

# Classification et estimation de densité de microstructures triplement périodiques avec des réseaux de neurones à convolution 3D

M. R. Guevara Garban<sup>1</sup>, Y. Chemisky<sup>1</sup>, E. Prulière<sup>2</sup>, M. Clément<sup>3</sup>

<sup>1</sup> Univ. Bordeaux, I2M, UMR 5295 351 Cours de la Libération, 33400 Talence, France, {yves.chemisky, manuel.guevara-garban}@u-bordeaux.fr

<sup>2</sup> École Nationale Supérieure d'Arts et Métiers, Campus de Bordeaux, I2M, Esplanade des Arts et Métiers, Etienne.PRULIERE@ENSAM.eu

<sup>3</sup> Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, 33400 Talence, France, michael.clement@labri.fr

---

**Résumé** — Le présent travail vise à proposer une contribution méthodologique pour l'identification et la prédiction de densité de microstructures en utilisant un réseau de neurones convolutif à trois dimensions (CNN 3D). L'algorithme de classification et d'estimation de densité travaille directement avec des données volumiques acquises par tomographie ou résonance magnétique, sans besoin de remaillage supplémentaire. Les expériences préliminaires sur des microstructures générées virtuellement sont prometteuses et ouvrent la voie à la prédiction de comportements mécaniques sur des données volumiques.

**Mots clés** — Réseaux de neurones convolutifs, prédiction de densité, microstructures.

---

## 1 Introduction

L'étude et l'ingénierie des matériaux poreux ont fait l'objet d'une attention particulière ces dernières années en raison de leur rôle crucial dans une multitude d'applications, notamment grâce à l'essor de la fabrication additive. En particulier, les surfaces minimales triplement périodiques (TPMS) [2] représentent une classe de structures poreuses qui présentent des propriétés géométriques et topologiques remarquables, ce qui les rend particulièrement intéressantes dans une optique d'allègement des structures, et également afin d'obtenir des propriétés physiques utiles pour des applications spécifiques (échangeurs de chaleur par exemple).

La conception et l'optimisation des matériaux architecturés exigent cependant un contrôle précis de leurs propriétés micro-structurelles, notamment de leur densité et de leur type de microstructure, afin d'adapter leurs performances à des applications spécifiques.

Il est alors nécessaire d'obtenir des données fiables des caractéristiques géométriques de ces structures, notamment post-fabrication, afin de déterminer précisément leurs propriétés mécaniques. L'analyse de microstructures dépend fortement de la méthode d'acquisition qui est souvent faite par microtomographie, la donnée brute extraite après acquisition est sous la forme d'un volume 3D, selon la résolution de l'appareil d'acquisition (souvent de l'ordre de  $2\mu\text{m}$  par voxel) ce volume peut être très lourd en mémoire et difficile à traiter. Afin de réaliser l'analyse de microstructure avec des outils d'analyse comme la méthode des éléments finis (FEM), le volume 3D doit être converti en maillage. Un maillage est la modélisation géométrique discrète d'un domaine par des éléments géométriques simples, avec le but de simplifier le domaine continu donnée en entrée pour la simulation des calculs.

Notons que les méthodes FFT [8] permettent de travailler directement sur des données issues de microtomographie, mais ces outils peuvent s'avérer peu fiables lorsqu'un contraste important de propriétés mécanique est présent dans la microstructure. Aussi, dans le cas de détermination de contrainte locale en surface, ces méthodes nécessitent une attention particulière.

En raison de ces difficultés de simulation, le développement des méthodes alternatives d'analyse de microstructures est prometteur pour le développement de structures multi-échelles complexes. Un algorithme de classification rapide des microstructures permettant d'accéder directement à des caractéristiques géométriques (classe de TPMS, densité, propriétés mécaniques) à partir de données de microtomographie permettrait de lever ce verrou.

Notons qu'en fonction des caractéristiques mécaniques visées, la densité des cellules architecturées peut fortement varier, ce qui présente une difficulté supplémentaire pour la caractérisation des microstructures. En effet la topologie des cellules est fortement affectée par la variation de densité. La prédiction de

la densité d'une microstructure et de sa classe de géométrie représente pourtant un enjeu important pour l'estimation des propriétés mécaniques.

