

Embedded product authentication codes in additive manufactured parts: Imaging and image processing for improved scan ability

Fei Chen¹, Jaime Zabalza², Paul Murray², Stephen Marshall², Jian Yu³, Nikhil Gupta^{1,*}

¹Composite Materials and Mechanics Laboratory, Mechanical and Aerospace Engineering Department, Tandon School of Engineering, New York University, Brooklyn, NY 11201 USA

²Department of Electronic and Electrical Engineering, University of Strathclyde, 204 George Street, Glasgow, G1 1XW, UK

³US Army Research Laboratory, CCRL-WMM-D, APG, MD 21005 USA

Abstract

The layer-by-layer printing process of additive manufacturing methods provides new opportunities to embed identification codes inside parts during manufacture. These embedded codes can be used for product authentication and identification of counterfeits. The availability of reverse engineering tools has increased the risk of counterfeit part production and new authentication technologies such as the one proposed in this paper are required for many applications including aerospace components and medical implants and devices. The embedded codes are read by imaging techniques such as micro-Computed Tomography (micro-CT) scanners or radiography. The work presented in this paper is focused on developing methods that can improve the quality of the recovered micro-CT scanned code images such that they can be interpreted by standard code reader technology. Inherent low contrast and the presence of imaging artifacts are the main challenges that need to be addressed. Image processing methods are developed to address these challenges using titanium and aluminum alloy specimens containing embedded quick response (QR) codes. The proposed techniques for recovering the embedded codes are based on a combination of Mathematical Morphology and an innovative denoising algorithm based on optimal image filtering techniques. The results show that the proposed methods are successful in making the codes scannable using readily available smartphone apps.

Key words: additive manufacturing; 3D printing; cyber-physical system; product authentication; image processing.

1. Introduction

Additive manufacturing (AM) is now established as one of the mainstream manufacturing methods. Parts can be additively manufactured in near-net or net shape due to the advancements in the capabilities of modern 3D printers [1, 2]. The aerospace industry is leading the way in using AM for part development and production due to the low production run of aircraft parts [3-6]. Due to the possibility of customizing every single part in a production run, the medical field is also rapidly adopting AM for developing customized, patient-specific implants and medical devices [7-11]. Many aerospace and medical grade materials (Ti-6-4 alloy, 316 stainless steel) are now available as feed materials for 3D printing to allow the manufacturing of parts that can be qualified for deployment [12-14]. Recent studies have focused on optimizing the processing

* Corresponding author. Phone: +1-646-997 3080, email: ngupta@nyu.edu.

parameters to improve the quality of the printed part, characterize defects, and develop secondary processes such as hot isostatic pressing [15, 16]. In addition, reduction in surface roughness of the manufactured part is also of interest so that the part does not require any polishing or finishing application before deployment [17, 18].

Reducing the weight of manufactured parts is an important consideration in aerospace applications and other fields. Parts with hierarchical structures have been designed using optimization methods to provide desired mechanical properties at very low weight [19, 20]. These highly optimized parts with multiscale features could not be manufactured by any approach other than AM methods. Significant design and optimization effort is invested in developing such parts. Similarly, patient specific implants are being developed in the medical field using computerized tomography (CT) scan and other imaging data of the patient and using solid modeling methods to develop the implant design [8, 10]. In other applications, 3D printed models of the brains of patients who have tumors are being used for planning complex surgeries [21].

A common theme among the examples of the current state-of-the-art in AM is that a significant effort is invested in developing the part design and any failures in the part or its manufacturing can jeopardize the safety and wellbeing of people as the ultimate end-users [22-26]. Hence, the quality of the part is critical in these applications. It can also be noted that the risks of counterfeit part production and reverse engineering are major threats in these areas [24]. Even genuinely acquired parts can be reverse engineered and then subjected to counterfeit production. As the availability of 3D printers increases and imaging, scanning and reverse engineering tools are being developed in parallel, new strategies are required to integrate security with the part design from the outset [27, 28].

1.1 Current security approaches

A number of security strategies have been proposed, which are summarized in a table presented in a recent article [28]. These strategies include use of quantum dots to create products with unique optical signature for identification of genuine products [29]. Other methods include securing the supply chain using blockchain to identify counterfeits [30]. As in the case of security, each method provides security options for certain products and supply chains. In our recent approach, tracking codes were designed in Computer Aided Design (CAD) models and embedded inside manufactured parts during 3D printing [28, 31]. The study used the example of a QR code, one of the most widely used tracking codes around the world. The layer-by-layer printing process allows embedding such information inside the part, which can be retrieved using a CT scan or other imaging technique. These codes should be small so that they do not compromise the integrity or mechanical strength of the part. Many AM methods, including powder based technologies such as selective laser sintering/melting, can print parts with resolutions as fine as a few microns and the embedded codes can be smaller than $1 \times 1 \text{ mm}^2$ size.

The QR code is only an example of the possible embedded codes but such codes can be designed to be much smaller with only 4 or 5 points strategically placed at geometrically identifiable locations giving enough identification information about the product. In addition, micro-QR codes can be a quarter in size compared to the QR codes. Since the corner squares in QR codes

are used only for positioning, these large sections can be eliminated to significantly reduce the number of points in parts, where mechanical strength may be a concern.

However, embedding such small codes is challenging for imaging methods to resolve. In micro-CT scanning, invariably there are some imaging artifacts that need to be removed in order to accurately recover the true embedded code for scanning. In addition, image contrast may be low in the code because features are also filled with the powder of the modeling material. As a further challenge for accurate imaging, the embedded codes are intentionally deconstructed into a number of parts and embedded in different layers during printing for the purpose of obfuscation and security. As a result, accurate extraction and reconstruction of the code requires aligning the imaging slices and viewing them together to extract and reconstruct the true embedded code before repairing any sectors that are not captured properly by the imaging methods. The present work is focused on developing new image analysis methods that can address these challenges and provide an accurate and legible QR code that can be scanned without error using a standard scanning device, such as a smartphone based app. While the present study uses QR codes as the embedded identification code, this work is more general in scope and the develop methods could be applied to other types of identification codes such as bar codes or other proprietary markings.

1.2 Threat Model

Understanding the threat vectors is an important aspect in developing security methods. Since all the steps, from the initial CAD model development through to the final part manufacturing using a 3D printer, are dependent on software, cybersecurity challenges are faced at every step [22, 27, 32-35]. However, not all threat vectors can be addressed by a single security scheme.

Cybersecurity methods such as file encryption, password protection, and network control cannot protect against reverse engineering from a genuinely acquired part or counterfeit production of a design that infringes the intellectual property. The risks of counterfeit and unauthorized parts can be significant [36-38]. For example, counterfeit parts produced using substandard materials may have weight and surface finish similar to the genuine part but the microstructure may be different leading to substandard performance. Furthermore, quality control and testing procedures may not have been rigorously applied to counterfeit parts before they are sold or deployed. It may be difficult to positively identify a counterfeit from a genuine part because they may have been produced on the same type of 3D printer. This kind of outcome may result in associated liability issues and a well-established and reputable company may suffer through no fault of their own due to the failure of substandard parts fraudulently produced by others.

