Trusted Integrity Verifier for Additive Manufacturing Systems

Sriharsha Etigowni, Sizhuang Liang*, Mehdi Javanmard, Saman Zonouz, Raheem Beyah*

Department of Electrical and Computer Engineering
Rutgers University, *Georgia Tech

Introduction

Motivating Scenario: STereoLithography (STL) files are one of the critical elements in additive manufacturing systems. Previous attacks on design files (STL files) have shown that adding small voids in critical locations can lead to failure of printed objects [1].

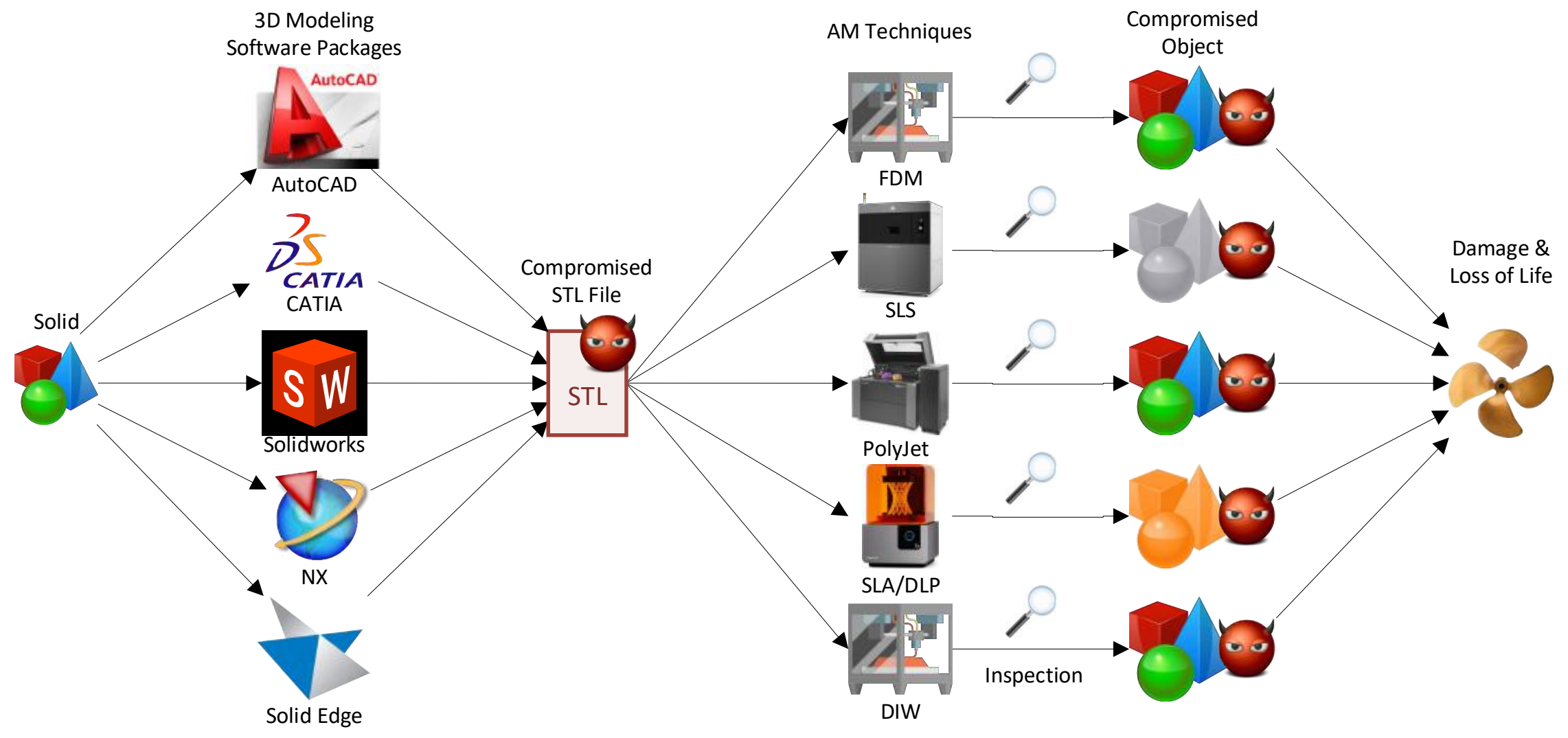


Figure: Additive manufacturing process and location of STL files in the process. A compromised STL file can lead to the failure of the printed object.

```
solid name
    facet normal n_{1,x} n_{1,y} n_{1,z}
        outer loop
            vertex v_{1,1x} v_{1,1y} v_{1,1z}
            vertex v_{1,2x} v_{1,2y} v_{1,2z}
            vertex v_{1,3x} v_{1,3y} v_{1,3z}
        endloop
    endfacet
    ...
    facet normal n_{N,x} n_{N,y} n_{N,z}
        outer loop
            vertex v_{N,1x} v_{N,1y} v_{N,1z}
            vertex v_{N,2x} v_{N,2y} v_{N,2z}
            vertex v_{N,3x} v_{N,3y} v_{N,3z}
        endloop
    endfacet
