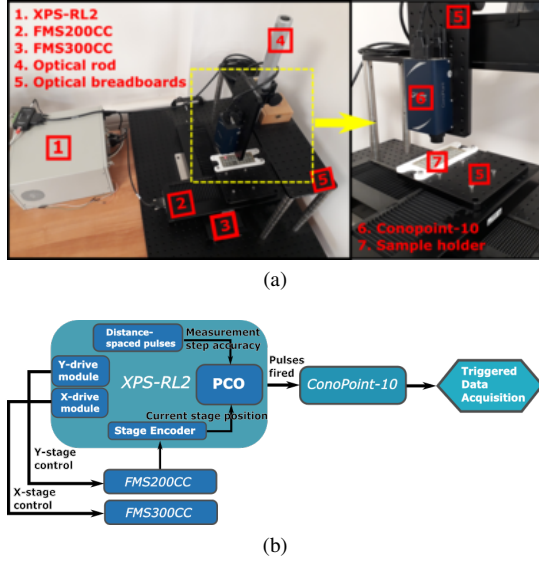
(a)

(b)

Fig. 4. **(a)** The left image is a snapshot of the scanning system where its basic components are highlighted, namely the motion controller (XPS-RL2), laser sensor (Conopoint-10), linear stages (FMS200CC and FMS300CC) and the parts (breadboards) for the mechanical integration of the system. The right image takes a closer look on the region in the yellow rectangle. **(b)** The operational flow of the scanning system where the motion controller drives the two linear stages X-stage and Y-stage using two drive modules. The Y-stage is configured as the scanning stage and its current stage position is used by the Position Compare Output (PCO) to fire the distanced-spaced pulses that trigger the data acquisition by the laser sensor.

our system we use the FMS200CC and FMS300CC [35] linear stages with travel ranges of $200mm$ and $300mm$ respectively.

Since inspection time is important for any production line, effort has been put in order to significantly decrease scanning time without losing accuracy. In this regard, a hardware triggering mechanism has been developed. The mechanism uses the Pulse Acquisition Mode of the sensor with measurement rate that is driven by distance spaced trigger pulses generated by the Position Compare Output (PCO) of the controller. The PCO provides $50nsec$ delay between the crossing of the positions defined by the sample step and sending out the signal. The maximum speed of $100mm/s$ of the stages produces $5nm$ uncertainty in position. The operational flow of the system is shown in 4b.
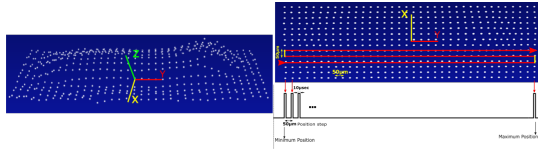

Fig. 5. The left pictures is a point cloud on XYZ space for glue type A, generated by scanning a region of $0.7mm \times 1.8mm$. On the top right image, the projection of the point cloud on XY plane is shown. The scanning step of $50\mu m$ in X and Y axis and the zig-zag scanning direction are marked. The bottom right image shows the pulse diagram of the even distanced pulses generated by the controller that trigger the sensor.

In the bottom right image of Figure 5 we see the pulse diagram for a $1.8mm$ range for Y-stage. The PCO generates a series of pulses that trigger the data acquisition at positions
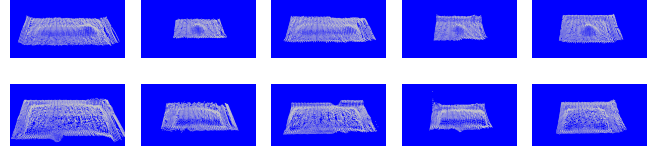

Fig. 6. Scans over different regions of the PCB. From left to right, scans of type *A*, *B*, *C*, *D* and *E* are shown both before (top) and after (bottom) dies are attached. Even though the dimension of these regions is below $1mm \times 2mm$ their 3D structure is well captured with a $20\mu m$ step.

that are defined by the sample step of $50\mu m$. A total of 36 pulses are generated in Y axis. After the last pulse is generated, the X-stage moves by $50\mu m$ without being homed and a new series of pulses are generated in the reverse direction in Y-axis. Indicatively, by adapting this zig-zag scanning procedure as shown in the top right image of Figure 5 and implementing the hardware triggering mechanism, the scanning time is decreased approximately 10-fold, compared to taking single measurements for every step along the motion. For all of the following experiments, this scanning strategy is followed.
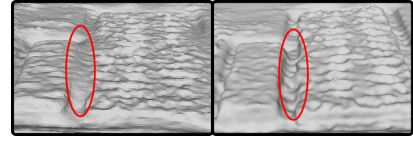

Fig. 7. Magnified views of the 3D surface after die attachment for same type regions where excessive glue is deposited (left) and insufficient glue is deposited (right). Notice the steeper gap between the die and the LCP substrate in the second case.