Embedding codes inside the manufactured parts serves several purposes. Firstly, the printed part will have a unique code for positive identification. Secondly, reverse engineering of the part using surface scanning methods would not produce the internal code. Thirdly, unless the required imaging facilities are also available to the counterfeit producers, such codes cannot be detected, identified and reproduced. Hence, embedding codes in multiple layers of manufactured parts can provide a robust measure to protect against such threat scenarios.

2. Materials and Methods

2.1 CAD model design

SolidWorks 2018 was used for solid modeling in this work. The QR code to be embedded was created by an online QR code generator to encode the website URL “engineering.nyu.edu/composites” as given in Figure 1. The QR code pattern was imported into SolidWorks to generate a 3D model using the “Sketch” and “Extrude” functions. This three-dimensional code was embedded inside a solid cube for the purpose of printing and testing. Figure 2a and b show the three-dimensional code inside the cube. As a possible obfuscation method, a second model was also developed where the code is sliced into three segments and these segments are embedded at different depths in the solid cube as shown in Figure 2c and d. The isometric view of the code shows three segments but the plan view reveals the entire code that is scannable. Following the standard AM process chain, the CAD files were exported to STL format [27]. The STL geometry was sliced and then converted to G-code for printing the parts. The one-layer code was printed using Ti Gr23(A) alloy, whereas the three-layer code was printed using AlSi10Mg alloy.



Figure 1. The QR code generated for this study for the website link engineering.nyu.edu/composites.

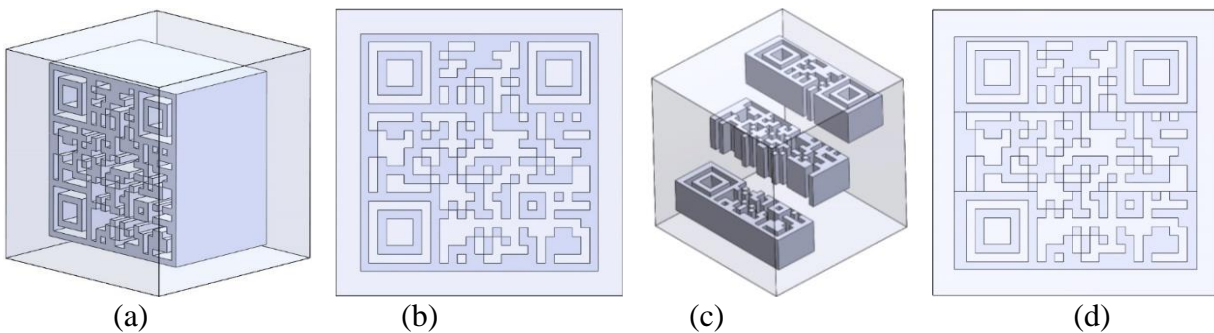


Figure 2. CAD models of cube ($10 \times 10 \times 10 \text{ mm}^3$) with embedded (a) one-layer QR code shown in isometric view and (b) front view, and (c) three-layer QR code with isometric view and (d) front view.

2.2 Additive manufacturing of specimens

The cubes containing a one-layer QR code were printed on a 3D Systems Prox DMP 300 metal printer using commercial grade Ti Gr23(A) titanium alloy. The cube size is $10 \times 10 \times 10 \text{ mm}^3$. The embedded QR code has a cross-sectional area of $7.71 \times 7.71 \text{ mm}^2$ with a 6 mm thickness. One of the printed cubes was cut open and cleaned to allow visual observation of the embedded code as shown in Figure 3. Six cubes were printed and some of these cubes were cut for physical verification of features and photography. Only one titanium alloy cube was CT-scanned and used in the experiments.

The cubes containing the three-layer QR code were printed using EOS 270 metal 3D printer and AlSi10Mg alloy. The size of the cube is $10\times 10\times 10\text{ mm}^3$. The QR code was built with $7.73\times 7.72\text{ mm}^2$ cross section area and 2 mm thickness and then sliced into three parts of 7.73×2.47 , 7.73×2.78 , and $7.73\times 2.47\text{ mm}^2$. Each part is offset by 3 mm from the adjacent part in the thickness direction. Again, six cubes were printed and some of them were cut for verification of physical features but only one aluminum alloy cube was CT-scanned in the experiments.

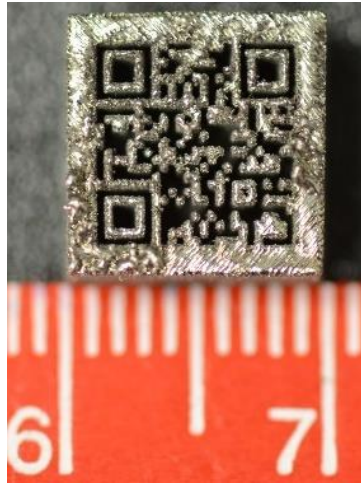


Figure 3. The cross section view of the 3D printed cube with Ti Gr23(A) alloy, showing the embedded QR code (one layer).

2.3 *Micro-CT scan imaging*

The micro-CT scanning was conducted using a Bruker SkyScan 1172 system. An Al+Cu filter was used to reduce any beam-hardening artifacts when scanning metallic parts and the X-ray voltage and intensity were maintained at 100 kV and $100\text{ }\mu\text{A}$, respectively. The scan resolution in one-layer Ti-alloy specimens was obtained as $10.08\text{ }\mu\text{m}/\text{pixel}$. The scan resolution is the smallest detail that can be captured during the scan. The Ti-alloy data consists of images with a spatial resolution of 2000×1332 pixels, and 499 images were acquired using a rotation step of 0.4° . The presence of unsintered powder of the same metal used for QR code causes low contrast in the micro-CT scan images since there is only a small density difference between the sintered and unsintered parts of the cube. A representative slice of the micro-CT scan image is presented in Figure 4, where a QR code pattern and the low contrast appearance of the code when imaged can be visualized.

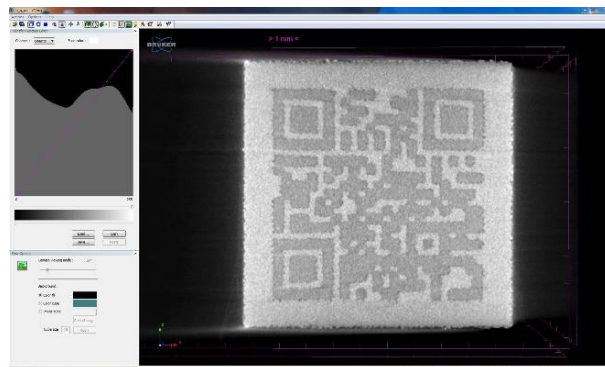


Figure 4. Cross-sectional micro-CT scan image of a 3D printed cube ($10\times 10\times 10\text{ mm}^3$) of Ti Gr23(A) alloy with embedded QR. code part ($7.7\times 7.7\times 6\text{ mm}^3$) displayed using CTVox volume rendering program.

The scan resolution in Al-alloy specimens was obtained as $8.72 \mu\text{m}/\text{pixel}$. The Al-alloy imaging data has a resolution of 2000×1332 pixels, with 333 images acquired using a rotation step of 0.6° while the specimen was subjected to in-plane rotation of 360° . A representative set of micro-CT scan image slices containing each of the three QR code segments is shown in Figure 5.

