

Graphes de textures procédurales pour le rendu par lancer de rayons

Charline Grenier
charline.grenier@etu.unistra.fr

Encadrement :
Basile Sauvage, Jean-Michel Dischler

1 Textures procédurales

Les **textures** permettent d'ajouter des détails visuels dans les mondes virtuels en simulant le **comportement lumineux** des surfaces.



FIGURE 1 - Géométrie et textures d'une scène. Les **textures procédurales** sont générées par ordinateur au moyens de procédures.

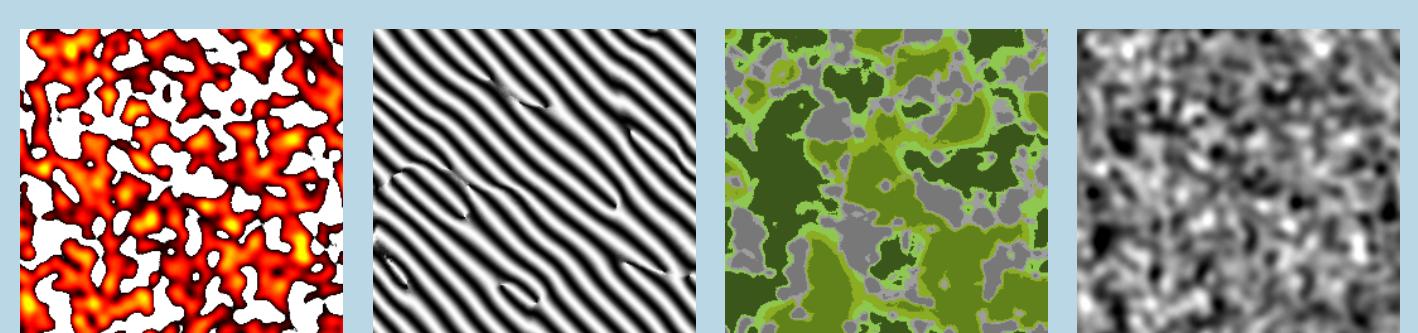


FIGURE 2 - Textures procédurales.

Ces procédures simulent des **processus Gaussiens**.

$$N(uv) = \sum_{i=1}^3 w_i(uv) E_i(uv) =$$

FIGURE 3 - Génération d'un bruit Gaussien N par mélange, pondéré par w_i , des tuiles E_i tirées dans une texture exemple [2].

2 Graphes de textures

Les **graphes** permettent de définir des textures complexes [1].

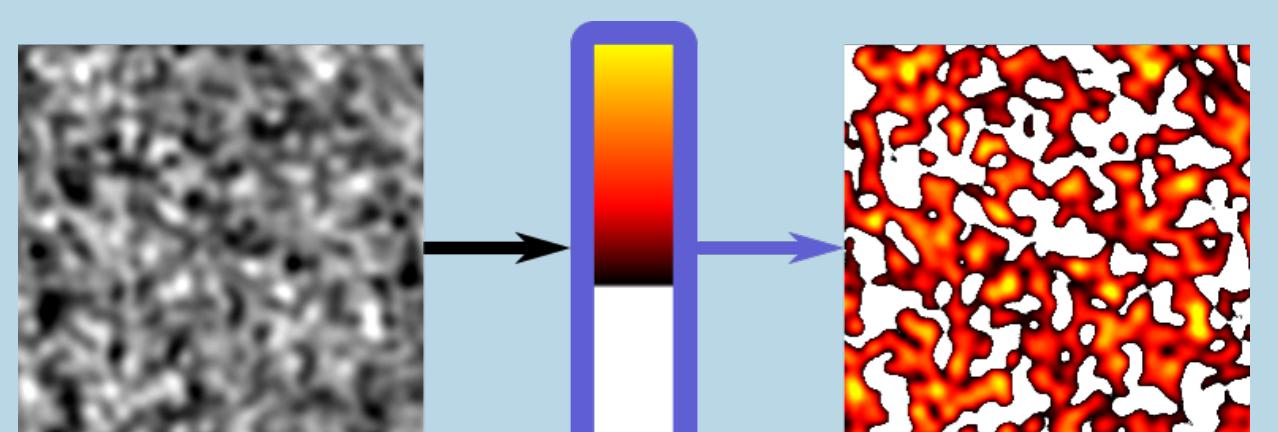


FIGURE 4 - Graphe de texture très simple.

Ils sont composés de champs scalaires en entrée (ici un, à gauche) et de noeuds fonctions de transfert (ici une carte de couleur, dans le cerclage bleu) pour obtenir la sortie (à droite).

3 Rendu

Le rendu consiste à calculer l'image d'une scène capturée par une caméra virtuelle en simulant les interactions lumineuses. Pour éviter les artefacts visuels (scintillement, moiré), chaque pixel prend la couleur moyenne des textures présentes dans son **empreinte**. Ce calcul de moyenne s'appelle le **filtrage**.