Les récentes avancées dans le domaine de l'apprentissage profond, en particulier les réseaux de neurones convolutifs (CNN), ont ouvert de nouvelles voies pour relever ces défis. Les réseaux de neurones convolutifs 3D semblent bien adaptés dans l'apprentissage automatique de caractéristiques à partir des volumes 3D [1], ce qui les rend adéquats pour l'analyse de microstructures complexes comme les TPMS.

La contribution principale de cet article est la proposition d'une méthodologie permettant de classifier et de prédire la densité de microstructures TPMS à partir des données volumiques brutes et sans remaillage supplémentaire. Avant de tester cette méthode sur des données de microtomographie, afin de tester la robustesse de l'algorithme proposé, la base de données utilisée pour l'entraînement, la validation et l'évaluation de la capacité prédictive de notre algorithme a été générée numériquement.

Dans les sections suivantes nous présenterons la méthodologie et l'architecture de CNN 3D utilisé, ainsi que des résultats expérimentaux sur la classification de microstructures et l'estimation de densité à partir de volumes 3D.

## 2 Présentation du modèle et méthodologie

### 2.1 Génération de la base de données d'apprentissage

Chaque type de TPMS ayant un comportement mécanique différent en fonction de leur type et de leur densité [2], nous allons générer des TPMS de différentes classes et avec des densités différentes pour apprendre un réseau de neurones à les classifier et à estimer leur densité à partir des volumes 3D. Ces structures ont été générées avec la bibliothèque open-source Microgen [9].

Une TPMS, est une surface minimisant localement son aire, invariante par translation dans les 3 dimensions de l'espace, qui peut être approximée par une fonction

$$f(x, y, z) \rightarrow C \mid \mathbb{R}^3 \rightarrow \mathbb{R}. \quad (1)$$

La plupart des TPMS peuvent être approchées par une combinaison de fonctions trigonométriques. Par exemple, l'expression mathématique de la TPMS de type *Gyroid* affichée en Figure 1 est la suivante :

$$f(x, y, z) = \sin(x) * \cos(y) + \sin(y) * \cos(z) + \sin(z) * \cos(x) = C \quad (2)$$

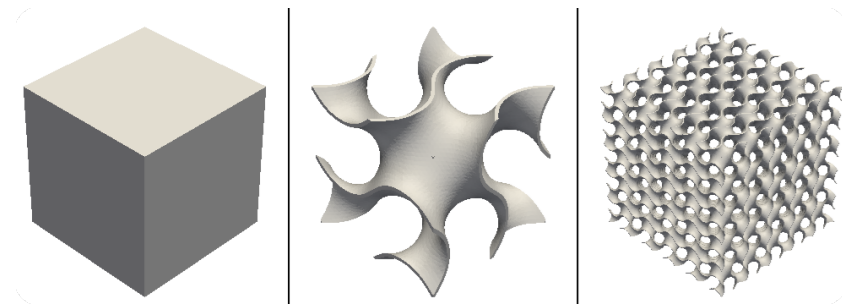


FIGURE 1 – Volume cubique plein à gauche, cellule unitaire de TPMS de type *Gyroid* au milieu et à droite TPMS répétée périodiquement sur un domaine cubique.

Cette fonction est évaluée sur un ensemble de coordonnées 3D  $x, y, z$  appartenant à un volume cubique plein comme celui affiché en figure 1 et uniquement les points satisfaisant l'équation  $f(x, y, z) = C$  seront conservés. La reconstruction de la surface sous la forme d'un maillage 3D, est réalisée en utilisant l'algorithme de reconstruction de surfaces *marching cubes* [6].

Afin d'obtenir une géométrie solide et pas uniquement une surface, deux surfaces sont générées, avec un décalage d'épaisseur noté  $\tau$  qui est intégré à la fonction 1 donnant ainsi :

$$f(x, y, z) = C + \frac{\tau}{2} \mid f(x, y, z) = C - \frac{\tau}{2} \quad (3)$$

En faisant varier le paramètre  $\tau$  on peut modifier la densité de la structure générée, comme affiché en figure 2.