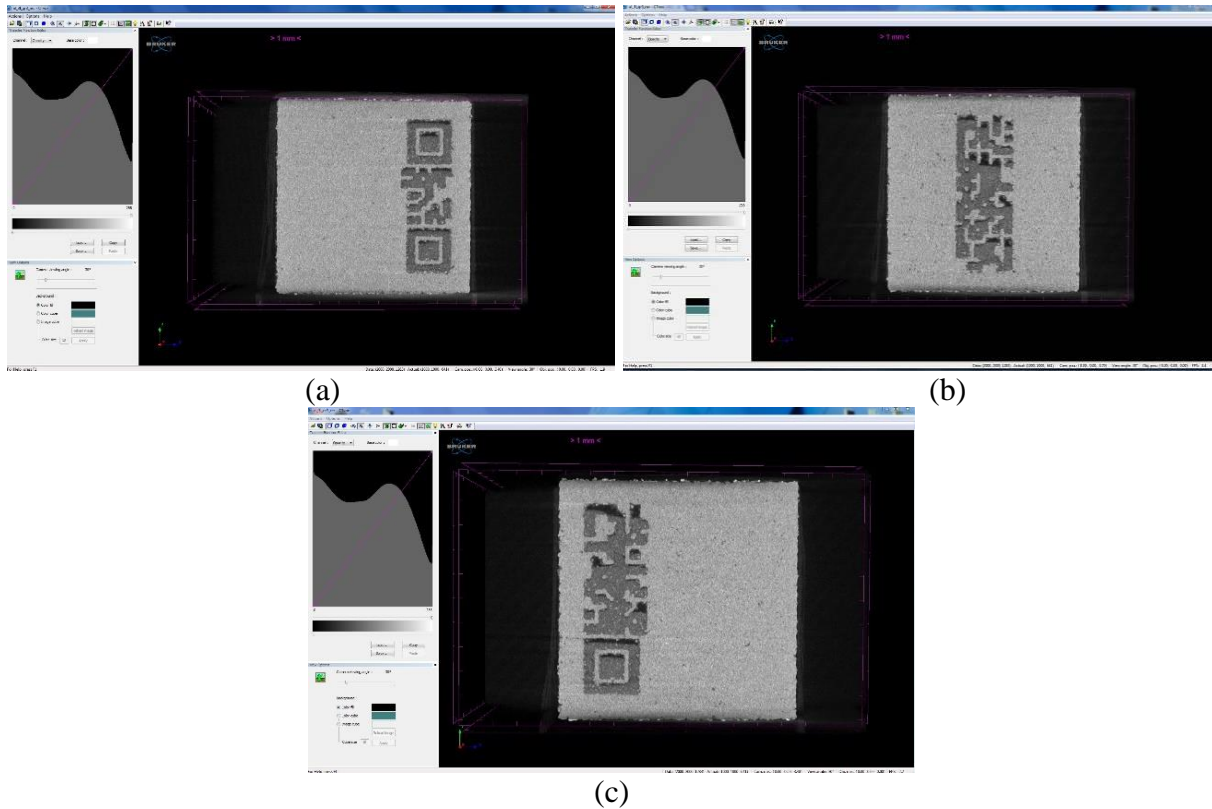


Figure 5. Cross-sectional micro-CT scan images of (a) QR segment I, (b) segment II, and (c) segment III of a 3D printed metal (AlSi10Mg) cube ($10 \times 10 \times 10 \text{ mm}^3$) with embedded QR (3 layers) code part ($7.7 \times 7.7 \times 2 \text{ mm}^3/\text{layer}$) as opened in CTVox volume rendering program.

3. Results and Discussion

This section first presents the image processing method that is developed for the CT scan images and then applies this method to the model datasets.

3.1 Image Processing

The previous image processing method proposed in [31] for extracting and reconstructing the QR Code was insufficient and standard QR reader technology could not recover the encoded link. After a closer analysis of the previous method in [31], one possible reason for such a high discrepancy is that the processed image is actually made from 3 different layers; each of them is placed at a different depth inside the cube similar to the three-layer code of the present work. Though the actual size of QR code at different layers is designed and 3D printed to be the same, the apparent sizes of these codes located at different layers appear to be different from their actual sizes when imaged using the process described. As a result, parts of the code printed on layers nearer to the sensor appear larger in the acquired images while layers further in the distance appear smaller when viewed from one direction. This difference may also occur because the specimen is rotated continuously with respect to the imaging sensor and the code may not be

directly in-line with the sensor orientation in a given image. The angle between the code and the sensor may provide a similar effect. The image presented in Figure 6 shows an example where features present in the left side of the code image appear to be smaller than those in the right side of the image. To address this issue, a new image processing methodology is proposed so that effective QR code recovery can be achieved. The proposed method performs well by resizing codes at different locations despite the fact that the code is embedded in different layers at different distances from the imaging sensor when positioned at the optimal viewing orientation.

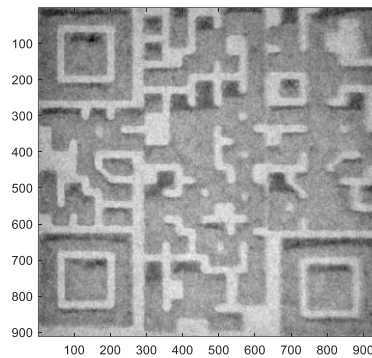


Figure 6. A QR code image obtained from micro-CT scan image stack. The features present in the left side of the image appear to be smaller than the features in the right side.

Implemented in MATLAB, our new image processing method has been designed to optimize performance without significantly increasing the complexity of the algorithm proposed in [28]. Our new image processing method has three stages: (i) data conditioning, (ii) main processing for blind QR code recovery, and (iii) advanced post-processing for de-noising. The first two stages are required for accurate performance while step (iii) can be considered optional and aims to refine the results and improve readability as required. The optional post-processing stage requires prior training and, if applied, means that the recovery process is no longer blind i.e. it requires knowledge of the original embedded QR code for training and correct operation. It is demonstrated that the proposed approach can be applied to recover readable QR codes from various materials with QR codes printed in varying numbers of layers. In all cases, the correct QR codes can be extracted successfully in such a way that they can be read using standard QR code readers thus significantly improving upon the approach described in [31].

3.1.1 Data conditioning

The first stage of the image processing method requires selection of an image from the micro-CT scan data for processing. The image should be selected such that the embedded QR code is oriented perpendicularly, or as close to being perpendicular as possible in the given data set, to the viewer. This ensures that the embedded QR code can be seen correctly and the image used is denoted the Optimally Oriented Image (OOI). For this work, the selection of the OOI was performed manually by visual inspection and the OOI was cropped manually to the extremes of QR code boundaries. The resulting grayscale image, originally stored in a 16-bit unsigned integer array, is then converted to a double-precision array (32-bit float) and normalized to the range [0-1] where black = 0 and white = 1. Figure 7 shows the selected OOI and the result of cropping of this image for further processing using the aluminum alloy data.