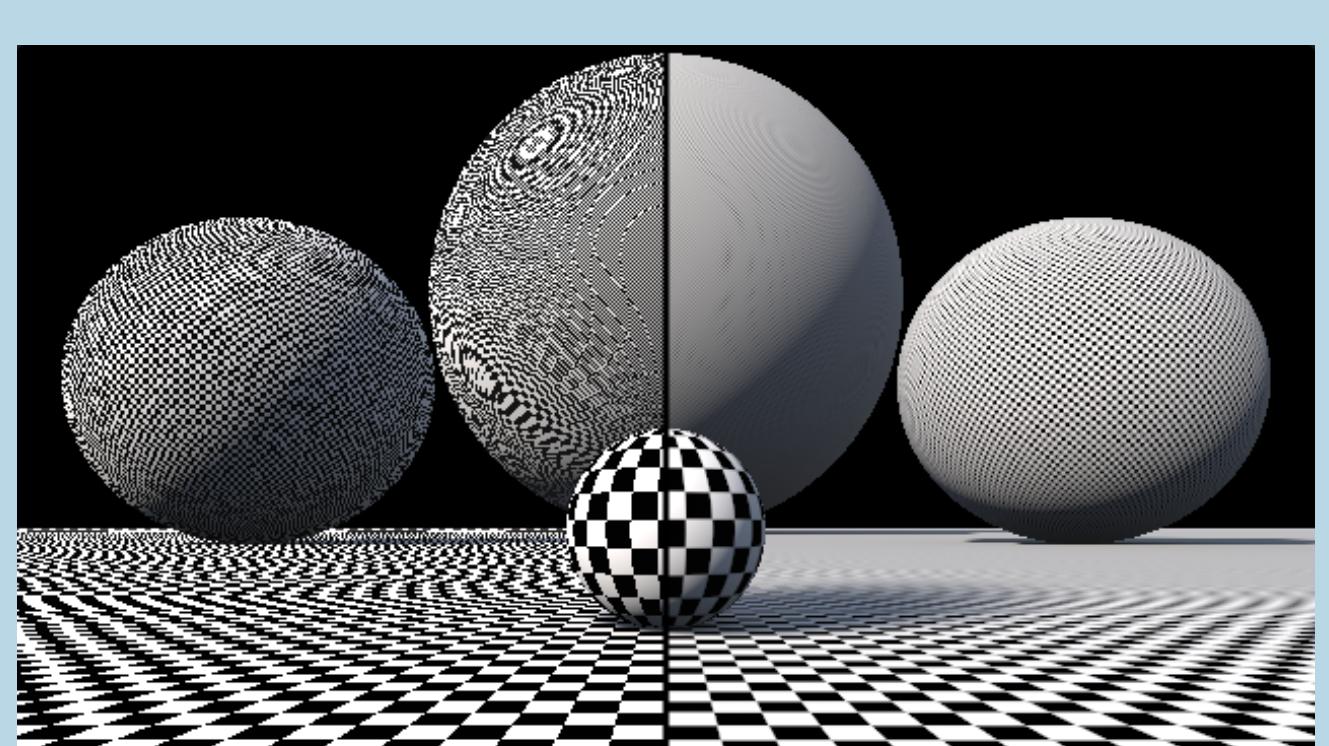


FIGURE 5 - Scène rendu sans (gauche) et avec (droite) filtrage [4].

Références

- [1] Cook, Robert L. *Shade trees*. Proceedings of the 11th annual conference on Computer graphics and interactive techniques. 1984.
- [2] Deliot, Thomas, and Eric Heitz. *Procedural stochastic textures by tiling and blending*. GPU Zen 2 (2019).
- [3] Heitz, Eric, et al. *Filtering Non-Linear Transfer Functions on Surfaces*. IEEE transactions on visualization and computer graphics 20.7 (2013): 996-1008.
- [4] Quilez, I. *filtering the checkerboard pattern*, <https://iquilezles.org/articles/checkerfiltering/>
- [5] Tricard, T., Efremov, S., Zanni, C., Neyret, F., Martinez, J., and Lefevre, S. *Procedural phasor noise*. ACM Transactions on Graphics (TOG) 38, 4 (2019), 1-13.

4 Problématiques étudiées

Défi 1 : L'utilisation des graphes est coûteuse en mémoire car elle nécessite le stockage de la texture résultat. Pour pouvoir faire l'évaluation et le rendu des graphes en **temps réel**, il faut pouvoir en faire le filtrage, ce qui est rendu difficile par leur complexité et leur non linéarité.