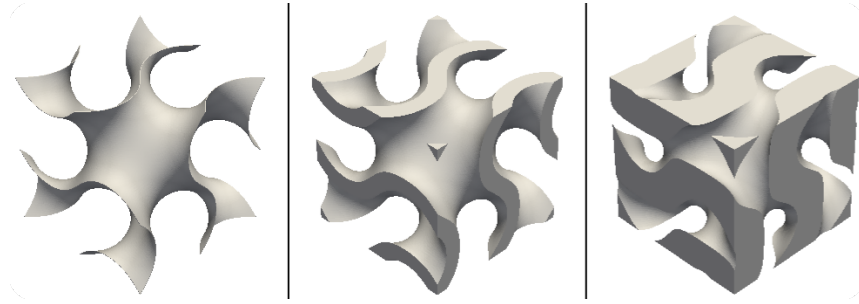


FIGURE 2 – Changement d'épaisseur croissant de gauche à droite sur une *Gyroid*

Le paramètre d'épaisseur  $\tau$  est intrinsèque à chaque type de TPMS, c'est à dire que deux ou plus TPMS peuvent avoir été générées avec le même paramètre d'épaisseur  $\tau$ , mais elles auront alors une densité différente. Afin d'avoir le contrôle sur les densités des TPMS générées nous avons approximé par des régressions polynomiales la relation densité et épaisseur d'une cellule unitaire de TPMS.

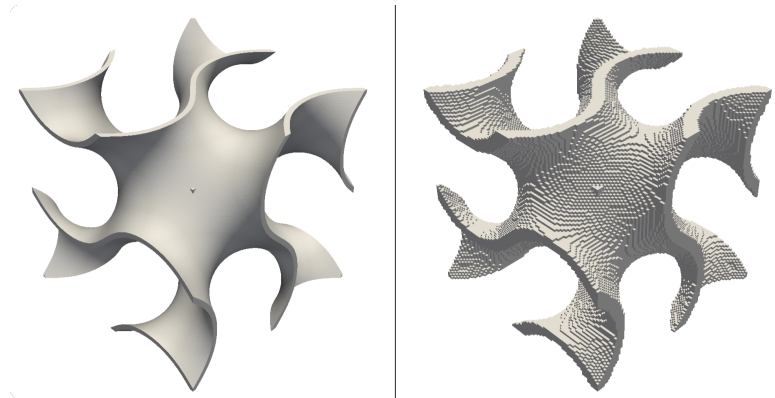


FIGURE 3 – Résultat de processus de voxélisation sur une cellule unitaire de type *Gyroid*

Étant donnée que les données acquises par microtomographie seront sous la forme de volumes 3D et que les données générées sont sous la forme de maillages, une étape de voxélisation est ici nécessaire. Le résultat de cette opération est affiché en figure 3 consiste à prendre chacun des points du maillage et les projeter sur une grille binaire à 3 dimensions.

Il est important de remarquer que les valeurs de densité de la base de données d'apprentissage sont calculées sur les maillages représentant les géométries. Cette précaution évite les erreurs d'approximation de volume par l'étape voxélisation. Cette étape ne respecte en effet pas correctement la densité des cellules en particulier sur les TPMS à faible épaisseur, comme on peut l'observer sur la figure 3.

## 2.2 Présentation de l'architecture CNN 3D utilisée

Dans cette étude, nous utilisons des réseaux de neurones à convolution 3D, qui sont une extension des réseaux de neurones convolutifs proposés par [5]. On peut voir dans la figure 4 un schéma simplifié de l'architecture CNN 3D implémentée, ce type de réseau utilise une grille de voxels à trois dimensions comme données d'entrée.