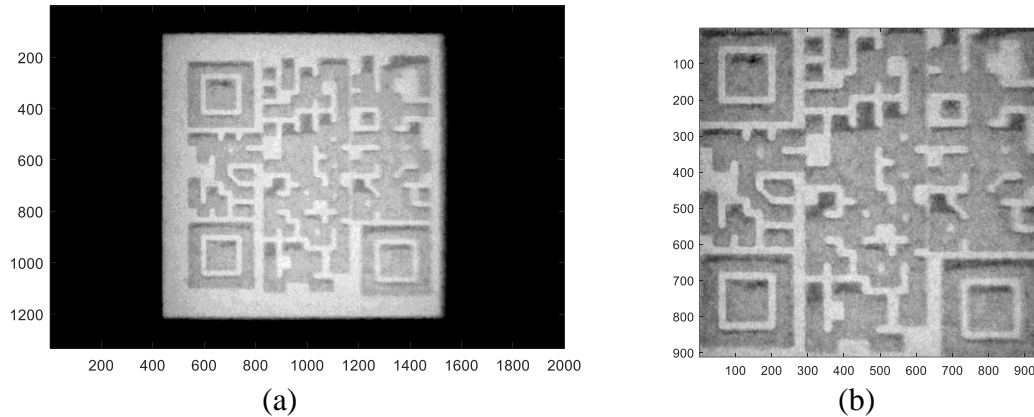


Figure 7. Conditioning stage for the aluminum data: (a) selected image (2000×1332 pixels) from the cube in which the QR code is oriented perpendicularly to the viewer, and (b) cropped region from the previous image adjusted to the QR code boundaries. This cropped OOI has an approximated size of 900×900 pixels.

3.1.2 Blind QR code recovery

After obtaining a cropped OOI, the main image processing tasks are applied – a workflow summarizing the algorithm is shown in Figure 8. First, Contrast-Limited Adaptive Histogram Equalization (CLAHE) [39] is applied to the resulting image from the first stage (Figure 7(b)). CLAHE is a technique used to achieve a more uniform distribution of the pixel values across the image range, which is [0-1] in this work. The new distribution highlights the separation between the pixels values, increasing the image contrast. Therefore, this equalization aims to optimize the image contrast and highlight the differences between black and white pixels in the QR code to be extracted. Next, each QR layer (or column as shown in Figure 8) in the equalized image is cropped and processed individually. This allows compensation for any discrepancy that may arise due to scaling in the three images obtained from different layers. Although the QR code in the Ti-alloy data was embedded in a single layer, the same procedure was applied for consistency in evaluations.

For each layer (column), a morphological operation known as Dilation [40] is applied to reinforce the division among black and white pixels given that, as can be seen in Figure 8, black pixels appear to be inflated in the image when compared to the original QR code. Dilation operation is generally used for adding pixels in the boundaries of objects, and here it simply aims to expand the size of the white pixels with relation to the black ones. This Dilation was implemented by using a 21×21 square structuring element in all cases. This structuring element defines the specific effect of Dilation on the image. Therefore, a square shape was selected because the QR code intrinsically lays on a square grid, and the size was obtained by visual inspection, looking for a balanced size between black and white pixels, as black pixels originally looked inflated.

Finally, each layer is resized from its current size (900×300 pixels approximately) to grids of 25×8, 25×9 and 25×8, respectively (the central layer is wider than other two), leading to a final 25×25 image (QR code size). The resizing process also includes a binarization, which means transforming the grayscale pixels with values within [0-1] into black = 0 or white = 1. This binarization process considers the value of all original grayscale pixels (after dilation) contributing to the resized (grid) pixels and compares them to a given threshold so that the resized gray pixels can be transformed in either black or white. The thresholds applied here are

obtained based on the related histograms (distribution of pixels across the grayscale range) and visual inspection, with all of them being between 0.8 and 0.95 (original scales are [0-1], black and white, respectively). The entire process is illustrated in Figure 8, which shows the inputs and outputs at each stage as well as an image histogram showing the distribution of pixel intensities and the threshold applied.

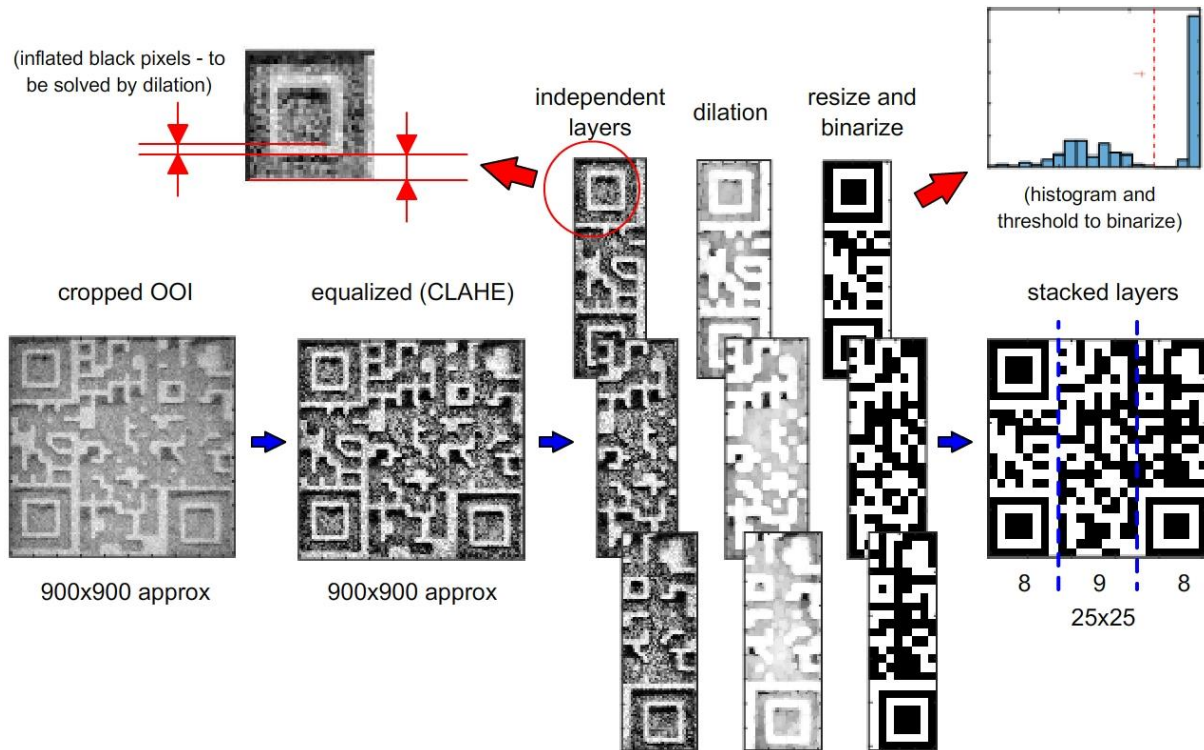


Figure 8. Workflow processing for blind QR code recovery. The cropped OOI from the conditioning stage is equalized for improved contrast. Then, each layer is treated individually, applying a Dilation (21×21 square) plus a resize with binarization step. Finally, layers are stacked to obtain the final 25×25 image.