endsolid name
```

Figure: The format of an STL file with N triangles.

Threat Model

- Untrusted - Design workstation and design toolchain.
- Trusted - AM operators, 3D printers, and their controllers (firmware and slicers), and material.

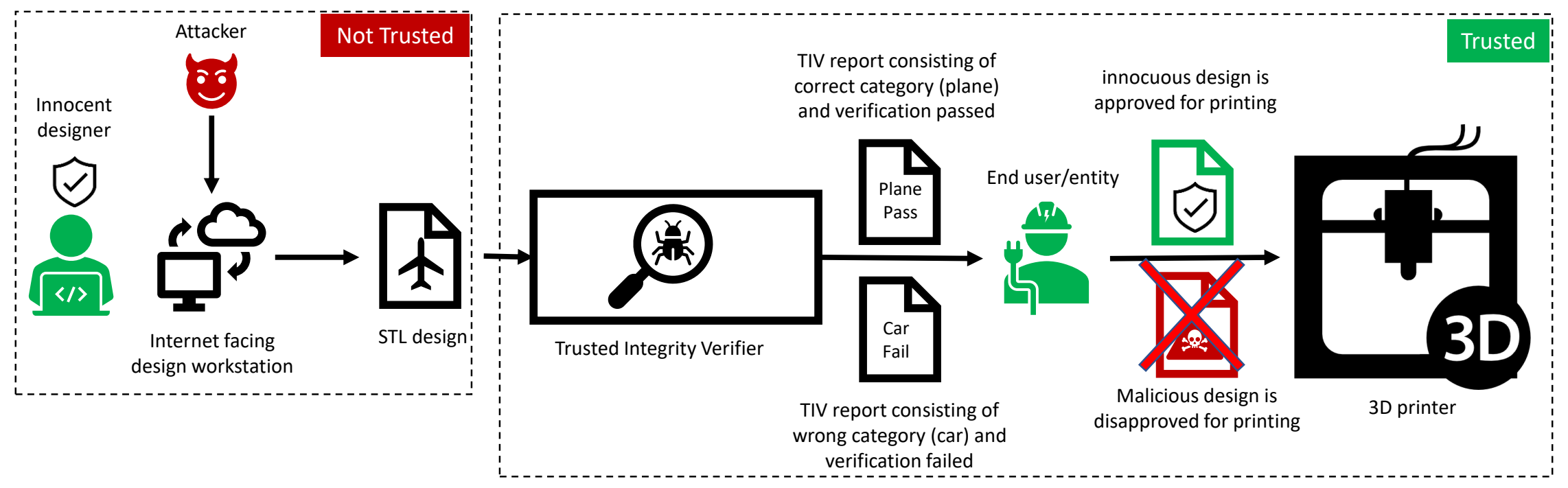

Not Trusted
Attacker
Innocent designer
Internet facing design workstation
STL design
Trusted
Trusted Integrity Verifier
TIV report consisting of correct category (plane) and verification passed
Plane Pass
Car Fail
TIV report consisting of wrong category (car) and verification failed
End user/entity
innocuous design is approved for printing
Malicious design is disapproved for printing
3D
3D printer

Figure: Threat model and application of the TIV framework.

TIV Architecture

The Trusted Integrity Verifier (TIV) consists of three modules.

1. Object classifier to determine the design matches with the actual object intended to print.
2. Suspicious feature detection to determine if there are any suspicious features present in the design that can lead to failure of the object.
3. Safety conditions verification to verify if the suspicious feature will lead to failure of the printed object.