Scans on a specific type of glue region have a constant range along the X, Y axes. Some indicative scans over different regions of a PCB are shown in Figure 6. The 3D structure of the regions is captured quite well both before die attachment, where the glue deposit is clearly visible, and after die attachment. What is interesting for the more challenging second case, is that the glue fillet that exceeds the die border is captured as well as the 3D structure of the actual die. To verify this, in Figure 7 two same type regions are shown with the first one corresponding to a case of excessive glue on the left of the PCB and the second one to insufficient glue on the right of the PCB. Notice the much sharper gap between the die and the LCP substrate in the second case.

B. Generating a dataset

To train and evaluate our monitoring framework an annotated point cloud dataset has been developed where the volumes of glue deposits have been analytically computed. To this end we used the circuits on the second PCB and second row of the bottom PCB in Figure 2 where no dies are attached and glue is clearly visible and assumed that the volume of deposited glue is equal for regions in the same column as explained in Section IV. For each regional scan, the plane equation for the LCP substrate has been calculated using the RANSAC algorithm [36] for plane fitting as implemented in PCL [37] and the 3D points corresponding to the glue
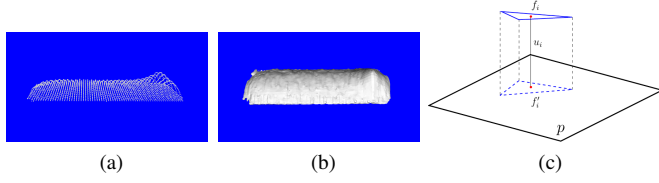
(a)   (b)   (c)

Fig. 8. **(a)** The cropped point cloud of the regional scan depicted in the upper left image of Figure 6 after its bottom is closed using the substrate plane. **(b)** The computed surface mesh after Poisson reconstruction is applied. **(c)** Denoting a random face of glue mesh as f_i, its projection on the LCP substrate plane p is represented as f_i' whereas u_i the distance between the centers of f_i, f_i'.

Glue Type	A	B	C	D	E
20 micrometer sampling step					
unatt. (train)	40500	25920	40500	25920	40500
unatt. (test)	13500	8640	13500	8640	13500
att. (train)	40500	32400	40500	32400	40500
att. (test)	13500	10800	13500	10800	13500
50 micrometer sampling step					
unatt. (train)	12960	6480	12960	6480	6480
unatt. (test)	4320	2160	4320	2160	2160
att. (train)	19440	9720	19440	9720	14580
att. (test)	6480	3240	6480	3240	4860

TABLE I
NUMBER OF TRAINING AND TEST SAMPLES.

mass have been manually cropped. In order to get a closed 3D surface, each 3D point of the cropped point cloud is projected on the estimated substrate plane as is illustrated on the example of Figure 8a. Having a closed 3D point cloud, Poisson reconstruction [38] is applied to compute the surface mesh of glue as is depicted in Figure 8b. Geometrically, having a surface S, its volume V can be computed using the formula,

$$V = \iiint_S \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z. \tag{1}$$

In our case an analytical representation of S is not available, therefore glue volume is approximated using its reconstructed mesh. More formally, let us denote the glue mesh as a set $F = \{f_1, f_2, \ldots, f_N\}$ of 3D points' triangles and the substrate plane as p. Additionally, we denote with f_i' the projection of f_i on p and with u_i the distance between the centers of f_i and f_i' as is more clearly depicted in Figure 8c. The glue volume is approximated as,

$$V = \sum_i |f_i'| u_i \tag{2}$$

where $|f_i'|$ is the area of the projected rectangle. Following this procedure, we end up with 108 annotated point clouds for each of the five different types of glue deposits on the circuits with no dies attached. We extend these volume estimations to the rest of the circuits where dies have been attached in order to annotate the entire PCB dataset. Naturally, this extension process as well as the volume approximation is error prone due to several factors such as sensor noise, discrepancies between glue deposits due to the glue dispenser and the induced approximation error. Nonetheless, as shown in Section VI these estimations captures the manufacturer's trend of progressively depositing more glue. This allows the correlation of the hidden glue after die attachment with actual measurements, thus enabling fault diagnosis even after dies are attached.

Of course the number of annotated samples is very small for a 3DCNN network, therefore a data augmentation step has been implemented. To this end and in order to also take into account measurement noise we scan each PCB five times, thus artificially raising the number of PCB samples to fifteen. Moreover, the range along the X, Y, Z is calculated for each regional scan. A cropping filter is applied on the $X - Y$ plane using a bounding box that is shifted along the X, Y axes. For
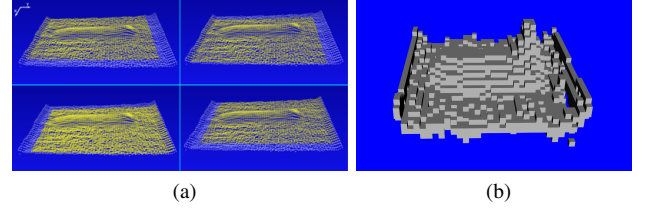

(a)   (b)