3.1.3 Advanced post-processing

In practice, the micro-CT acquisition of images can present some noisy/blurred regions which may not be truly representative of the original QR code. In these cases, the method described in Section 3.1.2 cannot always ensure a completely successful recovery simply because the “source” image is distorted by noise. Therefore, an additional post-processing step is designed and implemented to refine the results and, ultimately, recover the QR code with no error at all. Unlike the previous two stages, the final stage of post-processing uses *a priori* knowledge about the specific QR code that has been embedded within the manufactured part. While our post-processing step significantly improves the quality and readability of the recovered QR code, the requirement to use prior knowledge of the code itself means that this additional step cannot be applied to detect other QR codes without prior training. However, given the nature of the application and, since the true QR code (or other embedded pattern) would always be known in advance, it is prudent to exploit this additional information in order to improve performance.

From an image processing perspective, the errors that are present in the image following the initial processing can be considered as noise. By extension, the entire process of embedding QR codes into multiple layers of manufactured parts, imaging them using micro-CT scanners, and

extracting the embedded codes can be considered a noisy process. While many image de-noising filters exist to suppress noise e.g. Average Filter, Median Filter, Gaussian Smoothing, they all process a given image in the same way; regardless of the source or type of noise. For the present application, it is known exactly what the recovered image should look like. It is also known how the image data will be gathered to recover the QR code. Therefore, rather than using a generic filter like those listed above, an optimal filter [41] is designed here, which aims to fully restore the embedded QR code images.

An optimal filter [39] is a filter which output relies on statistics rather than in mathematical operations. In real life, noisy faxes or faded photocopies can be understood because the human brain has knowledge of the characters of the alphabet and can fill in gaps or ignore noise using experience and logic [39]. An optimal filter implements the same concept. This filter applies an evaluation window to each pixel in the image and studies the neighbor pixels, so that it can learn the structures normally found around misclassified pixels, where these structures are assumed to be inherent in the manufacturing processes and thus repeatable over time. Therefore, training data is required to build the filter so that it can learn about the inherent noisy structures.

The idea is that the optimal filter is trained in a way which aims to undo the system noise that is introduced during image acquisition. For this training, a set of noisy images is required. Ideally, training would be performed on a large set of images from a number of different manufacturing processes, including repetitions. However, this is not currently feasible due to the high time and financial costs associated with manufacturing and image parts like the ones described here. Therefore, the post-processing in this work uses synthetic data based on the assumption that the manufacturing processes involved in embedding QR codes introduce noise in the form of a distorted size ratio between black and white pixels. To obtain a set of representative noisy images, the same methodology described in Section 3.4.2 is applied. However, the original size of the structuring element involved in this method (21x21) is now replaced by disparate sizes, such as 1x1 and 41x41, leading to both insufficient and excessive Dilation (see Figure 9). As a result, the obtained images are not adequately dilated and present notable noise, which can be used to train the filter and learn the noise structure for this particular case.

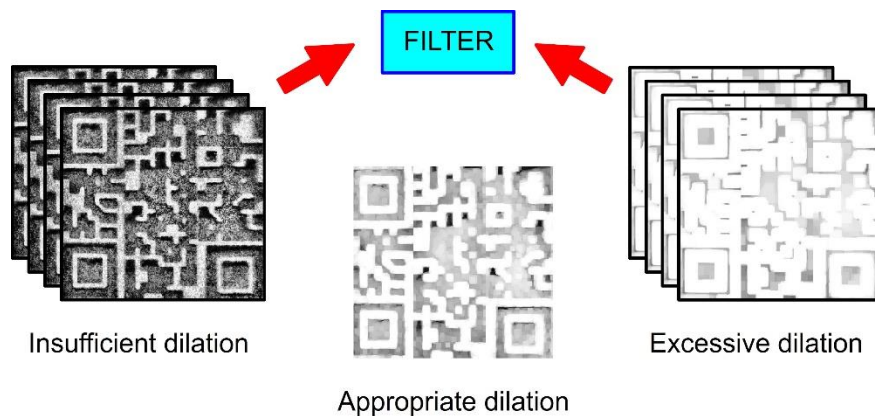


Figure 9. Representation of a set of synthetic images created for training the optimal filter. These images present either insufficient or excessive dilation, aimed to provide a representative noise structure.

To train the filter, the first step is to select a window in terms of size and shape (for this work, square windows are used (see Figure 10) and the performance of a number of windows of increasing size are evaluated). Next, the filter is translated to each point in the noisy images and the pixel coincident with window's origin is compared with the corresponding pixel in the original input image. Using this information, a table can be constructed where different rows correspond to different filter input patterns (as determined by the filter window and its location in the noisy images). As the filter is translated to each point in the noisy images, a record is kept for those times when, given a particular input pattern, the pixel coincident with the filter's origin was not correctly recovered according to the original QR code. By storing those patterns in which the recovered pixel during training was in error, the filter is able to perform an analysis of how the noisy system corrupts the original QR code that is embedded within the material, thus detecting a set of unreliable patterns for each position in the image. Using this information, it is possible to minimize the effects of noise using an optimal filter to correct pixels in the recovered images, based on the data seen during training.

Figure 10 provides an example of how the filter works after being trained. Here, the filter aims to determine whether the pixel denoted as C (central pixel) should be considered reliable or unreliable in the output obtained from the blind QR code recovery. To achieve this, the filter evaluates the pattern found in the neighbor pixels – a binary sequence of 9 values when the size of the window is 3×3. The filter checks for this pattern in the table of unreliable patterns for a given position in the image (generated during the training) and, depending on the search results, it changes the value (unreliable) or leaves it as it is (reliable).

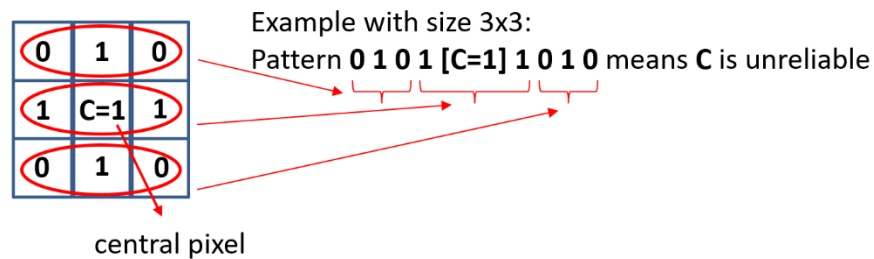


Figure 10. Schematic representation of how the optimal filter works. The pattern found in the neighbor pixels is used as input, so the value of C can be changed if it is considered unreliable based on previous training.

The larger the size $N \times N$ of the window, the larger binary sequence and, therefore, the larger possible number of combinations (2^N) and higher computation cost associated with training and deploying the filter. In this work, different sizes for the filter were implemented, leading to different results as shown in the next section.

3.2 Validation of signal processing method on CT scan data

In this section, the results of applying our new image processing method to reliably extract embedded QR codes from within multiple 3D printed layers of various manufactured parts are presented. To assess the performance, the discrepancy between the original true QR code and the codes recovered by the proposed technique after micro-CT imaging of the manufactured parts are evaluated. Comparisons are made in terms of pixel error (%) when comparing all 625 (25×25) pixels available in the true QR code and the one recovered from the micro-CT scan of each part. Firstly, results from blind recovery are shown; here, no prior information about the original code is used. Then, post-processing results for different configurations of the proposed

optimal filters are provided. In this case, the post-processing includes training using prior information about the specific QR code embedded within the parts being imaged. Finally, an analysis on selecting different (and contiguous) images as OOI in the data conditioning is reported to explore the robustness and usefulness of the proposed techniques.