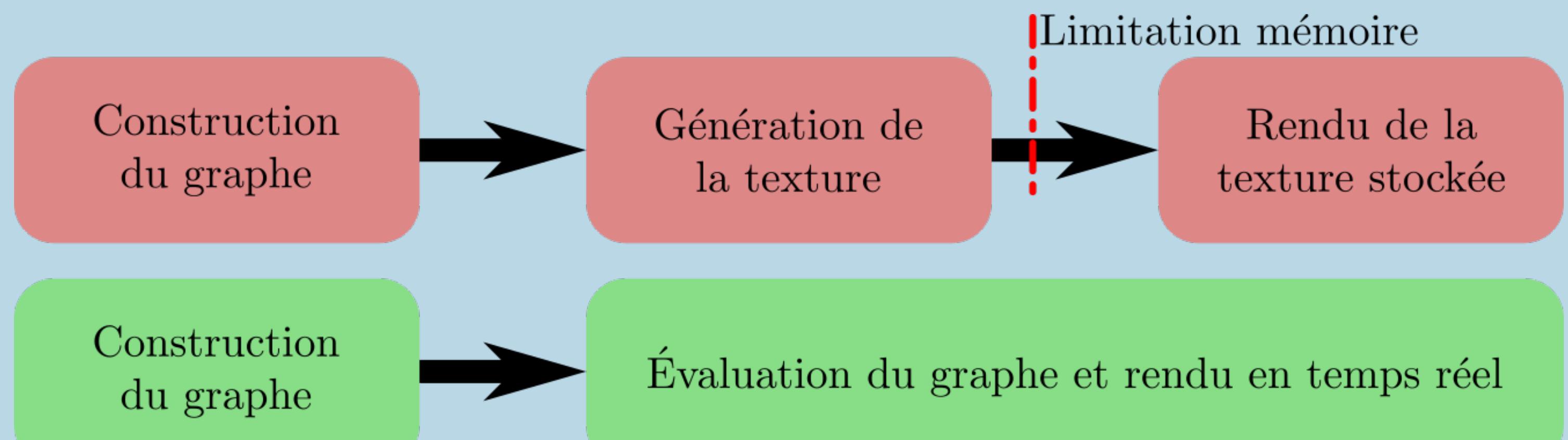


FIGURE 6 - Méthode d'utilisation des graphes en rendu, en rouge la méthode actuelle, en vert la méthode souhaitée.

Défi 2 : La génération et le filtrage en temps réel de **motifs structurés** procéduraux.

- Les bruits Gaussiens sont filtrables mais ne permettent pas de motifs structurés complexes.
- Certaines méthodes permettent des structures complexes mais ne sont pas filtrables.



FIGURE 7 - Motifs par composition avec une carte de couleur 1D (fig. 4).

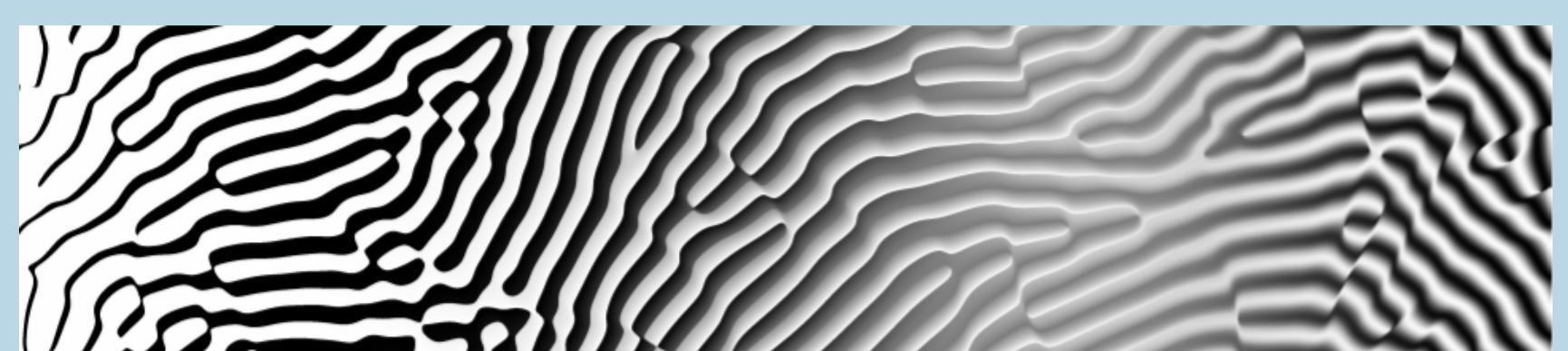


FIGURE 8 - Motifs obtenu par le *procedural phasor noise* [5].

5 Travail réalisé

Génération et rendu en temps réel de motifs procéduraux structurés, en utilisant des fonctions de transfert à deux dimensions.