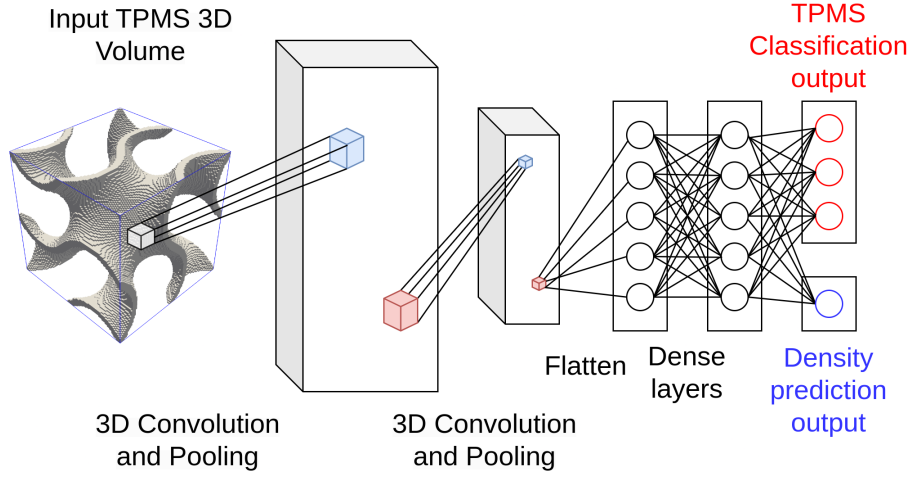


FIGURE 4 – Schéma de l'architecture CNN 3D mise en place, les filtres de convolution 3D sont représentés par les cubes oranges et bleus, ces filtres extraient les caractéristiques les plus importantes du volume pour la classification et l'estimation de densité, ces informations sont transmises via des couches convolutionnelles jusqu'à arriver aux couches denses qui seront ensuite connectées à deux couches de sorties indépendantes pour la classification et la prédiction de densité de TPMS

Les CNN 3D sont largement utilisés, par exemple dans le domaine médical pour l'analyse de données volumiques [1], telles que les images par résonance magnétique (IRM) et la scanographie. De même, les CNN 3D ont été prouvés efficaces pour l'identification en temps réel d'objets 3D représentés sous la forme de volumes [7].

On peut définir mathématiquement une couche d'un réseau de type CNN 3D tel que :

- $X$  soit le volume en entrée d'une couche de convolution :  $X \in \mathbb{R}^{D \times H \times W \times C_{in}}$ , où  $D, H, W$  représentent respectivement, la profondeur, la hauteur et la largeur du volume, et  $C_{in}$  le nombre des canaux du volume 3D.
- $K$  soit l'ensemble des filtres de convolution 3D sous la forme  $K \in \mathbb{R}^{D_k \times H_k \times W_k \times C_{in} \times C_{out}}$  où  $D_k, H_k, W_k$  représentent respectivement, la profondeur, la hauteur et la largeur, du filtre.  $C_{in}$  le nombre de canaux en entrée de la couche et  $C_{out}$  le nombre de canaux en sortie de la couche. On précise que  $C_{out}$  est égal au nombre des filtres de la couche convolutionnelle. La taille des filtres sur les 3 dimensions spatiales est généralement petite, par exemple  $(3 \times 3 \times 3)$  ou  $(5 \times 5 \times 5)$ .
- $b$  représente le terme de biais pour chaque canal de sortie :  $b \in \mathbb{R}^{C_{out}}$
- $\sigma$  représente la fonction d'activation.

L'opération de convolution réalisée par une couche de CNN 3D à la position  $(d, h, w, c_{out})$  peut être exprimée comme suit :

$$X_{out}(d, h, w, c_{out}) = \sigma \left( \sum_{i=1}^{D_k} \sum_{j=1}^{H_k} \sum_{k=1}^{W_k} \sum_{c=1}^{C_{in}} X(d+i, h+j, w+k, c) \cdot K(i, j, k, c, c_{out}) \right) + b(c_{out}) \quad (4)$$

Cette opération est appliquée pour chaque filtre de l'ensemble  $K$  de la couche de CNN 3D, afin d'obtenir un volume tri-dimensionnel à  $C_{out}$  canaux en sortie. Le volume en sortie sera de dimension  $(D \times H \times W \times C_{out})$ . Un *zéro-padding* est ajouté au volume  $X_{in}$  afin d'obtenir la même dimension spatiale en sortie.

La fonction d'activation utilisée pour cet étude est la *Leaky ReLU* représentée comme :  $\phi(X_i)$ , cette fonction est appliquée terme à terme sur le volume  $X_i$ .

$$\phi(X_i) = \begin{cases} X_i, & \text{if } X_i \geq 0 \\ \alpha X_i, & \text{if } X_i < 0 \end{cases} \quad (5)$$

où  $\alpha$  est une constante positive. Pour nos expériences, nous avons choisi une valeur de  $\alpha$  de l'ordre de 0.02.

Dans le contexte de réseaux de neurones à convolution, l'opération de *Max Pooling*, est fondamentale pour réduire la taille en dimension du volume en entrée en extrayant les caractéristiques les plus importantes de ce dernier.