3.2.1 Discrepancy error from blind recovery

Both titanium and aluminum alloy micro-CT scan images are processed with the methodology described in Section 3.1. Results with no post-processing are shown in Figure 11 and Figure 12 for the titanium and aluminum alloy data, respectively. In both cases, the embedded QR code was recovered with a small error, where just a few pixels (3-6) were in error. This led to errors of 0.96% and 0.48%, much smaller than the error reported in [31] (20.28%) and low enough to allow accurate reading using standard QR code reader apps [42, 43].

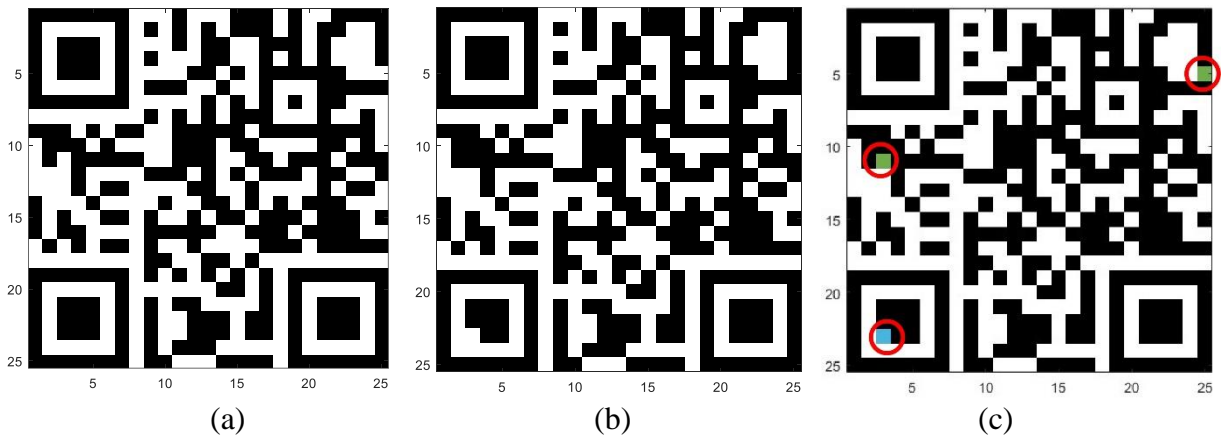


Figure 11. Results for the QR code recovery in titanium: (a) original QR code, (b) obtained QR code, and (c) discrepancy map between original and recovered code, where blue color represents pixels recovered as white when they were originally black, and green color indicates pixels recovered as black when they were actually white in the true embedded code.

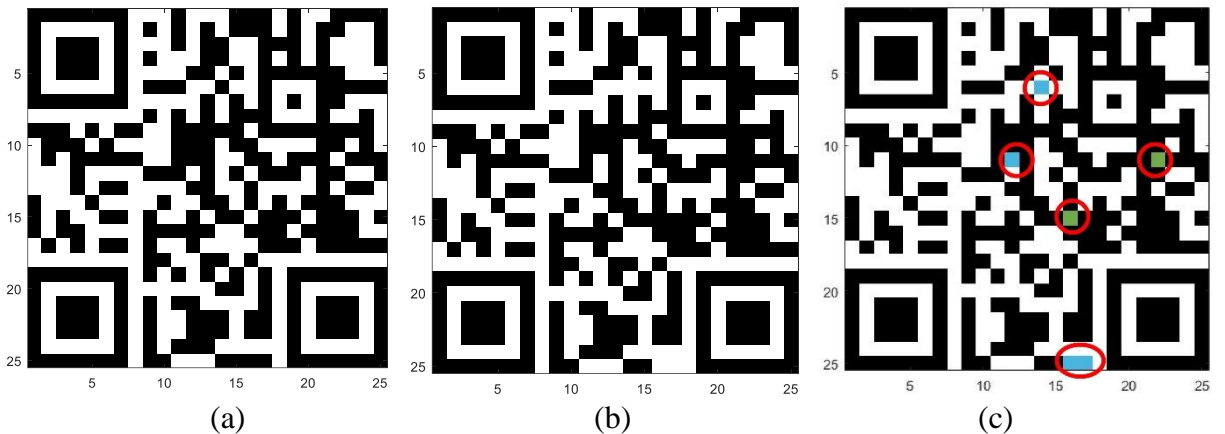


Figure 12. Results for the QR code recovery in aluminum: (a) original QR code, (b) obtained QR code, and (c) discrepancy map between original and recovered code, where blue color represents pixels recovered as white when they were originally black, and green color indicates pixels recovered as black when they were actually white in the true embedded code.

3.2.2 Discrepancy error after post-processing

In order to further refine these results and achieve a perfect recovery (0% error), our post-processing stage which uses an optimal filter is applied. For this application, the filter is trained using the original QR code image and micro-CT images of additive manufactured parts containing an embedded version of the same QR code. The micro-CT images are initially processed using the blind recovery methodology described in Section 3.1.2. Then, for the Dilation step, the selected size of the structuring element, originally of size 21, is varied to include sizes of [1, 3, 5, 7, 39, 41, 43 and 45]. These sizes are much different from the selected one, being excessively small (1, 3) and large (43, 45). As a result, the obtained images are not adequately dilated and present notable noise, leading to errors between 8% and 35%, which can be used to train the filter.

The filter is implemented using a range of different window sizes, including 3×3 , 5×5 and 7×7 in order that the optimal performing filter could be selected. Table 1 shows the error obtained after post-processing for both the aluminum and titanium alloy micro-CT images when different filter sizes were used. The results in Table 1 suggest that small filter sizes such as 3×3 can achieve a 0% error for the aluminum data cube. For the titanium 3×3 case, there is only one pixel in error (1/625) and it is located within a positioning square, having no negative effects.

Table 1. Discrepancy error (%) for both micro-CT images using no post-processing (blind analysis) and with post-processing based on optimal filtering. Results with post-processing depend on the filter size – larger windows are less effective in error reduction and require additional training and execution time.

Code type	No post-processing	Optimal filter as post-processing	
	error (%)	Filter size	Error (%)
Three-layer	0.96	3×3	0
		5×5	0.32
		7×7	0.96
One-layer	0.48	3×3	0.16*
		5×5	0.48
		7×7	0.48

* This error corresponds to only one pixel within a positioning square.

3.2.3 Analysis on the selection of the optimally oriented image (OOI)

For the experiments, the OOI, which best shows the QR code, is selected manually by visual inspection. This step required manual intervention as the rotation resolution in the data cubes is below one degree meaning that a number of images actually look similar to the ideal case. As a result, how the selection of the optimal image may affect the performance of the algorithm is further explored. In turn, by applying the algorithm to a number of images each containing the true QR code imaged from slightly different orientations, and under slightly different conditions, allows us to explore the robustness of the algorithm itself.