Fig. 9. **(a)** Augmentation example. In yellow we see the point cloud cropped from the original one at various augmentation steps. **(b)** The $32 \times 32 \times 32$ voxel grid of the regional scan depicted in the upper left image of Figure 6. The quantization of the 3D space along the Z axis is purposely more detailed since information along the vertical axis is more important for estimating the glue volume. Note for instance the magnified glue bump on the right of the glue deposit that occurred when glue dispensation finished.

all our experiments the bounding box covers 92% of the range on the X, Y axes and is shifted by a constant step s,

$$s = \max(0.02 * r, \epsilon) \tag{3}$$

where r is the respective axis range and ϵ is the minimum step of the linear stage which is either $20\mu m$ or $50\mu m$ for our experiments. This augmentation procedure, as is also shown in the example of Figures 9a, simulates the potential mislocalization of the scanning device. The dataset is further augmented by adding Gaussian noise along the Z axis coarsely approximating noise during the scanning procedure. In our experiments, 4 noise levels have been used by increasing the noise deviation by 3% with respect to the range on Z. Following this procedure for different step of the linear stages, namely $20\mu m$ and $50\mu m$, we get several thousands of point clouds for each of the different region types as is summarized in Table I. In the table we denote with att. and unatt. the cases with and without die attached and we adopt a 3 to 1 ratio for splitting between training and test sets.

C. 3DCNN for glue volume estimation

To automate glue volume estimation we train 10 instances of the proposed RNet architecture with each one corresponding to a particular type of glue before or after the respective die has been attached. Point clouds are parameterized using $0 - 1$ voxel grids similar to the works of [26], [39]. This simple representation works fine in our case since, in contrast to depth cameras, there are no occlusion issues caused by a
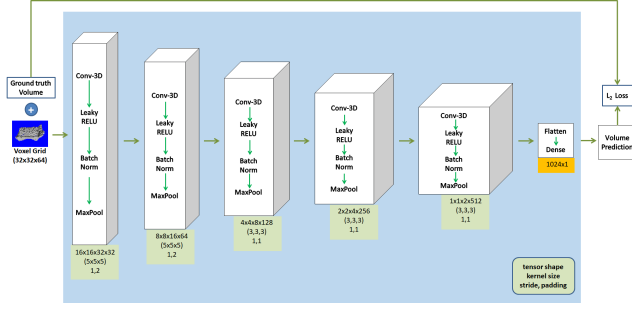
Fig. 10. The RNet architecture. Its input is a voxel grid that is sequentially passed through convolutional blocks of 3D convolution, leaky RELU, batch normalization and max pooling layers. To estimate glue volume a flattening and a dense layer are applied on the last convolutional block. Under each block we report the size of the output tensor, the kernel size of the convolutional filter as well as the stride and padding parameters.

single viewpoint while point cloud density is approximately constant. Throughout our experiments the size of voxel grids is $32 \times 32 \times 64$. The dimensions of a single voxel are constant for each type of glue. Specifically, along the X, Y axes the dimensions are equal to the range of point cloud divided by the number of voxels whereas it is set to $10\mu m$ along the Z axis. For glue types where the dimension along the Z is not adequate for covering at least 98% of the captured 3D points, the dimension of voxels increases in steps of $5\mu m$ until this criterion is fulfilled. An indicative voxel grid is shown in Figure 9b corresponding to the point cloud of the upper left image in Figure 6. To facilitate visualization we have reduced the number of voxels along Z to 32.

The architecture of RNet is shown in Figure 10 along with the specific parameters of each layer. The network's input is an occupancy grid and there are five network blocks consisting of sequences of 3D convolutional, leaky ReLU, batch normalization and max pooling layers that progressively drop the spatial dimensions of the input tensor while increasing the number of channels. Concretely the dimension of the input tensor for the first block is $32 \times 32 \times 64 \times 1$ and the output tensor is $16 \times 16 \times 32 \times 32$ which is then propagated to the following block. After these five network blocks, the last one generates a $1 \times 1 \times 2 \times 512$ tensor that is subsequently flattened to a 1024 feature vector which is forwarded to a 1024×1 dense layer that produces the final volume estimation. Figure 10 includes the dimension of the output tensor under each network block along with the kernel size, stride and padding of the respective convolutional layer. The size of the max pooling window is $2 \times 2 \times 2$ with a stride of 2 for all network blocks. It should be mentioned that we have also experimented with deeper architectures without noticing substantial accuracy improvement. The cost function that is minimized is the l_2 error between ground truth and estimation. Namely, if the glue volume of grid g is denoted as $V(g)$ and the volume estimation as $V(g)'$ the network loss is defined as,

$$L = \mathbb{E}[|V(g) - V(g)'|^2]. \qquad (4)$$

The underlying optimization problem is solved using Adam optimization algorithm [40] as is implemented in the PyTorch

framework [41].

VI. EXPERIMENTS

This section presents a series of experiments that were carried out in order to examine the performance of our framework in different aspects and compare it with alternative state of the art methods.