Étant donné un volume d'entrée  $X \in \mathbb{R}^{D \times H \times W \times C_{in}}$  et une fenêtre de pooling de taille  $P_D \times P_H \times P_W$ , pour chaque position  $(d, h, w)$  dans le volume en sortie, la valeur du voxel est obtenue en prenant la valeur maximale entre la fenêtre de pooling et le volume en entrée  $X$ , la taille de cette fenêtre est généralement petite, par exemple la taille de fenêtre  $(2 \times 2 \times 2)$ , permet de diviser par 2 la taille du volume passé en entrée.

Suivant la dernière couche de convolution 3D, le volume sortant  $X_{out}$  est aplati dans un vecteur à 1 dimension de taille  $(D \times H \times W \times C_{out})$ , ce vecteur est donné en entrée à une suite de couches de réseau de neurones denses. Le vecteur sortant des couches denses sera utilisé comme entrée de deux couches de sortie indépendantes,

- une couche de sortie de classification contenant  $M$  neurones, correspondant au nombre de classes de microstructures à prédire, cette sortie utilise une fonction d'activation de type *Softmax* qui convertit le vecteur sortant de la couche de classification vers un vecteur de probabilités où la somme de ce dernier est égale à 1,
- une couche de sortie de régression contenant uniquement un seul neurone destiné à la prédiction de densité. La fonction d'activation utilisée pour cette sortie est la fonction *Sigmoid*.

Afin d'accélérer la vitesse et la stabilité de convergence nous avons utilisé des couches de *Batch Normalization* [3], entre chaque couche interne du réseau.

Le modèle implémenté contient 6 couches convolutionnelles cachées avec 16, 16, 32, 32, 64, 64 filtres de convolution de taille 3, 3, 3 respectivement, des couches de *Max Pooling* sont placées toutes les deux couches de convolution 3D. Le réseau dense utilisé après les couches de convolution contient 3 couches cachées de 16 neurones chacune.

## 2.3 Entraînement du CNN 3D

L'entraînement du réseau se fait par une descente de gradient stochastique par *batches*, cette stratégie consiste à diviser l'ensemble du jeu de données d'entraînement en sous-lots de données chacun de taille  $N$ , l'algorithme d'optimisation utilisé pour nos expériences est *Adam* [4].

Pour répondre au problème de classification de microstructures et d'estimation de densité, nous proposons un CNN 3D à deux sorties, on note :

- $M$  le nombre de classes de microstructures à prédire
- $N$  la taille de chaque *batch* de données :  $N \in \mathbb{Z}^+$
- $i$  l'observation  $i \in N$  d'un *batch* de données.
- $d_i$  la densité vérité terrain :  $d_i \in \mathbb{R}$  et  $0 < d_i < 1$
- $\hat{d}_i$  la densité prédite par le réseau de neurones :  $\hat{d}_i \in \mathbb{R}$  et  $0 < \hat{d}_i < 1$
- $y_i$  la vérité terrain pour le  $i$ ème échantillon  $i \in N$  dans un *batch* de données, sous la forme d'un indicateur binaire contenant une seule valeur non-nulle pour une classe  $m \in M$  donnée :  $y_i = \delta_{mk}$ ,  $k \in M$ ,  $\delta_{mk}$  le symbole de Kronecker où  $\delta_{mk} = 1$  si  $m = k$ ,  $\delta_{mk} = 0$  si  $m \neq k$ .
- $\hat{y}_i$  le vecteur de probabilités en sortie de la couche de classification :  $\hat{y} \in \mathbb{R}^M$ .

On propose une fonction de perte multi-objectif notée  $L$  :

$$L = L_c + \lambda L_d \quad (6)$$

où  $\lambda$  est un hyperparamètre du réseau qui équilibre l'influence entre la classification de microstructures et l'estimation de densités. La valeur de  $\lambda$  est fixée à 1 pour les prédictions présentées dans ce travail.