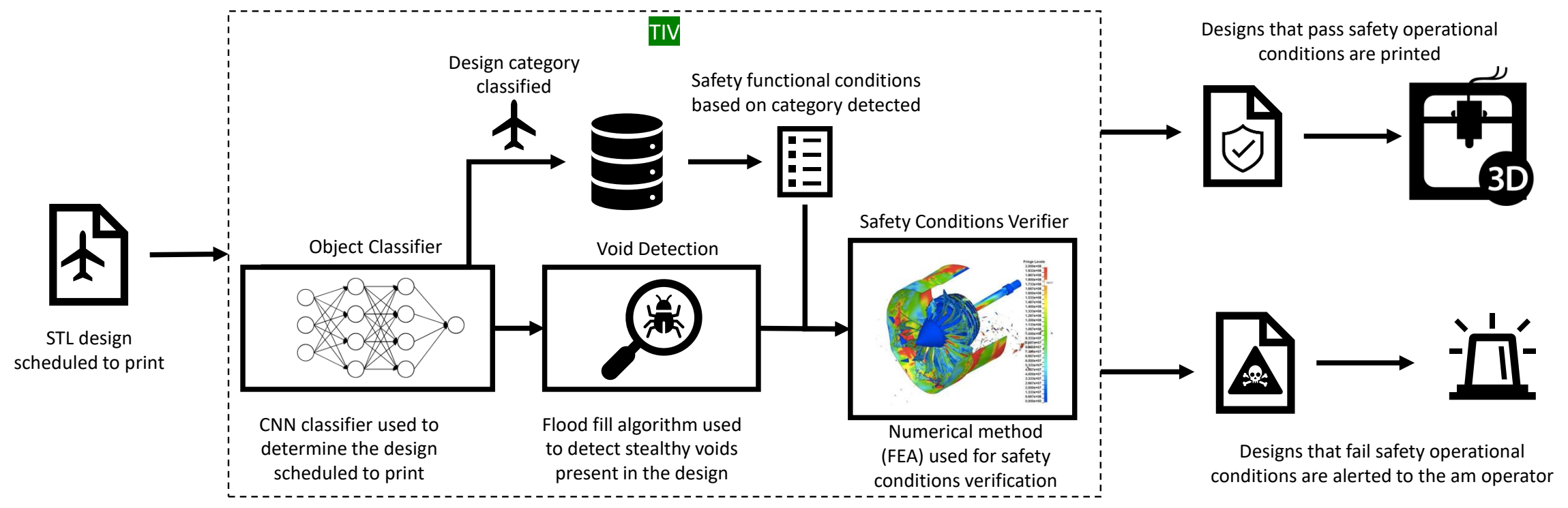

TIV
STL design scheduled to print
Object Classifier
CNN classifier used to determine the design scheduled to print
Design category classified
Safety functional conditions based on category detected
Void Detection
Flood fill algorithm used to detect stealthy voids present in the design
Safety Conditions Verifier
Numerical method (FEA) used for safety conditions verification
Designs that pass safety operational conditions are printed
3D
Designs that fail safety operational conditions are alerted to the am operator

Figure: Structure of the TIV framework.

Object Classification

Octree Convolution Neural Network (O-CNN) [2] is used to classify to determine if there is any major changes in the design of STL file.