A. Experimental setup

The complexity of the PCB with its different types of glue regions and dies provide a challenging evaluation setup for the proposed framework. The conducted experiments have been divided in two categories based on the dataset described in Section V-B. In the first category dies have not been attached and glue deposits are clearly visible while in the second one dies have been placed and only glue on the sides of dies is partially visible. Another important experimental parameter that is examined is inspection time using different scanning step. In this regard, two different sets of experiments are presented where the sampling step for the scanning system of Section V-A is 20 and 50 micrometers.

For each of these sets of experiments, 10 separate instances of RNet are trained corresponding to the five different glue types before and after die attachment. For each instance, point clouds are split in training and test set under a $75\% - 25\%$ ratio. To avoid biasing the evaluation by the data augmentation described in Section V-B, the training and validation sets correspond to scans from different regions on the PCBs. Concretely, as is also explained in Figure 3, on each circuit module the top three glue deposits of each type are used for training and the last one for testing.

Each RNet instance has been trained for 100 epochs using a batch size of 128, a learning rate of 0.0001 and having 4908865 trainable parameters. Training and validation mean square error remained approximately constant after 80 epochs thus we assume that the training procedure converges to a stable minimum in all cases. Training time was up to 15 hours for the cases with the largest training set and was performed on two Tesla K40m GPUs. For parameter initialization all convolutional layers are initialized using a Gaussian distribution of mean 0 and standard deviation of 0.02 while batch normalization layers are initialized using a mean of 1 and standard deviation of 0.02.

B. Data augmentation analysis

In this paragraph we examine the contribution of the data augmentation step. In this regard, Table II contains the achieved Mean Square Error (MSE) on the validation set when training with and without data augmentation. The rows denoted as *no IC* and *IC* correspond to before and after die attachment. In all cases it is apparent that the data augmentation step on Section V-B clearly improves the generalization of the trained model, achieving lower MSE by at least an order of magnitude.
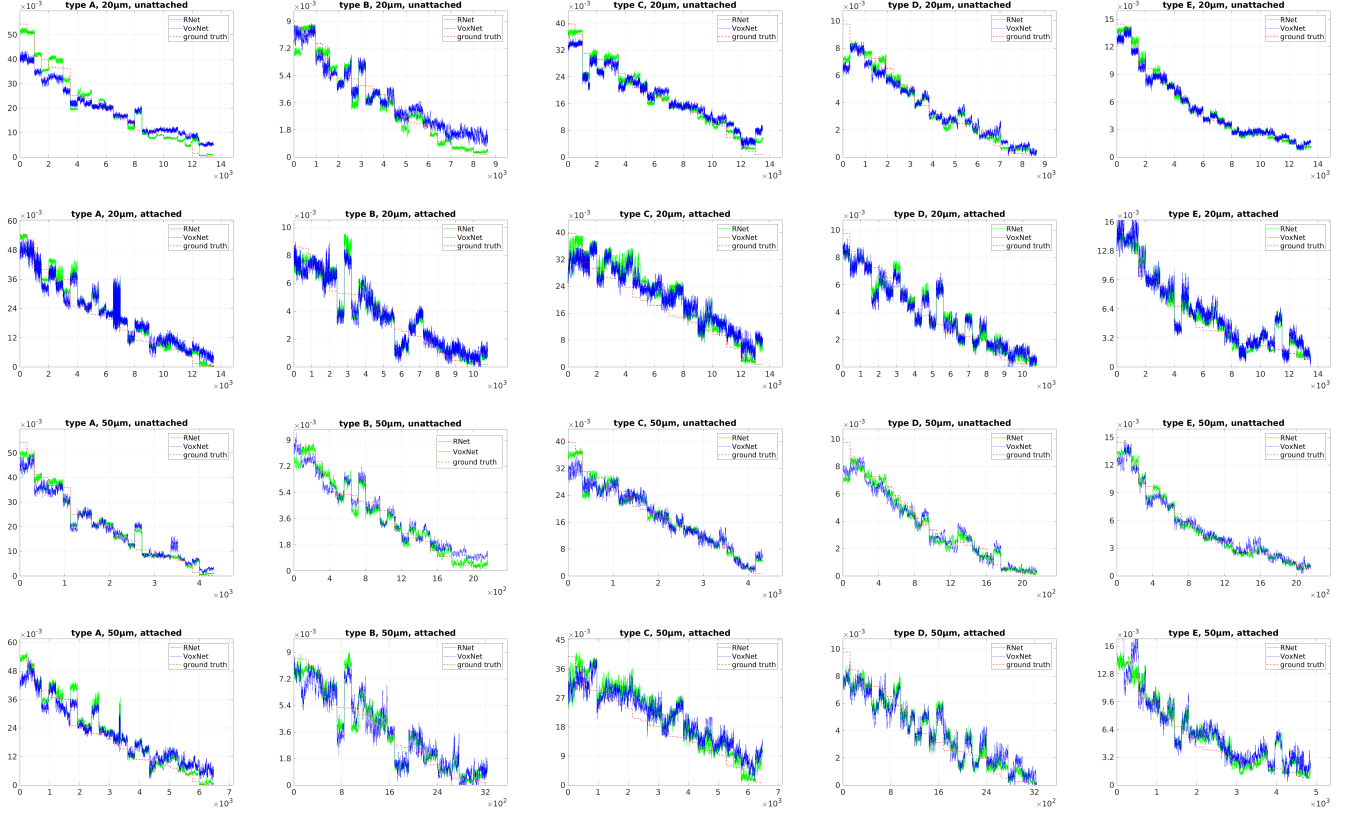
Fig. 11. Comparison of 3DCNN predictions and ground truth. The vertical axis corresponds to the glue volume and the horizontal axis to the index of the test sample. For each model testing samples are sorted according to the ground truth glue volume, and results for sampling steps of $20\mu m$ and $50\mu m$ are reported. Even in the more challenging cases where dies have been attached, the predictions of the 3DCNN models (in green) follow the ground truth curves (in red).