L'erreur de classification du réseau est mesurée en utilisant la fonction de perte "*Cross Entropy Loss*" qui est définie comme suit :

$$l_c(i) = - \sum_{m=1}^M y_i^m \log(\hat{y}_i^m) \quad (7)$$

L'erreur totale de classification pour un batch donnée est la moyenne des fonctions de perte  $l_c$  individuelles :

$$L_c = \frac{1}{N} \sum_{i=1}^N l_c(i) \quad (8)$$

L'erreur de prédiction sur la densité dans un *batch* a été mesurée selon l'erreur quadratique moyenne comme suit :

$$L_d = \frac{1}{N} \sum_{i=1}^N (d_i - \hat{d}_i)^2 \quad (9)$$

### 3 Expériences

Pour cette étude nous avons utilisé 9 classes de TPMS affichées en Figure 5, la base de données utilisée pour l'entraînement de réseaux de neurones, contient 9000 échantillons de cellules unitaires de TPMS (1000 échantillons par type de TPMS) de résolution  $139 \times 139 \times 139$ , où la densité de chaque structure est tirée aléatoirement suivant une distribution uniforme, où  $0.01 < d < 1$ ,  $d \in \mathbb{R}$  étant la densité.

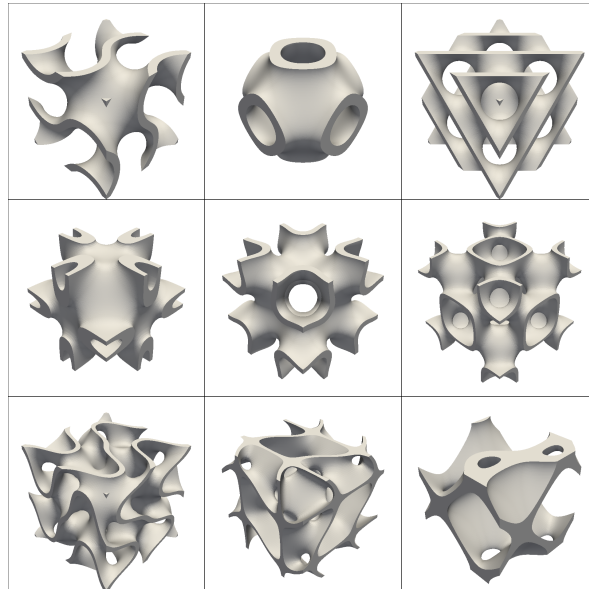


FIGURE 5 – Les 9 types de cellules unitaire de TPMS utilisées pour l'apprentissage. (Gyroid, Schwarz P, Schwarz D, Neovious, Schoen IWP, Schoen FRD, Fischer Koch S, PMY, Honeycomb)

Le jeu de données est reparti en 6300 échantillons (70%) pour le jeu d'entraînement et 2700 échantillons (30%) pour le jeu de test, découpés uniformément pour chaque classe. Nous avons rajouté un processus d'augmentation de données en faisant des rotations aléatoires sur chaque cellule unitaire de TPMS sur les 3 axes de référence.

Après 25 *epochs* d'apprentissage sur le jeu d'entraînement, le réseau obtient un taux de bonne classification de 0.99 sur le jeu de test et une erreur moyenne absolue de 0.04 sur la prédiction de densité.

En Figure 6 on peut trouver la matrice de confusion qui montre les erreurs de classification faites par le modèle proposé sur les données de test, on note ici que le réseau a une tendance à confondre les TPMS de types *Neovious*, *Schoen FRD*, *Schoen IWP* et *Schoen IWP*, cette erreur pourrait s'expliquer à cause d'une ressemblance géométrique entre ces trois types de TPMS quand la densité est importante.

Affiché en Table 1 on peut voir l'erreur d'estimation de densité par classe.