A number of contiguous images from the micro-CT data cube are selected for this experiment. Figure 13 shows the results for both the aluminum and titanium data, where the results of measurement error between the recovered code and the ground truth present a clear trend. That is, the discrepancy error is increased for both data cubes when the distance from the OOI is

increased in either direction. Interestingly, the post-processing retains its ability to reduce the error (with some fluctuations) regardless of the image shift.

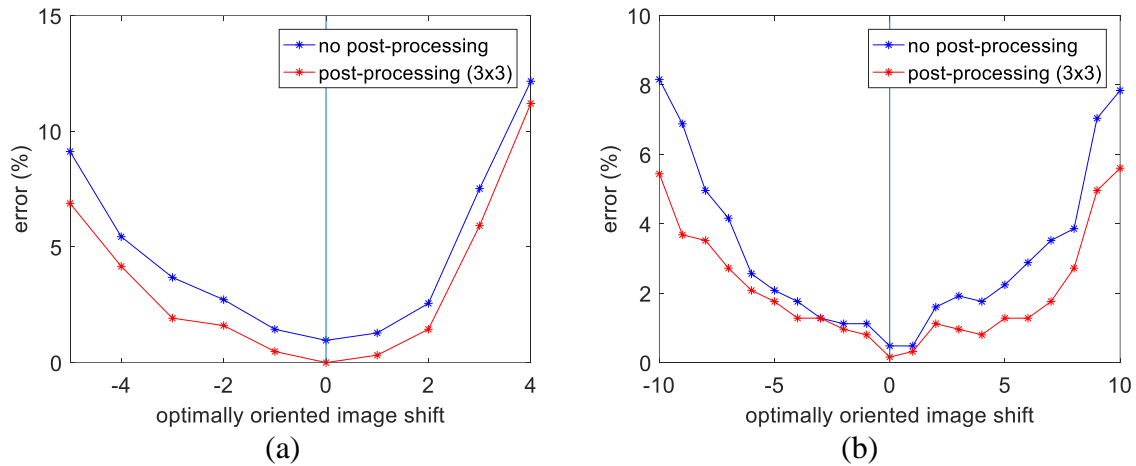


Figure 13 . Discrepancy error (%) for different input images selected as optimally oriented: (a) plot for the aluminum data cube using a 3x3 filter size for post-processing, and (b) plot for the titanium data cube using a 3x3 filter size for post-processing.

The vertical line (no shift) in Figure 13 represents the best image according to visual/manual selection. The shift resolution in the titanium case is greater due to the higher rotation resolution (step of 0.4° instead of 0.6°) during acquisition. Although this is a simple analysis, the same concept could be used in future for an automated selection of the OOI in a given data cube.

4. Conclusions

Identification codes can be embedded inside additive manufactured parts for product authentication purposes. However, imaging of such codes by micro-CT scanning presents challenges because such codes have poor contrast and imaging artifacts. A method is developed in this work to process the obtained images to ensure that the acquired images provide a scannable code. In one of the obfuscation methods, the code is sliced into three parts and embedded in different layers during additive manufacturing. The developed algorithm is demonstrated to work well on both the datasets that have been evaluated. The method applies some pre-processing before the QR code is extracted and an optional post processing step can be applied to suppress noise in the recovered image. Once extracted and processed using our new image processing method, the extracted code can be read using standard QR code scanner technology. While our approach has been demonstrated to be successful, it has only been tested on two datasets (titanium and aluminum) due to the time and financial costs associated with printing and imaging the manufactured parts used in this study. Exploring the generality of our proposed techniques when applied to a number of manufactured parts constructed using the same materials, with the same embedded codes undergoing the same image acquisition and processing will be the subject of further work. Regardless, we have shown in this paper that due to recent advances in AM it is possible to embed hidden codes within multiple layers of manufactured parts and to successfully image and recover readable codes using blind algorithms and advanced supervised techniques.

Acknowledgment

NYU Global Seed Grant for Collaborative Research to Drs. Nikhil Gupta and Khaled Shahin is acknowledged. Authors thank Khulood Alawadi at NYUAD for printing the specimens or testing. NIH SBIR grant 1R43FD006133-01 is also acknowledged for partially funding the study. Profs. Ramesh Karri of ECE Department, NYU, and Khaled Shahin of NYUAD are thanked for useful discussions.

References

1. Guo, N. and Leu, M.C., *Additive manufacturing: technology, applications and research needs*. *Frontiers of Mechanical Engineering*, 2013, **8**(3), 215-243.
2. Julien Gardan, *Additive manufacturing technologies: state of the art and trends*. *International Journal of Production Research*, 2016, **54**(10), 3118-3132.
3. Shapiro, A.A., Borgonia, J.P., Chen, Q.N., Dillon, R.P., McEnerney, B., Polit-Casillas, R., and Soloway, L., *Additive manufacturing for aerospace flight applications*. *Journal of Spacecraft and Rockets*, 2016, **53**(5), 952-959.
4. Ryan Dehoff, Chad Duty, William Peter, Yukinori Yamamoto, Wei Chen, Craig Blue, and Cory Tallman, *Case study: Additive manufacturing of aerospace brackets*. *Advanced Materials and Processes*, 2013, **171**(3), 19-22.
5. Liz Nickels, *AM and aerospace: An ideal combination*. *Metal Powder Report*, 2015, **70**(6), 300-303.
6. Joshi, S.C. and Sheikh, A.A., *3D printing in aerospace and its long-term sustainability*. *Virtual and Physical Prototyping*, 2015, **10**(4), 175-185.
7. Tuomi, J., Paloheimo, K.-S., Vehviläinen, J., Björkstrand, R., Salmi, M., Huotilainen, E., Kontio, R., Rouse, S., Gibson, I., and Mäkitie, A.A., *A novel classification and online platform for planning and documentation of medical applications of additive manufacturing*. *Surgical Innovation*, 2014, **21**(6), 553-559.
8. Salmi, M., Tuomi, J., Paloheimo, K.S., Björkstrand, R., Paloheimo, M., Salo, J., Kontio, R., Mesimäki, K., and Mäkitie, A.A., *Patient-specific reconstruction with 3D modeling and DMLS additive manufacturing*. *Rapid Prototyping Journal*, 2012, **18**(3), 209-214.
9. Liz Nickels, *World's first patient-specific jaw implant*. *Metal Powder Report*, 2012, **67**(2), 12-14.
10. Jardini, A.L., Larosa, M.A., Filho, R.M., Zavaglia, C.A.d.C., Bernardes, L.F., Lambert, C.S., Calderoni, D.R., and Kharmandayan, P., *Cranial reconstruction: 3D biomodel and custom-built implant created using additive manufacturing*. *Journal of Cranio-Maxillofacial Surgery*, 2014, **42**(8), 1877-1884.
11. Sutradhar, A., Park, J., Carrau, D., Nguyen, T.H., Miller, M.J., and Paulino, G.H., *Designing patient-specific 3D printed craniofacial implants using a novel topology optimization method*. *Medical & Biological Engineering & Computing*, 2016, **54**(7), 1123-1135.
12. William E. Frazier, *Metal additive manufacturing: A review*. *Journal of Materials Engineering and Performance*, 2014, **23**(6), 1917-1928.
13. Leuders, S., Thöne, M., Riemer, A., Niendorf, T., Tröster, T., Richard, H.A., and Maier, H.J., *On the mechanical behaviour of titanium alloy TiAl6V4 manufactured by selective laser melting: Fatigue resistance and crack growth performance*. *International Journal of Fatigue*, 2013, **48**, 300-307.
14. Froes, F.H. and Dutta, B., *The additive manufacturing (AM) of titanium alloys*. *Advanced Materials Research*, 2014, **1019**, 19-25.