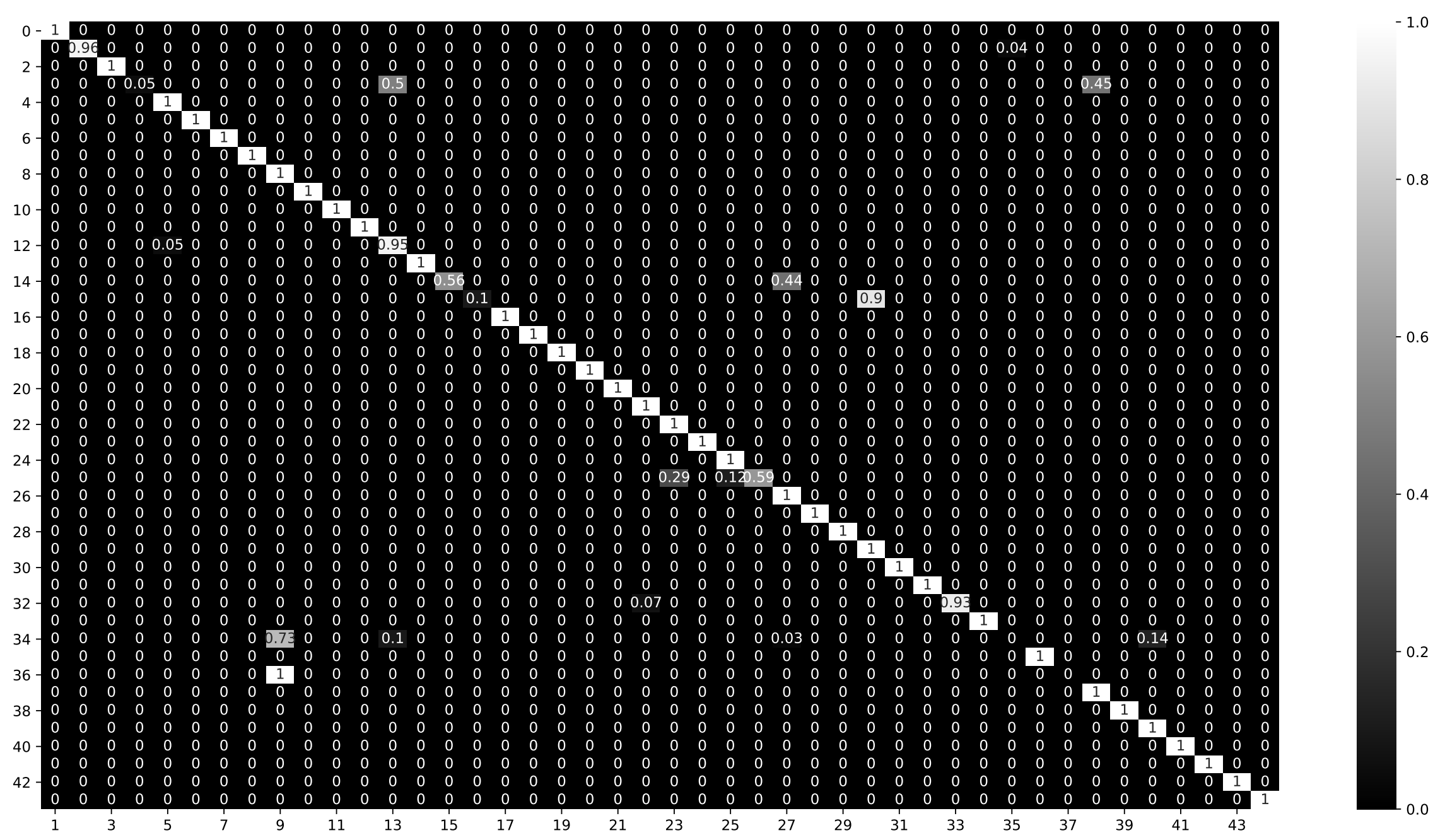


Figure: Confusion matrix for classification of objects into 44 different categories. Diagonal 1's indicates that most of the objects are classified correctly.

Suspicious Feature Detection

Malicious features which could lead to printed object's structural failure are detected by the flood fill algorithm.