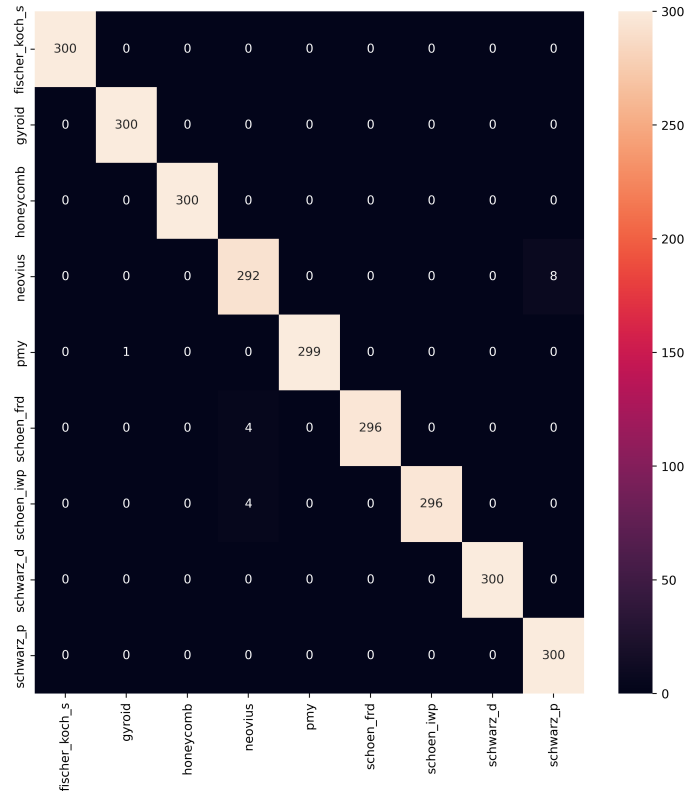


FIGURE 6 – Matrice de confusion montrant les résultats de classification sur le jeu de données de test

TABLE 1 – Erreur d’estimation de densité par classe de TPMS sur le jeu de données de test, *MAE* (*Mean Absolute Error*) étant l’erreur moyenne absolue et *MSE* (*Mean Squared Error*) l’erreur moyenne au carrée.

Classe	Métrique	
	MAE	MSE
Fischer Koch	0.0466	0.0040
Gyroid	0.0417	0.0030
Honeycomb	0.0363	0.0026
Neovius	0.0459	0.0035
PMY	0.0418	0.0029
Schoen FRD	0.0398	0.0028
Schoen IWP	0.0425	0.0030
Schwarz D	0.0402	0.0030
Schwarz P	0.0448	0.0034

Malgré le fait d’avoir obtenu des résultats de classification assez élevés, il existe un risque d’avoir un biais d’apprentissage lié à la distribution de données générées, vu qu’elles proviennent du même générateur, les données d’entraînement et d’évaluation peuvent se ressembler de façon conséquente. Des expériences avec des données expérimentales acquises par microtomographie devraient être réalisées afin de valider le modèle proposé.

## 4 Conclusion et perspectives

Dans cette étude, nous avons proposé un réseau de neurones CNN 3D afin de servir comme un outil de classification de microstructures et estimation de densité à partir des volumes 3D. Ce modèle a été développé grâce à la génération d'une base de données synthétique de TPMS, cependant la méthodologie mise en place reste compatible pour être appliquée à d'autres types de microstructures. Dans nos expériences nous avons utilisé uniquement des cellules unitaires de TPMS, il serait pertinent par la suite, d'entraîner le CNN 3D avec des volumes composés par plusieurs types de microstructures en localisant un volume élémentaire représentatif du matériau architecturé.

Les résultats présentés ouvrent la voie à la prédiction d'autres informations liées à la microstructure, comme la matrice de rigidité, sans besoin d'une étape de remaillage ou de calcul par FEM.

### Références

- [1] ÇIÇEK, Ö., ABDULKADIR, A., LIENKAMP, S. S., BROX, T., AND RONNEBERGER, O. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (2016).
- [2] FENG, J., FU, J., YAO, X., AND HE, Y. Triply periodic minimal surface (TPMS) porous structures: from multi-scale design, precise additive manufacturing to multidisciplinary applications. *International Journal of Extreme Manufacturing* 4, 2 (2022), 022001.
- [3] IOFFE, S., AND SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)* (2015).
- [4] KINGMA, D. P., AND BA, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)* (2015).
- [5] LECUN, Y., BOTTOU, L., BENGIO, Y., AND HAFNER, P. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
- [6] LORENSEN, W. E., AND CLINE, H. E. Marching cubes: A high resolution 3d surface construction algorithm. In *Conference on Computer Graphics and Interactive Techniques, (SIGGRAPH)* (1987), pp. 163–169.
- [7] MATURANA, D., AND SCHERER, S. VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In *International Conference on Intelligent Robots and Systems (IEEE/RSJ)* (2015), pp. 922 – 928.
- [8] MOULINEC, H., AND SUQUET, P. A numerical method for computing the overall response of nonlinear composites with complex microstructure. *Computer Methods in Applied Mechanics and Engineering (CMAME)* 157, 1 (1998), 69–94.
- [9] PRULIÈRE, E., AND CHEMISKY, Y. 3MAH : Un ensemble de bibliothèques pour analyser le comportement complexe de matériaux hétérogènes. In *Colloque National en Calcul des Structures (CSMA)* (2022), pp. 1–8.