15. Shamsaei, N., Yadollahi, A., Bian, L., and Thompson, S.M., *An overview of Direct Laser Deposition for additive manufacturing; Part II: Mechanical behavior, process parameter optimization and control*. Additive Manufacturing, 2015, **8**, 12-35.
16. Mohamed, O.A., Masood, S.H., and Bhowmik, J.L., *Optimization of fused deposition modeling process parameters: a review of current research and future prospects*. Advances in Manufacturing, 2015, **3**(1), 42-53.
17. Townsend, A., Senin, N., Blunt, L., Leach, R.K., and Taylor, J.S., *Surface texture metrology for metal additive manufacturing: a review*. Precision Engineering, 2016, **46**, 34-47.
18. Strano, G., Hao, L., Everson, R.M., and Evans, K.E., *Surface roughness analysis, modelling and prediction in selective laser melting*. Journal of Materials Processing Technology, 2013, **213**(4), 589-597.
19. Huang, R., Riddle, M., Graziano, D., Warren, J., Das, S., Nimbalkar, S., Cresko, J., and Masanet, E., *Energy and emissions saving potential of additive manufacturing: the case of lightweight aircraft components*. Journal of Cleaner Production, 2016, **135**, 1559-1570.
20. Seabra, M., Azevedo, J., Araújo, A., Reis, L., Pinto, E., Alves, N., Santos, R., and Pedro Mortágua, J., *Selective laser melting (SLM) and topology optimization for lighter aerospace components*. Procedia Structural Integrity, 2016, **1**, 289-296.
21. Ploch, C.C., Mansi, C.S.S.A., Jayamohan, J., and Kuhl, E., *Using 3D printing to create personalized brain models for neurosurgical training and preoperative planning*. World Neurosurgery, 2016, **90**, 668-674.
22. Zeltmann, S.E., Gupta, N., Tsoutsos, N.G., Maniatakos, M., Rajendran, J., and Karri, R., *Manufacturing and security challenges in 3d printing*. JOM, 2016, **68**(7), 1872-1881.
23. Thomas Kurfess and William J. Cass, *Rethinking additive manufacturing and intellectual property protection*. Research-Technology Management, 2014, **57**(5), 35-42.
24. Gupta, N., Chen, F., Tsoutsos, N.G., and Maniatakos, M., *ObfusCADE: Obfuscating additive manufacturing CAD models against counterfeiting: Invited*, in *Proceedings of the 54th Annual Design Automation Conference 2017*. 2017, ACM: Austin, TX, USA. 1-6.
25. Yampolskiy, M., Andel, T.R., McDonald, J.T., Glisson, W.B., and Yasinsac, A. *Intellectual property protection in additive layer manufacturing: Requirements for secure outsourcing*. in *Proceedings of the 4th Program Protection and Reverse Engineering Workshop*. December 9, 2014. New Orleans, LA, USA: ACM.
26. Yampolskiy, M., King, W.E., Gatlin, J., Belikovetsky, S., Brown, A., Skjellum, A., and Elovici, Y., *Security of additive manufacturing: Attack taxonomy and survey*. Additive Manufacturing, 2018, **21**, 431-457.
27. Chen, F., Mac, G., and Gupta, N., *Security features embedded in computer aided design (CAD) solid models for additive manufacturing*. Materials & Design, 2017, **128**, 182-194.
28. Chen, F., Yu, J.H., and Gupta, N., *Obfuscation of embedded codes in additive manufactured components for product authentication*. Advanced Engineering Materials, 2019, doi: 10.1002/adem.201900146.
29. Ivanova, O., Elliott, A., Campbell, T., and Williams, C.B., *Unclonable security features for additive manufacturing*. Additive Manufacturing, 2014, **1-4**, 24-31.
30. Mandolla, C., Petruzzelli, A.M., Percoco, G., and Urbinati, A., *Building a digital twin for additive manufacturing through the exploitation of blockchain: A case analysis of the aircraft industry*. Computers in Industry, 2019, **109**, 134-152.

31. Chen, F., Luo, Y., Tsoutsos, N.G., Maniatakos, M., Shahin, K., and Gupta, N., *Embedding tracking codes in additive manufactured parts for product authentication*. Advanced Engineering Materials, 2018, Article #201800495.
32. Mai, J., Zhang, L., Tao, F., and Ren, L., *Customized production based on distributed 3D printing services in cloud manufacturing*. The International Journal of Advanced Manufacturing Technology, 2016, **84**(1), 71-83.
33. Susan M. Bridges, Ken Keiser, Nathan Sissom, and Sara J. Graves, *Cyber security for additive manufacturing*, in *Proceedings of the 10th Annual Cyber and Information Security Research Conference*. 2015, ACM: Oak Ridge, TN, USA. 1-3.
34. Sturm, L.D., Williams, C.B., Camelio, J.A., White, J., and Parker, R., *Cyber-physical vulnerabilities in additive manufacturing systems: A case study attack on the .STL file with human subjects*. Journal of Manufacturing Systems, 2017, **44**, Part 1, 154-164.
35. Kietzmann, J., Pitt, L., and Berthon, P., *Disruptions, decisions, and destinations: Enter the age of 3-D printing and additive manufacturing*. Business Horizons, 2015, **58**(2), 209-215.
36. J. Stradley and D. Karraker, *The electronic part supply chain and risks of counterfeit parts in defense applications*. IEEE Transactions on Components and Packaging Technologies, 2006, **29**(3), 703-705.
37. Maruchek, A., Greis, N., Mena, C., and Cai, L., *Product safety and security in the global supply chain: Issues, challenges and research opportunities*. Journal of Operations Management, 2011, **29**(7-8), 707-720.
38. Wilcock, A.E. and Boys, K.A., *Reduce product counterfeiting: An integrated approach*. Business Horizons, 2014, **57**(2), 279-288.
39. K. Zuiderveld, *Contrast limited adaptive histogram equalization*. Graphics Gems IV, 1994, 474-485.
40. Gonzalez, R.C., Woods, R.E., and Eddins, S.L., *Vol. 2. Digital image processing using MATLAB*. 2009, Knoxville: Gatesmark Publishing.
41. Marshall, S., *Logic-Based Nonlinear Image Processing (1st ed.)*. Vol. 72. 2007, SPIE – The International Society for Optical Engineering, Bellingham, Washington, USA. : SPIE Press.
42. *QR Code Reader by Scan (Version 2.3) [Mobile application software]*. Available from: <http://www.scan.me>.
43. *QR Code Reader by TWMobile (Version 2.9.9) [Mobile application software]*. Available from: <http://www.mobile.tw>.