Type		A	B	C	D	E
20 micrometer sampling step						
no IC	aug.	8.21	0.45	5.37	0.39	0.21
	no aug.	88.52	1.06	45.02	0.78	2.24
IC	aug.	21.54	1.25	25.19	1.02	1.37
	no aug.	68.52	2.07	79.93	1.91	5.26
50 micrometer sampling step						
no IC	aug.	7.99	0.35	5.29	0.47	0.25
	no aug.	33.91	0.98	19.05	0.90	1.21
IC	aug.	26.89	1.00	28.40	1.00	1.55
	no aug.	99.16	2.27	80.56	2.05	6.47

TABLE II
VALIDATION MSE IN $e-06$ $(mm^3)^2$ WITH(OUT) AUGMENTATION.

Type		A	B	C	D	E
20 micrometer sampling step						
no IC	RNet	8.21	0.45	5.37	0.39	0.21
	VoxNet	29.55	0.67	9.89	0.71	0.38
IC	RNet	21.54	1.25	25.19	1.02	1.37
	VoxNet	28.46	1.07	25.28	1.04	2.26
50 micrometer sampling step						
no IC	RNet	7.99	0.35	5.29	0.47	0.25
	VoxNet	16.05	0.54	8.71	0.58	0.60
IC	RNet	26.89	1.00	28.40	1.00	1.55
	VoxNet	28.12	1.21	30.72	1.29	3.07

TABLE III
VALIDATION MSE IN $e-06$ $(mm^3)^2$ FOR RNET AND VOXNET.

Type		A	B	C	D	E
20 micrometer sampling step						
no IC	PointNet	2.343	33.968	0.618	1.623	0.800
IC	PointNet	39.57	0.265	22.025	0.094	1.840

TABLE IV
VALIDATION MSE IN $e-03$ $(mm^3)^2$ FOR POINTNET.

C. Volume estimation evaluation

Although the volume estimation task that we focus on can have a wider industrial applicability, to the best of our knowledge it has not been studied before. Thus there are no obvious alternative deep architectures that we can compare with. To this end, we have adjusted two well known methods, namely VoxNet [26] and PointNet [29], to our volume estimation task. For VoxNet we have replaced its final classification layer with a fully connected layer in order to produce a single volume estimation whereas for PointNet we followed the same procedure for its classification sub-network. Also since PointNet takes as input 1024 3D points we have uniformly subsampled each of our $20\mu m$ 3D scans to sample this number of points.

The MSE over the different validation sets is reported in Tables III and IV. Notice that the results for PointNet (Table IV) are much worse compared to VoxNet and RNet (Table III). Although PointNet has shown excellent results for 3D

Scanning Step	Scanning Time	Prediction Time	Total Time
$20\mu m$	2620	470	**3090**
$50\mu m$	1181	470	**1651**

TABLE V
PCB TOTAL SCANNING & INSPECTION TIME IN *secs*.

object classification, it performs poorly in the current task and its architecture needs to be further altered. Probably in its current form it cannot encode accurately the subtle differences between different quantities of glue from direct point cloud input. RNet outperforms VoxNet as well in almost all experiments.

As is anticipated the error is significantly larger after dies have been attached since glue mass can not be directly scanned and inevitably there are discrepancies in the ground truth values as explained in Section V-B. Nonetheless, test error is not substantially affected by having a larger sampling step, showcasing the ability of RNet and to a lesser degree VoxNet to infer the volume of glue even from a sparser point cloud. Having a larger sampling step is important for accelerating the scanning process and not delaying the PCB manufacturing process. The required scanning and prediction time for a single PCB with a step of $20\mu m$ or $50\mu m$ is shown in Table V. During prediction point clouds are examined one by one and not in a batch allowing an earlier notification in case of a defect. In both cases the total time is below the critical time threshold of one hour after which glue viscosity drops and any repair activity becomes much more difficult.