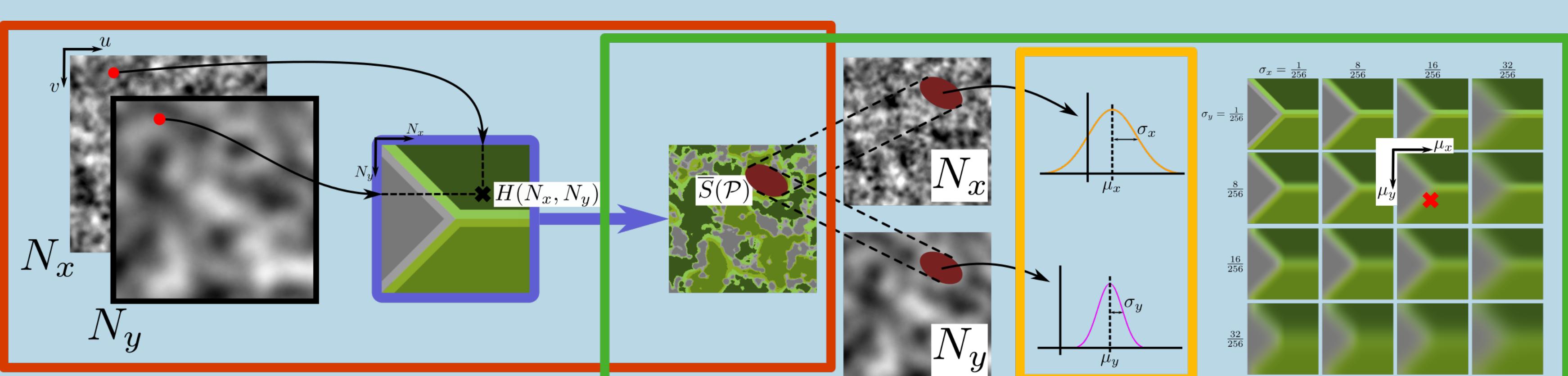


FIGURE 9 - Résumé des contributions.

Contribution 1 : Cadre rouge - Proposition d'un nouveau modèle de noeud représenté par une fonction de transfert bi-dimensionnelle, permettant la génération de nombreux **motifs structurés**.

Contribution 2 : Cadre vert - Implémentation de la méthode de **filtrage** de Heitz et al. [3] adaptée au noeud 2D.

Contribution 3 : Cadre jaune - Estimation en temps réel des **statistiques** d'ordre 2 (variance et covariance), nécessaires au filtrage, en utilisant l'équation de la fig. 3.

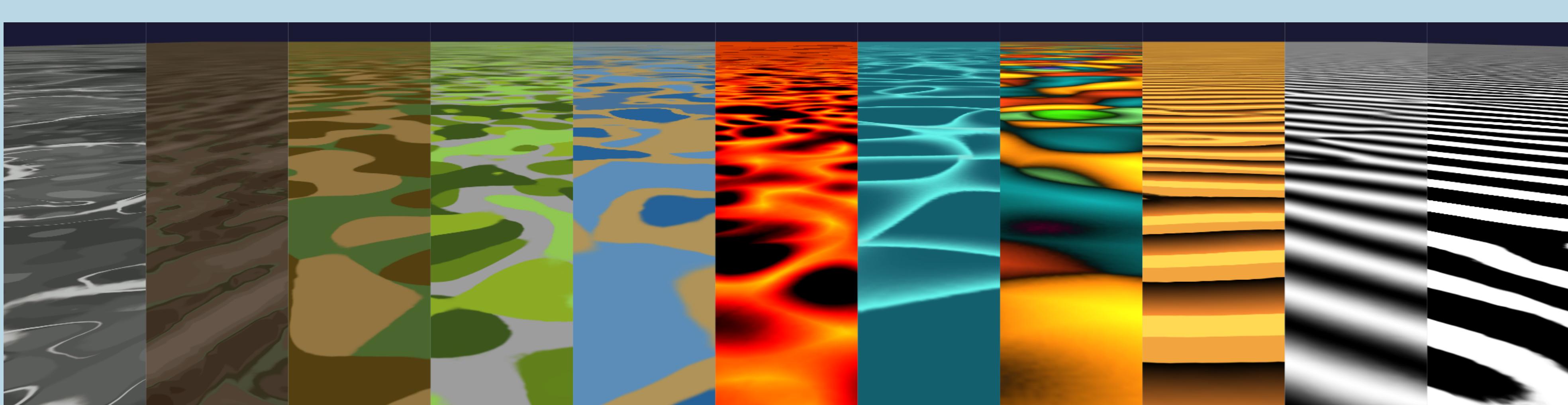


FIGURE 10 - Rendu en temps réel des motifs obtenus.

Remerciements

Merci à Sylvain Thery pour l'implémentation du rendu en temps réel.
Travaux financés par le projet *ReProcTex* de l'Agence Nationale de la Recherche (ANR-19-CE33-0011-01).