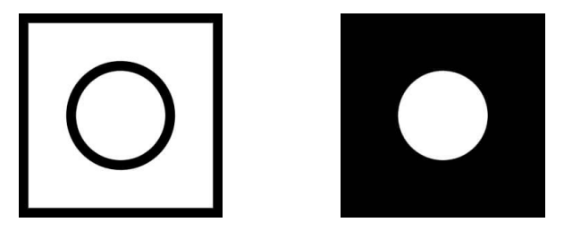
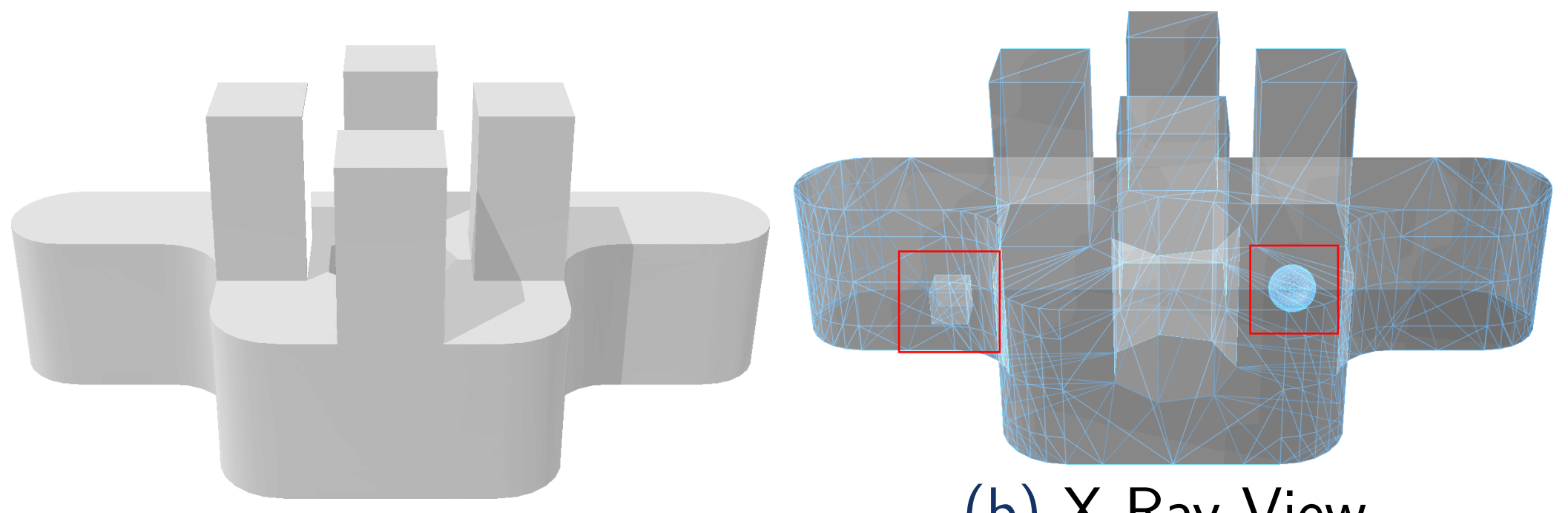


Figure: An image before and after applying the flood fill algorithm.

Automated Attack on Design: The automated attack was performed to get the ground truth of the attack objects. We use the ray casting algorithm to trace the object and add malicious features of random sizes and shapes such that they are not visible externally after being printed.



(a) Normal View. (b) X-Ray View.

Figure: Example of a manually attacked STL file.