While Tables III, IV and V are useful for comparing different experimental setups, from the end-user's point of view it is more important to examine the ability of the trained models to discern the fluctuations of glue quantity. In this regard, Figure 11 shows the performance of VoxNet and RNet for each test set in comparison with the ground truth. The vertical axis refers to the volume of glue, while the lateral one to the index of the test set sample. On each plot the red curve shows the ground truth glue volume while the green and blue curves the predictions of RNet and VoxNet respectively. We have excluded PointNet from this comparison since its high MSE would hinder visualization. Each test set has been sorted according to the annotated quantity of glue in the respective samples, thus the red curves are monotonically decreasing. The diagrams of Figure 11 verify Table III since there is no substantial difference when increasing the scanning step while accuracy drops after die attachment. Also RNet obviously follows more tightly ground truth in several instances (e.g. type E unattached) whereas in other cases is covered by VoxNet due to VoxNet's wider fluctuations. However in general, the trained models can indeed capture the different levels of glue since their predictions follow the ground truth curves. Of course when dies are attached there are stronger deviations from the ground truth, especially for type B and D glue depositions where the scanned area is smaller and deposited quantity of glue is less. Nonetheless the general downward trend is efficiently captured in all cases. Even after die attachment RNet can infer glue volume from the exceeding glue around the die and from the elevation of the die above
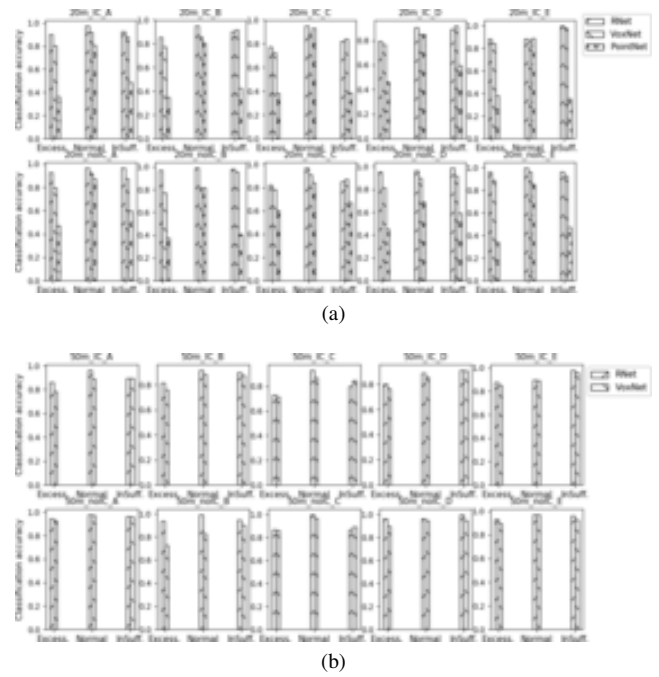

(a)

(b)

Fig. 12. Classification results for RNet, VoxNet and PointNet. RNet outperforms the other architectures in almost all cases.

the LCP substrate due to the intermediate glue layer, therefore providing meaningful predictions for fault diagnosis.

D. Fault diagnosis evaluation

Volume estimations can be used for automated fault diagnosis based on predefined threshold of normal quantities. To this end, using expert knowledge we have defined such threshold and evaluated the performance of RNet, VoxNet and PointNet for all experiment scenarios. In Figures 12a and 12b we see the classification accuracy for RNet using a $20\mu m$ or $50\mu m$ scanning step. Again RNet outperforms VoxNet and PointNet with an average classification accuracy of 91.82 % compared to 86.42 % and 58.33 % respectively.

VII. CONCLUSIONS

In this paper a fault diagnosis system is proposed for inspecting glue dispensation and die attachment on PCBs. To realize this, a custom scanning module has been built using a commercially available laser scanner and two high-accuracy linear stages while a hardware triggering mechanism has been implemented in order to accelerate the scanning process. This paper also introduces RNet architecture in order to monitor a geometric quantity, namely the volume of glue from the scanned point clouds corresponding to different regions of interest on a PCB. To train and deploy the 3DCNN data parameterization and augmentation steps have been introduced.

The efficiency of the system is demonstrated through several experiments on different types of glue regions. A main finding of our work is that glue volume can be estimated even after dies have been attached and most of the glue is occluded. Although the error increases in this case, the predictions of RNet are still useful for quality control. This enables the

automatic inspection at a later stage, after dies are attached thus postponing a time consuming scanning process that might affect the viscosity of glue especially in even more complex PCBs. Another interesting outcome of our research is that prediction accuracy is not significantly affected by a larger scanning step as similar performance is noticed for sparser point clouds at a significantly reduced scanning time.

As a next step it would be interesting to deploy and evaluate the proposed framework in other industrial use cases where a geometric parameter needs to be monitored and further research challenges would arise that have not been addressed in the current work. Such challenges could be related to the misalignment of the sensor and specimen that could result in partial scans and the detection of multi-scale defects that would require adjustments in both data parameterization and network architecture.

References

[1] A. Aakerberg, K. Nasrollahi, and T. Heder, "Improving a deep learning based rgb-d object recognition model by ensemble learning," in *International Conference on Image Processing Theory, Tools and Applications (IPTA)*, 2017.

[2] M. R. Loghmani, S. Rovetta, and G. Venture, "Emotional intelligence in robots: Recognizing human emotions from daily-life gestures," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[3] B. Liu, X. Yu, A. Yu, and G. Wan, "Deep convolutional recurrent neural network with transfer learning for hyperspectral image classification," *Journal of Applied Remote Sensing*, vol. 12, 2018.

[4] T. H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "Pcanet: A simple deep learning baseline for image classification?" *IEEE Transactions on Image Processing*, vol. 24, no. 12, 2015.

[5] J. Lee, T. Kim, J. Park, and J. Nam, "Raw waveform-based audio classification using sample-level CNN architectures," *CoRR*, vol. abs/1712.00866, 2017. [Online]. Available: http://arxiv.org/abs/1712.00866

[6] A. Vafeiadis, D. Kalatzis, K. Votis, D. Giakoumis, D. Tzovaras, L. Chen, and R. Hamzaoui, "Acoustic scene classification: from a hybrid classifier to deep learning," in *Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE2017)*, 2017.

[7] P. Ghamisi, B. Höfle, and X. X. Zhu, "Hyperspectral and lidar data fusion using extinction profiles and deep convolutional neural network," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, DOI 10.1109/JSTARS.2016.2634863, no. 6, 2017.

[8] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2774777, no. 7, 2018.

[9] O. Costilla-Reyes, P. Scully, and K. B. Ozanyan, "Deep neural networks for learning spatio-temporal features from tomography sensors," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2716907, no. 1, 2018.

[10] P. Li, Z. Chen, L. T. Yang, Q. Zhang, and M. J. Deen, "Deep convolutional computation model for feature learning on big data in internet of things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, 2018.

[11] W. Sun, R. Zhao, R. Yan, S. Shao, and X. Chen, "Convolutional discriminative feature learning for induction motor fault diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, 2017.

[12] L. Yao and Z. Ge, "Deep learning of semisupervised process data with hierarchical extreme learning machine and soft sensor application," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2733448, no. 2, 2018.

[13] W. Yan, D. Tang, and Y. Lin, "A data-driven soft sensor modeling method based on deep learning and its application," *IEEE Transactions on Industrial Electronics*, vol. 64, DOI 10.1109/TIE.2016.2622668, no. 5, 2017.

[14] J. Pan, Y. Zi, J. Chen, Z. Zhou, and B. Wang, "Liftingnet: A novel deep learning network with layerwise feature learning from noisy mechanical data for fault classification," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2767540, no. 6, 2018.

[15] C. Shi, G. Panoutsos, B. Luo, H. Liu, B. Li, and X. Lin, "Using multiple-feature-spaces-based deep learning for tool condition monitoring in ultraprecision manufacturing," *IEEE Transactions on Industrial Electronics*, vol. 66, DOI 10.1109/TIE.2018.2856193, no. 5, 2019.

[16] B. Luo, H. Wang, H. Liu, B. Li, and F. Peng, "Early fault detection of machine tools based on deep learning and dynamic identification," *IEEE Transactions on Industrial Electronics*, vol. 66, DOI 10.1109/TIE.2018.2807414, no. 1, 2019.

[17] M. He and D. He, "Deep learning based approach for bearing fault diagnosis," *IEEE Transactions on Industry Applications*, vol. 53, no. 3, 2017.

[18] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric locomotive bearing fault diagnosis using a novel convolutional deep belief network," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2745473, no. 3, 2018.

[19] M. Zhao, M. Kang, B. Tang, and M. Pecht, "Deep residual networks with dynamically weighted wavelet coefficients for fault diagnosis of planetary gearboxes," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2762639, no. 5, 2018.

[20] H. Oh, J. H. Jung, B. C. Jeon, and B. D. Youn, "Scalable and unsupervised feature engineering using vibration-imaging and deep learning for rotor system diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2752151, no. 4, 2018.

[21] F. Chen and M. R. Jahanshahi, "Nb-cnn: Deep learning-based crack detection using convolutional neural network and naive bayes data fusion," *IEEE Transactions on Industrial Electronics*, vol. 65, DOI 10.1109/TIE.2017.2764844, no. 5, 2018.

[22] Z. Hu and P. Jiang, "An imbalance modified deep neural network with dynamical incremental learning for chemical fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 66, DOI 10.1109/TIE.2018.2798633, no. 1, pp. 540–550, 2019.

[23] T. Nakazawa and D. V. Kulkarni, "Wafer map defect pattern classification and image retrieval using convolutional neural network," *IEEE Transactions on Semiconductor Manufacturing*, vol. 31, no. 2, 2018.

[24] G. Tello, O. Y. Al-Jarrah, P. D. Yoo, Y. Al-Hammadi, S. Muhaidat, and U. Lee, "Deep-structured machine learning model for the recognition of mixed-defect patterns in semiconductor fabrication processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 31, no. 2, 2018.