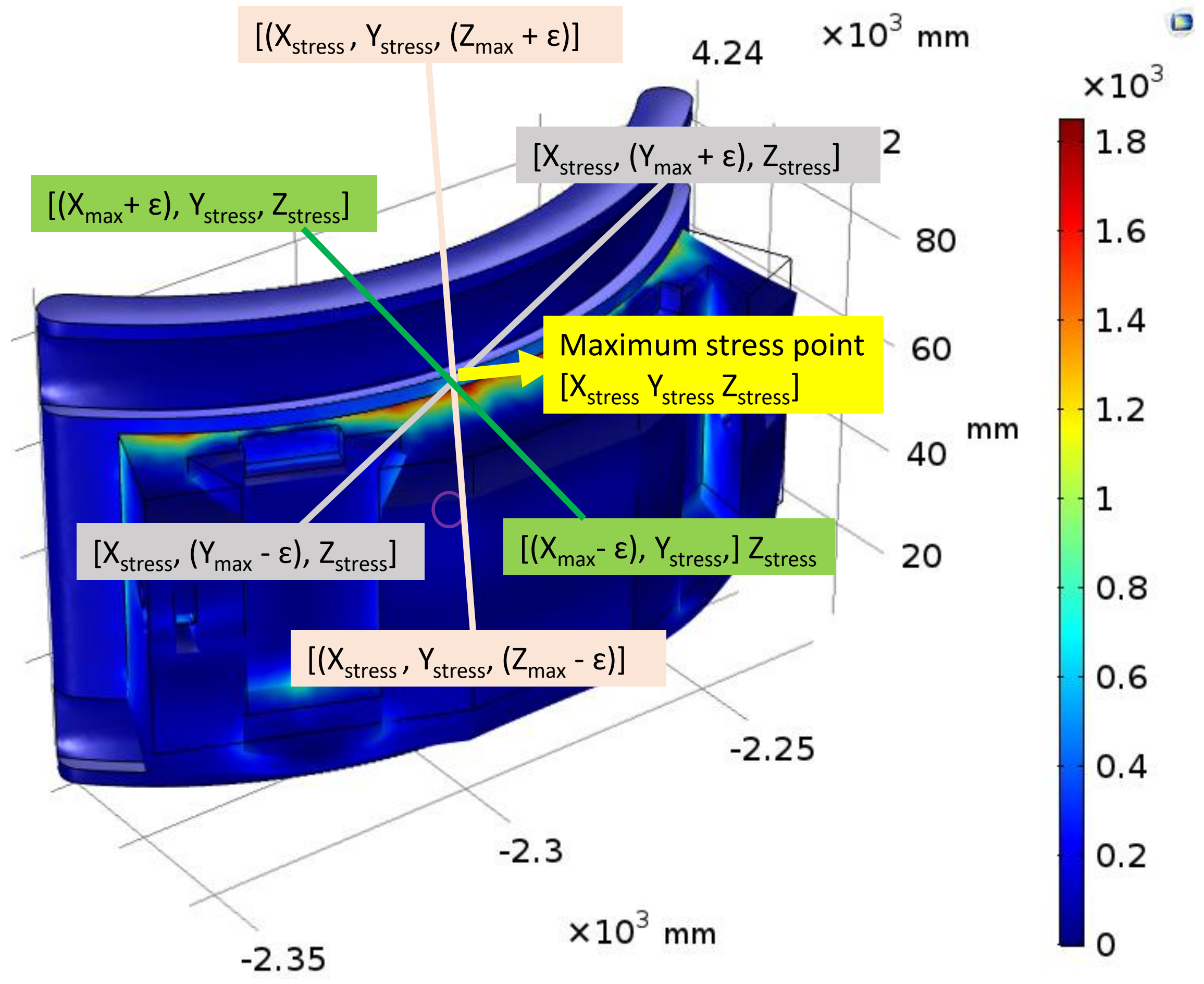

[(X_stress , Y_stress, (Z_max + ε)]
[X_stress, (Y_max + ε), Z_stress]
[(X_max+ ε), Y_stress, Z_stress]
Maximum stress point [X_stress Y_stress Z_stress]
[X_stress, (Y_max - ε), Z_stress]
[(X_max- ε), Y_stress,] Z_stress
[(X_stress , Y_stress, (Z_max - ε)]
4.24
×10^3 mm
2
80
60
40
20
mm
-2.25
-2.3
-2.35
×10^3 mm
×10^3
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0