[25] K. B. Lee, S. Cheon, and C. O. Kim, "A convolutional neural network for fault classification and diagnosis in semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 30, no. 2, 2017.

[26] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[27] X. Xu, A. Dehghani, D. Corrigan, S. Caulfield, and D. Moloney, "Convolutional neural network for 3d object recognition using volumetric representation," in *2016 First International Workshop on Sensing, Processing and Learning for Intelligent Machines (SPLINE)*, 2016.

[28] J. Huang and S. You, "Point cloud labeling using 3d convolutional neural networks," in *International Conference on Pattern Recognition (ICPR)*, 2016.

[29] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, DOI 10.1109/CVPR.2017.16, 2017.

[30] P. Wang, Y. Liu, Y. Guo, C. Sun, and X. Tong, "O-cnn: Octree-based convolutional neural networks for 3d shape analysis," *ACM Transactions on Graphics*, vol. 36, no. 4, 2017.

[31] L. Shao and P. Xu, "Neural network for 3d object recognition," Stanford University, Computer Science Department, Tech. Rep., 2016.

[32] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, DOI 10.1109/CVPR.2015.7298801, 2015.

[33] "Conopoint-10, 3d non contact measurement," http://www.optimet.com/conopoint-10.php, accessed: 2018-07-04.

[34] "Xps-rl universal high-performance motion controller/driver," https://www.newport.com/f/xps-rl-universal-high-performance-motion-controller, accessed: 2018-07-04.

[35] "Linear metrology stages," https://www.newport.com/f/fms-metrology-stages, accessed: 2018-07-04.

[36] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, 1981.

[37] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[38] M. B. M. Kazhdan and H. Hoppe, "Poisson surface reconstruction," in *Eurographics Symposium on Geometry Processing*, 2006.

[39] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[40] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[41] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *Conference and Workshop on Neural Information Processing Systems*, 2017.

Nikolaos Dimitriou received the diploma in electical and computer engineering from the Aristotle University of Thessaloniki (AUTh) in 2007. From the same institution he received the PhD degree in 2014. Since the July of 2014, he works a postdoctoral research associate for the Informatics and Telematics Institute of the Centre for Research and Technology Hellas in Thessaloniki. His research focuses on the areas of machine vision and deep learning with a focus on industrial applications.

Lampros Leontaris received the diploma in electical and computer engineering from the Aristotle University of Thessaloniki (AUTh) in 2016. He works as a research assistant at the Information Technologies Institute of the Centre for Research and Technology Hellas in Thessaloniki (CERTH/ITI) since March 2017.His research interests include computational intelligence methods, machine learning and fuzzy systems.

Thanasis Vafeiadis received his diploma in mathematics in 2002, master?s degree in Statistics and Operation Research in 2004 and PhD in Departmental Investigation of Linear Trend in Time Series in 2011, from Aristotle University of Thessaloniki (AUTh). He is currently working as postdoctoral research associate in Informatics and Telematics Institute of the Centre for Research and Technology Hellas in Thessaloniki. His main research interests focus on the analysis of linear trend in time series, in machine learning, signal processing and data analytics.

Dimosthenis Ioannidis received his diploma in electrical and computing engineering in 2000 master?s degree in advanced communication systems and engineering in 2005 from Aristotle University of Thessaloniki (AUTh). He is working from October 2006 till now as a Research Associate in Informatics and Telematics Institute of the Centre for Research and Technology Hellas in Thessaloniki. His main research interests are in the areas of biometrics (gait and activity-related recognition), 3-D data processing, and web semantics.
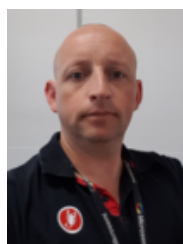
Tracy Wotherspoon graduated from Birmingham City University in 1984 with a HND in Electrical and Electronic Engineering. Following graduation he worked in the process control industry before transferring to the design of broadcast quality sound mixers with Audio Developments (Walsall) and latterly Coomber Electronic Equipment (Worcester). Following Microsemi's, then Zarlink's, move into medical electronic assembly he was awarded an MSc in Clinical Engineering from Cardiff University specialising in medical RF. He has patents in the field of energy harvesting, as well as several publications and book chapters in medical implant communications and harsh environment electronics.

Gregory Tinker graduated with a HNC in Electronics and Communications from Plymouth University in 1993. Following graduation he worked at Meggitt electronics (Holsworthy) as a production engineer until 1998 and subsequently moved to Mitel (south Wales) as an equipment engineer. The company has changed name several times since and so have the roles he has fulfilled in 20 years of service including SMT process engineer, equipment and facilities senior engineer and currently holds the position of senior process engineer.

Dimitrios Tzovaras (SM13) received the Diploma and the Ph.D. degree in electrical and computer engineering from Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1992 and 1997, respectively. He is the Director at the Information Technologies Institute of the Centre for Research and Technology Hellas. His main research interests include visual analytics, 3D object recognition, search and retrieval, behavioral biometrics, assistive technologies, information and knowledge management, computer graphics, and virtual reality.