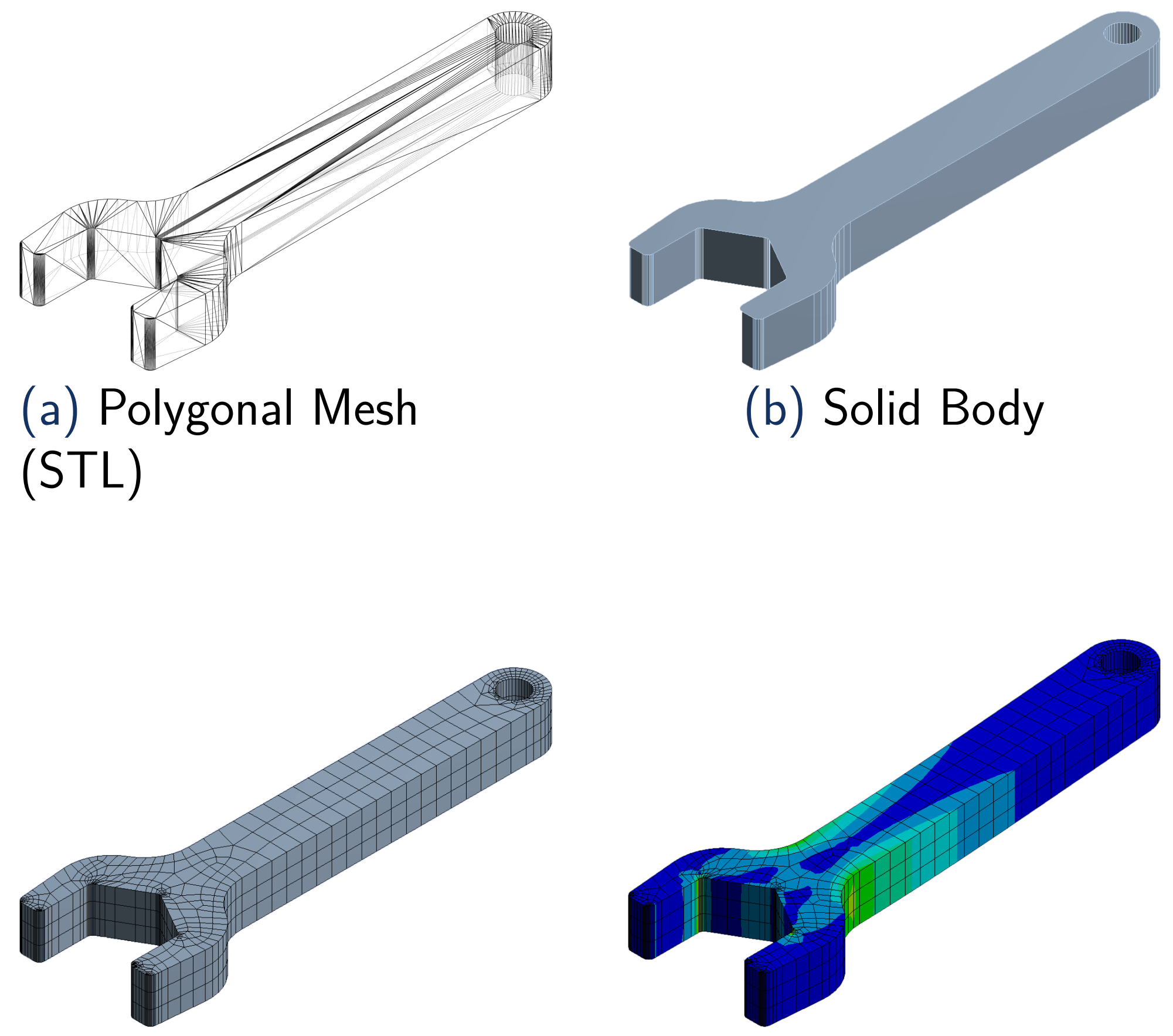
Figure: Ray casting algorithm being used to determine the geometrical boundaries to determine the size of the void that should be inserted.

The inputs for the ray casting algorithm are in the matrix shown below.

$$\phi_{\max} = \begin{bmatrix} X_{\max}+\epsilon & Y_{\text{stress}} & Z_{\text{stress}} \\ X_{\text{stress}} & Y_{\max}+\epsilon & Z_{\text{stress}} \\ X_{\text{stress}} & Y_{\text{stress}} & Z_{\max}+\epsilon \end{bmatrix} \quad (1)$$

$$\phi_{\min} = \begin{bmatrix} X_{\min}-\epsilon & Y_{\text{stress}} & Z_{\text{stress}} \\ X_{\text{stress}} & Y_{\min}-\epsilon & Z_{\text{stress}} \\ X_{\text{stress}} & Y_{\text{stress}} & Z_{\min}-\epsilon \end{bmatrix} \quad (2)$$

Safety Conditions Verification



(a) Polygonal Mesh (STL) (b) Solid Body

(c) Volumetric Mesh (d) Structural Property

Figure: Polygonal mesh, solid body, volumetric mesh, and structural property of a spanner.

References

[1] Sofia Belikovetsky, Mark Yampolskiy, Jinghui Toh, Jacob Gatlin, and Yuval Elovici.
dr0wned – cyber-physical attack with additive manufacturing.
In *11th USENIX Workshop on Offensive Technologies (WOOT 17)*, Vancouver, BC, 2017. USENIX Association.

[2] Donald JR Meagher.
Octree encoding: A new technique for the representation, manipulation and display of arbitrary 3-d objects by computer.
Electrical and Systems Engineering Department Rensseiaer Polytechnic Institute Image Processing Laboratory, 1980.

Acknowledgements

We would like to thank our sponsor:
National Science Foundation (NSF)